Topology Optimization of Multicomponent Beam Structure via Decomposition-Based Assembly Synthesis

Naesung Lyu[1]
Graduate Student,
Research Assistant
e-mail: nlyu@umich.edu

Kazuhiro Saitou[2]
Associate Professor
e-mail: kazu@umich.edu

Department of Mechanical Engineering,
University of Michigan,
Ann Arbor, MI 48109-2125

This paper presents a method for synthesizing multicomponent beam structural assemblies with maximum structural performance and manufacturability. The problem is posed as a relaxation of decomposition-based assembly synthesis, where both topology and decomposition of a structure are regarded as variables over a ground structure with nonoverlapping beams. A multiobjective genetic algorithm with graph-based crossover, coupled with FEM analyses, is used to obtain Pareto optimal solutions to this problem, exhibiting trade-offs among structural stiffness, total weight, component manufacturability (size and simplicity), and the number of joints. Case studies with a cantilever and a simplified automotive floor frame are presented, and representative designs in the Pareto front are examined for the trade-offs among the multiple criteria.
[DOI: 10.1115/1.1814671]

Keywords: Design for Manufacturing, Assembly Synthesis, Structural Design, Genetic Algorithms, Topology Optimization

1 Introduction

Most structural products have complex geometry to meet customer's demand of high functionality with enhanced structural stability. However, manufacturing those products in one piece requires sophisticated methods of process that will increase the total production cost. For this reason, most structural products are multicomponent structures; they are made of a number of components, and these components are assembled into the final structure. Designing a multicomponent structural product often requires designers to decompose overall product geometry at some point during the design process. The decomposition will determine the component set to be assembled into the final product.

For instance, the automotive industry utilizes a handful of basic decomposition schemes of a vehicle, taking into account of geometry, functionality, and manufacturing issues. However, those decomposition schemes are usually nonsystematic and have remained more or less unchanged for decades. This is because the desired form, functionality, materials, joining methods, and overall weight distribution of mass-production vehicles have not changed much for decades. However, the conventional decomposition schemes may no longer be valid for the vehicles with new technologies, such as space frame, ultralightweight materials, and fuel-cell- or battery-powered motors, which would require dramatically different structural properties, weight distribution, and packaging requirements. This motivates the development of a systematic decomposition methodology presented in this paper.

In our previous work [1–3], we have termed *assembly synthesis* as the decision of which component set can achieve a desired function of the end product when assembled together, and assembly synthesis is achieved by the decomposition of product geometry. Since the assembly process generally accounts for more than 50% of manufacturing costs and also affects the product quality [4], assembly synthesis would have a large impact on the quality and cost of the end product. In [3], we proposed a systematic method for decomposing a given product geometry, considering the structural stiffness of the end product, where joints are modeled as torsional springs. During the work, it was observed that the structural integrity (e.g., stiffness) of the end product is heavily influenced by the choice of a particular decomposition as well as the given topology of the structure provided as an input of decomposition. This observation led us to a natural relaxation of the problem where both topology and decomposition of a structure are regarded as variable. This is the problem addressed in the present paper.

In this paper, topology and decomposition of a structure are simultaneously optimized over a ground structure with nonoverlapping beams, for overall structural performance and manufacturability. As in [3], the joints between components are modeled as torsional springs. A multiobjective genetic algorithm [5,6] with graph-based crossover [7–9], coupled with FEM analyses, is used to obtain Pareto optimal designs, exhibiting trade-offs among structural stiffness, total weight, component manufacturability (size and simplicity), and the number of joints. Case studies with a cantilever and a simplified automotive floor frame are presented, and representative designs in the Pareto front are examined for the trade-offs among the multiple criteria.

2 Related Work

2.1 Structural Topology Optimization. Structural optimization can be classified into three categories: topology optimization, shape optimization, and size optimization [10]. Among these three categories, topology optimization is considered as the most general optimization problem with the largest design space that can produce solutions with no prior assumptions. As one of the topology optimization methods, the ground structure approach was first proposed by Dorn et al. [11]. In the ground structure approach, optimal substructures can be found as a subset of a predefined large set of discrete beam elements in an extended design domain (i.e., ground structure). Extensive research has been done to develop numerical methods for the topology design using ground structures: layout theory for frames and flexural systems [12,13], an approach using branch and bound algorithm [14], and genetic algorithm [15]. Recently, Beckers [16] successfully

[1]Currently Post-doc Research Fellow, Department of Mechanical Engineering, University of Michigan.

[2]To whom correspondence should be addressed.

Contributed by the Design Automation Committee for publication in the JOURNAL OF MECHANICAL DESIGN. Manuscript received April 21, 2003; revised April 19, 2004. Associate Editor: G. M. Fadel.

solved a large-scale compliance minimization problem using a dual approach. More detailed history of development on the ground structure approach can be found in Refs. [17] and [18].

Another class of topology optimization method assumes structures made from solid continuum, rather than from discrete beams, where the topology optimization problem is formulated as a material distribution problem within an extended design domain. The Homogenization Design Method (HMD) is a representative of such ''continuum-based'' topology optimization methods [19], where material inside an extended design domain is treated as a composite material made of microstructures consisting of material and void. HMD has been applied to a broad range of problems, including multiple loading problems [20], compliant mechanism design problems [21], multiple constraints problems [22], and topology optimization problems with composite material [23]. More closely related to the present work, several researchers investigated the homogenization-based topology optimization of multicomponent structures [24–27]. These approaches, however, require overlapping extended design domains for each component and each joint as a predefined input.

2.2 Design for Assembly/Manufacturing and Assembly Sequence Design. Boothroyd and Dewhurst [28] are widely regarded as major contributors in the formalization of the design for assembly (DFA) and design for manufacturing (DFM) concepts. In their work [29], assembly costs are first reduced by the reduction of part count, followed by the local design changes of the remaining parts to enhance their assembleability and manufacturability. This basic approach is adopted by most subsequent works on DFA/DFM. In most cases of DFA/DFM procedures, assembleability or manufacturability analysis requires a targeting product to be decomposed into features containing one assembly or manufacturing meaning, such as surfaces, dimensions, tolerances, and their correlations [30]. Therefore, DFA/DFM methods require predetermined components set with given geometries to improve existing designs by modifying the geometries of given components. Regarding the aspect that the earlier attention on the manufacturability and the shape and geometry of components can save total cost and production time, DFA shears the idea of the decomposed-based assembly synthesis concept proposed in this paper. However, while DFA is mainly analyzing or improving a proposed design (i.e., already ''decomposed'' product design with ''given'' topology) from the view point of assembly and manufacturability by modifying geometry of given components, the method proposed in this paper can start without any prescribe components and generate an optimized components set considering manufacturability and structural characteristic of the assembled structure.

Many researchers have investigated the integration of DFA and assembly sequence planning [31,32], where assembly sequence planning is proposed as the enumeration of geometrically feasible cut-sets of a liaison graph, an undirected graph representing the connectivity among components in an assembly. While graph representation has been extensively used in the assembly planning to illustrate the correlation between given components and to find the sequence of assembly process, the application of graph representation to generate the possible components is relatively new [33,34]. In the proposed research, a decomposition-based components set generation method is adopted to generate all possible components set of a given overall structure.

3 Approach

This section describes our method for synthesizing multicomponent structural assemblies with maximum structural performance and manufacturability. Topology of a structure is represented as a subset of a ground structure consisting of a large set of nonoverlapping beams (we call them *basic members*) within a given design domain. Joints within a structure are modeled as torsional springs, which can be placed only at the intersections of basic members in the ground structure. Joints are assumed to be less stiff than the beam elements and therefore reduce the overall structural rigidity. The torsional spring rates are considered to represent the structural characteristic of the corresponding joints. If the joint contains a joining between two components, then physical welds need to be placed between two components, resulting changed structural characteristics at the joint. This structural change can be represented as the changed torsional spring rates in the structural model.

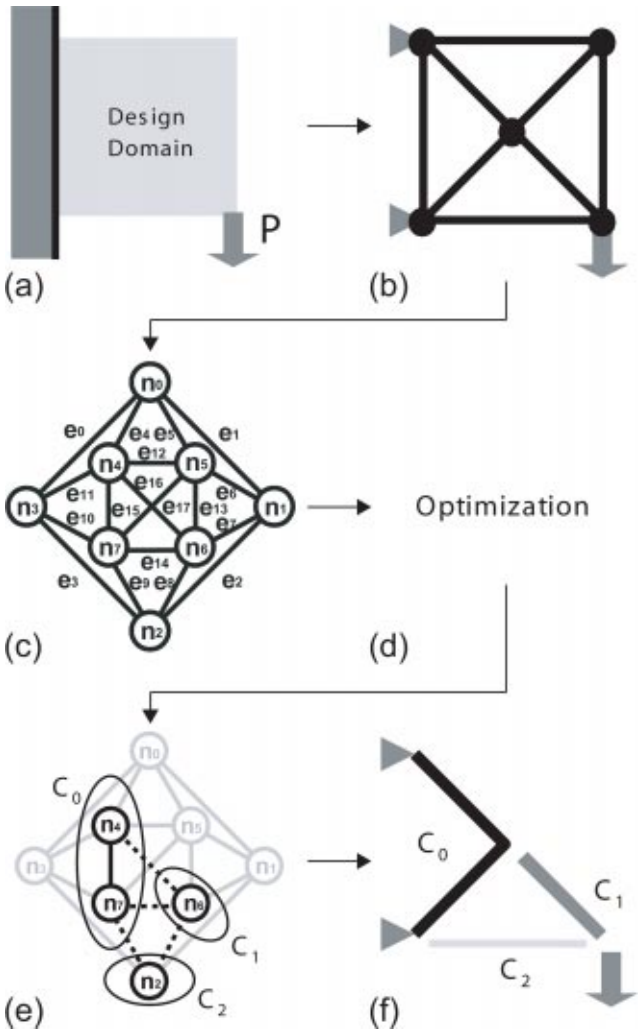

Fig. 1 Outline of the approach (*a*) design domain, (*b*) ground structure consisting of basic members and potential joint locations, (*c*) the ground topology graph G_g, (*d*) optimization, (*e*) best product topology graph G (subgraphs representing components are annotated as C_1-C_3, and edges in joint set J is shown in dashed lines), and (*f*) optimal multicomponent structure

Topology of a ground structure can be represented by the *ground topology graph* $G_g=(V_g,E_g)$ with node set V_g of the basic members defined by all beam elements in the ground structure and edge set E_g of the intersections of the basic members (i.e., potential joints in the ground structure). Similarly, we represent the topology of a multicomponent structural by the *original topology graph* $G_0=(V,E)$, a subgraph of G_g where $V\subseteq V_g$ and $E\subseteq E_g$. Using these notations, the following steps outline the approach (Fig. 1):

1. Given a design domain with boundary and loading conditions (Fig. 1(*a*)), define the ground structure (Fig. 1(*b*)).
2. Construct the ground topology graph $G_0=(V_0,E_0)$ for the ground structure (Fig. 1(*c*)).

3. Using an optimization algorithm (Fig. 1(*d*)), obtain the product topology graph $G=(V,E,J)$ that gives the best structural performance and manufacturability (Fig. 1(*e*)).
4. Construct the multicomponent structure corresponding to G (Fig. 1(*f*)).

3.1 Definition of Design Variables. In order to uniquely specify a topology from the ground structure, a binary vector $\boldsymbol{t}$, named as the *topology vector*, is defined. The *topology vector* $\boldsymbol{t}$ represents the existence of each basic member (a node in ground topology graph G_g) in the original topology graph $G_0=(V,E)$:

$$\boldsymbol{t}=(t_0,t_1,\ldots,t_i,\ldots,t_{gnn-2},t_{gnn-1}) \tag{1}$$

where $gnn=|V_g|$=the number of nodes in the G_g and

$$t_i=\begin{cases}1 & \text{if basic member } i \text{ exists in the structure}\\ 0 & \text{otherwise}\end{cases} \tag{2}$$

For a given $\boldsymbol{t}$, therefore, node set $V\subseteq V_g$ and edge set $E\subseteq E_g$ of G_0 can be defined as

$$V=\{n_i\in V_0 | x_i=1\} \tag{3}$$

$$E=\{e|e=\{u,v\}\in E_0,u\in V,v\in V\} \tag{4}$$

Note that $\boldsymbol{t}$ can only define the existence of beams in the topology without defining the size of the beams in the topology because the elements in the vector $\boldsymbol{t}$ are binary values. The size of the beams in the given topology defined by $\boldsymbol{t}$ is determined by defining another vector $\boldsymbol{w}$, named as the *beam size vector*:

$$\boldsymbol{w}=(w_0,w_1,\ldots,w_i,\ldots,w_{gnn-2},w_{gnn-1}) \tag{5}$$

where w_i represents the cross sectional dimension for the ith beams in the ground structure.

Figure 2 illustrates a sample topology defined by $\boldsymbol{t}$ and $\boldsymbol{w}$ from the ground topology graph G_g in the Fig. 1(*b*). Note that the gray nodes in Fig. 2(*d*) indicate the corresponding basic members with $t_i=0$ and also only when $t_i=1$, w_i (the ith element of $\boldsymbol{w}$) can be realized.

The structure defined by the *original topology graph* G_0 is considered as a one-component structure. Therefore, a method that can define the components set from G_0 is required. In this paper, we used joint library method to define the components set. In the joint library method [35], $\boldsymbol{JL}_i$, a joint library for the ith possible joint location is defined at each possible joint location

$$\boldsymbol{JL}_i=(JL_{i\_}C_0,JL_{i\_}C_1,\ldots,JL_{i\_}C_j,\ldots,JL_{i\_}C_{cnji-1}) \tag{6}$$

where $cnji$ is the number of configurations in the ith joint library and $JL_{i\_}C_j$ is the jth configuration of the ith joint library. However, the structure of each possible joint location changes with the topology vector $\boldsymbol{t}$. Therefore the joint library of each possible location $\boldsymbol{JL}_i$ also changes with $\boldsymbol{t}$. As an example, the joint library at the possible joint location $\mathbf{J}_1$ in Fig. 3 does not exist because the topology defined by the topology vector $\boldsymbol{t}$ contains no beams around $\mathbf{J}_1$. To define the entire joint configuration $\boldsymbol{JL}$, named as the *joint library vector*, is defined as follows:

$$\boldsymbol{JL}\in\boldsymbol{JL}_0\times\boldsymbol{JL}_1\times\ldots\times\boldsymbol{JL}_{gjn-1} \tag{7}$$

where gjn is the number of possible joint locations in the ground structure.

Note that the elements of $\boldsymbol{JL}$ are defined in the ground topology graph G_g. In this method, the portions in the G_g that correspond with each possible joint location are modified according to the selected joint library case ($JL_{i\_}C_j$). Therefore, the edges to be cut in G_g (or G_0) can be obtained by considering all elements JL_i in the joint library vector $\boldsymbol{JL}$ (Fig. 3).

Finally, the attributes of the joining between components are assigned to each joint location. The *individual joint attribute vector* $\boldsymbol{A\_J}_i$ is a vector whose elements are assigned to the beam elements that merge to the joint location:

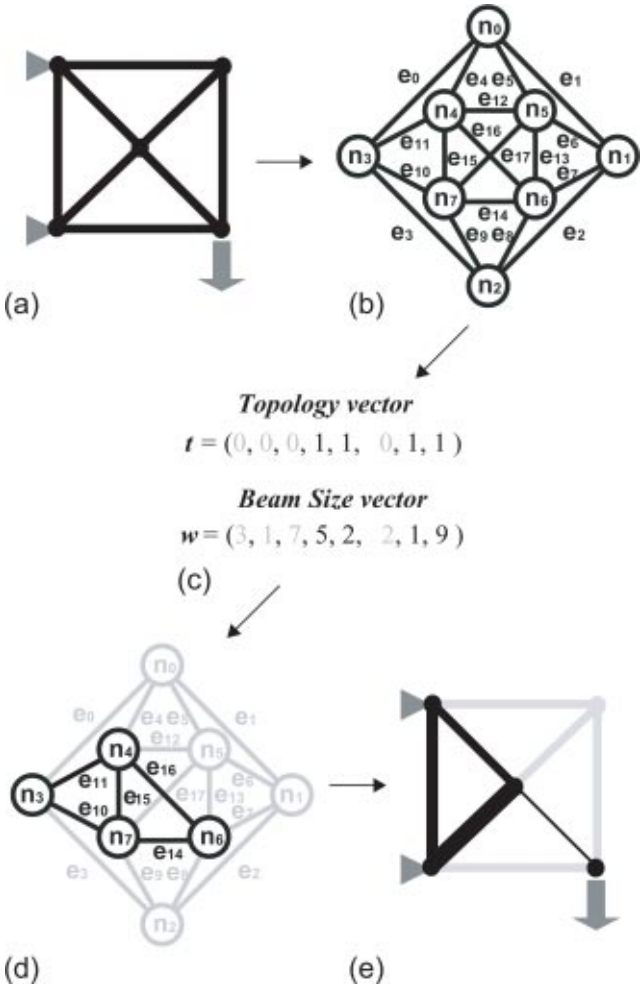

Fig. 2 A sample topology defined by a topology vector t: (*a*) the ground structure, (*b*) G_g the ground topology graph, (*c*) a sample topology vector t and beam size vector w, (*d*) the original topology graph G_0, defined by t and w, and (*e*) the corresponding topology. Note that only the black-colored elements of w are realized in (*e*).

$$\boldsymbol{A\_J}_i=(A\_J_{i\_}B_0,A\_J_{i\_}B_1,\ldots,A\_J_{i\_}B_{bnji-1}) \tag{8}$$

where $bnji$ is the number of beams at the ith possible joint location and $A\_J_{i\_}B_j$ is the joint attribute between the jth beam of the ith possible joint location and the joint. To define the entire joint attributes using $\boldsymbol{A\_J}_i$, the *joint attribute vector* $\boldsymbol{A\_J}$ is defined

$$\boldsymbol{A\_J}=(\boldsymbol{A\_J}_0,\boldsymbol{A\_J}_1,\ldots,\boldsymbol{A\_J}_{gjn-1}) \tag{9}$$

where gjn is the number of possible joint location in the ground structure.

Figure 4 illustrates the joint model and the joint attributes at a specific location ($\mathbf{J}_2$). The FE model shown in the Fig. 4(*b*), the torsional spring elements $k_1\sim k_3$ are used to represent the structural characteristics of the joining between the component and the joint.

In the following case studies, joint attribute $\boldsymbol{A\_J}_i$ represents the rates (spring constants) of torsional springs [Nmm/rad] of the corresponding joint. Because the loading conditions and corresponding displacements considered in the case studies are two-dimensional (2D), only the torsional spring rate normal to this 2D plane (rotation about z-axis in the case studies) is considered as the design variable.

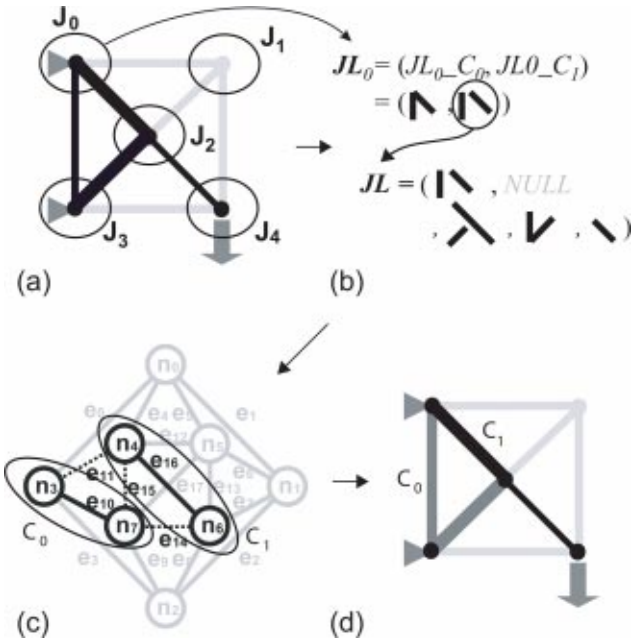

Fig. 3 Components set defined by the joint library method: (*a*) topology defined by *t* and *w*, (*b*) a sample joint library vector *JL*. Note that the joint library JL_1 at the location J_1 does not exist (marked as *NULL* in *JL*), (*c*) corresponding subgraph, and (*d*) physical components set of (*c*) with two components C_0 and C_1.

3.2 Definition of Constraints. Topology of the structure defined by the topology vector ***t*** must satisfy the following constraints to avoid infeasible topologies as a mechanical structure:

- Connectivity Constraint 1: All beams should be connected to at least one other beam element, i.e., product topology graph G_0 should be connected (Figs. 5(*a*) and (*b*)).
- Connectivity Constraint 2: The following points should be connected to at least one beam element (Figs. 5(*c*)–5(*e*)):

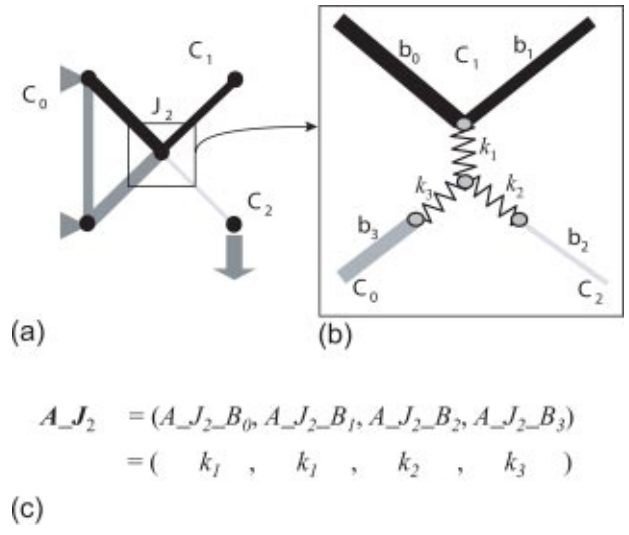

Fig. 4 A sample joint model and modified joint assignment method used in this paper to assign joint attributes: (*a*) a sample structure with three components and (*b*) FE model for the J_2 in (*a*), where each component is connected to the center node by using the torsional spring element (*c*) the modified individual joint attribute vector A_J_2 that assigned to J_2

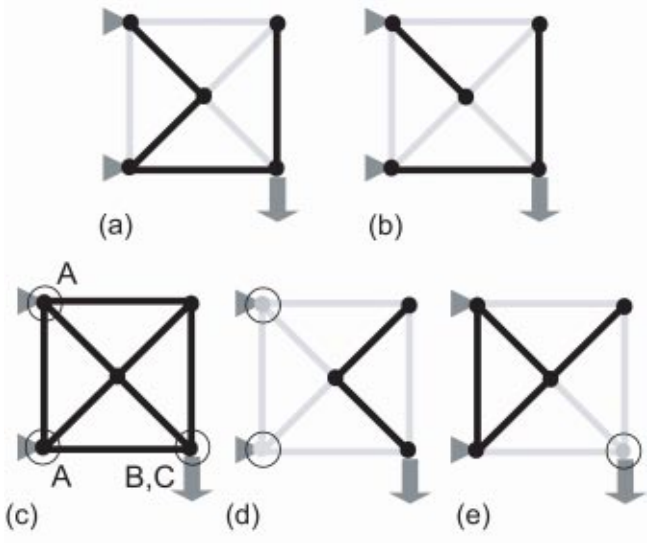

Fig. 5 Constraints for feasible topology: (*a*) feasible structure, (*b*) infeasible structure violating Connectivity Constraint 1, (*c*) points considered in Connectivity Constraint 2 (A: boundary condition, B: loading, and C: displacement), (*d*) infeasible topology violating Connectivity Constraint 2 (point A not connected), and (*e*) infeasible topology violating Connectivity Constraint 2 (point B–C not connected)

- at least one of the points at which boundary conditions are defined
- points at which loads are applied
- points at which displacements are measured for the evaluation of structural performance

Connectivity Constrains 1 can be formally written as

$$\text{IS\_CONNECTED}(G_g, \boldsymbol{t}) = \text{IS\_CONNECTED}(G_0) = \text{TRUE} \tag{10}$$

where IS_CONNECTED is a function that checks the connectivity of the original topology graph G_0, defined by the ground topology graph G_g and the topology vector ***t***. This function is implemented as the function that checks the number of subcomponents in the G_0. Only when this number is equal to 1, IS_CONNECTED function returns TRUE. Connectivity Constraint 2 can be written as

$$\left(\sum_{i=1}^{N_{BP}}\left(\sum_{j\in S_{BPi}} t_j\right)\right)\cdot\left(\prod_{i=1}^{N_{LP}}\sum_{j\in S_{LPi}} t_j\right)\cdot\left(\prod_{i=1}^{N_{DP}}\sum_{j\in S_{DPi}} t_j\right)\neq 0 \tag{11}$$

where N_{BP}, N_{LP}, and N_{DP} are the number of points at which boundary conditions are defined, loads are applied, and displacements are measured, respectively. S_{BPi}, S_{LPi}, and S_{DPi} are sets of the indices of basic members attached to the *i*th point at which boundary conditions are defined, loads are applied, and displacements are measured, respectively.

Note that not all of the points for boundary locations are required to be connected. Only the boundary points that can keep the structure statically determinate are required to be connected [36]. Because here we assumed that the boundary points are fixed in all 6 dof, the first term in Eq. (11) checks if there exist any connected boundary point in the structure while the other two terms check all the points for loading and displacement measuring are connected to the structure. Also note that when we measure the stiffness of the structure using the mean compliance, then the points **B** (points at which loads are applied) and points **C** (points at which displacements are measured) in Fig. 5 are same and Connectivity Constraints can be simplified.

3.3 Definition of Objective Functions. A multicomponent structure represented by a topology vector ***t***, a beam size vector ***w***,

the joint library vector ***JL*** for the components set, and finally a joint attribute vector ***A_J*** is evaluated according to the following four criteria: (i) stiffness of the structure, (ii) weight of the structure, (iii) manufacturability of each component in the structure, and (iv) numbers of joints (torsional springs) in the structure.

Stiffness of a structure can be measured as the negative of the compliance in the structure

$$f_{\text{stiffness}} = \text{stiffness} = -\text{COMPLIANCE}(G_g, \boldsymbol{t}, \boldsymbol{w}, \boldsymbol{JL}, \boldsymbol{A\_J}) \quad (12)$$

where COMPLIANCE is a function that returns the compliance in the multicomponents structure, using finite element analyses. This function is composed of three modules: (i) *FE model generator module* generating a FE input model composed of beam and torsional spring elements. Existences of beam elements are determined by topology vector ***t*** and their cross sectional properties are determined using the beam size vector ***w***. Torsional spring elements are placed at all possible joint locations and their spring rates are determined by the joint attribute vector ***A_J***. (ii) *FE analysis module* performing FE analysis. (iii) *Compliance evaluating module* collecting the displacements at the given loading points and evaluating compliance C using following equation:

$$C = \tfrac{1}{2}\{f\}^T\{u\} \quad (13)$$

where $\{f\}$ is the vector of the external loads and $\{u\}$ is the global vector of the nodal displacements calculated from the FE analysis.

Weight of a structure can be calculated as the inner product of topology vector ***t*** and the beam weight vector $\boldsymbol{w}_b$ of the weights of the basic members in the ground structure

$$f_{\text{weight}} = \boldsymbol{w}_b \cdot \boldsymbol{t} \quad (14)$$

where the vector $\boldsymbol{w}_b$ is defined as

$$\boldsymbol{w}_b = (w_{b\,0}, w_{b\,1}, \ldots, w_{b\,i}, \ldots, w_{b\,gnn-2}, w_{b\,gnn-1}) \quad (15)$$

Here, $gnn = |V_g|$ and $w_{b\,i}$ = weight of ith basic member defined by the beam size vector ***w***. For an example, if w_i (the ith element of beam size vector ***w***) defines the diameter of the cylindrical beam cross section and the density of the beam is ρ, then $w_{b\,i} = 2\pi w_i l_i \rho$. Note that l_i indicates the length of the ith beam element.

Estimating manufacturability will require specific approaches for each manufacturing process. In this paper, manufacturability of components (to be maximized) is evaluated considering the total cost of producing components in the structure (to be minimized) represented by a product topology graph. It is assumed the components are made from sheet metals working, whose cost is estimated as the cost of stamping and blanking dies. The die costs consist of die set cost and die machining cost, which are functions of die usable area Au and shearing perimeter P, respectively [29]. For each component, Au is approximated as the convex hull area of given component and P is calculated as the outer perimeter of the component (Fig. 6(*c*)). Hence, larger size of the component results in higher value of Au requiring larger die set with higher cost. Also, complex geometry of component increases the P value accompanied by higher die machining cost. The following equation is used to calculate entire manufacturability of a structure:

$$f_{\text{manufacturability}} = -\text{total manufacturing cost} = -\sum_{i=1}^{cn} \text{DIECOST}_S(i, SubG_i) \quad (16)$$

where cn is the number of the components (=number of sub graphs defined by G_0 and ***JL***) and $SubG_i$ is the ith subgraph defined by G_0 and ***JL***. Finally, DIECOST_S is the function that calculates the stamping die cost of ith component and is defined as follows:

$$\text{DIECOST}_S(i, SubGi) = \text{COST}_{Au}[Au(SubG_i)] + \text{COST}_P[P(SubG_i)] \quad (17)$$

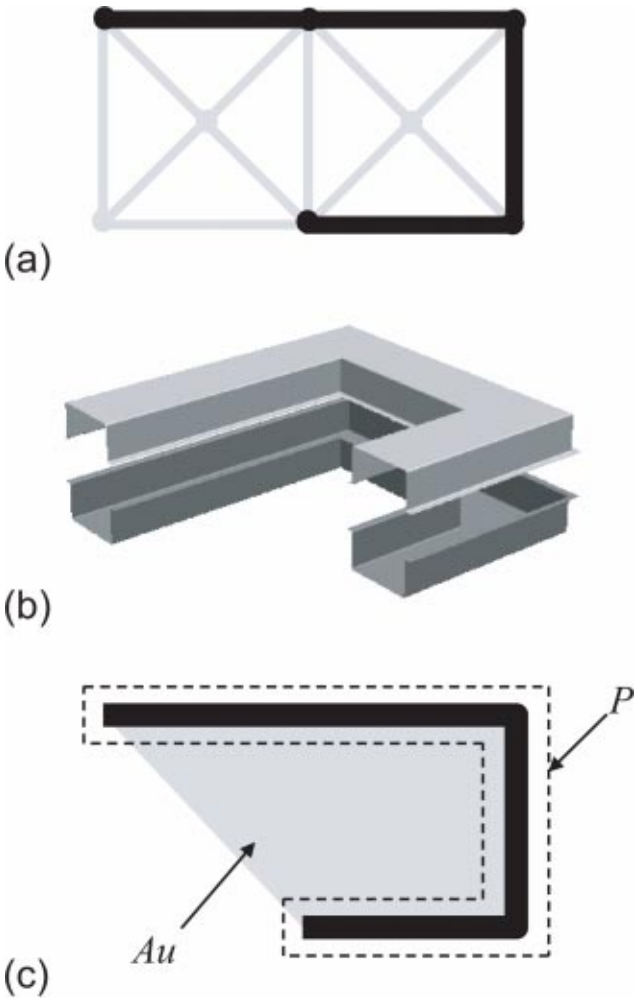

Fig. 6 Manufacturability calculation considering sheet metal working: (*a*) a beam component defined, (*b*) corresponding sheet metal components to be joined into the beam component defined in (*a*), and (*c*) die useable area *Au* calculated from the convex hull area and shearing parameter *P* for (*b*)

where $Au(SubG_i)$ and $P(SubG_i)$ calculate the die usable area Au and the shearing parameter P of the ith component defined by subgraph $SubG_i$. COST_{Au} and COST_{p} are the functions than convert Au and P into the cost. Qualitatively, maximizing $f_{\text{manufacturability}}$ would result in a structure consisting of components in smaller sizes and in simpler geometries.

Components at each joint are assumed to be joined with spot welds which are done one by one. Since the cost of spot welding for a structure is proportional to the number of weld spots in the structure and the number of weld spots in a joint is approximately proportional to the torsional stiffness of the joint; the welding cost is estimated by the sum of the rates of torsional springs [Nm/rad] in the finite element model of the structure

$$f_{\text{assembleability}} = -\text{total assembly cost} = -\sum_{i=1}^{gjn} \text{COST}_{RSW}[\text{SWELD}(i, G_0, \boldsymbol{JL}, \boldsymbol{A\_J})] = -\sum_{i=1}^{gjn} \text{COST}_{RSW}[\text{WSRATE} \times \text{SRATE}(i, G_0, \boldsymbol{JL}, \boldsymbol{A\_J})] \quad (18)$$

where G_0 is the original topology graph. ***JL*** is the joint library vector and ***A_J*** is the joint attribute vector. Function $SWELD(i, G_0, \boldsymbol{JL}, \boldsymbol{A\_J})$ returns the total number of welds at the ith possible joint location and $COST_{RSW}$ function calculates the cost of Resistance Spot Welding procedures by using the number of spot welds. Finally, gjn is the total number of possible joint locations.

In summary, the multiobjective optimization problem to be solved can be stated as follows:

$$\begin{aligned} &\text{Maximize} && \{f_{\text{stiffness}}, -f_{\text{weight}}, f_{\text{manufacturability}}, f_{\text{assembleability}}\} \\ &\text{Subject to} && \text{Connectivity Constraint 1 and 2 defined as} \\ & && \text{Eqs. (10) and (11)} \end{aligned} \tag{19}$$

3.4 Optimization Algorithm. Due to the multiobjective formulation (as opposed to, e.g., weighted sum of multiple objectives) and the complexity of the underlying graph partitioning problem [37], the above optimization problem is solved using a modified version of NSGA-II [4,5], whose basic steps [5] are outlined below:

1. Create a population ***P*** of n chromosomes (an encoded representation of design variables) and evaluate their values of objective functions.
2. Rank each chromosome ***c*** in ***P*** according to the number of other chromosomes dominating **c** in Pareto sense (rank 0 is Pareto optimal). Store the chromosomes with rank 0 into set ***O***. Also, create an empty subpopulation ***Q***.
3. Select two chromosomes $\boldsymbol{c}_i$ and $\boldsymbol{c}_j$ in ***P*** with probability proportional to $n\text{-}rank(\boldsymbol{c}_i)$ and $n\text{-}rank(\boldsymbol{c}_j)$.
4. Crossover $\boldsymbol{c}_i$ and $\boldsymbol{c}_j$ to generate two new chromosomes $\boldsymbol{c}_i'$ and $\boldsymbol{c}_j'$ with a certain high probability.
5. Mutate $\boldsymbol{c}_i'$ and $\boldsymbol{c}_j'$ with a certain low probability.
6. Evaluate the objective function values of $\boldsymbol{c}_i'$ and $\boldsymbol{c}_j'$ and store them ***Q***. If ***Q*** contains less than m new chromosomes, go to 3.
7. Let $\boldsymbol{P} \leftarrow \boldsymbol{P} \cup \boldsymbol{Q}$ and empty ***Q***, Rank each chromosome in ***P*** and remove m chromosomes with lowest ranks from ***P***.
8. Update set ***Q*** and increment the generation counter. If the generation counter has reached a pre-specified number, terminate the process and return ***O***. Otherwise go to 3.

A chromosome ***c*** (an encoded representation of design variables) used in this study is a simple list of the four design variables

$$\boldsymbol{c} = (\boldsymbol{t}, \boldsymbol{w}, \boldsymbol{JL}, \boldsymbol{A\_J}) \tag{20}$$

Crossover in step 4 combines ''genetic materials'' of two parent chromosomes to produce two offspring chromosomes. The role of crossover is to combine high-quality partial solutions (building blocks) in parent chromosomes to produce higher quality offspring [38]. Since information in ***t***, ***w***, ***JL***, and ***A_J*** are linked in a nonlinear fashion as defined in the ground topology graph, the conventional one point or multiple point crossover for linear chromosomes will not effectively preserve the building blocks. For this type of problem, graph-based crossover has been successfully applied for improved performance of GA [6–8], which is adapted to fit to our problem as described below:

1. Draw an arbitrary crossover line **L** on two parent structures **P1** and **P2**, and use the line to ''cut'' **P1** and **P2** into two substructures **S1** and **S2** (Fig. 7(*a*)).
2. Partition product topology graphs of **P1** and **P2** into two sub graphs **G1** and **G2** corresponding to **S1** and **S2** defined in the step 1 (Fig. 7(*b*)).
3. Assemble the graph of the offspring **C1** with **G1** of parent **P1** and **G2** of parent **P2**. Also, assemble the graph of offspring **C2** with **G2** of parent **P1** and **G1** of parent **P2**. During the assembly process, edges between two nodes came from different parents are randomly assigned (Fig. 7(*c*)).
4. Construct offspring structures **C1** and **C2** using the assembled graphs (Fig. 7(*d*)).

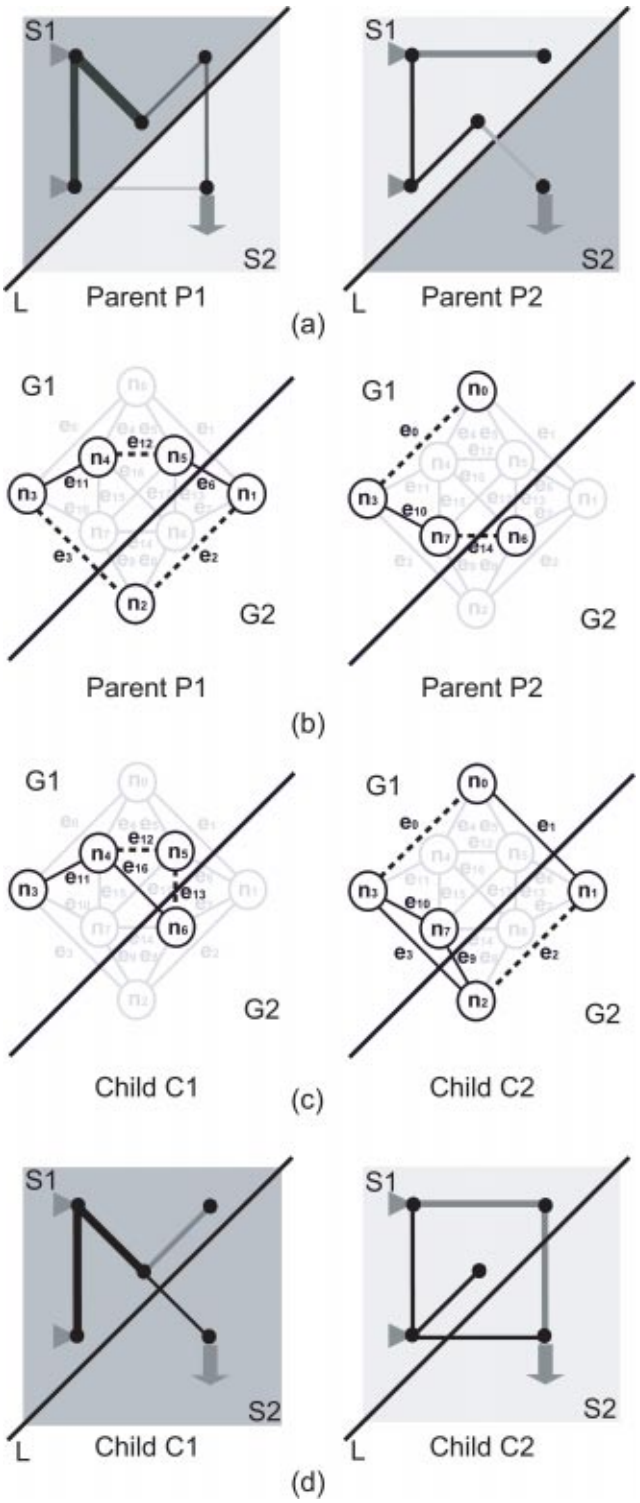

Fig. 7 Graph-based crossover operation: (*a*) parent structures P1 and P2 cut by crossover line L, (*b*) corresponding partitioning of P1 and P2 in graph representation, (*c*) assembly of offspring graphs C1 and C2. Note that in C1, edges e_{11} and e_{12} are copied from parent P1 because nodes n_3, n_4, and n_5 are from P1. Edges e_{16} and e_{13} are randomly assigned because n_6 is from P2 while n_4 and n_5 are from P1. (*d*) Offspring structures C1 and C2 constructed from their graphs. Both C1 and C2 have 2 components.

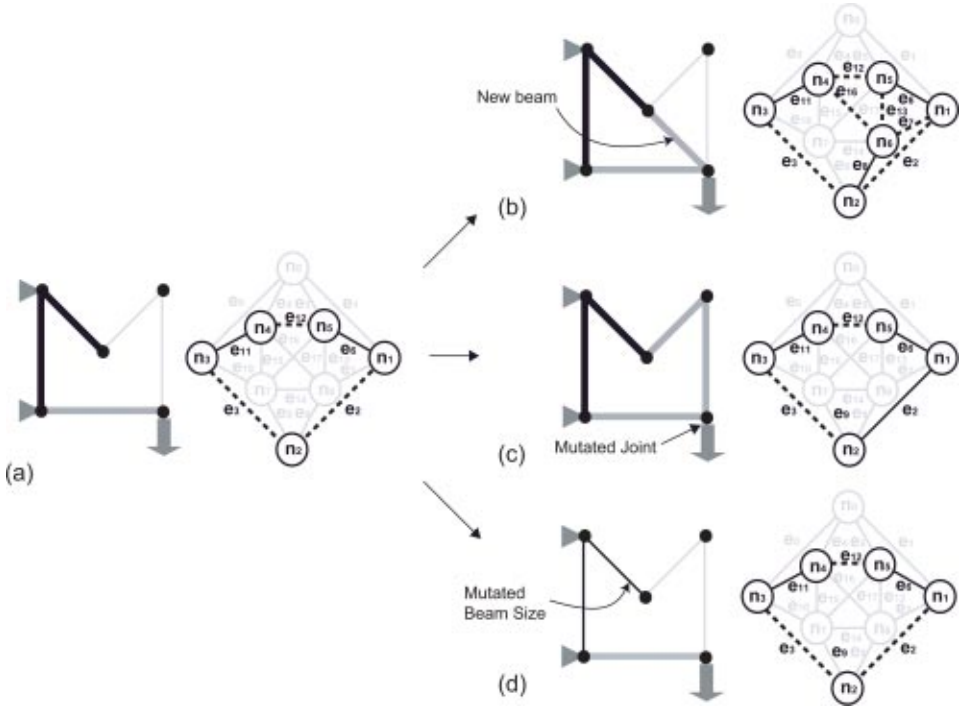

Fig. 8 Mutation operation (*a*) the original structure and graph (*b*) topology mutation (*c*) altering joint locations and (*d*) altering beam size

Crossover line **L** is selected in the geometrical space (where the physical structures belong) rather than in the topological space (where the product topology graphs belong) to realize the effective preservation of smaller high-quality substructures—building blocks for our problem. Even though both parent graphs are connected, the crossover may yield an offspring graphs **C** that are disconnected. In such cases, a repair operator is applied to reconstruct the connectivity, where Dijkstra's algorithm [39] is used to find the shortest path on the ground topology graph between the disconnected subgraphs of **C**, and the nodes and edges on the shorted path are added to **C**. This operator makes every individual in the current population to satisfy the Connectivity Constraints defined in Section 3.2 and accelerated the convergence of optimization.

Mutation modifies a structure in the following steps:

1. Mutate topology vector ***t*** by random bit flipping. This will add or remove basic members (Fig. 8(*b*)).
2. If the resulting structure is disconnected, apply the above repair operator to reconstruct connectivity.
3. Mutate decomposition of the graph by changing the joint library ***JL*** by random. This will alter the location of joints (Fig. 8(*c*)).
4. Mutate beam size vector ***w***. This will alter the cross sectional size of components (Fig. 9(*d*)).
5. Mutate joint attribute vector ***A_J***. This will alter the torsional spring rate of the corresponding joint.

In addition to the above-customized crossover and mutation operators, the implementation of multiobjective genetic algorithms used in the following examples utilizes linear fitness scaling, niching based on the distances in objective function space, and stochastic universal sampling [5]. Also, the population is initialized to contain only chromosomes that satisfy both Connectivity Constraints 1 and 2. Figure 7 shows the flowchart of multicomponent structure synthesis. Software implementation,

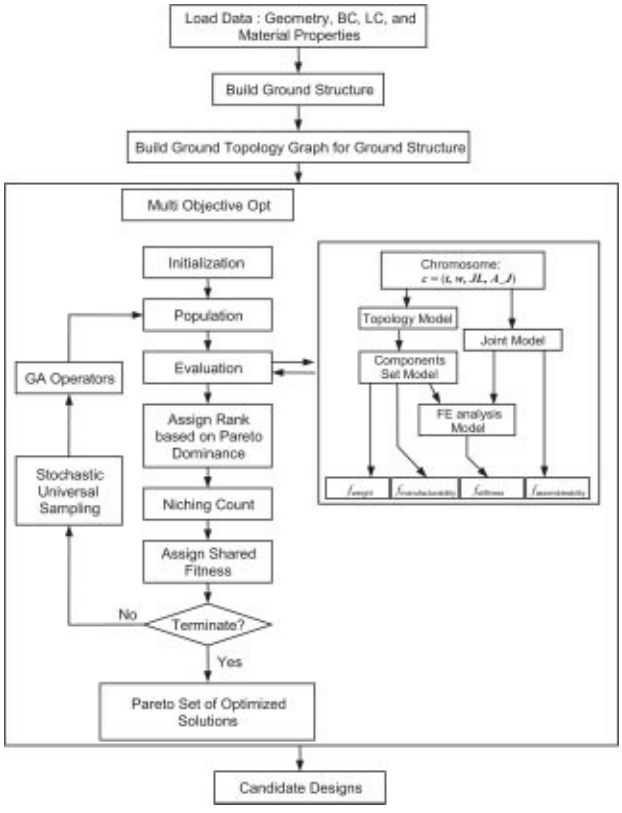

Fig. 9 Flowchart of multicomponent structure synthesis

Table 1 Parameter values for GAs used in the case studies

property	value	
	Case Study 1	Case Study 2
number of generation	100	100
number of population	2000	2000
replacement rate (m/n)	0.5	0.5
crossover probability	0.9	0.9
mutation probability for t	0.01	0.1
mutation probability for w	0.1	0.1
mutation probability for JL	0.1	0.1
mutation probability for A_J	0.1	0.1

Table 2 Material properties of sheet metal used in the case studies

Property	value
Thickness [mm]	2.0
Young's modulus [GPa]	200.0
Poisson's ratio	0.3
Density [kg/mm3]	$8.0*10^{-6}$

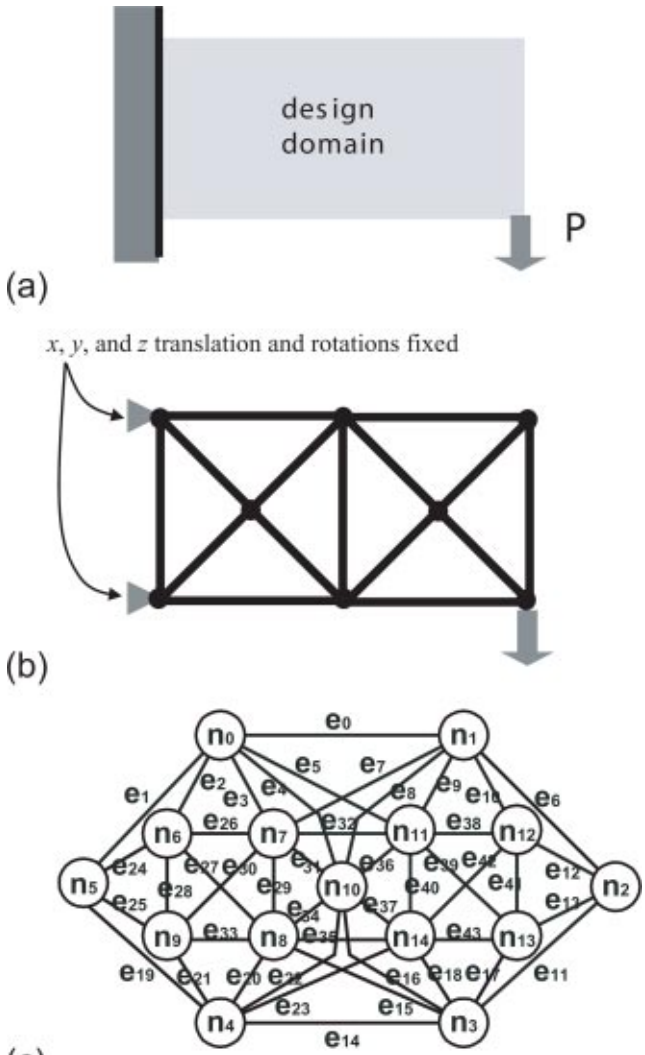

Fig. 10 Case Study 1 model: (*a*) design domain, (b) ground structure with 15 beam elements, and (*c*) ground topology graph of 15 nodes (n_0~n_{14}) and 44 edges (e_0~e_{43})

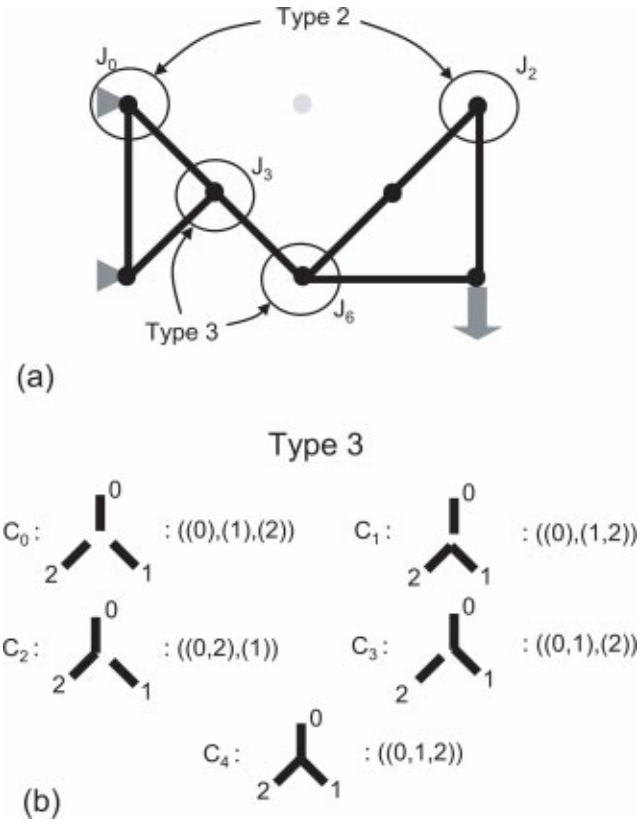

Fig. 11 Joint Library Type for a given topology: (*a*) Joint Location J_0 and J_2 have the same joint library Type 2, and (*b*) Joint Library of the Type 3 Joint. Total 5 cases (C_0~C_4) are in the Type 3 Joint Library.

including GAs code, is done in the C++ programming language. LEDA[1] library was used for graph algorithm and an in-house FEM code[2] is used to obtain $f_{\text{stiffness}}$.

4 Case Studies

Multi-component structure synthesis proposed in this paper is applied to two case studies: a cantilever structure and a simplified automotive floor frame. Table 1 shows the parameter values for GAs runs used in both case studies. Components used in the both case studies are considered to be made of thin-walled beams with rectangular cross sections where width and height are in same length. This length is considered as the elements of beam size vector w. Table 2 lists the material properties of the sheet metal (steel) used in the both case study.

[1]Developed by Algorithmic Solution (http://www.algorithmic-solutions.com).
[2]Developed by Karim Hamza.

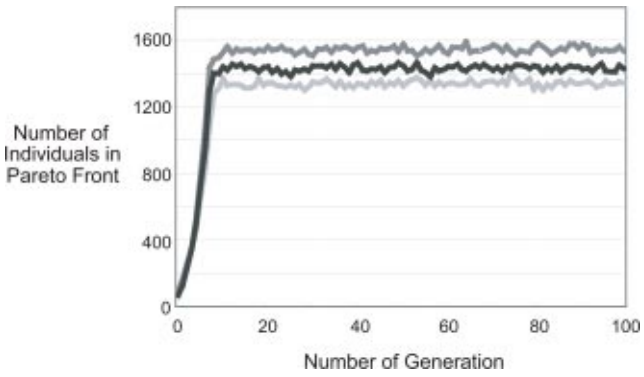

Fig. 12 Typical convergence histories of GA runs with three different mutation probabilities for *y* (black line: 0.005, dark gray line: 0.05, and light gray line: 0.1)

4.1 Case Study 1: Cantilever Structure. For the first case study, a cantilever structure is modeled as a design domain in Fig. 10(*a*), with length 200 mm and height 100 mm. The left side of the domain is fixed on the wall and a vertical load **P** (=100 [N]) is applied at the lower right corner of the domain. The displacement is measured at the loading point to calculate the stiffness of the structure and used to calculate the compliance. Figure 10(*b*) shows the ground structure with 15 nonoverlapping beam elements, each of which are regarded as a basic member. Figure 10(*c*) shows the ground topology graph of the ground structure in Fig. 10(*b*), containing 15 nodes and 44 edges.

In this case study, elements of beam size vector ***w*** are the cross-sectional height (=width) of each rectangular cross sectional beam components ranged from 5.0 to about 25.0 mm. Figure 11 illustrates an example of the joint library assignment for a given topology. The joint libraries used in this case study define the configuration of the possible joint locations. Joint libraries are classified by the number of the beams merging at the joint location:

$$\textbf{Type}\ \text{i: When total } i \text{ numbers of beams merge at the joint location} \tag{21}$$

As illustrated in the Fig. 11(*a*), types of the joint library at $\mathbf{J}_0$ and $\mathbf{J}_2$ are Type 2 even though the real structural shapes are not same with each other.

Considering the ground structure in Fig. 10(*b*), the most complex joint can have maximum five beams. Therefore, total of five joint type libraries (Type 1~Type 5) are built and used for this case study, considering all possible cases of each joint type only excluding the overlapping components configurations. The elements of the joint attribute vector ***A_J*** are the torsional spring rates in normal direction of the design domain plane ranged from $1.0\times10^4\sim10.0\times10^4$ Nmm/rad.

Figure 12 shows the typical convergence histories of GA runs with three different mutation probabilities for ***t***. All three plots indicate the increase in the size of Pareto set (number of Pareto optimal designs) as the number of generation increases. Note that as mutation probability decreases, the number of individuals in the Pareto Front converges closer to the total number of population (=2000). In these optimizations, the number of generation (=100)

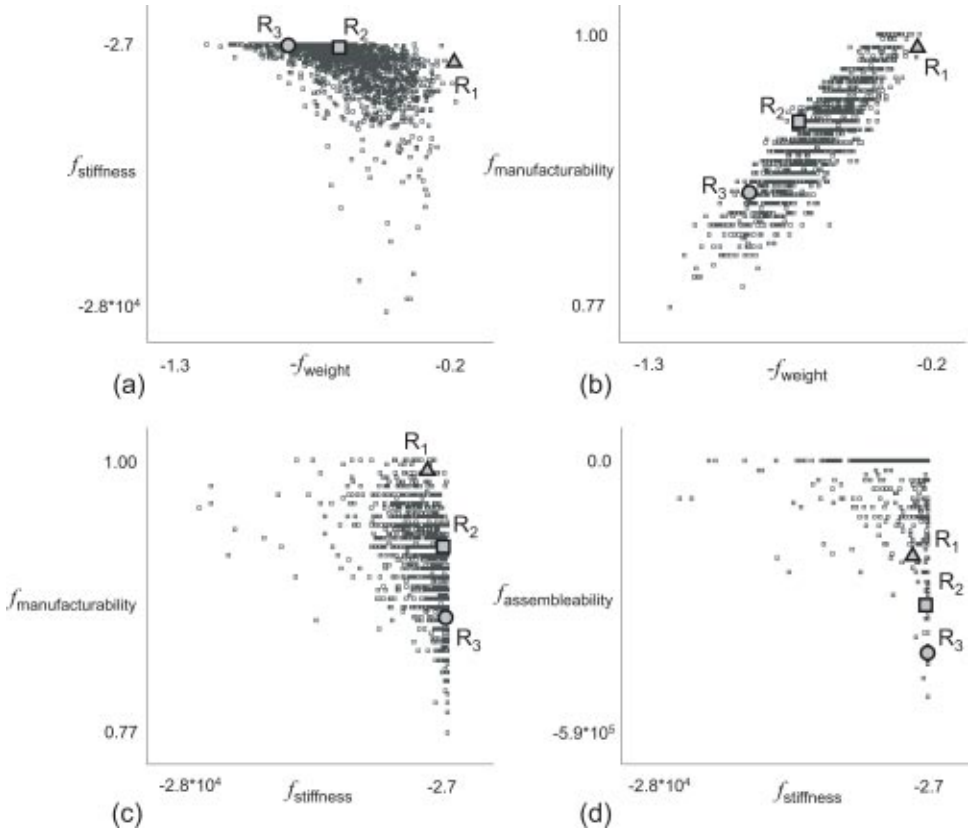

Fig. 13 Distribution of designs at generation=100 for Case Study 1. In all plots, the utopia points are at the upper-right corner. Black-marked ones are designs in the Pareto Set with respect to all four objectives. Three representative Pareto optimal designs R_1, R_2, and R_3 are shown in Fig. 14.

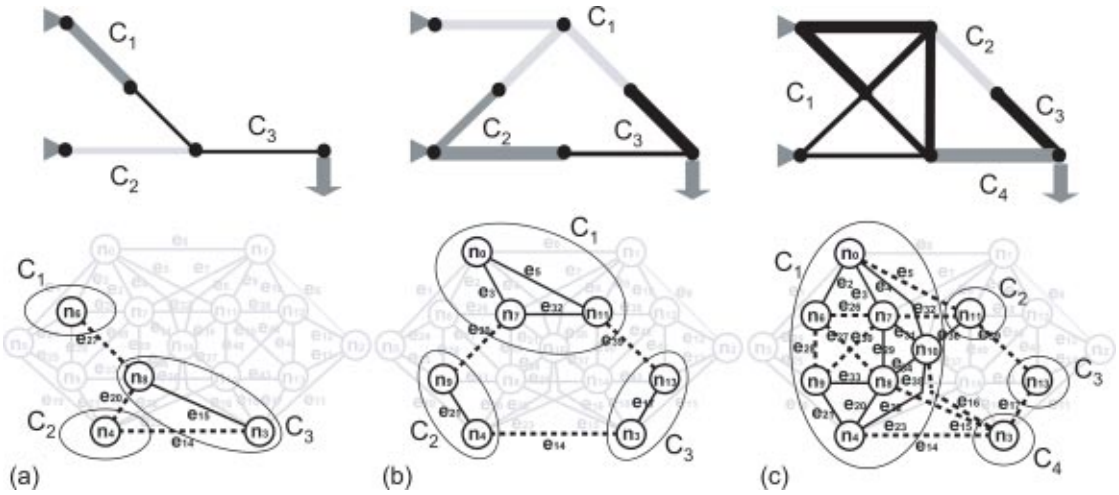

Fig. 14 Representative Pareto optimal designs for Case Study 1: (*a*) R_1, (*b*) R_2, and (*c*) R_3. R_1, R_2, and R_3 have 3, 3, and 4 components, respectively. Thickness of beams represents the size (width and height) of cross-sectional design.

Table 3 Objective function values for R_1, R_2, and R_3 in Case Study 1

	$f_{\text{stiffness}}$ [Nmm]	f_{weight} [Kg]	$f_{\text{manufacturability}}$ [$]	$f_{\text{assembleability}}$ [$]
R_1	-1841.3	**0.192**	**-16.6**	**-12.0**
R_2	-3.4	0.721	-26.0	-30.0
R_3	**-3.7**	0.941	-35.4	-42.0

was used as the termination condition. Using a PC with hyper-threaded Pentium 4, 3.07 GHz, one optimization run takes approximately 3.5 h.

Figure 13 illustrates objective function spaces obtained at the terminal generation (=100). Because there are four objective functions, $f_{\text{stiffness}}$, f_{weight}, $f_{\text{manufacturability}}$, and $f_{\text{assembleability}}$, the resulting 4D space is projected on to four 2D spaces as shown in Figs. 13(*a*)–13(*d*). Each 2D plot shows points for structural designs in the Pareto front with respect to the chosen two objectives only, ignoring the values of the remaining two objectives. In all plots, the utopia points are located at the upper right corner. The following observations can be made from these plots:

- Observation 1: In $f_{\text{weight}}-f_{\text{stiffness}}$ space (Fig. 13(*a*)), designs are concentrated on the upper-right portion.
- Possible explanation: Higher weight implies designs with more beams or thicker beams, which tends to increase stiffness.

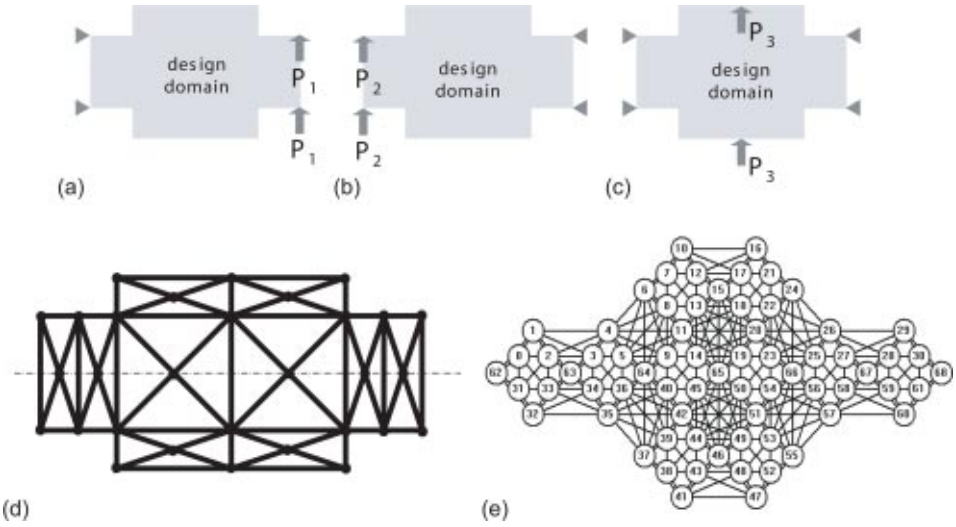

Fig. 15 Case study 2 model: (*a*)–(*c*) Design domain with three loading and boundary conditions, (*d*) ground structure with 69 beam elements, and (*e*) ground topology with 69 nodes (n_0~n_{68}) and 284 edges. Edge numbers (e_0~e_{283}) are not shown in (*e*) due to the space limitation. Due to the symmetric design assumption, only the basic members in the right half plane in (*d*) contain the design variables. Left half plane has the symmetric design of the right half-plane design.

- Observation 2: $f_{\text{weight}} - f_{\text{manufacturability}}$ space (Fig. 13(*b*)) shows a linear trend between the total weight and manufacturing cost.
- Possible explanation: Two elements that determine $f_{\text{manufacturability}}$ [die usable area (Au) and shearing perimeter (P)] are highly related to the f_{weight} because as a design's total weight increases by having more number of beams, at least one of Au or P will be increased. As a result, higher total weight implies lower manufacturability (less Au and P).
- Observation 3: Designs with lower stiffness show higher manufacturability (Fig. 13(*c*)).
- Possible explanation: Higher manufacturability implies smaller components (smaller Au), which would require more joints, which, in turn, tends to reduce stiffness.

Three representative Pareto optimal designs, annotated as $\mathbf{R}_1$, $\mathbf{R}_2$, and $\mathbf{R}_3$ in Fig. 13, are shown in Fig. 14, and their objective function values are listed in the Table 3. The geometry of each structure exhibits its unique characteristics allowing the following interpretations:

- Structure $\mathbf{R}_1$ (Fig. 14(*a*)) is a very light structure with three simple components connected by two joints. This structure shows good manufacturability by having small number of simple-shaped components.
- Structure $\mathbf{R}_2$ (Fig. 11(*b*)) shows balanced performances in the most objectives. The structure is fairly stiff thanks to the clever arrangement of beams (including a triangular internal structure) while most components are relatively small sized. This triangular shape seems to impose mostly axial loading in each beam, thereby avoiding bending of joints.
- Structure $\mathbf{R}_3$ (Fig. 11(*c*)) has four components and is the stiffest among the three structures. It contains the component $\mathbf{C}_1$, which is more complex and larger than the components of $\mathbf{R}_1$ and $\mathbf{R}_2$. By having numbers of internal triangular structures, this component seems to help in increasing the stiffness of the structure. However, due to the size and complexity of this component, entire manufacturability is relatively low.

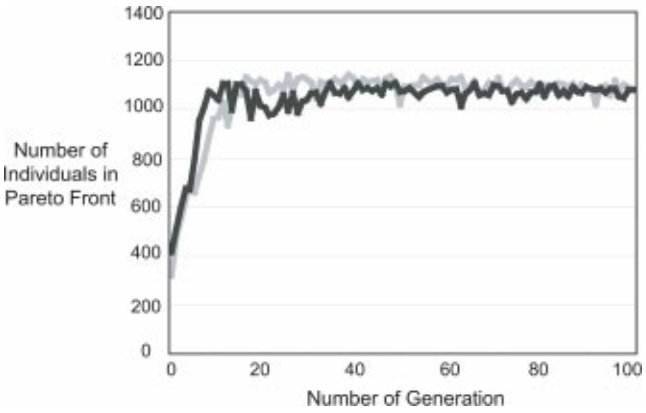

Fig. 16 Typical convergence histories of two GA runs

4.2 Case Study 2: Simplified Automotive Floor Frame Under Multiple Loadings. For the second case study, a simplified automotive floor frame under multiple loadings is modeled as a design domain in Figs. 15(*a*)–15(*c*), with length 3000 mm and width 1600 mm seen from the above. The structure is subject to the following three loading cases:

1. Front wheel locations are fixed on the ground, and a horizontal load $\mathbf{P}_1 (= 1000.0\ \text{N})$ is applied at each of the right-end points of the domain that represent rear wheel locations (Fig. 15(*a*)).

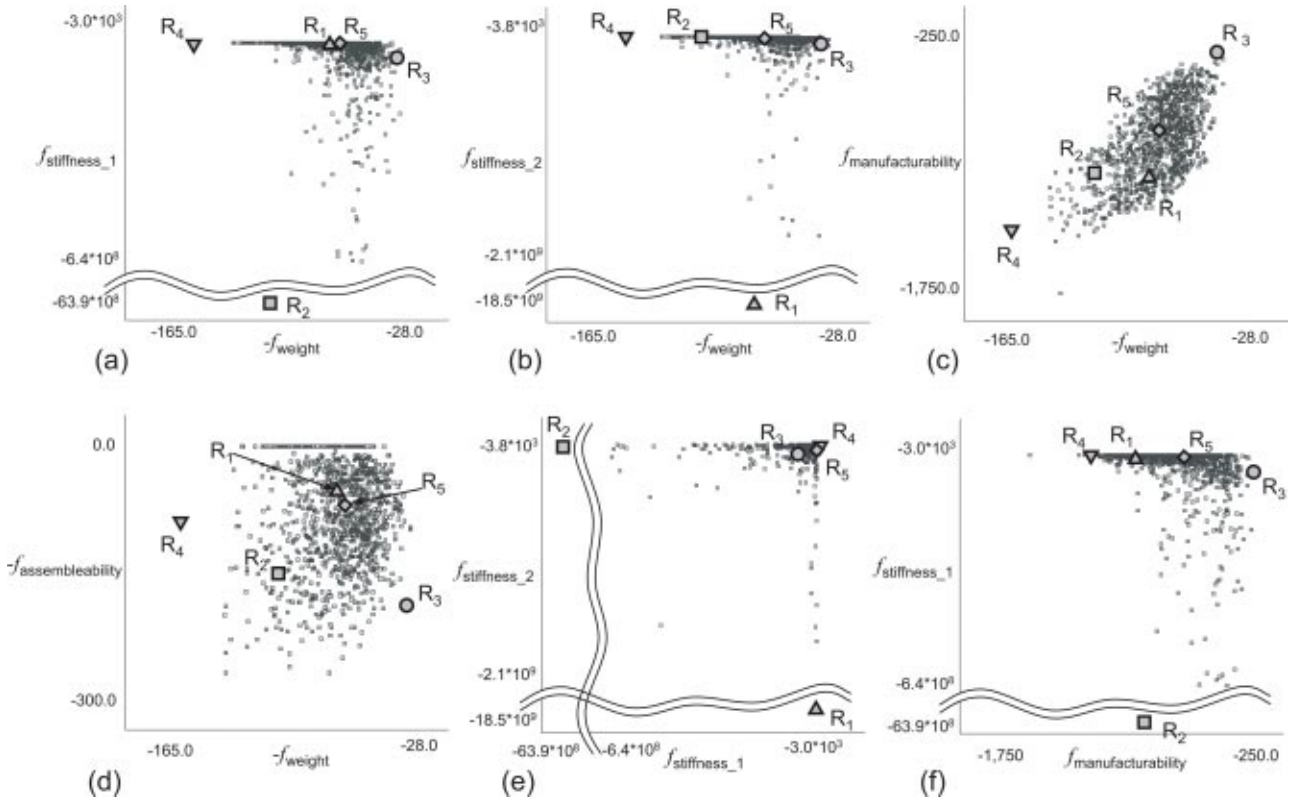

Fig. 17 Distribution of designs at generation=100 for Case Study 2. In all plots, the utopia points are at the upper-right corner. Black-marked ones are designs in the Pareto Set to all six objectives. Five representative Pareto optimal designs $\mathbf{R}_1$, $\mathbf{R}_2$, $\mathbf{R}_3$, $\mathbf{R}_4$, and $\mathbf{R}_5$ are shown in Fig. 18.

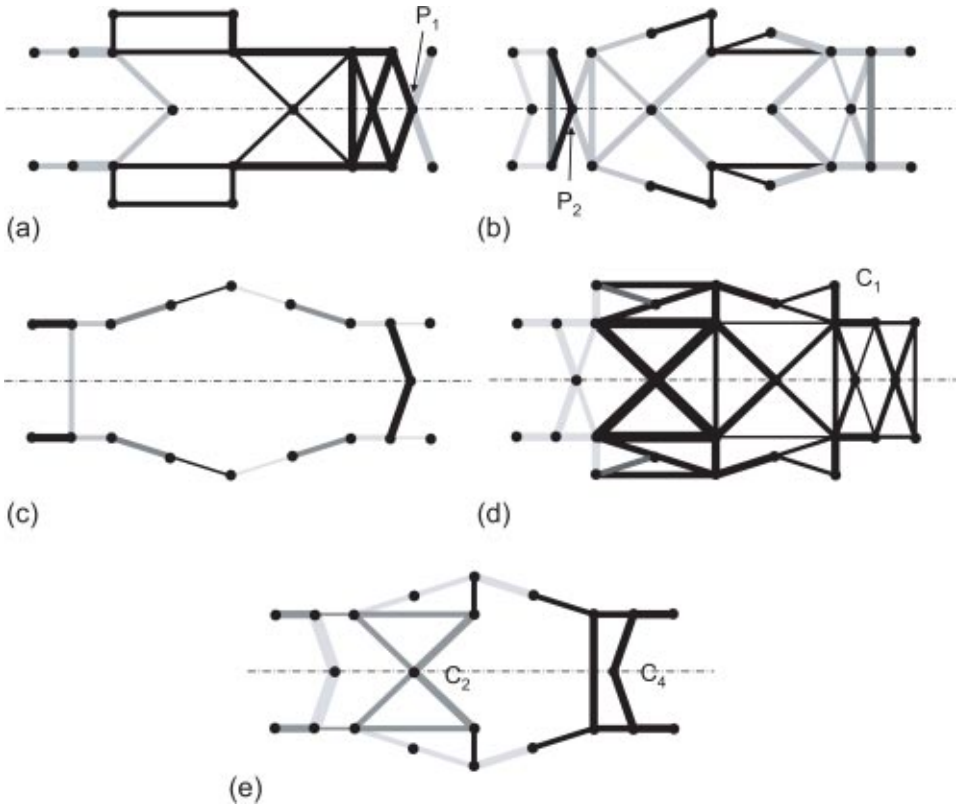

Fig. 18 Representative Pareto optimal designs: (*a*) R_1 (3 components), (*b*) R_2 (8 components), (*c*) R_3 (14 components), (*d*) R_4 (4 components), and (*e*) R_5 (7 components). Thickness of components represents the size (width and height) of cross-sectional design.

2. Rear wheel locations are fixed on the ground, and a horizontal load $\mathbf{P}_2(=1000.0\ \mathrm{N})$ is applied at each of the left-end points of the domain that represent front wheel locations (Fig. 15(*b*)).
3. Both of the front and rear wheel locations are fixed, and a horizontal load $\mathbf{P}_3(=1000.0\ \mathrm{N})$ is applied at the middle of the domain (Fig. 15(*c*)).

These loading and boundary conditions are simulating various lateral loading conditions an automotive floor frame can encounter. The result of final topology can be interpreted as the reenforced parts in the automotive floors. Compliances are measured at the loading points to calculate the stiffness of the structure. Since there are three values of displacements corresponding to the three loading cases, the number of objective functions becomes 6 (3 stiffness, 1 weight, 1 manufacturability, and 1 assembleability function). Figure 15(*d*) illustrates the ground structure with 69 nonoverlapping beam elements, each of which are regarded as a basic member. Due to the symmetric design nature of the automotive floor frame, only right half basic members of the floor frame will be considered as design variables and the left side of the floor frame will have the mirror image of the right side. Figure 15(*e*) shows the ground topology graph of the ground structure in Fig. 15(*d*), containing 69 nodes and 284 edges.

As in the first case study, elements of beam size vector $\boldsymbol{w}$ are the cross-sectional height (=width) of each beam components of the right side ranged from 25.0 to about 125.0 mm. Considering the ground structure in Fig. 15(*d*), the most complex joint can have a maximum of eight beams. Therefore, a total of eight joint-type libraries (Type 1~Type 8) are built and used for this case study. Note that first 5 libraries (Type 1~Type 5) are the same libraries used in the first case study. The elements of $\boldsymbol{A\_J}_i$ (Eq. (8)) are the torsional spring rates in normal direction of the design domain plane ranged from $1.0\times10^4\sim10.0\times10^4$ Nmm/rad.

Figure 16 shows the typical convergence histories of two optimization runs of this case study. As in the first case study, this plot indicates the increase in the size of Pareto set (number of Pareto optimal designs) as the number of generation increases. In these optimizations, the number of generation (100) was used as the termination condition. Using a PC with hyperthreaded Pentium 4, 3.07 GHz, one optimization run takes approximately 30 h 45 min. Compared to the first case study, the increased computational costs mainly came from (i) 3 FEM analyses to calculate $f_{\mathrm{stiffness\_1}}$, $f_{\mathrm{stiffness\_2}}$, $f_{\mathrm{stiffness\_3}}$ for each evaluation and (ii) the increased design domain size. For more complex structural problem, parallel processing computation for evaluating each individual design can be incorporated with the GA runs.

Six selected objective function spaces obtained at the terminal generation (=100) are illustrated in Fig. 17. Because there are six objective functions ($f_{\mathrm{stiffness\_1}}$, $f_{\mathrm{stiffness\_2}}$, $f_{\mathrm{stiffness\_3}}$, f_{weight}, $f_{\mathrm{manufacturability}}$, and $f_{\mathrm{assembleability}}$), the resulting 6D space is projected onto 2D spaces as shown in Figs. 17(*a*)–17(*f*) as in the first case study. The following observations can be made from these plots.

- Observation 1: As in the Case Study 1, optimal designs are concentrated on the upper-right portion in the $f_{\mathrm{weight}}-f_{\mathrm{stiffness\_1}}$ and $f_{\mathrm{weight}}-f_{\mathrm{stiffness\_2}}$ spaces (Figs. 17(*a*) and 17(*b*)).
- Possible explanation: Higher weight implies designs with more beams or thicker beams, which tends to increase stiffness.
- Observation 2: $f_{\mathrm{weight}}-f_{\mathrm{manufacturability}}$ space (Fig. 17(*c*)) shows a proportional trend between the weight and total manufacturing cost.
- Possible explanation: As in the Case Study 1, die usable area Au and shearing perimeter P determining f_{manufac} are highly related to the f_{weight} because as a design's total weight in-

Table 4 Objective function values for $R_1 \sim R_5$ in the Case Study 2

	$f_{\text{stiffness\_1}}$ [Nmm]	$f_{\text{stiffness\_2}}$ [Nmm]	$f_{\text{stiffness\_3}}$ [Nmm]	f_{weight} [Kg]	$f_{\text{manufacturability}}$ [$]	$f_{\text{assembleability}}$ [$]
R_1	$-2.153*10^6$	$-1.858*10^{10}$	$-3.271*10^3$	74.73	-1,027.1	**-46.0**
R_2	$-6.394*10^9$	$-9.190*10^6$	$-2.309*10^5$	110.25	-967.3	-156.0
R_3	$-4.168*10^7$	$-5.520*10^7$	$-4.557*10^5$	**32.90**	**-249.4**	-190.0
R_4	$\mathbf{-2.953*10^3}$	$\mathbf{-2.127*10^5}$	$\mathbf{-1.886*10^2}$	161.54	-1,363.3	-90.0
R_5	$-1.055*10^6$	$-1.946*10^7$	$-1.006*10^5$	71.71	-731.6	-70.0

creases by having more number of beams, at least one of Au or P will be increased. As a result, higher total weight implies lower manufacturability (less Au and P).

- Observation 3: Designs with lower stiffness values (larger compliance values) show relatively higher manufacturability (Fig. 17(*f*)).
- Possible explanation: Higher manufacturability implies smaller (smaller Au) and simpler (smaller P) components, which would require more joints, which, in turn, tends to reduce stiffness.

Five representative Pareto optimal designs, annotated as $\mathbf{R}_1$, $\mathbf{R}_2$, $\mathbf{R}_3$, $\mathbf{R}_4$, and $\mathbf{R}_5$ in Fig. 17, are shown in Fig. 18, and their objective function values are listed in the Table 4. Objective function values in Table 4 are plotted on a spider diagram in Fig. 19. The geometry of each structure exhibits its unique characteristics allowing the following interpretations:

- Structure $\mathbf{R}_1$ (Fig. 18(*a*)) is a structure with three components. By having a pivot design near the tail (Point $\mathbf{P}_1$), it shows worst stiffness in the second loading case (smallest $f_{\text{stiffness\_2}}$) while the other two stiffness values are relatively good compared with the other four designs (Fig. 19).
- Structure $\mathbf{R}_2$ (Fig. 18(*b*)) is a structure with eight components. As in $\mathbf{R}_1$, this design also contains a pivot design near the front of structure (Point $\mathbf{P}_2$) resulting a significant deflection (larger compliance value) in the first loading condition.
- Structure $\mathbf{R}_3$ (Fig. 18(*c*)) is a light and simple structure with 14 straight beam components. This structure has one of the simplest topology that satisfies the given topology constraints defined in Section 3.2. The structure shows the best manufacturability by having all straight components (Au value is minimized) except one, and also this is the lightest design compared with the other four selected designs (Fig. 19). However, it shows relatively poor stiffness characteristics by having 14 joining location (worst $f_{\text{assembleability}}$).
- Structure $\mathbf{R}_4$ (Fig. 18(*d*)) is the heaviest structure among these five structures and contains one big component ($\mathbf{C}_1$) that are more complex and larger than the ones in the other designs. Due to this complex component, manufacturability of this structure is worst among the five designs. However, this component seems to increase the stiffness of the structure in all three loading cases, resulting the best stiffness values (smallest compliance values).
- Structure $\mathbf{R}_5$ (Fig. 18(*e*)) contains relatively simple seven components. Only two components ($\mathbf{C}_5$ and $\mathbf{C}_5$) are relatively complex, and these two components seem to increase the stiffness of the structure with relatively low total weight. In Fig. 19, Structure $\mathbf{R}_5$ shows a well-balanced performance on every objective function.

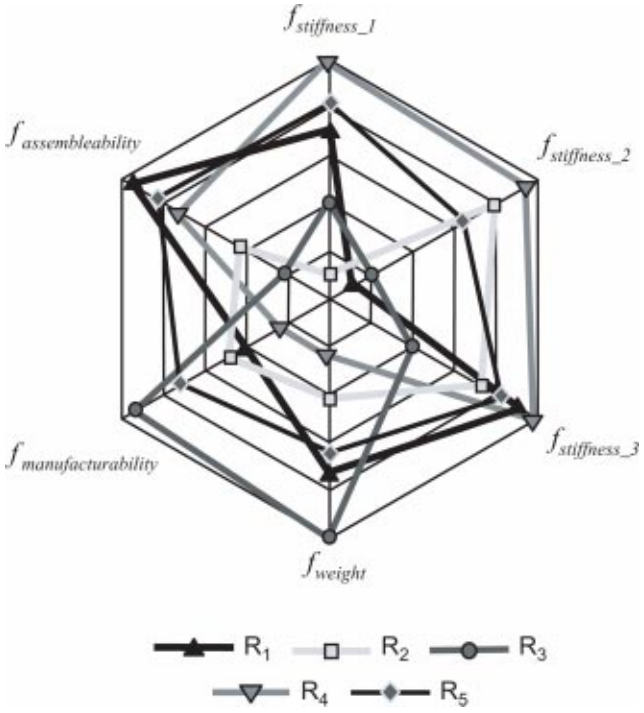

Fig. 19 A spider diagram for the objective function values of the representative Pareto optimal designs ($R_1 \sim R_5$) in Case Study 2. Note that R_5 shows a balanced performance in all six objective functions.

5 Summary and Future Works

This paper described a method for synthesizing multicomponent structural assemblies, where the topology and decomposition of a structure is simultaneously optimized over a ground structure for stiffness, weight, component manufacturability, and assembleability. A multiobjective genetic algorithm, coupled with finite element analyses, was employed to efficiently obtain Pareto optimal designs for the four objectives. Two simple case studies were presented to demonstrate the effectiveness of the proposed method.

While the obtained results are inspiring, we believe it would be possible to extend the present approach to continuum-based topology optimization by extending the framework of, for example, the Homogenization Design Method [19]. The developments in these directions are currently in progress and will be reported at other future opportunities.

Acknowledgments

The authors acknowledge funding provided by Toyota Motor Corporation and National Science Foundation under CAREER Award (DMI-9984606) for this research. We thank Karim Hamza at Discrete Design Optimization Laboratory at the University of Michigan for providing his FEM code. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

[1] Yetis, A., and Saitou, K., 2002, "Decomposition-Based Assembly Synthesis Based on Structural Considerations," ASME J. Mech. Des., **124**, pp. 593–601.
[2] Cetin, O. L., and Saitou, K., 2001, "Decomposition-Based Assembly Synthe-

sis for Maximum Structural Strength and Modularity,'' *Proc. 2001 ASME Design Engineering Technical Conferences*, Pittsburgh, Pennsylvania, September 9–12, DETC2001/DAC-21121. An extended version accepted to ASME Journal of Mechanical Design.
[3] Lyu, N., and Saitou, K., 2003, ''Decomposition-Based Assembly Synthesis for Structural Stiffness,'' ASME J. Mech. Des., **125**, pp. 452–463.
[4] Lotter, B., 1989, *Manufacturing Assembly Handbook*, Butterworths, London.
[5] Deb., K., Agrawal, S., Pratab, A., and Meyarivan, T., 2000, ''A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II,'' KanGAL Report 200001, Indian Institute of Technology, Kanpur, India.
[6] Coello, C. A., Veldhuizen, D. A., and Lamont, G. B., 2002, *Evolutionary Algorithms for Solving Multi-Objective Problems*, Kluwer Academic/Plenum Publishers, New York.
[7] Saab, Y., and Rao, V., 1990, ''Stochastic Evolution: A Fast Effective Heuristic for Some Genetic Layout Problems,'' *27th ACM/IEEE Design Automation Conference*, ACM/IEEE, New York, pp. 26–31.
[8] Laszewski, G., 1991, ''Intelligent Structural Operators for the K-Way Graph Partitioning Problem,'' *Fourth International Conference on Genetic Algorithms*, Morgan Kaufmann, San Francisco, pp. 45–52.
[9] Pereira, F., Machado, P., Costa, E., and Cardoso, A., 1999, ''Graph Based Crossover—A Case Study With the Busy Beaver Problem,'' *Proc. 1999 Genetic and Evolutionary Computation Conference*, Morgan Kaufmann, San Francisco, pp. 1149–1155.
[10] Howell, L.L., 2001, *Compliant Mechanisms*, John Wiley & Sons, New York, p. 312.
[11] Dorn, W. C., Gomory, R. E., and Greenberg, H. J., 1964, ''Automatic Design of Optimal Structures,'' J. Mec., **3**, pp. 25–52.
[12] Rozvany, G. I., 1976, *Optimal Design of Flexural Systems*, Pergamon, London.
[13] Rozvany, G. I., 1992, *Shape and Lay-Out Optimization in Structural Design*, CISM Lecture Notes No. 325, Springer, Vienna.
[14] Fleury, C., 1993, *Discrete Valued Optimal Design Problems*, Bendsøe, M. P., and Mota Soares, C. A. (eds.), *Topology Optimization of Structure*, pp. 81–96, Kluwer, Dordrecht.
[15] Hajela, P., Lee, E., and Lin, C. Y., 1993, *Genetic Algorithms in Structural Topology Optimization*, Bendsøe, M. P., and Mota Soares, C. A. (eds.), *Topology Optimization of Structure*, pp. 117–134, Kluwer, Dordrecht.
[16] Beckers, M., 1999, ''Topology Optimization Using a Dual Method With Discrete Variables,'' Struct. Optim., **17**, pp. 14–24.
[17] Kirsch, U., 1989, ''Optimal Topologies of Structures,'' Appl. Mater. Res., **42**, pp. 223–249.
[18] Bendsøe, M. P., 1994, ''Optimization Methods for Truss Geometry and Topology Design,'' Struct. Optim., **7**, pp. 141–159.
[19] Bendsøe, M. P., and Kikuchi, N., 1988, ''Generating Optimal Topologies in Structural Design Using a Homogenization Method,'' Comput. Methods Appl. Mech. Eng., **71**, pp. 197–224.
[20] Diaz, A. R., and Bendsøe, M. P., 1992, ''Shape Optimization of Structures for Multiple Loading Conditions Using a Homogenization Method,'' Struct. Optim., **4**, pp. 17–22.
[21] Ananthasuresh, G. K., Kota, S., and Kikuchi, N., 1994, ''Strategies for Systematic Synthesis of Compliant MEMS,'' DSC-Vol. 55-2, *1994 ASME Winter Annual Meeting*, pp. 677–686.
[22] Jiang, T., and Papalambros, P. Y., 1996, ''Optimal Structural Topology Design Using Homogenization Method With Multiple Constraints,'' Eng. Optimiz., **27**, pp. 87–108.
[23] Rodrigues, H., Guedes, J. M., and Bendsøe, M. P., 2002, ''Hierarchical Optimization of Material and Structure,'' Struct. Multidisciplinary Optim., **24**, pp. 1–10.
[24] Chickermane, H., and Gea, H. C., 1997, ''Design of Multi-Component Structural System for Optimal Layout Topology and Joint Locations,'' Eng. Comput., **13**, pp. 235–243.
[25] Johanson, R., Kikuchi, N., and Papalambros, P., 1994, ''Simultaneous Topology and Material Microstructure Design,'' *Advances in Structural Optimization*, B. H. V. Topping and M. Papadrakakis (ed.), Civil-Comp Ltd., Edinburgh, Scotland, pp. 143–149.
[26] Jiang, T., and Chirehdast, M., 1997, ''A Systems Approach to Structural Topology Optimization: Designing Optimal Connections,'' J. Mech. Des., **119**, pp. 40–47.
[27] Li, Q., Steven, G. P., and Xie, Y. M., 2001, ''Evolutionary Structural Optimization for Connection Topology Design of Multi-Component Systems,'' Eng. Comput., **18**(3/4), pp. 460–479.
[28] Boothroyd, G., and Dewhurst, P., 1983, *Design for Assembly Handbook*, University of Massachusetts, Amherst.
[29] Boothroyd, G., Dewhurst, P., and Knight, W., 1994, *Product Design for Manufacturing and Assembly*, Marcel Dekker, New York.
[30] Gupta, S. K., Regli, W. C., and Nau, D. S., 1994, ''Integrating DFM With CAD Through Design Critiquing,'' Concurrent Eng. Res. Appl., **2**, pp. 85–95.
[31] De Fanzio, T., and Whitney, D., 1987, ''Simplified Generation of All Mechanical Assembly Sequences,'' IEEE Trans. Rob. Autom., **3**(6), pp. 640–658.
[32] Ko, H., and Lee, K., 1987, ''Automatic Assembling Procedure Generation From Mating Conditions,'' Comput.-Aided Des., **19**, pp. 3–10.
[33] Wang, C. H., and Bourne, D., 1997, ''Concurrent Decomposition for Sheet-Metal Product,'' *Proc. of 1997 ASME Design Technical Conference*, Sacramento, CA, September, 1997.
[34] Yetis, A., and Saitou, K., 2000, ''Decomposition-Based Assembly Synthesis Based on Structural Considerations,'' ASME J. Mech. Des., **124**, pp. 593–601.
[35] Cetin, O., and Saitou, K., 2003, ''Decomposition-Based Assembly Synthesis of Multiple Structures for Minimum Production Cost,'' *Proc. 2003 ASME International Mechanical Engineering Congress and R&D Exposition*, Washington, D.C., Nov 15–21, IMECE2003-43085.
[36] Bendsøe, M. P., and Sigmund, O., 2003, *Topology Optimization: Theory, Methods and Applications*, Springer, Berlin.
[37] Garey, M. R., and Johnson, D. S., 1979, *Computers and Intractability, a Guide to the Theory of NP-Completeness*, Freeman, New York.
[38] Goldberg, D., 1989, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Massachusetts.
[39] Dijkstra, E. W., 1959, ''A Note on Two Problems in Connection With Graph,'' Numer. Math., **1**, pp. 269–271.