Three-Dimensional Assembly Synthesis for Robust Dimensional Integrity Based on Screw Theory

Byungwoo Lee
Mechanical Engineer
Product Realization Lab,
GE Global Research,
Niskayuna, NY 12309
e-mail: leeb@research.ge.com

Kazuhiro Saitou[1]
Associate Professor
Department of Mechanical Engineering,
University of Michigan,
Ann Arbor, MI 48109-2125
e-mail: kazu@umich.edu

This paper presents a three-dimensional (3D) extension of our previous work on the synthesis of assemblies whose dimensional integrity is insensitive to the dimensional variations of individual parts. Assuming that assemblies can be built in the reverse sequence of decomposition, the method recursively decomposes a given product geometry into two subassemblies until parts become manufacturable. At each recursion, joints are assigned to the interfaces between two subassemblies to ensure the two criteria for robust dimensional integrity, in-process dimensional adjustability, and proper part constraints. Screw theory is utilized as a unified 3D representation of the two criteria. A case study on an automotive space frame is presented to demonstrate the method.
[DOI: 10.1115/1.1904048]

1 Introduction

Structural enclosures of modern mechanical products, such as ship hulls, airplanes, and automotive bodies, typically are made of hundreds or thousands of parts due to their geometric complexity and sizes. As the number of parts increases, however, achieving the dimensional integrity of the final assembly becomes more difficult due to the inherent variations in manufacturing and assembly operations.

A solution is to adjust critical dimensions during assembly when parts are located and fully constrained in fixtures. This in-process dimensional adjustment is typically facilitated by slip planes, mating surfaces at joints that allow a small amount of relative motions. For example, Fig. 1 shows two designs of a rectangular box. In contrast to design in (*a*) with no in-process adjustability of the critical dimensions (length between sections 1 and 3), design in (*b*) provides slip planes such that relative location of parts can be adjusted along the critical dimension.

The dimensional integrity of an assembly is also affected by the postassembly distortion due to the internal stress induced by joining parts with dimensional mismatches. A solution is to ensure parts to be properly constrained at each assembly step. For example, part 1 in Fig. 2(*a*) is not properly constrained and therefore the postassembly distortion might occur, if the length of sections 2 and 4 are slightly different due to manufacturing variation. With two slip planes perpendicular to each other, the design in (*b*) can absorb manufacturing variations within section 1 and 2-3-4 provided that variations in angles are negligible.

In addition to the decomposition of product geometry and the assignment of joint types at part interfaces, the assembly sequence also influences the in-process dimensional adjustability and proper part constraints. In the assembly sequence in Fig. 3(*a*), the critical dimension (total length) is not adjustable since there is no slip plane when the total length is determined with the addition of part 1. On the other hand, the sequence shown in (*b*) provides the slip plane at the assembly step where the critical dimension is achieved, absorbing a variation in length. As another example, the sequence in Fig. 4(*b*), where each critical dimension is independently adjusted at each step, is more desirable than the sequence in (*a*), where both dimensions are adjusted by one step, inevitably requiring a compromise between two critical dimensions.

Figure 5 illustrates an effect of the assembly sequence on proper part constraints. The sequence in (*a*) causes improper constraint of part 1 at the second step, whereas all parts are properly constrained at all steps in the sequence in (*b*).

As illustrated so far, the in-process adjustability and proper part constraints are effective tools for achieving high dimensional integrity of an assembly without requiring tight part tolerances [3]. The use of these tools in complex assemblies can be a very tedious task due to the coupling between the product decomposition, joint assignments, and assembly sequences. As a remedy, we have previously designed a correct and complete algorithm to fully enumerate feasible solutions for any two-dimensional (2D) enclosure geometry [1]. Assuming that assemblies can be built in the reverse sequence of decomposition, the algorithm recursively decomposes a given product geometry into two subassemblies until parts become manufacturable. At each recursion, joints are assigned to the interfaces between two subassemblies to ensure in-process dimensional adjustability and proper part constraints.

This paper presents a three-dimensional (3D) extension of the algorithm, where the screw theory [4] is utilized as a unified 3D representation of in-process adjustability and proper part constraints. Dissimilar to our previous work that assumes joints with arbitrary mating angles, they are selected from a library of feasible joints specific to the application domain. A case study on an automotive space frame is presented to demonstrate the method.

2 Related Works

Previous works related to assembly synthesis in general are reviewed in Ref. [1]. Due to the space limit, this section focuses on the works relevant to the 3D extension of the method.

The advantages of properly constrained assemblies are well known to practitioners in precision machinery design and several methods have been proposed in literatures including: kinematic design [5], minimum constraint design [6], and exact constraint design [3,7]. These works describe disadvantages of overconstraint through examples and provide good practices as well as analytical methods to compute constraints. In these works, the most commonly cited merit of properly constrained design is repeatability which leads to high precision. Recently, there has been a trial to analyze and classify key features that enables properly constrained design [8].

A universal analytical method for motion and constraint analysis dates back to the screw theory, a pioneering work by Ball [4]. Since then, the screw theory has been applied to areas of mechanism, robotics, and machine design. Among others, Waldron [9]

[1]Author to whom correspondence should be addressed.

Contributed by the Mechanisms and Robotics Committee of ASME for publication in the JOURNAL OF MECHANICAL DESIGN. Manuscript received May 10, 2004; final manuscript received October 24, 2004. Assoc. Editor: Gordon R. Pennock.

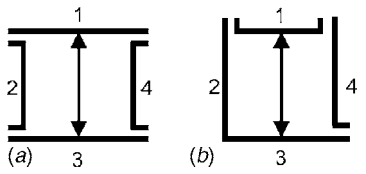

Fig. 1 Two box designs (*a*) without and (*b*) with adjustable height during assembly (see Ref. [1])

utilized the the screw theory to build a general method which can determine all relative degrees of freedom (DOF) between any two rigid bodies making contacts to each other. Recently, Konkar and Cutkosky [10] have proposed a recursive algorithm which computes motions allowed by mating features within mechanisms. Adams and Whitney [11] have extended this method by providing a dual method to compute the state of constraint of parts and applied it to rigid body assemblies with mating features such as pin-slot joint.

While these works provide tools for analyzing constraints in a given assembly and simple design guidelines, they do not address a systematic synthesis of an assembly with desired constraint characteristics such as in-process dimensional adjustability and proper part constraints, as discussed in this paper.

3 Terminology (Previously Defined in Ref. [12])

Since the assembly synthesis approach deals with objects yet to be decomposed into an assembly of separate parts, related terms need to be defined to avoid confusion with generic meanings used in other literatures.

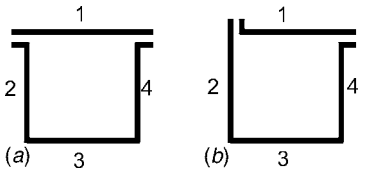

Fig. 2 Two box designs (*a*) without and with (*b*) proper constraints (see Ref. [1])

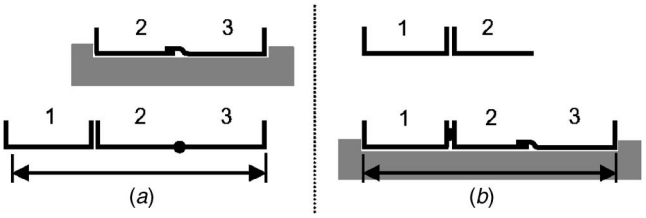

Fig. 3 Assembly sequences (*a*) without and (*b*) with in-process adjustability (modified from (Ref. [2])

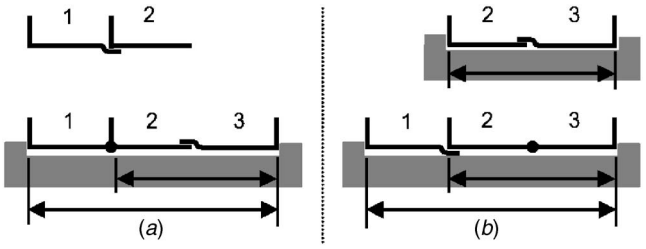

Fig. 4 Assembly sequences where two dimensions are adjusted (*a*) at one step and (*b*) independently at two steps (modified from Ref. [2])

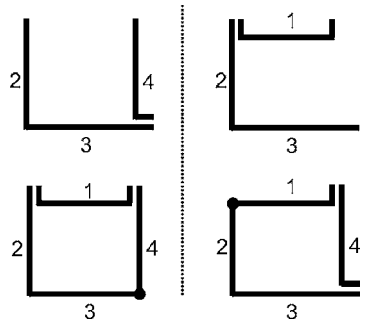

Fig. 5 Assembly sequences (*a*) without and (*b*) with proper constraints (see Ref. [1])

- A *product geometry* is a geometric representation of a whole product as one piece (Fig. 6(*a*)).
- A *member* is a section of a product geometry allowed to be a separate part. A pair of members is *connected* when they meet at a certain point in the product geometry.
- A *configuration* is a group of members which are connected to at least one member within the group. A product geometry is a configuration, so as a part (as defined later).
- The *key characteristics* (KCs) are defined by in Ref. [13] as product features, manufacturing process parameters, and assembly features that significantly affect a product's performance, function and form. In this paper, KC refers to a critical dimension to be achieved in assemblies.
- A configuration graph (or simply configuration if unambiguous in the context) is a triple

$$C = (M, E, A) \tag{1}$$

where M, E, and A are the sets of nodes representing members, edges representing connections between two members, and edges representing KCs, respectively (Fig. 6(*b*)).
- A *decomposition* is a transition of a configuration into two or more subconfigurations by removing connections between two members.
- A *part* is a configuration that is not decomposed further under given criteria, e.g., a minimum part size. A part may consist of one or more members.
- A *joint library* is a set of joint types available for a specific application domain (Fig. 7).
- An *(synthesized) assembly* is a set of parts and joints that connect every part in the set to at least one of other parts in the set.
- *Assembly synthesis* is a transformation of a product geometry into an assembly.

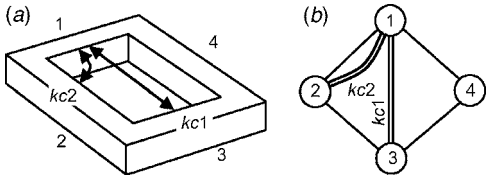

Fig. 6 (*a*) Product geometry of a beam based product and (*b*) its configuration graph

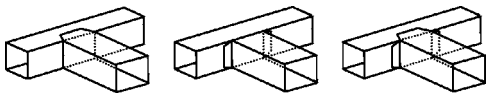

Fig. 7 An example of joint library for 3D beam based assemblies consisting of lap, butt, and lap-butt

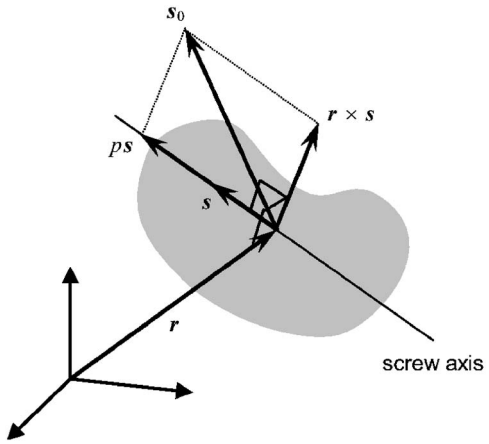

Fig. 8 Representation of a screw using screw coordinates

4 Screw Theory (Summarized From Refs. [4,10,14] and [15])

In the screw theory, a *screw* is defined as a pair of a straight line (*screw axis*) in a 3D Cartesian space and a scalar (*pitch*). A screw is commonly represented by *screw coordinates*, a pair of two row vectors $\Sigma=(\boldsymbol{s};\boldsymbol{s}_0)$ in 3D Cartesian coordinates (see Fig. 8), where $\boldsymbol{s}$ is a unit vector parallel to the screw axis and

$$\boldsymbol{s}_0 = \boldsymbol{r} \times \boldsymbol{s} + p\boldsymbol{s} \tag{2}$$

In the equation, $\boldsymbol{r}$ is the position vector of a point on the screw axis and p is the pitch, which can be recovered using the following equation:

$$p = \frac{\boldsymbol{s} \cdot \boldsymbol{s}_0}{\boldsymbol{s} \cdot \boldsymbol{s}} \tag{3}$$

A screw with an infinite pitch does not follow Eq. (2). Instead, it is denoted by using zero vector for $\boldsymbol{s}$ and having $\boldsymbol{s}_0$ represent the unit vector parallel to the screw axis.

Two types of screws, a twist and a wrench, are utilized in this paper. A *twist* is a screw representing a motion of a rigid body simultaneously rotating around and translating along an axis. Using screw coordinates, it is denoted as $T=(\boldsymbol{\omega};\boldsymbol{\nu})$, where $\boldsymbol{\omega}$ is the angular velocity and $\boldsymbol{\nu}$ is the linear velocity of a point on the body (or its extension) located at the origin of global reference frame. A *wrench* is a screw representing a force and a moment along an axis exerted on a rigid body. Using screw coordinates, it is denoted as $\Omega=(\boldsymbol{f};\boldsymbol{m})$, where $\boldsymbol{f}$ is the force and $\boldsymbol{m}$ is the moment that a point on the body (or its extension) located at the origin of global reference frame should resist.

Two screws, $\Sigma_1=(\boldsymbol{s}_1;\boldsymbol{s}_{01})$ and $\Sigma_2=(\boldsymbol{s}_2;\boldsymbol{s}_{02})$, are *reciprocal* to each other, if and only if they satisfy

$$\boldsymbol{s}_1 \cdot \boldsymbol{s}_{02} + \boldsymbol{s}_{01} \cdot \boldsymbol{s}_2 = 0 \tag{4}$$

If a twist T is a reciprocal of wrench Ω (or vice versa), Ω does no "work" to a rigid body moving according to T.

When a body undergoes linear combination of several screws (either twist or wrench), this set of screws are typically represented as a matrix where each screw in the set forms a row vector of the matrix. This matrix is called a *screw matrix*. As its row space is the screw space, the rank of a screw matrix is equal to the dimension of the screw space.

The function reciprocal ($\mathbf{S}$) returns a screw matrix, of which row space includes all reciprocal screws to screws contained in $\mathbf{S}$. It can be obtained by exchanging the former three columns and the letter three columns of the null space of $\mathbf{S}$.

The *union* of screw matrices represents the sum of screw spaces

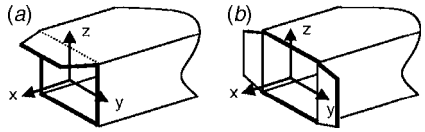

Fig. 9 Lap (*a*) and lap-butt joint (*b*) of a beam based model and the local coordinate frames for twists

and it can be obtained by simply "stacking" them on top of one another

$$\bigcup_{i=1}^{n} \mathbf{S}_i \equiv \begin{pmatrix} \mathbf{S}_1 \\ \mathbf{S}_2 \\ \vdots \\ \mathbf{S}_n \end{pmatrix} \tag{5}$$

The *intersection* of screw matrices is the set of screws common to the screw matrices, and it can be computed through double reciprocals

$$\bigcap_{i=1}^{n} \mathbf{S}_i \equiv \text{reciprocal}\left(\bigcup_{i=1}^{n} \text{reciprocal}(\mathbf{S}_i)\right) \tag{6}$$

Since a twist and a wrench are also screws, the definitions of reciprocal, union, and intersection hold.

Woo and Freudenstein [16] present kinematic properties of various joint types in screw coordinates, which are borrowed to build twist matrices of beam joint types.

Figure 9(*a*) shows a typical lap joint found in beam-based structures. When it is attached to another beam, the tab allows planar motion parallel to x-y plane. Also, if we assume that the tab is very short compared to the beam, it can be treated as line contact along y axis, thus allowing rotation about y axis. Therefore, a lap joint at its local coordinate frame can be modeled as a twist matrix

$$\mathbf{T}_{\text{lap}} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \tag{7}$$

Similarly, a butt joint in Fig. 9(*b*) allows motion parallel to y-z plane, and it can be modeled as

$$\mathbf{T}_{\text{butt}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{8}$$

In twist matrices in Eq. (7) and (8), each row represents an independent motion, and each nonzero number represents rotation or translation along a corresponding axis—ω_x, ω_y, ω_z, ν_x, ν_y, or ν_z. For example, the first row in Eq. (7) has 1 at the second column, which means the lap joint allows rotational motion about y axis. In the third row, it has 1 at the fourth column, meaning translation along the x axis. As these matrices are used only to give information on which DOFs are constrained for a joint type, amplitude of each twist (row) of these twist matrices, in this paper, does not have significant meaning.

Once a twist matrix is obtained for a joint type, the reciprocal wrench matrix can be computed as described earlier, and the wrench matrices corresponding to twist matrices in (7) and (8) are

$$\mathbf{W}_{\text{lap}} = \text{reciprocal}\ (\mathbf{T}_{\text{lap}}) = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \tag{9}$$

$$\mathbf{W}_{\text{butt}} = \text{reciprocal}\ (\mathbf{T}_{\text{butt}}) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{10}$$

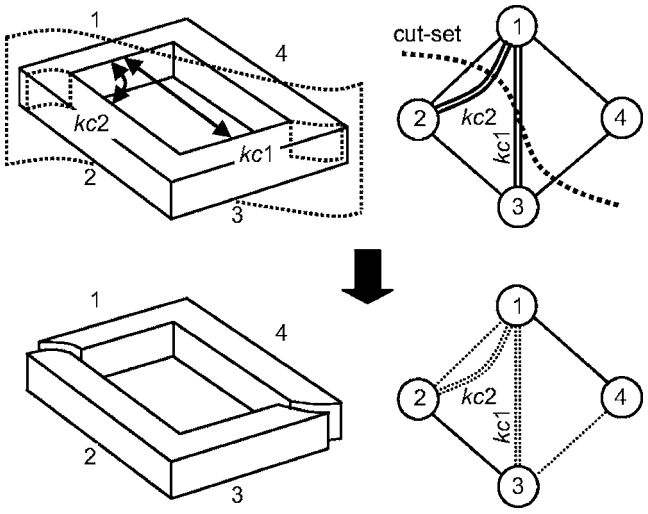

Fig. 10 A binary decomposition in product geometry (left) and configuration graph (right)

Each nonzero number now represents force or moment along a corresponding axis—f_x, f_y, f_z, m_x, m_y, or m_z—that the joint can constrain. For example, the first row in Eq. (9) has 1 at the third column, which means the lap joint can support a force along z axis.

5 3D Assembly Synthesis

5.1 Binary Decomposition. The assembly synthesis algorithm [1] adopted in this paper assumes every assembly step combines two subassemblies. Conversely, the algorithm decomposes a configuration into two (sub)configurations, by removing some connections, which is equivalent to finding a *cut-set* (In a configuration graph, edges representing KCs are not counted to a cut-set.) [17] of the configuration graph. In the following, CS_d and KC_d denote the cut-set and the set of KCs broken by a decomposition d, respectively. For the decomposition shown in Fig. 10, $CS_d = \{(1,2),(3,4)\}$ and $KC_d = \{kc1, kc2\}$.

Any configuration $C_a = (M_a, E_a, A_a)$ decomposed to two subconfigurations, $C_b = (M_b, E_b, A_b)$ and $C_c = (M_c, E_c, A_c)$, must satisfy the following conditions:

1. $M_b \neq \emptyset$ and $M_c \neq \emptyset$
2. (M_a, E_a), (M_b, E_b) and (M_c, E_c) are connected
3. $M_a = M_b \cup M_c$
4. $M_b \cap M_c = \emptyset$ (11)

The first condition states subconfigurations should be nonempty. The second condition states the configurations must be connected before and after decomposition. The third and fourth conditions specify two subconfigurations should not share any members.

A joint is assigned to each connection broken by a binary decomposition, which can be represented as a mapping $\gamma_d : CS_d \mapsto JL$, where JL is a joint library. With the joint assignment, a (binary) decomposition d can be uniquely specified as $d = (M_a, \gamma_d, (M_b, M_c))$. See Fig. 11 for an example. Note that feasible joint types for a particular joint location may depend on the local geometry at the joint location. For example, feasible joint types between two perpendicular beams would be different from that for two coaxial beams.

5.2 The First Decomposition Rule for In-Process Dimensional Adjustability. Let us consider how to assign appropriate joint types for those decompositions that have at least one broken KC. Recall Fig. 3, which has a slip plane between parts 2 and 3

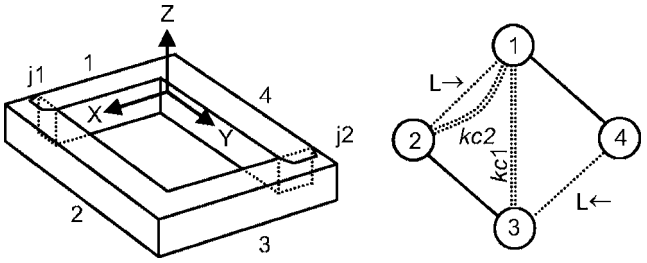

Fig. 11 Joint types assigned to the subconfigurations in Fig. 10. The "*L*→" represents a lap joint from a lower-index node to a higher-index node.

such that the KC can be delivered. The assembly sequence in Fig. 3(*b*) shows that it is desirable that a slip plane is provided at the very assembly operation where KC is realized, no matter how subassemblies are assembled before. This can be stated in the reverse course as follows: no matter how a subconfiguration is decomposed further, when KCs are broken by a decomposition, joints assigned to the cut-set, in combination, should allow motions compatible with the KCs. This statement has been referred to as the first decomposition rule for in-process dimensional adjustability [1].

A KC, in this paper, is assumed to be a critical dimension between parts only achieved by adjustment during assembly of the parts. Thus the dimension noted as a KC will be constrained by a fixture, according to which parts being assembled will be adjusted. In this context, a KC constrains relative DOFs between two parts; hence, it is natural to model a KC as a wrench matrix. The approach taking tolerance relations as constraints can be found in the area of computer-based tolerance modeling, and a recent study by Wu et al. [18] shows the number and the type of DOFs constrained for each tolerance relation in standard tolerance classes. In this paper, we consider only distance and angularity between lines (beams' axes). The distance between lines constrains only one translational DOF between two points where the KC is anchored, thus it is modeled as a wrench whose axis passes these points. The angularity between lines constrains only one rotational DOF between two lines and it is modeled as a wrench with infinite pitch whose axis is the vector product of the two lines' direction vectors.

The first decomposition rule for in-process dimensional adjustability, in other words, states that the DOFs constrained by KCs should not be constrained by the joints, thus avoiding conflicts. Once wrench matrices are associated to joints and KCs broken by a decomposition, this rule can be stated in the screw theory's terminology: for a decomposition, the wrenches representing joints and KCs should not constrain the same DOF, thus satisfying

$$\left(\underset{e \hat{\in} CS_d}{\mathbf{U}} \mathbf{W}_{g_d(e)}\right) \mathbf{I} \left(\underset{a \hat{\in} KC_d}{\mathbf{U}} \mathbf{W}_a\right) = \mathbf{O} \tag{12}$$

Since the rank of the intersection of the joint and KC matrices should be zero as shown in Eq. (12), by the theorem from linear algebra, it is obvious that the rank in Eq. (12) should be merely summation of ranks of joint and KC matrices

$$\operatorname{rank}\left(\left(\underset{e \hat{\in} CS_d}{\mathbf{U}} \mathbf{W}_{g_d(e)}\right) \mathbf{U} \left(\underset{a \hat{\in} KC_d}{\mathbf{U}} \mathbf{W}_a\right)\right) = \operatorname{rank}\left(\underset{e \hat{\in} CS_d}{\mathbf{U}} \mathbf{W}_{g_d(e)}\right) + \operatorname{rank}\left(\underset{a \hat{\in} KC_d}{\mathbf{U}} \mathbf{W}_a\right) \tag{13}$$

Furthermore, as the proper constraint design in its rigorous definition avoids underconstraints as well as over-constraints, the combined constraints from joints and KCs should cover six DOFs, such that no DOF could be left unconstrained when two parts are being assembled. In other words, the dimension of the combined wrench space, i.e., the rank of the union of joint and KC wrench

matrices, should be equal to six. Combined with Eq. (13), we can now conclude the first rule of decomposition for in-process dimensional adjustability with

$$\text{rank}((\underset{e\hat{I}CS_d}{\cup}\mathbf{W}_{g_d(e)})\cup(\underset{a\hat{I}KC_d}{\cup}\mathbf{W}_a)) = \text{rank}(\underset{e\hat{I}CS_d}{\cup}\mathbf{W}_{g_d(e)}) + \text{rank}(\underset{a\hat{I}KC_d}{\cup}\mathbf{W}_a) = 6 \tag{14}$$

Consider the product geometry decomposed in Fig. 10 and joint assignment shown in Fig. 11, which has two lap joints, $j1$ and $j2$ for edges cut by the decomposition. Suppose the location of $j1$ and $j2$ in global reference frame X-Y-Z are (3, 0, 0) and (0, 4, 0). Then, based on the local coordinate frame of lap joint shown in Fig. 9 and orientation of $j1$ and $j2$, $\mathbf{W}_{\text{lap}}$ (Eq. (9)) can be transformed to $j1$ and $j2$ in global reference frame. Then the union of joint wrench matrices can be computed

$$\underset{e\in CS_d}{\cup}\mathbf{W}_{\gamma_d(e)} = \mathbf{W}_{j1}\cup\mathbf{W}_{j2} \sim \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \tag{15}$$

(The result has been reduced to the row reduced echelon form for easy interpretation.)

The wrench matrix in (15) has 1 at the third, fourth, and fifth column, meaning that it supports force along Z axis, moments about X and Y axis, respectively. On the other hand, the decomposition in Fig. 10 has broken two KCs, $kc1$ and $kc2$. The union of these KCs is

$$\underset{a\in KC_d}{\cup}\mathbf{W}_a = \mathbf{W}_{kc1}\cup\mathbf{W}_{kc2} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 1.5 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{16}$$

Note that $\mathbf{W}_{kc1}$ (upper row) represents the distance KC between member 1 and 3 (translation along Y axis) and $\mathbf{W}_{kc2}$ (lower row) represents the angularity KC between member 1 and 2 (rotation about Z axis).

The union of the joint twist matrix (Eq. (15)) and KC twist matrix (Eq. (16)) is

$$(\underset{e\in CS_d}{\cup}\mathbf{W}_{\gamma_d(e)})\cup(\underset{a\in KC_d}{\cup}\mathbf{W}_a) \sim \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}. \tag{17}$$

It shows that the parts are not constrained in X axis by either joints or KCs. Although it does not satisfy Eq. (14), it does satisfy Eq. (13), which implies that, at least, there is no conflict between joints and KCs. As this decomposition does not satisfy Eq. (14), the assembly synthesis process will discard it.

5.3 The Second Decomposition Rule for in-Process Dimensional Adjustability. As discussed in Fig. 4, when multiple KCs in the same direction are realized at an assembly step, the adjustment of one KC will affect the dimension of the other KCs. Viewing KCs as constraints, this happens when two or more KCs constrain the same DOF of a subassembly at an assembly step. However, for complex assemblies, detecting overconstrained tolerance relationship is not straightforward from the engineering drawings because tolerances are specified on parts, not subassemblies, which are defined by assembly sequences. Therefore, clumsy assembly planning might cause a subassembly's DOF to be constrained by several KCs. In order to avoid this situation, one should plan assembly steps such that, in every assembly step, subassemblies being assembled are free of overconstraining KCs.

Accordingly, the second decomposition rule for in-process dimensional adjustability in Ref. [1] states a decomposition can break only KCs independent to each other. (In 2D cases in the previous works, only KCs perpendicular to each other were allowed to be stricter.) In other words, KCs broken by a decomposition, i.e., the KCs in KC_d, should not constrain the same DOF more than once. In such cases, the intersection of the wrench matrix corresponding to any subset of KC_d and the wrench matrix of its complement set must result in the zero matrix

$$''K\acute{I}\text{KC}_{d'}(\underset{a\hat{I}K}{\mathbf{U}}\mathbf{W}_a)\mathbf{I}(\underset{a\hat{I}KC_d\backslash K}{\mathbf{U}}\mathbf{W}_a) = \mathbf{O} \tag{18}$$

which is also equivalent to

$$\text{rank}(\underset{a\hat{I}KC_d}{\mathbf{U}}\mathbf{W}_a) = \underset{a\hat{I}KC_d}{\mathbf{\text{å}}}\text{rank}(\mathbf{W}_a). \tag{19}$$

The two KCs shown in Fig. 10, each with single wrench make the KC matrix of rank 2 as shown in Eq. (16), thus satisfying Eq. (19).

5.4 The Decomposition Rule for in-Process Proper Constraint. In Fig. 5, it has been shown that joints should be perpendicular to each other to have subassemblies being assembled properly constrained. Similarly to drawing the first and second decomposition rule for in-process dimensional adjustment, this assembly rule has been inversed to the decomposition rule for nonforced fit in our previous work [1], which allows only mutually perpendicular joints to be broken by a decomposition. This rule is simplified and limited to two-dimensional space, assuming overconstraints in rotation are minimal.

The idea of this rule is that there should be no overconstraint at each assembly step, hence, the decomosition rule (renamed as the *decomposition rule for in-process proper constraint*) should not allow any combination of joints yielding overconstraint of parts. In other words, joints placed for connections broken by a decomposition, i.e., the joints corresponding to CS_d, should not constrain the same DOF more than once. Except that joints serve as constraints, instead of KCs, this rule is identical to the second rule of in-process dimensional adjustability, thus satisfying

$$''C\acute{I}\text{CS}_{d'}(\underset{e\hat{I}C}{\mathbf{U}}\mathbf{W}_{g_d(e)})\mathbf{I}(\underset{e\hat{I}CS_d\backslash C}{\mathbf{U}}\mathbf{W}_{g_d(e)}) = \mathbf{O} \tag{20}$$

which is also equivalent to

$$\text{rank}(\underset{e\hat{I}CS_d}{\mathbf{U}}\mathbf{W}_{g_d(e)}) = \underset{e\hat{I}CS_d}{\mathbf{\text{å}}}\text{rank}(\mathbf{W}_{g_d(e)}). \tag{21}$$

For the decomposition depicted in Fig. 11, each of the two joints $j1$ and $j2$ has rank 2 (Eq. (9)). However, the union of corresponding wrench matrices has rank 3, which does not satisfy Eq. (21). In order to check what DOFs are overconstrained, we can intersect the wrench matrices

$$\mathbf{W}_{j1}\cap\mathbf{W}_{j2} = \text{recip}(\text{recip}(\mathbf{W}_{j1})\cup\text{recip}(\mathbf{W}_{j2})) = \text{recip}(\mathbf{T}_{j1}\cup\mathbf{T}_{j2}) = [0\ 0\ 1\ 4\ -3\ 0] \neq \mathbf{O} \tag{21'}$$

The results states that the joints overconstrain the translational DOF along Z axis, which yields locked moment about X axis with unit of 4 and moment about Y axis with unit of −3. It occurs because $j1$ itself constrains parts both in translation along Z axis and the moment about X axis at the same time $j2$ combined with $j1$ constrain the moment about X axis again. And $j2$ and $j1$ cooperate in the same way to result in the locked moment about Y axis.

5.5 Unified Decomposition Rule for in-Process Proper Constraints. According to Eq. (19), the set of KCs related to a decomposition should be linearly independent. Similarly, the set of joints assigned for broken connections should be linearly independent according to Eq. (21). Further, as these sets should be linearly independent to each other and of full rank when unionized by Eq. (14), these three equations in combination requires the independency of constraints, regardless of whether they are KCs or joints, and full rank when unionized. Thus, combining Eqs. (14), (19), and (21), we can unify three decomposition rules into

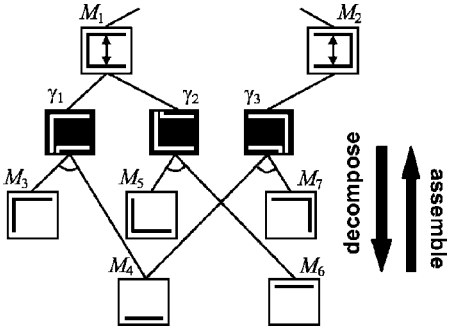

Fig. 12 A partial AND/OR graph of the 2D rectangular box in Fig. 1

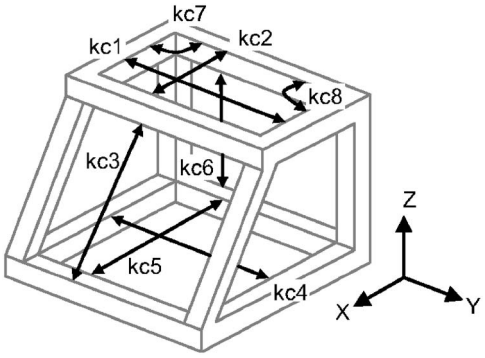

Fig. 13 A frame structure with eight KCs

$$\text{rank}\Big(\Big(\mathop{\cup}_{e\hat{I}CS_d}\mathbf{W}_{g_d(e)}\Big)\cup\Big(\mathop{\cup}_{a\hat{I}KC_d}\mathbf{W}_a\Big)\Big)=\mathop{\sum}_{e\hat{I}CS_d}\text{rank}(\mathbf{W}_{g_d(e)})+\mathop{\sum}_{a\hat{I}KC_d}\text{rank}(\mathbf{W}_a)=6 \quad (22)$$

Finally, a predicate of a decomposition $d=(M_a,\gamma_d,(M_b,M_c))$ for complying the all three rules is given as $\text{de}:2^{M_0}\times(2^{E_0}\mapsto JL)\times(2^{M_0}\times 2^{M_0})\mapsto\{true,false\}$ where $\text{de}(M_a,\gamma_d,(M_b,M_c))$ is *true* if and only if conditions in (11) and Eq. (22) are satisfied. However, it is often the case that a underconstraints are unavoidable during assembly synthesis due to the limited choice of joints, Eq. (22) may be relaxed to abandon the full rank.

5.6 Part Manufacturability. The decomposition stops when the resulting subconfigurations become manufacturable by a chosen manufacturing process. In the following case study on frame structures, components are assumed to be extruded and bent. Therefore, a predicate of a configuration M_a for stopping decomposition is given as $\text{stop\_de}:2^{M_0}\mapsto\{true,false\}$, where $\text{stop\_de}(M_a)$ is false (i.e., decomposition continues) if and only if any of the following conditions are satisfied:

- M_a has a KC (assuming KCs cannot be achieved by the tolerances of extrusion and bending)
- M_a has a closed loop (cannot extrude such parts)
- M_a has a connection point where three or more members meet (cannot extrude such parts)
- M_a has members lie on more than one plane (difficult to handle/fixture)

The product geometry shown in Fig. 6 has two KCs and a closed loop, thus stop_de returns false, subject to further decomposition.

5.7 AND/OR Graph of Assembly Synthesis. The AND/OR graph [19] is adopted to facilitate the assembly synthesis, in which multiple trees share common nodes. Although the AND/OR has been previously used to enumerate assembly sequences for a given assembly design [20], it is augmented in this paper in order to embody joint assignments. Figure 12 shows a partial AND/OR graph of assembly synthesis [1] for the 2D rectangular box shown in Fig. 1. Each node in white background contains a subset of members ($M_a\subseteq M_0$) and each node in black background contains joint assignment $\gamma_i:CS_i\mapsto JL$. A set of three lines which connects a configuration M_a, joint assignment γ_i, and two subconfigurations (M_b,M_c) is a hyperedge, represented as $(M_a,\gamma_i,(M_b,M_c))$ which is also the representation of a decomposition defined earlier. The AND/OR graph of assembly synthesis is then represented as a triple

$$AO=(S,J,F) \quad (23)$$

where S is a set of nodes representing configurations, J is a set of nodes representing joint assignments, and F is a set of hyperedges $(M_a,\gamma_i,(M_b,M_c))$ satisfying the following necessary conditions:

1. $\text{stop\_de}(M_a)=\text{false}$
2. $\text{de}(M_a,\gamma_i,(M_b,M_c))=\text{true}$ (24)

Then $AO=(S,J,F)$ is recursively defined as:

1. If $\text{stop\_de}(M_0)=\text{false}$, $M_0\in S$
2. For $\forall M_a\in S$, if $\exists\gamma_i,M_b,M_c$ such that $f=(M_a,\gamma_i,(M_b,M_c))$ satisfies necessary conditions (24), then $\gamma_i\in J,M_b,M_c\in S$and $f\in F$
3. No element is in S, J and F, unless it can be obtained by using rules 1 and 2 (25)

The recursive definition in Eq. (25) can be easily transformed to an algorithm build_ AO that generates AO from initial configuration $C_0=(M_0,E_0,A_0)$ and joint library JL by recursively decomposing a configuration into two subconfigurations, whose details are omitted due to space limitation. Using stop_de and de as defined earlier, one can run build_ AO with any 3D configurations to enumerate all possible assemblies (decompositions and joint assignments) and accompanying assembly sequence that satisfy the rules for in-process dimensional adjustability and proper part constraint.

6 Case Study

A frame structure in Fig. 13 is decomposed using the joint types in Fig. 14. As initial attempt according to Eq. (22) and definitions in (24) and (25) yields no assembly synthesis without underconstraints, Eq. (22) is relaxed to allow underconstraints. The reason for this is that the choice of decomposition and joints is limited. While a decomposition can break KCs, of which summed DOFs will vary, the joint types constraining a small number of DOFs of various kinds are not available to fill the unconstrained DOFs in various situations (see Fig. 14). And because every joint type in Fig. 14 constrains at least three DOFs, decompositions cutting more than two points will certainly results in overconstrained DOF(s). While a large joint library with joint types constraining various DOFs would be able to obtain exactly constrained solutions, we will stay with the joint types in Fig. 14, as these are the

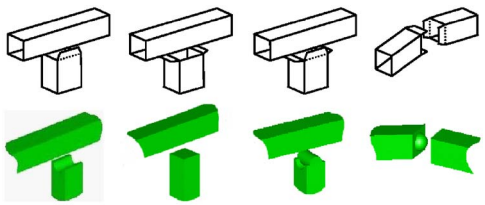

Fig. 14 (Top) Joint types for frame sturcture and (bottom) their graphical representation

Table 1 Nondominated cost vector and the number of corresponding nondominated solution trees for the frame structure shown in Fig. 14

Objectives		
No. of parts	No. of under-constraints	No. of nondominated solution trees
7	2	48
Total number of solution trees		$\approx 6.6 \times 10^9$

most typical joints found in beam structures. As allowing underconstraints would certainly increase the number of solutions, among joint assignments with underconstraints for a given decomposition, those with minimum underconstraints are included in the AND/OR graph.

The relaxed rule produced the AND/OR graph of assembly synthesis with 524 nodes representing configurations and 6762 hyperedges, which contains around seven billion trees. Building the AND/OR graph took 36.4 s with a 3.2 GHz Pentium IV personal computer (PC). When there are too many solutions as in this case study, we can search the AND/OR graph to identify optimum solutions. While there would be many other potential objectives in designing assemblies, we have chosen the number of parts and

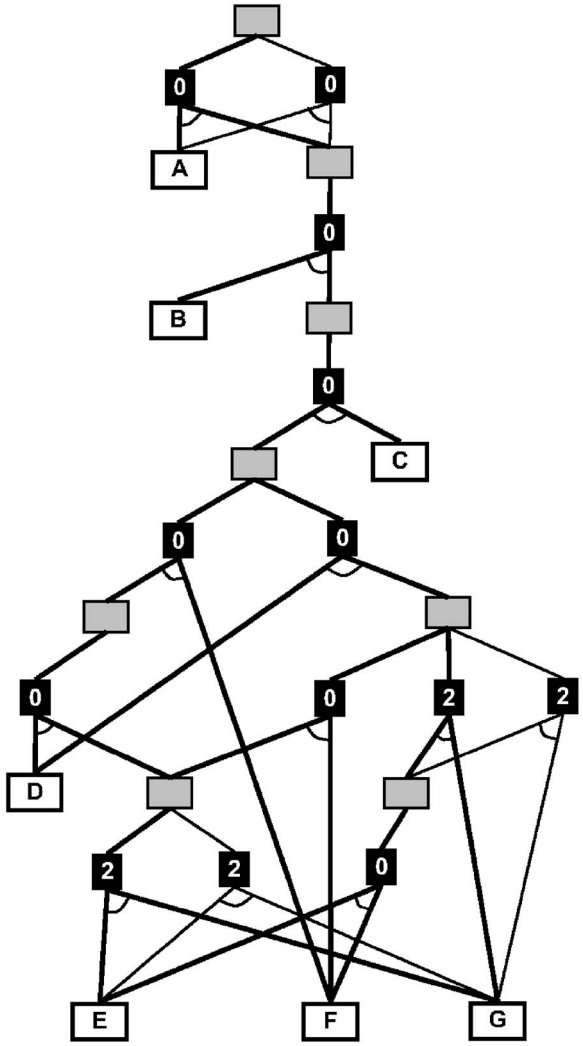

Fig. 15 Half of optimal AND/OR graph of assembly synthesis for the product geometry in Fig. 13

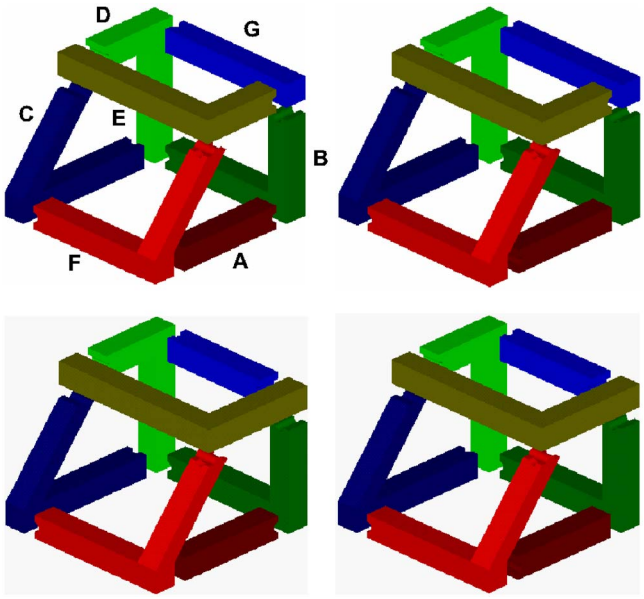

Fig. 16 All assembly designs existing in the optimal AND/OR graph shown in Fig. 15, where corresponding assembly sequences can be found

total underconstraints as objectives, which form a two-element cost vector. Using brute search starting from multiple terminal nodes satisfying stop_de, we have found only 48 trees are nondominated, (If no element in the cost vector of a solution tree T_1 is smaller than the corresponding element in the cost vector of a solution tree T_2, then T_1 is dominated by T_2, in a minimization problem. If a solution tree is not dominated by any other solution

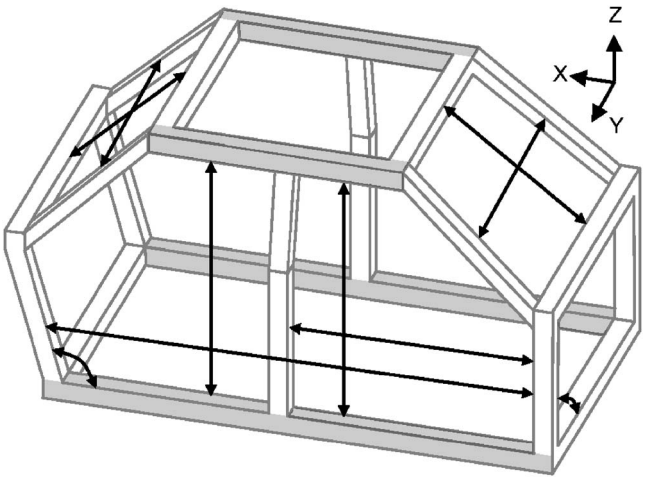

Fig. 17 Passenger area of an automotive space frame

Table 2 Nondominated cost vectors and the number of corresponding nondominated solution trees for the automotive space frame structure shown in Fig. 17

Objectives		
No. of parts	No. of under-constraints	No. of nondominated solution trees
15	14	13,602,816
16	13	1,843,200
Total		15,446,016
No. of solution trees		$\approx 4.0 \times 10^{18}$

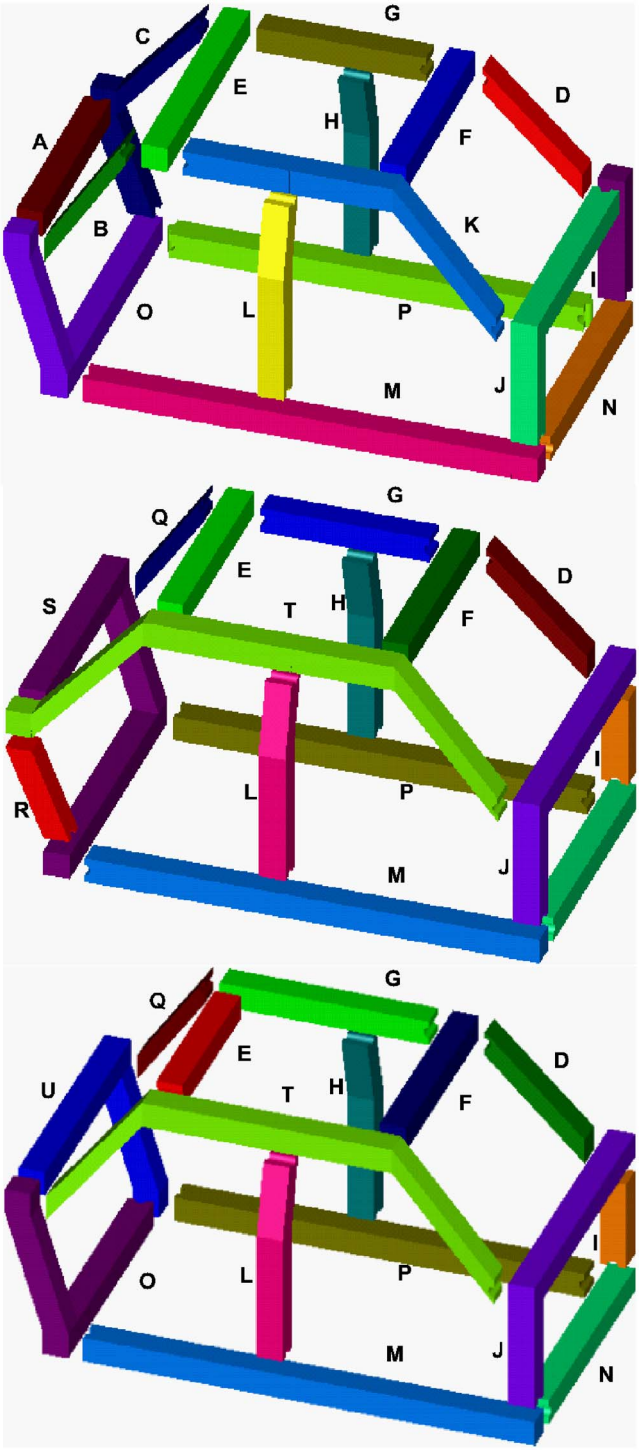

Fig. 18 Three optimal assembly designs synthesized for the automotive frame in Fig. 17

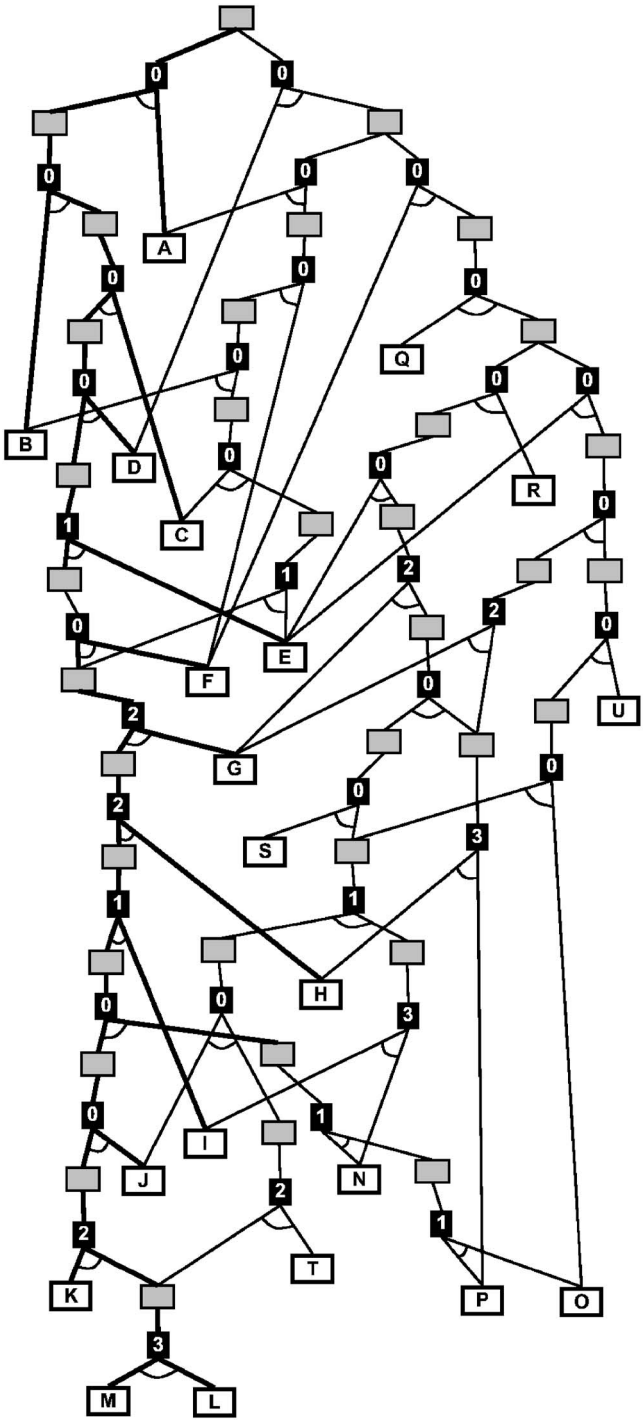

Fig. 19 A partial optimal AND/OR graph of assembly synthesis for the automotive space frame in Fig. 17

tree, it is called a nondominated solution tree, and its cost vector is called a nondominated cost vector.) and all these trees have seven parts and two underconstraints throughout their assembly sequences. The results is summarized in Table 1. Due to the space limit, 12 among 48 trees that have the are depicted in Fig. 15 in the form of AND/OR graph. In Fig. 15, white nodes are parts and grey nodes are subassemblies. Joint assignments are represented as black nodes with numbers, which represent the number of underconstraints related to the assembly step.

Figure 16 shows all assembly designs in Fig. 15. While the AND/OR graph has 12 trees, Fig. 16 shows only one way of part decomposition with four variations in joint assignments, which is because there are three different assembly sequences for each assembly design in Fig. 16. For example, the assembly design at top left in Fig. 16 has three assembly sequences, which are thick lined in Fig. 15. One of these assembly sequences is as follows:

1. Assemble E and G to achieve $kc2$ and $kc8$. Underconstrained in Y translation and Y rotation.
2. Assemble D and E-G to achieve $kc1$ and $kc7$. Properly constrained.

3. Assemble *F* and *D-E-G* to achieve *kc*3. Properly constrained.
4. Assemble *C* and *D-E-F-G* without achieving any KC. Properly constrained.
5. Assemble *B* and *C-D-E-F-G* to achieve *kc*5 and *kc*6. Properly constrained.
6. Assemble *A* and *B-C-D-E-F-G* to achieve *kc*4. Properly constrained.

Figure 17 shows the passenger area of an automotive space frame with 18 KCs (KCs at the far side of *X-Z* plane are symmetrical and omitted). Each horizontal beam at side (shaded area) has an intersection with a vertical beam in the middle, however, we have not allowed them to part, in other to reflect practices and reduce the size of solution. With the same relaxed rule permitting underconstraints, the AND/OR graph contains 4726 nodes representing configurations and 67,099 hyperedges, which contains around 4.0×10^{18} solution trees. It took 209.5 min with a 3.2 GHz Pentium IV PC to build the AND/OR graph. Based on the two objectives, 1,5446,016 trees are found to be nondominated. Associated cost vectors for these nondominated solution trees are listed in Table 2. Three of these optimal assembly designs are shown in Fig. 18 and the corresponding AND/OR graph of assembly synthesis is shown in Fig. 19. The first assembly design consists of 16 parts and has two different assembly sequences in the AND/OR graph shown, which yield 13 underconstraints during assembly steps. The other designs have 15 parts and each has one assembly sequence in the graph, yielding 14 underconstraints.

7 Summary and Discussion

This paper presented a 3D assembly synthesis method for in-process dimensional adjustability and proper part constraint based on the screw theory. The method has been successfully applied to beam based structures with predefined joint library and optimal solutions have been identified using graph search. While this paper focuses only on in-process adjustability and proper part constraints and suggests ideal designs for these criteria, other criteria, such as part symmetry and structural stiffness, also drive assembly synthesis in practice and narrow down the solutions. When there are too many nondominated solution trees as in the second case study, it would be unpractical for a human designer to choose one. The present method, therefore, would most effectively be integrated in the design process if it is applied to subassemblies of a product first decomposed by a human designer based on the current practice.

Acknowledgments

This work has been supported by the National Science Foundation with a CAREER Award (DMI-9984606) and Toyota Motor Company.

References

[1] Lee, B., and Saitou, K., 2003, "Decomposition-Based Assembly Synthesis for In-Process Dimensional Adjustability," ASME J. Mech. Des., **125**(3), pp. 464–473.
[2] Whitney, D. E., Mantripragada, R., Adams, J. D., and Rhee, S. J., 1999, "Designing Assemblies," Res. Eng. Des., **11**, pp. 229–253.
[3] Blanding, D. L., 1999, *Exact Constraint: Machine Design Using Kinematic Principles*, ASME Press, New York.
[4] Ball, R. S., 1900, *A Treatise on the Theory of Screws*, Cambridge University Press, Cambridge.
[5] Whitehead, T. N., 1954, *The Design and Use of Instruments and Accurate Mechanism*, Dover, New York.
[6] Kamm, L. J., 1990, *Designing Cost-Effective Mechanisms*, McGraw-Hill, New York.
[7] Kriegel, J. M., 1995, "Exact Constraint Design," Mech. Eng. (Am. Soc. Mech. Eng.), **117**(5), pp. 88–90.
[8] Downey, K., Parkinson, A. R., and Chase, K. W., 2002, "Smart Assemblies for Robust Design: A Progress Report," *Proceedings of the 2002 ASME Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Montreal, Canada, 29 Sep.–2 Oct., Paper No. DETC2002/DAC–34135.
[9] Waldron, K. J., 1966, "The Constraint Analysis of Mechanisms," J. Mech., **1**(2), pp. 101–114.
[10] Konkar, R., and Cutkosky, M., 1995, "Incremental Kinematic Analysis of Mechanisms," ASME J. Mech. Des., **117**, pp. 589–596.
[11] Adams, J. D., and Whitney, D. E., 1999, "Application of Screw Theory to Constraint Analysis of Assemblies of Rigid Parts," *Proceedings of the 1999 IEEE International Symposium on Assembly and Task Planning*, Porto, Portugal, July 1999, pp. 69–74.
[12] Lee, B., and Saitou, K., 2003, "Assembly Synthesis with Subassembly Partitioning for Optimal In-Process Dimensional Adjustability," *Proceedings of the 2003 ASME Design Engineering Technical Conferences*, Chicago, Illinois, 2–6 Sep., Paper No. DETC2003/DAC–48730.
[13] Lee, D. J., and Thornton, A. C., 1996, "The Identification and Use of Key Characteristics in the Product Development Process," *Proceedings of the 1996 ASME Design Engineering Technical Conferences*, Irvine, California, Paper No. 96-DETC/DTM–1506.
[14] Hunt, K. H., 1978, *Kinematic Geometry of Mechanisms*, Oxford University Press, New York.
[15] Roth, B., 1983, "Screws, Motors, and Wrenches that can not be Bought in a Hardware Store," *Robotics Research, The First Symposium*, MIT Press, Cambridge, MA, pp. 679–735.
[16] Woo, L., and Freudenstein, F., 1970, "Application of Line Geometry to Theoretical Kinematics and the Kinematic Analysis of Mechanical Systems," J. Mech., **5**, pp. 417–460.
[17] Foulds, L. R., 1991, *Graph Theory Applications*, Springer-Verlag, New York.
[18] Wu, Y., Shah, J. J., and Davison, J. K., 2003, "Computer Modeling of Geometric Variations in Mechanical Parts and Assemblies," J. Comput. Inf. Sci. Eng., **3**(1), pp. 54–63.
[19] Nilsson, N. J., 1980, *Principles of Artificial Intelligence*, Tioga, Palo Alto, Calif.
[20] Homem de Mello, L. S., and Sanderson, A. C., 1990, "AND/OR Graph Representation of Assembly Plans," IEEE Trans. Rob. Autom., pp. **6**(2), pp. 188–199.