Applications of symmetry in point–line–plane frameworks for CAD

Patrick W. Fowler¹, Simon D. Guest²,* and John C. Owen³

¹Department of Chemistry, University of Sheffield, Sheffield S3 7HF, UK; ²Department of Engineering, University of Cambridge, Trumpington Street, Cambridge CB2 1PZ, UK and ³Siemens, Francis House, 112 Hills Road, Cambridge CB2 1PH, UK

*Corresponding author. E-mail: sdg@eng.cam.ac.uk http://orcid.org/0000-0002-0152-6579

Abstract

Computer-aided design (CAD) typically deals with geometries (points, lines, and planes) subject to constraints on distances and angles. Simple counting of freedoms and constraints, as used in the analysis of engineering structures, also provides a useful condition on the residual freedoms of a CAD drawing. Here, we derive general symmetry-extended counting equations to account more fully for the balance of freedoms and constraints in 2D point–line and 3D point–line–plane frameworks. General forms are given for symmetries of the freedoms of points, lines, and planes, and constraints based on distances and angles. The resulting toolkit can be used to give stronger conditions on dimensioning of CAD drawings. This importation to CAD of a physical point of view, in which residual freedoms correspond to the mechanisms and redundant constraints to the states of self-stress of a structure composed of bodies and joints, can often reveal hidden freedoms and redundancies in CAD systems. Point-group symmetry is not a panacea: Mechanisms that depend on specific geometries may escape detection by symmetry alone. One systematic limitation of this type is proved for polyhedra with planar faces and prescribed edge lengths.

Keywords: computational geometry; geometric modeling; Maxwell counting; Point Group Symmetry; Rigidity; Constraints solving; Quadratically solvable

1. Introduction

Constraint solving is an important aspect of computer-aided design (CAD), and there is a considerable literature that deals with movability of kinematic mechanisms/linkages and geometric constraint solving (Owen, 1991; Bouma et al., 1995; Kim et al., 1998; Barton et al., 2009; Müller, 2016; Sitharam et al., 2019). This paper takes a cross-disciplinary approach to the consideration of systems of points, lines, and planes subject to constraints that can be specified in terms of distances and angles. Such configurations give representations of physical systems that may be simple but are sufficient to encompass many engineering structures and mechanisms and they are already versatile enough to allow description of the balance of geometric freedoms and constraints involved in setting up and dimensioning engineering drawings. There is a close analogy between these two types of applications. For example, sets of points with distance constraints are precisely analogous to bar-and-joint frameworks in structural engineering. Symmetry analysis of such frameworks and generalizations to other combinations of bodies and joints have proved fruitful (Fowler & Guest, 2000; Guest & Fowler, 2005; Guest et al., 2010; Owen & Power, 2010). Here, we adapt and extend this analysis to the general case of a set of points, lines, and planes that is subject to constraints.
The basic objects in the present analysis are geometries (points, lines, and planes), which are “decorated” as needed with motifs to represent the applicable constraints mathematically in a way that avoids singularities. For example, signed distances are used in 2D to give a non-singular description of the point–line distance constraint that continues smoothly to the case when the point lies on the line; in the drawings used in this paper, this constraint would be represented by decorating the line with a perpendicular displacement arrow. In effect, we are treating lines in 2D and planes in 3D as half-spaces (Jackson & Owen, 2016).

In the spirit of Maxwell counting (Maxwell, 1864; Calladine, 1978) in engineering, for the present application to drawings we first establish the freedoms associated with geometries and the constraints imposed by the various distance and angle conditions. We count these in order to establish the number of residual freedoms in a drawing, which would correspond in the engineering context to the net mobility of a structure. Then, we find symmetry extensions to Maxwellian and Calladinean counting rules for systems of points and lines in 2D and of points, lines, and planes in 3D. The theory of group representations when applied to the symmetry-extended freedom count can often detect the presence of redundant constraints and excess freedoms in a drawing, even when this is not detected by the simple freedom count.

Such symmetry techniques have often been used in an engineering context where the requirement is to describe a system that has mobility (Fowler & Guest, 2005). Each application requires some tailoring to the types of object under consideration, whether bars, bodies, and joints as in engineering, or geometries and constraints, as here. For applications in CAD, the perspective is further extended, in that we are also interested in finding solutions to the constraint equations (i.e. determining the coordinates of the geometries) for given values for the constraints (such as the distances between pairs of points and lines or the angle between pairs of lines; Owen, 1991). Typically, we may be given approximate coordinates for the geometries and required to find precise coordinates nearby that satisfy the constraints exactly. A consequence of the inverse function theorem (Spivak, 1965) is that for every excess freedom detected by symmetry there is a corresponding locally redundant constraint, which means it can be removed without affecting the solution space in the neighborhood of the configuration.

An important property of the constraint equations is that they may be quadratically solvable, by which we mean they can be solved using only the successive solution of quadratic equations, equivalent to the classical notion of constructibility with ruler and compasses (Jackson & Owen, 2019). The fact that some constraints are locally redundant can make it easy to determine that the constraint equations are quadratically solvable and hence to solve them, as we show in examples below.

The plan of the paper is as follows: In Section 2, the groundwork is laid for the guiding analogy between concepts in CAD and structural engineering, and an example shows the use of symmetry in recognition of potential problems with dimensioning of drawings. Counting rules, both scalar and symmetry extended, are worked out for systems of points, lines, and planes (Section 3). In Section 4, general expressions for the symmetries of the respective freedoms are obtained (equations 3–6, and 11). In Section 5, symmetries for the various types of distance and angle constraints are tabulated (Figs 3 and 4). As described in Section 6, this tabulation gives a toolkit for use with the overall mobility equation (14) in applications to specific systems. Examples where symmetry-extended equations detect hidden freedoms and redundant constraints in CAD systems are given in Section 7. There are some systematic limitations on the information that a pure symmetry approach can provide. A general result in Section 8 shows cancellation of symmetries of freedoms and constraints on geometries in polyhedra with planar faces and fixed edge lengths. A detailed treatment is given in Section 9 of freedoms and constraints for the frequently encountered case of the cubic polyhedral cage (3D embedding of a cubic polyhedral graph), of relevance to CAD, structural mechanics, and design of chemical force fields. The paper ends with some brief conclusions.

2. CAD, Structural Engineering and Symmetry

The upper panel of Fig. 1 shows two similarly dimensioned drawings, which were created in the commercial design system NX from Siemens. A drawing is said to be fully dimensioned if it has enough dimensions and annotations to enable precise replication on a drawing board or in another CAD system. If any small, but non-zero, change to each dimension value in turn (keeping the others unchanged) determines another dimensioned drawing, then the drawing is said to be independently dimensioned. A drawing that is both fully and independently dimensioned is said to be well dimensioned.

Comparison of the upper and lower drawings in Fig. 1a shows that they are not fully dimensioned, since the height of the horizontal construction line is different in the two drawings. The reader can easily verify that, for any reasonable positioning of the horizontal construction line, the drawing can be made by ignoring any one of the dimensioned line segments in the left-hand part of the drawing and any one in the right-hand part of the drawing. The drawing can then be constructed sequentially (using only ruler and compasses). On completion, the removed segments will be found to have the required length. This shows that the drawing is neither fully nor independently dimensioned.

The lower panel, Fig. 1b, shows two slightly modified drawings in which the reflection symmetries of the drawing have been broken in distinct ways. In the upper drawing, the overall reflection symmetry is maintained and the internal reflection symmetry within the component pieces is broken. In the lower drawing, the symmetry of the component pieces is maintained but the overall symmetry is broken. The reader is invited to determine which of these, if either, is well dimensioned.

In fact, the upper drawing is well dimensioned whereas the lower drawing is not. The methods to be described in this paper can be used to prove that the lower drawing is neither independently nor fully dimensioned. All variants of the lower drawing can be constructed sequentially with ruler and compasses using the method described above.

The upper drawing is almost certainly not constructible with ruler and compasses (Owen, 1991) and requires a CAD system that has a numerical constraint solving capability. This drawing was made using the 2D Dimensional Constraint Manager (DCM) constraint solver, which was developed by one of the authors and colleagues and is incorporated into the NX product from Siemens and into many other commercial CAD products. The fact that the upper drawing is well dimensioned is easily verified in 2D DCM by showing that the Jacobian matrix of the system of constraint equations has independent rows and maximum rank.
Figure 1: Four CAD drawings of the same basic design with different dimension values. Of these, only the upper drawing of (b) is well dimensioned.
The group representation methods used in this paper to determine the balance of freedoms and constraints were first applied in the analysis of the rigidity of structures (Kangwai & Guest, 1999; Guest & Fowler, 2005, 2007). As this field uses terminology rather different from that used in CAD and constraint solving, we will describe the relationship between them with the aid of our example.

From a CAD perspective, the drawing consists of eight line segments (each with a length specified by the dimensions) and one construction line. Four of the line segments have an end point whose coordinates can be specified independently as \((-20, 0)\) and \((20, 0)\) and the other four have an end point that has a zero \(y\)-coordinate. A completely free line segment has three freedoms corresponding to rigid-body motions. Hence, four of the segments have one freedom and the other four have two freedoms giving a total of twelve freedoms. Two of the segments are automatically connected because their end points include a pair with the same coordinates. Connectivity of the ends of other segments, which lie on the \(x\)-axis, introduces two constraints and connectivity of the segments at the remaining four points introduces eight further constraints. The construction line ensures that four of the common segment ends are collinear, and since any two points are collinear this introduces two further constraints. There are a total of 12 freedoms and 12 constraints, which gives a residual freedom count of zero and suggests that the drawing might be fully dimensioned.

This process of counting freedoms and constraints is automated in dimensional constraint solving software. We describe briefly how this is done in the 2D DCM software component. A line segment is broken into two points and a line. Each point is constrained to be coincident with the line, and the length of the segment is specified by a distance dimension between the two points. Line segments with a common end point share the same point. Construction lines are represented as lines and the \(x\)- and \(y\)-axes are also represented as “fixed” lines. Each unfixed point and line has two freedoms. Each of the four drawings in Fig. 1 has eight points, eleven lines (two of which are fixed), twenty four point–line coincident constraints, two point–line distance constraints, and eight point–point distance constraints. Again, this gives a residual freedom count of zero.

In structural engineering terms, this design corresponds to a bar-joint framework with two slider joints. Each point corresponds to a joint and each dimension between two points corresponds to a bar. Two of the joints are pinned. The construction line and the \(x\)-axis each corresponds to a slider joint. Each unpinned joint has two positional freedoms (in 2D) and each bar imposes one constraint. The slider joints each imposes two constraints. The net mobility of the structure is zero.

The difference between the number of freedoms and constraints is known as the net mobility of a structure and corresponds to the residual freedoms in a drawing. A drawing is fully dimensioned if the corresponding structure is rigid, and well dimensioned if the corresponding structure is minimally rigid.

The Jacobian matrix in constraint solving is called the rigidity matrix in structural engineering (with each row of the matrix divided by 2). This matrix has a column for each freedom and a row for each constraint. The number of residual freedoms, which corresponds to the net mobility, is the difference between the number of rows and columns. Vectors in the kernel of the rigidity matrix are called flexes (or infinitesimal mechanisms), and vectors in the cokernel are called self-stresses (named after the corresponding internal equilibrium forces that can therefore exist). The kernel of the rigidity matrix has a subspace of internal flexes that are orthogonal to the infinitesimal rigid-body motions. The dimension of the space of self-stresses is denoted by \(s\) and that of the space of internal flexes by \(m\). The rigidity matrix determines the first order or infinitesimal rigidity properties of the structure, which is said to be infinitesimally rigid if \(m = 0\), and infinitesimally independent if \(s = 0\). A basic theorem of linear algebra states that the difference between the numbers of columns and rows of a matrix is equal to the difference between the dimensions of its kernel and cokernel. This is known in structural engineering as the Maxwell/Calladine relation (or counting rule). If \(s = 0\), then rigidity and infinitesimal rigidity are equivalent. If \(m = 0\), then \(s\) gives the number of locally redundant constraints. A structure is called isostatic if both \(m = 0\) and \(s = 0\).

As the examples in Fig. 1 show, a drawing may not be well dimensioned even though it has a residual freedom count of zero. This happens for specific reasons of the geometry. Similar examples are studied in robotics. For example, a Stewart-Gough platform is known to have an “internal” motion if the upper joints lie on a conic and the lower joints also lie on a conic and these two conics are in projective equivalence (Nawratil, 2013). In our case, we use methods from group theory to show that drawings with no residual freedoms may not be well dimensioned, owing to the interplay between symmetry in the dimensioning scheme and symmetry in the positions of the geometries. This property also leads to redundant dimensions and often allows the simple construction of drawings that would otherwise be difficult without numerical methods.

The essential idea we use is that if a structure or drawing (or part of a drawing or structure) is symmetrical then both the geometries and the constraints can be viewed separately under each symmetry operation (Kangwai & Guest, 1999). A geometry is unshifted if its coordinates are left unchanged by the symmetry operation. A pairwise constraint is unshifted if either both geometries connected by the constraint are unshifted, or both are of the same type and their coordinates are exchanged. It is clear from linear algebra that a structure cannot be infinitesimally rigid if a count of the residual freedoms is greater than zero and from the inverse function theorem that it cannot be rigid if in addition it has no states of stress. Similarly, it can be shown that the structure cannot be infinitesimally rigid if a count of the residual freedoms (suitably defined) of only the unshifted geometry and constraints is greater than zero (Owen & Power, 2010). In this case, group representation theory may also imply the symmetry of the flexes and determine whether they preserve the original symmetry of the structure. If the original symmetry is preserved and there are no self-stresses with the same symmetry (or which are fully symmetric), then the structure supports a finite motion (a continuous flex) (Schulze, 2010) and the corresponding dimensioned drawing is not fully dimensioned.

The freedom count that is assigned to a geometry under a symmetry operation is the trace of the transformation matrix that describes the effect of the operation on the geometry coordinates. For example, a point in 2D has zero freedom count under a reflection (which reverses one of a pair of orthogonal translations and preserves the other), whereas it has a count of \(-2\) under a half-turn rotation (which reverses translations in both \(x\)- and \(y\)-directions). Similarly, the effective number of freedoms removed by a constraint may depend on the type of constraint and the symmetry operation. In the following sections, we show how these numbers
are computed and provide a complete tabulation for points, lines, and planes in 2D and 3D and for various constraints between them.

3. Points, Lines, and Planes

3.1. Scalar counting

Each point in 2D has two degrees of freedom and each point in 3D has three. Lines in 2D have two degrees of freedom (a translation perpendicular to the line direction and a rotation in the plane). Lines in 3D have four degrees of freedom (translations along, and rotations about, two independent axes perpendicular to the line). For lines, these freedoms can be considered as the set of all possible rigid-body motions with subtraction (in 2D) of the translation along the line, or (in 3D) of this translation and the rotation about the line. Another justification for the four-parametric nature of the family of 3D lines is via Plücker coordinates, which are six-tuples subject to the quadratic constraint known as Klein’s quadric (Plücker, 1865a, b; Klein, 1870).

A plane has three degrees of freedom, which we may consider as one freedom to translate along the normal to the plane and two freedoms to rotate about independent in-plane axes. We may also regard these freedoms as derived from the six freedoms of rigid-body motion by exclusion of the single rotation about the plane normal and the pair of translations orthogonal to the plane normal.

Hence, the Maxwell/Calladine counting rule for the net mobility of a system of points, lines, and planes is (in 3D and 2D) as follows:

\[
\begin{align*}
3D: & \quad m - s = 3P + 4L + 3S - C - 6. \\
2D: & \quad m - s = 2P + 2L - C - 3.
\end{align*}
\]

where \(P\), \(L\), and \(S\) are the numbers of points, lines, and planes, and \(C\) is the total number of freedoms removed by constraints. Constraints may be imposed on distances \(D\) of types labeled \(PP\), \(PL\), \(PS\), \(LL\), \(LS\), and \(SS\) in 3D, or \(PP\), \(FL\), and \(LL\) in 2D, or on angles \(A\) of types labeled \(LL\), \(LS\), and \(SS\) in 3D, or \(LL\) in 2D. A constraint typically removes a single degree of freedom, but may in some settings remove more. Finally, the constant terms of \(-6\) and \(-3\) on the right-hand sides of (1) and (2) account for removal of rigid-body motions of the whole system.

The mobility equations (1) and (2) give the balance \(m - s\), but do not determine \(m\) and \(s\) separately. Further information on \(m\) and \(s\) may often be obtained by extending pure counting equations such as these to account for symmetries of mechanisms and states of self-stress (Fowler & Guest, 2000; Guest & Fowler, 2005).

3.2. Symmetry-extended counting

In CAD, a constrained system of points, lines, and planes is usually described by a graph in which the vertices are labeled with the type of geometry (point, line, or plane) that they represent, and likewise the edges are labeled with the type of constraint (distance or angle) that they represent.

A constrained system of points, lines, and planes has a point group \(G\) consisting of the symmetry operations that leave the system as a whole unchanged. The general principle behind the construction of symmetry-extended versions of counting relations of the Maxwell–Calladine type is that all components of an object occur in sets within which they have well-defined behavior under the symmetries of the object itself. This extends to decorations of the object with sets of scalars, vectors, or other local symbols and motifs. Consideration of these symmetries allows construction of a single symmetry-extended counting rule that embodies the ordinary scalar Maxwell/Calladine count as a special case, and in favorable cases adds further conditions on the separate contributions of \(m\) and \(s\) to the generalized mobility count.

In the present application, the sets of structural components, freedoms, and constraints have characters \(\chi(g)\) under the various symmetry operations \(g \in G\), which define their (typically reducible) representations \(\Gamma\). [For definitions and use of terms, see our earlier papers and standard texts (Altmann & Herzig, 1994; Bishop, 1973).] Symmetry operations do not mix geometries or constraints of different types.

An illustration for a simple case is given in Fig. 2. The point group considered is \(C_{2n}\), with a list of four symmetry elements, consisting of the identity \(E\); an axis of two-fold rotation, \(C_2\); and two orthogonal mirror planes, \(\sigma_1\) and \(\sigma_2\) (Fig. 2a). The character table (Fig. 2b) is generated from the group multiplication table by standard methods (see e.g. Griffith, 2009), and here has four irreducible representations, \(A_1\), \(A_2\), \(B_1\), and \(B_2\). Hence, the possible symmetry behavior of any set of structural components or local motifs of a \(C_{2n}\)-symmetric object is described by some linear combination of these. The figure shows the contrasting nodal behavior of scalar functions of each type (Fig. 2c), and then illustrates (Fig. 2d–g) the ways in which they combine to give reducible representations for sets of scalars and vectors associated with vertices and edges of a simple geometrical object in a \(C_{2n}\) setting. Representations can be added and multiplied, character by character, according to standard rules (Bishop, 1973; Altmann & Herzig, 1994).

Our aim here is to establish relations between the symmetries of geometries, freedoms, and constraints to act as symmetry-extended parallels to the scalar counting rules. A comprehensive list of the various reducible representations needed for this task is calculated in the next two sections.

4. Freedoms

We consider in turn the symmetries of freedoms of points, lines, and planes.
4.1. Points

Following Fowler and Guest (2000), the representation of the freedoms of a set of points is simply

3D:
\[ \Gamma_{\text{freedom}}(P) = \Gamma(P) \times \Gamma_T. \]  

2D:
\[ \Gamma_{\text{freedom}}(P) = \Gamma(P) \times \Gamma(T_x, T_y). \]  

where \( \Gamma(P) \) is the permutation representation of the points, with character \( \chi(g) \) equal to the number of points that are left unshifted by the operation \( g \). \( \Gamma_T \) is the representation of the set of translations, \( \Gamma_T = \Gamma(T_x, T_y, T_z) \), and \( \Gamma(T_x, T_y) \) is its restriction to motion in the \( x, y \) plane.
opposite sign under improper operations, we can write the freedoms of the set of planes span by

\[ \Gamma_{\text{freedom}}(L) = \Gamma_{\perp}(L) \times (\Gamma_\perp + \Gamma_\parallel) - \Gamma_\parallel(L). \] (5)

\[ \Gamma_{\text{freedom}}(L) = \Gamma(L) \times (\Gamma(L) + \Gamma(R)) - \Gamma(L). \] (6)

where \( \Gamma_\parallel \) is the representation of rotations in 3D and \( \Gamma(R) \) is its restriction to motion in the \( x, y \) plane. Also, \( \Gamma(L) \) is the permutation representation of the lines and \( \Gamma_\parallel(L) \) is the representation of a set of vectors directed along the lines, one per line. \( \Gamma_\parallel(L) \) is the totally symmetric representation \( \chi(\phi) = +1 \) for \( g \) and \( \chi(\phi) = -1 \) for improper \( g \). The multiplication of \( \Gamma_\parallel \) by \( \Gamma_\perp \), in (5) accounts for the pseudo-vector nature of the rotation about the direction of the line.

An equivalent form of \( \Gamma_{\text{freedom}}(L) \) that avoids the subtractions in (5) and (6) is

\[ \Gamma_{\text{freedom}}(L) = \Gamma_\perp(L) \times (\Gamma_\perp + \Gamma_\parallel). \] (7)

\[ \Gamma_{\text{freedom}}(L) = \Gamma(L) \times (\Gamma(L) + \Gamma(R)) + \Gamma(L). \] (8)

where (in 3D) \( \Gamma_\perp(L) = \Gamma(L) \times \Gamma_\perp - \Gamma_\parallel(L) \) is the representation of a set of vectors transverse to the lines, one orthogonal pair per line, or (in 2D) \( \Gamma(L) \) is the representation of a set of vectors orthogonal to the lines, one per line. The subscripts \( \perp \) and \( \perp \perp \) are chosen to reflect the physical distinction in the number of vectors orthogonal to a line in 3D and 2D. The number of freedoms [i.e. the trace of \( \Gamma_{\text{freedom}}(L) \) under the identity] is therefore 4L in 3D and 2L in 2D, as in the scalar relations (2) and (1).

The forms of equations (5)–(8) can be justified by considering the transformation properties of the freedoms of a line under operations of the maximum possible site symmetry group (the group \( G \) of operations that leaves the line in place). These calculations are shown in Tables 1 and 2 for the simplest case of the single line, where \( G = D_{\text{ab}} \) in 3D and \( C_{\infty h} \) in 2D, and the permutation representation \( \Gamma(G) \) is by definition \( \Gamma(1) \). Reduction of \( \Gamma_{\text{freedom}}(L) \) gives \( \Gamma_{\text{freedom}}(L) = \Gamma_\perp(L) \times (\Gamma_\perp + \Gamma_\parallel) \) in 3D and \( \Gamma_\perp(L) \times (\Gamma_\perp + \Gamma_\parallel) \) in 2D, which are, respectively, the symmetries of translations along and rotations about the two directions orthogonal to the line (in 3D), and the symmetries of the in-plane rotation and the translation perpendicular to the line (in 2D).

4.3. Planes

It is straightforward to include planes within the symmetry formalism. Let the permutation representation of the set of planes be \( \Gamma(S) \). We define \( \Gamma_\perp(S) \) as the representation of a set of vectors, one along the normal for each plane, and \( \Gamma_\parallel(S) \) as the representation of a set of circular arrows, one around each plane normal. The freedoms of the set of planes span

\[ \Gamma_{\text{freedom}}(S) = \Gamma_\perp(S) + \Gamma(S) \times \Gamma_\parallel - \Gamma_\parallel(S). \] (9)

where the first term on the RHS represents the translational freedoms along the normal, and the remainder represents the pairs of rotational freedoms. Taking advantage of the fact that the characters \( \Gamma_\perp(S) \) and \( \Gamma_\parallel(S) \) are identical under proper operations but of opposite sign under improper operations, we can write

\[ \Gamma_\parallel(S) = \Gamma_\perp(S) \times \Gamma_\parallel \] (10)
Table 2: Calculation of $\Gamma_{\text{freedom}}(L)$ for a single line $L$ in 2D. Operations $\sigma_\parallel$ and $\sigma_\perp$ are, respectively, reflections in $L$ and in the normal to $L$. The table reports the calculation according to (6), and the check that (9) gives an identical result.

| $\mathcal{G} = C_{2v}$ | $E$ | $C_2$ | $\sigma_\parallel$ | $\sigma_\perp$ | $\sigma_\parallel \equiv \sigma_1$ | $\sigma_\perp \equiv \sigma_2$ |
|------------------------|-----|--------|-------------------|----------------|------------------|------------------|
| $\Gamma(T_x,T_y)$      | 2   | $-2$   | 0                 | 0              | 0                | 0                |
| $\Gamma(R_x)$          | 1   | $1$    | $-1$              | $-1$           | 0                | 0                |
| $\times \Gamma(L)$     | 3   | $-1$   | 1                 | 1              | 1                | 1                |
| $\Gamma(L)$            | 1   | $1$    | $-1$              | $-1$           | 0                | 0                |
| $\Gamma(L) \times \Gamma(R_x)$ | 1 | 1 | $-1$ | 1 | 0 | 0 |
| $\Rightarrow \Gamma_{\text{freedom}}(L)$ | 2 | 0 | $-2$ | 0 | |

Table 3: Calculation of $\Gamma_{\text{freedom}}(S)$ for a single plane $S$. Operations are as defined in Table 1 but now the principal axis lies along the normal to the plane.

| $\mathcal{G} = D_{\text{ch}}$ | $E$ | $2C_{\infty}(\phi)$ | $C_2$ | $\infty \sigma_\parallel$ | $\sigma_\perp$ | $2S_{\infty}(\phi)$ | $l$ | $\infty C_2$ |
|-------------------------------|-----|---------------------|--------|---------------------------|----------------|-------------------|----|------------|
| $\Gamma_L$                    | 3   | $1 + 2\cos \phi$   | $-1$   | $-1$                      | $1$            | $1 - 2\cos \phi$ | 3  | $-1$      |
| $+\Gamma_L$                   | 1   | $1$                 | $1$    | $1$                       | $1$            | $-1$              | $-1$ | $-1$      |
| $-\Gamma_L$                   | $-1$ | $-1$                | $1$    | $1$                       | $-1$           | $-1$              | $-1$ | $1$      |
| $\times \Gamma(S)$            | 3   | $1 + 2\cos \phi$   | $-1$   | $1$                       | $-3$           | $-1 - 2\cos \phi$ | 1  | $-1$      |
| $\Rightarrow \Gamma_{\text{freedom}}(S)$ | 3 | $1 + 2\cos \phi$ | $-1$ | $1$ | $-3$ | $-1 - 2\cos \phi$ | 1 | 1 |

and hence

$$3D: \quad \Gamma_{\text{freedom}}(S) = \Gamma(S) \times \Gamma_L \times \Gamma_{\perp} \times (\Gamma_0 - \Gamma_\parallel).$$

(11)

Table 3 gives the calculation for the maximum $D_{\text{ch}}$ symmetry, where $\Gamma_{\text{freedom}}(S)$ for a single plane spans $A_{1u} + E_{1g} \equiv \Sigma^+_0 + \Pi_3$.

Note that, in the purely rotational point groups that apply to chiral configurations, $\Gamma_\parallel = \Gamma_0$ and $\Gamma_L = \Gamma_{\perp}$, so that the representation of plane freedoms reduces in these groups to

$$3D \text{ chiral:} \quad \Gamma_{\text{freedom}}(S) = \Gamma(S) \times \Gamma_{\perp}.$$  

(12)

In a case where the planes define faces of a polyhedron, we have $\Gamma_{\perp}(S) = \Gamma(S)$ and, on invoking the spherical-shell decomposition into radial and tangential vector components of the product of a permutation representation and the translational representation (Quinn et al., 1984; Fowler & Quinn, 1986), (11) reduces to

$$\text{polyhedron:} \quad \Gamma_{\text{freedom}}(S) = \Gamma(S) \times \Gamma_L \times \Gamma_{\parallel} \times (\Gamma_0 - \Gamma_\parallel).$$

(13)

for any polyhedron, chiral or not. The final step follows from the invariance property of the $\pi$ representation ($\Gamma_\parallel \times \Gamma_\parallel = \Gamma_\parallel$). We will exploit (13) in an example later.

5. Constraints

Representations of the constraints are also straightforwardly derived, even if not always easy to describe compactly. In this section, we consider the representation of individual constraints in their maximum site symmetry. In the examples considered in later sections, site symmetries are typically lowered, and the representations appearing in our equations will be used with descent in symmetry to describe orbits of constraints of a given type.

A note on special configurations that we exclude from consideration is as follows: As a pragmatic decision, we rule out distance constraints between pairs of coincident points, lines, or planes, because this would cause difficulties with constraints that do not have a finite derivative at distance zero. We do, however, allow parallel constraints between geometrically coincident lines or planes,
although care must then be taken about how the actions of symmetry operations are defined (Schulze, 2010). We take the actions at coincidence to be the limit of the actions at small distance.

5.1. Constraints in 2D

We first consider the 2D case. In 2D, we have points and lines, and constraints are of the following four basic types:

(i) distances between points;
(ii) distances between points and lines;
(iii) distances between lines;
(iv) angles between lines. Each type of constraint is illustrated in Fig. 3 by the set of motions that it forbids, along with a table describing the character of the constraint under the symmetry operations that leave unshifted the points and lines involved.

A constraint of type (i) prevents changes in a pairwise distance between two points. As a scalar quantity, it transforms as a bar. The reducible representation spanned by all constraints of type D_pp is therefore the permutation representation of a set of notional bars. With our definition, we are implicitly excluding bars of zero length associated with coincidence of points.

A constraint of type (ii) prevents change in the signed perpendicular distance from a line to a point and transforms as a vector perpendicular to the line. The reducible representation spanned by constraints of type D_ll is that of the set of such vectors for line–point pairs. We use the signed distance because it behaves linearly even in the limit where the point lies on the line. In 2D, the constraint has the same dimension whether P is on or off the line L; this is no longer true in 3D.

A constraint of type (iii) prevents change in the pairwise distance between two lines, and in 2D is relevant only when the lines have already been constrained to be parallel (by an angular constraint A_{LL}). The additional distance constraint transforms as a bar. The reducible representation of a set of such constraints is the permutation representation of a set of notional bars.

A constraint of type (iv) prevents infinitesimal rotations between the two lines. We distinguish between cases where the lines meet, and where the lines are parallel. The representation for the full set of type (iv) constraints contains contributions from all those of types A_{LL} and A_{LL}..

5.2. Constraints in 3D

Similar reasoning applies to the symmetry description of the constraints in 3D. In 3D, the types are (i) to (iv), plus those involving (v) distance of a point to a plane, (vi) distance of a line to a plane, (vii) distance between planes, (viii) angle between a line and a plane, and (ix) angle between two planes. Each type of constraint is illustrated in Fig. 4 by the set of motions that it forbids, along with a table describing the character of the constraint in maximum possible local site symmetry.

In 3D, the constraint of type (i) is exactly as described in 2D, and again contributes to the permutation representation of the set of notional bars connecting constrained pairs.

In 3D, for constraints of type (ii) we distinguish two cases. If the point P is not on the line L, the constraint removes one freedom and transforms as a vector along the perpendicular from L to P, which is totally symmetric in the site group. If P lies on L, however, the constraint removes two freedoms and transforms as a pair of orthogonal vectors perpendicular to L. Similar doubling up at coincidence applies to line–plane (vii) and plane–plane angle (viii) constraints.

In 3D, for constraints of type (iii) we distinguish between lines that are parallel and lines that are not. If the lines are not parallel, the constraint is of dimension one, preventing changes in the pairwise distance between the lines, and transforms as a bar along the mutual perpendicular. The direction of the perpendicular remains defined even when the distance between the lines tends to zero, but we note that the case of intersecting lines does require more careful treatment. In that case, we are effectively considering the symmetry of a system of two lines, and their corresponding displacement vectors that point in opposite directions along the perpendicular. New symmetry operations that appear in the limit of zero separation may or may not exchange lines together with their displacements, and may or may not reverse the directions of those displacements. The combined effect leads to the characters seen in the tables.

For type (iii) where the lines are already constrained to be parallel, the distance constraint is of dimension one. We are implicitly excluding the case where the distance between two parallel lines is constrained to zero: In this case, the lines would become identical and a distance constraint of dimension two would be required.

In 3D, for constraints of type (iv) there is also a distinction between parallel and non-parallel cases. For non-parallel lines, there is a unique definition of the angle between lines defined in the plane normal to the mutual perpendicular, and the symmetry analysis follows a similar course to the 2D case. For constrained systems of type (iv), where the lines are parallel, the constraint is of dimension two. The two lines define a plane (as we are not considering coincident lines), and both in-plane and out-of-plane angular constraints apply.

For types (iii) and (iv), the characters for the case of non-zero separation can be derived by elimination of symmetry operations (descent in symmetry). For type (iii), the operations that disappear at finite separation are all those with character −1 in the higher symmetry group; the displacement that increases the separation from zero is fully symmetric in the lower group. For type (iv), this is not the case: Some of the symmetry operations that are retained have character −1; this is because the motion associated with separation is not the angular freedom that is forbidden by the constraint. Similar considerations apply to the constraints (v) to (viii) that involve planes.
6. Net Mobility

Collection of the partial results gives the total representation of the balance of freedoms and constraints. The final form of this representation is

$$\Gamma(m) - \Gamma(s) = \Gamma_{\text{freedom}} - \Gamma_{\text{constraint}} - \Gamma_{\text{rigid}}$$

(14)

where $\Gamma_{\text{freedom}}$ is defined by the sum of representations of point, line, and plane freedoms, and $\Gamma_{\text{constraint}}$ is defined by the sum over constraints of the representations of the freedoms removed by each type: $\Gamma(D_{PP}), \Gamma(D_{PL}), \Gamma(D_{PL}), \Gamma(D_{PLL}), \Gamma(D_{LL})$. 

---

Figure 3: Contribution to the constraint representation for single constraints of types (i)-(iv) in 2D. Arrows indicate the displacements away from the constrained geometry that are forbidden by the constraint. In each case, the maximum site symmetry is shown. To maintain compatibility between 2D and 3D notation, mirror lines are labeled $\sigma$, indicating that they have their normal along the $\alpha$-axis.
Type (i): Point-point distance constraint

Type (ii): Point-line distance constraint, point off the line

Point-line distance constraint, point on the line

Type (iii): Line-line distance constraint, lines in general position

Distance non-zero

Distance zero

Line-line distance constraint, lines perpendicular

Distance non-zero

Distance zero

Line-line distance constraint, lines parallel

Distance zero

Figure 4: Contribution to the constraint representation for single constraints of type (i)–(ix) in 3D. Arrows indicate the displacements away from the constrained geometry that are forbidden by the constraint. In each case, the maximum possible site symmetry is shown. Particular settings for the reduction $D_{\infty h} \rightarrow D_{2h}$ are given for types (vii) and (ix).
Applications of symmetry in point–line–plane frameworks for CAD

Figure 4: (Continued.)

Type (iv): Line-line angle constraint, lines in general position

\[
\begin{array}{c|cccc}
D_2 & E & C_{2x} & C_{2y} & C_{2z} \\
\Gamma(A_{LL}) & +1 & +1 & +1 & +1 \\
\end{array}
\]

Distance non-zero

\[
\begin{array}{c|cccc}
D_{2h} & E & C_{2x} & C_{2y} & i \\
\Gamma(A_{LL}) & +1 & +1 & +1 & +1 \\
\end{array}
\]

Distance zero

\[
\begin{array}{c|cccc}
\sigma_x, C_{2z} & E & 2C_2 & 2C_2' & 2S_4 \\
\Gamma(A_{LL}) & -1 & -1 & -1 & -1 \\
\end{array}
\]

Line-line angle constraint, lines perpendicular

\[
\begin{array}{c|cccc}
D_{2d} & E & 2C_4 & C_2 & 2C_2' \\
\Gamma(A_{LL}) & +1 & +1 & +1 & +1 \\
\end{array}
\]

Distance non-zero

\[
\begin{array}{c|cccc}
D_{2h} & E & 2C_4 & C_2 & 2C_2' \\
\Gamma(A_{LL}) & +1 & -1 & -1 & -1 \\
\end{array}
\]

Distance zero

\[
\begin{array}{c|cccc}
\sigma_x, C_{2z} & E & 2C_4 & C_2 & 2S_4 \\
\Gamma(A_{LL}) & -1 & -1 & -1 & -1 \\
\end{array}
\]

Line-line angle constraint, lines parallel

\[
\begin{array}{c|cccc}
D_{2h} & E & C_{2x} & C_{2y} & i \\
\Gamma(A_{LL}) & +1 & -1 & -1 & -1 \\
\end{array}
\]

Distance non-zero

\[
\begin{array}{c|cccc}
D_{2h} & E & C_{2x} & C_{2y} & i \\
\Gamma(A_{LL}) & -1 & -1 & -1 & -1 \\
\end{array}
\]

Distance zero

\[
\begin{array}{c|cccc}
\sigma_x, C_{2z} & E & 2S_4 & C_2 & 2S_4 \\
\Gamma(A_{LL}) & -1 & -1 & -1 & -1 \\
\end{array}
\]

Type (v): Point-plane distance constraint, point off plane

\[
\begin{array}{c|cc}
C_{\infty} & E & 2C_{\infty}(\phi) \\
\Gamma(D_{PS}) & +1 & +1 \\
\end{array}
\]

Point-plane distance constraint, point in plane

\[
\begin{array}{c|cc}
C_{\infty} & E & 2C_{\infty}(\phi) \\
\Gamma(D_{PS}) & +1 & +1 \\
\end{array}
\]

Type (vi): Line-plane distance constraint, line off the plane

\[
\begin{array}{c|cc}
C_{2y} & E & C_{2x} \\
\Gamma(D_{LS}) & +1 & +1 \\
\end{array}
\]

Line-plane distance constraint, line on the plane

\[
\begin{array}{c|cccc}
D_{2h} & E & C_{2x} & C_{2y} & i \\
\Gamma(D_{LS}) & +1 & -1 & -1 & -1 \\
\end{array}
\]
Type (vii): Line-plane angle constraint, general position

\[
\begin{array}{cccc}
C_{2h} & E & C_{2v} & i & \sigma_h \\
\Gamma(A_{LS}) & +1 & +1 & +1 & +1 \\
\end{array}
\]

Line-plane angle constraint, line in plane

\[
\begin{array}{cccccccc}
D_{2h} & E & C_{2v} & C_{2v} & C_{2v} & i & \sigma_z & \sigma_z & \sigma_y \\
\Gamma(A_{LS}) & +1 & -1 & +1 & -1 & +1 & -1 & +1 & -1 \\
\end{array}
\]

Line-plane angle constraint, line normal to plane

\[
\begin{array}{cccccccc}
D_{2h} & E & C_{2v} & C_{2v} & C_{2v} & i & \sigma_x & \sigma_y \\
\Gamma(A_{LS}) & +1 & -1 & +1 & -1 & +1 & -1 & +1 \\
\end{array}
\]

General setting:

\[
\begin{array}{cccccccc}
D_{\infty h} & E & 2C_{\infty} & C_2 & \infty \sigma_x & \sigma_h & 2S_{\infty}(\phi) & i & \infty C_2' \\
\Gamma(A_{LS}) & +2 & 2 \cos(\phi) & -2 & 0 & -2 & -2 \cos \phi & +2 & 0 \\
\end{array}
\]

Orthorhombic subgroup

\[
\begin{array}{cccccccc}
D_{2h} & E & C_{2v} & C_{2v} & C_{2v} & i & \sigma_z & \sigma_z & \sigma_y \\
yz-plane & +1 & -1 & +1 & -1 & +1 & -1 & +1 & -1 \\
xz-plane & +1 & -1 & -1 & +1 & -1 & +1 & -1 & +1 \\
\Gamma(A_{LS}) & +2 & -2 & 0 & 0 & +2 & -2 & 0 & 0 \\
\end{array}
\]

Type (viii): Plane-plane distance constraint

\[
\begin{array}{cccccccc}
D_{\infty h} & E & 2C_{\infty}(\phi) & C_2 & \infty \sigma_h & \sigma_h & 2S_{\infty}(\phi) & i & \infty C_2' \\
\Gamma(D_{SS}) & +1 & +1 & +1 & +1 & +1 & +1 & +1 \\
\end{array}
\]

Type (ix): Plane-plane angle constraint, general position

\[
\begin{array}{cccccccc}
D_{\infty h} & E & 2C_{\infty}(\phi) & C_2 & \infty \sigma_x & \sigma_h & 2S_{\infty}(\phi) & i & \infty C_2' \\
\Gamma(A_{SS}) & +2 & 2 \cos(\phi) & -2 & 0 & +2 & 2 \cos \phi & -2 & 0 \\
\end{array}
\]

Plane-plane angle constraint, planes parallel

\[
\begin{array}{cccccccc}
D_{\infty h} & E & 2C_{\infty}(\phi) & C_2 & \infty \sigma_x & \sigma_h & 2S_{\infty}(\phi) & i & \infty C_2' \\
\Gamma(A_{SS}) & +2 & 2 \cos(\phi) & -2 & 0 & +2 & 2 \cos \phi & -2 & 0 \\
\end{array}
\]

General setting:

\[
\begin{array}{cccccccc}
D_{\infty h} & E & 2C_{\infty}(\phi) & C_2 & \infty \sigma_x & \sigma_h & 2S_{\infty}(\phi) & i & \infty C_2' \\
\Gamma(A_{SS}) & +2 & 2 \cos(\phi) & -2 & 0 & +2 & 2 \cos \phi & -2 & 0 \\
\end{array}
\]

Orthorhombic subgroup

\[
\begin{array}{cccccccc}
D_{2h} & E & C_{2v} & C_{2v} & C_{2v} & i & \sigma_z & \sigma_z & \sigma_y \\
yz-plane & +1 & -1 & -1 & +1 & -1 & +1 & -1 & +1 \\
xz-plane & +1 & -1 & +1 & -1 & -1 & +1 & -1 & +1 \\
\Gamma(A_{SS}) & +2 & -2 & 0 & 0 & -2 & +2 & 0 & 0 \\
\end{array}
\]

Figure 4: (Continued.)
Applications of symmetry in point–line–plane frameworks for CAD

The mechanisms preserve are points or lines that are constrained to have a fixed relative angle with an angle constraint of type \(\gamma\). The constant representation, \(\Gamma_{\text{rigid}}\), is the representation of rigid-body motions in the space of the appropriate dimension. This general equation can now be applied to any particular case, using the once-and-for-all tables for freedoms and constraints, as derived in Sections 4 and 5.

7. Examples

We now consider some examples from 2D and 3D where symmetry yields extra information, and some telling examples where it does not.

7.1. 2D

7.1.1. Constrained crossing four-bar linkage

We begin with a 2D example based on the well-known four-bar linkage, expressed as a CAD problem. The relevant point–line system consists of four points, \(P_1\) to \(P_4\), connected in a cycle by four distances of type \(D_{P}\) corresponding to the four bars of the linkage. Two lines \(L_1\) and \(L_2\) are each constrained to pass through a pair of antipodal points by four constraints of type \(A_{LL}\). The number of constraints and freedoms is unchanged, but the linkage has a mechanism that is revealed when the symmetry-extended counting rule (2) is applied.

As shown in Fig. 5b, the underlying graph of geometries and constraints is a weighted complete bipartite graph \(K_{2,3}\), where vertices are points \(P\) or lines \(L\) and edges correspond to distance and angle constraints. If points, lines, point–point and point–line distance constraints, and line–line angular constraints are, respectively, counted by \(P\), \(L\), \(D_{P}\), \(D_{L}\), and \(A_{LL}\), the number of freedoms is \(F = 2P + 2L = 8 + 4 = 12\), and the number of constraints is \(C = D_{P} + D_{L} + A_{LL} = 4 + 4 + 1 = 9\), hence the mobility count is the isostatic value: \(m - s = F - C = 12 - 3 - 3 = 6\).

Considering the symmetry-extended rule (14) for 2D point–line systems, the three constraint types have the following characters under the symmetry operations of \(C_{2v}\):

\[
\begin{array}{cccccc}
G & = & C_{2v} & E & C_2 & \sigma_y & \sigma_x \\
\Gamma(D_{P}) & = & 4 & 2 & 0 & 2 & 2A_1 + 2A_2 + B_2 \\
\Gamma(D_{L}) & = & 4 & 0 & 0 & 0 & A_1 + A_2 + B_1 + B_2 \\
\Gamma(A_{LL}) & = & 1 & -1 & -1 & 1 & B_2 \\
\end{array}
\]

The irreducible representations in the final column of this table can be obtained from standard tables for representations in \(C_{2v}\) (Atkins et al., 1970). Summing the representations for the three constraint types,

\[
\Gamma_{\text{constraint}} = 3A_1 + 3A_2 + B_1 + 3B_2.
\]  

(15)

The freedoms of the points and lines, calculated from (4) and (6) span

\[
\Gamma_{\text{freedom}} = 3A_1 + 3A_2 + 3B_1 + 3B_2 = 3\Gamma_{\text{reg}}.
\]  

(16)

where \(\Gamma_{\text{reg}}\) is the regular representation of \(C_{2v}\) [with character \(\chi(E) = |G|\) and \(\chi(g) = 0\) otherwise]. Hence, with \(\Gamma_{\text{rigid}} = A_2 + B_1 + B_2\),

\[
\Gamma(m) - \Gamma(s) = B_1 - B_2.
\]  

(17)

revealing a mechanism of \(B_1\) symmetry, which is balanced and cancelled out in the scalar count by a state of self-stress of \(B_2\) symmetry. The mechanism preserves \(\sigma_y\) but destroys \(\sigma_x\), so that the initial \(C_{2v}\) point group descends to \(C_v\), in which \(\Gamma(m)\) is \(A'\) and \(\Gamma(s)\) is \(A''\), in
the infinitesimally distorted configuration. As the representations $\Gamma(m)$ and $\Gamma(s)$ do not share a symmetry ($A \neq A'$), the mechanism is not blocked by the state of self-stress; the initial symmetric configuration is therefore free to distort under the specified constraints. The state of self-stress detected by symmetry corresponds to the forbidden simultaneous lengthening and shortening of the bars $P_1 P_4$ and $P_2 P_3$. Numerical computation of the rank of the Jacobian matrix of the corresponding system of constraint equations shows that this is the only state of stress and hence that the mechanism is finite (Kangwai & Guest, 1999; Guest & Fowler, 2007).

If the angular constraint between the lines is removed, the $B_2$ self-stress is eliminated, leaving the $B_1$ mechanism unchanged. This shows that the angular constraint is locally redundant. The 2D DCM constraint solver that is described in Section 2 can be used to exhibit the path of this finite mechanism by replacing this angular constraint with a distance constraint between $P_1$ and $P_3$. The constraint graph is no longer three-connected and hence is quadratically solvable (Owen, 1991; Owen & Power, 2006). The constraint equations are easily solved for a sequence of small changes in the value for the new distance constraint to obtain a sequence of non-congruent configurations that satisfy all of the original constraints (including the locally redundant angular constraint). A similar method can be used to exhibit all of the finite mechanisms found in other examples in this paper.

If we were to add an extra constraint $D_{Ll}$ on the separation of the parallel lines, $\Gamma_{\text{constraint}}$ would contain an additional $A_1$ term, indicating the presence of an extra state of self-stress of $A_1$ symmetry. On descent to $C_2$, $\Gamma(m) = \Gamma(s)$ would reduce to $-A'$, indicating that the $A$ mechanism has become symmetry undetectable. The mechanism detected at the high-symmetry point is now infinitesimal, blocked by a fully symmetric state of self-stress in which the vertical bars $P_1 P_4$ and $P_2 P_3$ are in an equal state of tension that is balanced by the compression carried by the $D_{Ll}$ constraint.

This example also explains why each of the drawings in Fig. 1a has a redundant dimension. If we add a parallel constraint between the $x$-axis and the horizontal construction line, both the left and right subdiagrams are isomorphic to Fig. 5a. This shows that they both induce the parallel constraint and that without the additional constraint any one of the distant dimensions in the subdiagrams is redundant. This argument does not require a symmetry between the left and right subdiagrams, which explains why the lower drawing in Fig. 1b also has a redundant dimension. The corresponding subdiagrams in the upper drawing of Fig. 1b have no symmetry but the whole drawing has $C_2$ reflection symmetry about the $y$-axis. We now show that the overall $C_2$ symmetry alone does not predict that the drawing is not well dimensioned.

We use the procedure for counting residual freedoms that is used by 2D DCM, described in Section 2. The geometry comprises eight points and eleven lines. Two of the lines correspond to the $x$- and $y$-axes and one corresponds to the construction line. These lines are all unshifted by the reflection symmetry. The remaining eight lines that correspond to the displayed line segments, and all the points, are shifted by the reflection. There are 24 point-line coincident constraints and 2 point-line distance dimensions that are all type $D_{PP}$ and 8 point-point distance dimensions that are all type $D_{PP}$. These are all shifted by the reflection. There is also a perpendicular constraint, type $A_{LL}$, between the lines that correspond to the $x$- and $y$-axes. This constraint is unshifted by the reflection symmetry. The character table calculation below shows the full cancellation of the characters for the identity and the reflection operations.

| $\mathcal{G} = C_2$ | $E$ | $\sigma$ | Representation |
|---------------------|-----|----------|----------------|
| $-\Gamma(D_{PP})$   | $-8$ | $0$      | $-4A' - 4A$    |
| $-\Gamma(D_{LL})$   | $-26$ | $0$      | $-13A' - 13A$  |
| $-\Gamma(A_{LL})$   | $-1$  | $1$      | $-A$           |
| $-\Gamma_{\text{constraint}}$ | $-35$ | $1$      | $17A' - 17A$   |
| $-\Gamma_P - \Gamma_R$ | $-3$  | $1$      | $-A - 2A$     |
| $-\Gamma(P)$        | $16$  | $0$      | $8A + 8A'$     |
| $-\Gamma(L)$        | $22$  | $-2$     | $-10A + 12A'$  |
| $-\Gamma_{\text{freedom}}$ | $38$  | $-2$     | $18A + 20A'$   |
| $\Gamma(m) - \Gamma(s)$ | $0$   | $0$      | $\emptyset$    |

This complete cancellation implies that there is no reason arising from the overall reflection symmetry for the design not to be well dimensioned. Numerical calculation of the Jacobian matrix of the constraint equations in 2D DCM shows, as expected, that $m = 0$ and $s = 0$ for the configuration shown in the upper panel of Fig. 1b, but $m = 1$ and $s = 1$ for the other three configurations.

7.1.2. Non-crossing four-bar linkage
Fig. 5c shows an alternative configuration for the four-bar linkage, in which the bars do not cross but instead form the four sides of a quadrilateral. The two lines are diagonals of the quadrilateral and are connected by an angular constraint. This linkage is generically isostatic.

In the high-symmetry special configuration shown in the figure, where the four points are on the corners of a square, the sets of points, lines, and constraints have $C_{4v}$ symmetry where the two mirror planes $\sigma_x$ each contains one of the lines. The corresponding graph of geometries and constraints is still $K_{3,3}$, as for the parallel-constrained linkage (Fig. 5b), but now tick marked to indicate equality of all four black edges. The number of freedoms and constraints is the same as in the previous example.

In this configuration, the three constraint types have the following characters under the symmetry operations of $C_{4v}$:
Applications of symmetry in point–line–plane frameworks for CAD

| \( G = C_{4v} \) | E | 2C\(_4\) | C\(_2\) | \( 2\sigma_v \) | \( 2\sigma_d \) | Representation |
|----------------|---|---------|------|----------|----------|----------------|
| \( \Gamma(D_{2h}) \) | 4 | 0       | 0    | 0        | 2        | \( A_1 + B_2 + E \) |
| \( \Gamma(D_{1h}) \) | 4 | 0       | 0    | −2       | 0        | \( A_2 + B_2 + E \) |
| \( \Gamma(A_{1h}) \) | 1 | −1      | 1    | −1       | 1        | \( B_2 \) |

Summing the representations for the three constraint types gives

\[
\Gamma_{\text{constraint}} = A_1 + A_2 + 3B_2 + 2E. \tag{18}
\]

The freedoms of the points and lines span \( A_1 + A_2 + B_1 + B_2 + 2E \) and \( A_2 + B_2 + E \), respectively. Hence,

\[
\Gamma_{\text{freedom}} = A_1 + 2A_2 + B_1 + 2B_2 + 3E. \tag{19}
\]

and, with \( \Gamma_{\text{rigid}} = A_2 + E \), the net mobility is

\[
\Gamma(m) - \Gamma(s) = B_1 - B_2. \tag{20}
\]

revealing a mechanism of \( B_1 \) symmetry, balanced in the scalar count by a state of self-stress of \( B_2 \) symmetry. The mechanism is not blocked by the state of self-stress and is finite. The removal of the \( A_{1h} \) constraint has eliminated the \( B_2 \) self-stress without affecting the mechanism with \( B_1 \) symmetry. Hence, the angular constraint is locally redundant, which also follows from the well-known theorem in geometry that the two diagonals of a square are perpendicular. Deletion of this constraint means that the constraint graph is no longer three-connected and the constraint equations are then recognized as quadratically solvable.

From the entries for \( B_1 \) in the character table [with \( \chi(C_4) = -1, \chi(\sigma_d) = -1, \chi(R) = +1 \) otherwise], it follows that a mechanism of this symmetry preserves the two \( \sigma_v \) reflection planes, removes \( \sigma_d \), and distorts the square into a rhombus with \( C_{2v} \) symmetry, where the three constraint types then have the following characters under the symmetry operations:

| \( G = C_{4v} \) | E | \( C_2 \) | \( \sigma_v \) | \( \sigma_d \) | Representation |
|----------------|---|---------|------|----------|----------------|
| \( \Gamma(D_{2h}) \) | 4 | 0       | 0    | 0        | \( A_1 + A_2 + B_1 + B_2 \) |
| \( \Gamma(D_{1h}) \) | 4 | 0       | 0    | −2       | \( 2A_2 + B_1 + B_2 \) |
| \( \Gamma(A_{1h}) \) | 1 | 1       | 1    | −1       | \( A_2 \) |

Summing the representations for the three constraint types gives

\[
\Gamma_{\text{constraint}} = A_1 + 4A_2 + 2B_1 + 2B_2. \tag{21}
\]

The freedoms of the points span \( 2(A_1 + A_2 + B_1 + B_2) \) in \( C_{2h} \) and those of the lines \( 2A_2 + B_1 + B_2 \). Hence,

\[
\Gamma_{\text{freedom}} = 2A_1 + 3A_2 + 2B_1 + 2B_2. \tag{22}
\]

and, with \( \Gamma_{\text{rigid}} = A_2 + B_1 + B_2 \), the net mobility is

\[
\Gamma(m) - \Gamma(s) = A_1 - A_2. \tag{23}
\]

revealing a mechanism of \( A_1 \) symmetry, which is balanced in the scalar count by a state of self-stress of \( A_2 \) symmetry. The mechanism is not blocked by the state of self-stress and is finite. The removal of the \( A_{1h} \) constraint eliminates the self-stress of \( A_2 \) symmetry without affecting the \( A_1 \) mechanism. Hence, the angular constraint remains locally redundant. This reflects the well-known theorem of geometry that the two diagonals of a rhombus are always perpendicular.

The addition of a distance constraint between two diagonally opposite points in the rhombus adds an extra term \( A_1 \) to \( \Gamma_{\text{constraint}} \) and hence destroys the mechanism with \( A_1 \) symmetry without changing the symmetry of the \( A_2 \) state of self-stress, to give \( \Gamma(m) - \Gamma(s) = -A_2 \).

7.2 3D

7.2.1 Non-crossing four-bar linkage in 3D

We can rework the preceding example in 3D by constraining the four points to lie on a plane. The group of symmetries, \( D_{4h} \), now includes a reflection in the plane and four half-turn rotations around \( \sigma_v \) and \( \sigma_d \) axes. There is an additional plane symmetry element and four additional point–plane coincidence constraints. The tabular character calculation is shown below. Note the conventional reversal of labelings between \( \sigma_v \) and \( \sigma_d \) mirror lines in 2D and corresponding mirror planes in 3D (see Figs 3 and 4).
This calculation shows that \( \Gamma(m) - \Gamma(s) = B_{2g} - B_{1g} \). The \( B_{2g} \) mechanism is finite as it cannot be blocked by a \( B_{1g} \) stress. The angular constraint has \( \Gamma(A_{1L}) = B_{1g} \) and is locally redundant.

An alternative way to force the quadrilateral to be planar is to constrain the two lines to intersect in three-space. This is achieved by deleting the plane and the four point–plane constraints and adding a distance-zero constraint between the two lines. As the table also shows, the characters for \( \Gamma(D_{2L}) \) and \( \Gamma(D_{4L}) \) are equal (irreducible representation \( B_{1L} \)).

If we now delete the \( D_{4L} \) constraint, we get an additional finite mechanism with \( B_{1L} \) symmetry that takes the four points out of coplanarity into a configuration with \( D_{2L} \) symmetry where the axis of symmetry is along the line that is perpendicular to both lines (see Fig. 4ii for the case of a line–line distance constraint with the lines perpendicular and non-zero distance).

After restoring the line–line distance constraint, the row for \( \Gamma(m) - \Gamma(s) \) in the character table has entries 0, 0, –2, 0, and 2 in the columns for \( E \), \( C_{2} \), \( 2C_{1} \), \( 2S_{4} \), and \( 2\sigma_{d} \), respectively. This shows that \( \Gamma(m) - \Gamma(s) = B_{2} - B_{1} \); in \( D_{2d} \) symmetry and we have a finite mechanism with \( B_{2} \) symmetry and a stress with \( B_{1} \) symmetry. The line–line angle constraint has \( \Gamma(A_{1L}) = B_{1} \) and is again redundant. This reflects the geometric theorem that the diagonals of an equilateral quadrilateral are perpendicular even if the points are not coplanar and the diagonals do not intersect.

In fact, all of these properties for non-crossing four-bar mechanisms reflect an underlying rigidity property that any polygon of bars and joints in 3D, where the joints are constrained alternately to lie on a pair of orthogonal lines (which need not intersect), is dependent. For a four-bar mechanism in the plane, this follows from the properties of an orthodiagonal quadrilateral. The extension to a \( (2n) \)-bar mechanism in 3D, where the lines need not intersect, is straightforward. Our symmetry results demonstrate symmetric projections of the general result.

7.2.2. Examples based on the dodecahedron

The following four examples are based on the regular dodecahedron, chosen as a moderately complicated three-dimensional structure of points, lines, and planes. A dodecahedral configuration has 20 points representing the vertices, 12 planes representing the faces, and 30 lines representing the edges. The appropriate point symmetry group of the regular Platonic dodecahedron is \( \text{Ih} \), consisting of 120 operations based on the 60 rotations about \( C_{5} \), \( C_{3} \), and \( C_{2} \) axes through opposite face centers, vertices, and edge midpoints, compounded with inversion symmetry to generate the 60 improper operations of the group, including 15 reflection planes.

The four examples that follow explore the consequences for mobility of changing various constraints on the geometries. Note that there is an exact cancellation between the characters for the freedoms of a line (Table 1) and those for two \( D_{4h} \) constraints (Fig. 4) for any setting in which the line passes through two distinct points (this also follows from the fact that there is a unique line through two distinct points). Hence, we omit these characters from the tables.

Example (a): Consider a dodecahedron as a set of 20 points arranged on congruent planar faces with 30 edges of equal length, all in full \( \text{Ih} \) symmetry. Is an assembly of geometries with these constraints rigid or flexible? Scalar counting gives exact cancellation of 96 freedoms by 90 constraints and 6 rigid-body motions. The dodecahedron is illustrated in Fig. 6a. A full tabular calculation using all 10 classes of symmetry operations of \( \text{Ih} \) to classify the symmetries of the freedoms of points and planes, together with those of the 30 constraints on point–point distances and the 60 constraints on point–plane distances, gives (with \( \phi = (\sqrt{5} + 1)/2 \):
Applications of symmetry in point–line–plane frameworks for CAD

Figure 6: Constraints on a dodecahedron. In (a) all faces are planar and all edges are of the same fixed length. In the other panels, distance constraints on two opposite edges (dotted) are replaced by (b) fixed chord lengths on one face, (c) point–line distance constraints to one of the missing edges, and (d) parallel constraints on the pair of missing edges, respectively.

| \( \tilde{\gamma} = I_h \) | \( E \) | \( 12C_5 \) | \( 12C_5^2 \) | \( 20C_3 \) | \( 15C_2 \) | \( i \) | \( 12S_{10} \) | \( 12S_{10}^T \) | \( 20S_{10} \) | \( 15\sigma \) |
|-----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| \( \Gamma(P) \times \Gamma_T \) | 20  | 0   | 0   | 2   | 0   | 0   | 0   | 0   | 0   | 4   |
| \( \Gamma(S) \times \Gamma_R \) | 60  | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 4   |
| \( \Gamma(m) - \Gamma(s) \) | 96  | 2\phi | -2\phi^{-1} | 0   | 0   | 0   | 0   | 0   | 0   | 4   |

Note the exact cancellation of intrinsic point freedoms and point–plane constraints that arises from the symmetry equivalence in this case between \( \Gamma(D_{5h}) \) and \( \Gamma(P) \times \Gamma_T \). This also follows from the fact that there is a unique intersection point for three pairwise non-parallel planes. The final result that \( \Gamma(m) - \Gamma(s) \) spans the null representation implies that the system is either rigid or has mechanisms masked by equisymmetric states of self-stress. Numerical computation of the rank of the Jacobian matrix of the corresponding constraint equations confirms the structure as rigid.

**Example (b):** In the configuration illustrated in Fig. 6b, two distance constraints on antipodal edges have been replaced by chordal distance constraints on a face, preserving the isostatic property. There are two distinct ways to place the chords to be compatible with a reduced point-group symmetry of \( C_5 \), where only the columns for the identity \( E \) and a single mirror plane \( \sigma \) are retained from the earlier tabular calculation. The representations of freedoms and rigid-body motions from the table for the icosahedrally symmetric configuration (a) reduce to \( \{ \chi(E), \chi(\sigma) \} = \{ 96, 8 \} \) (freedoms), \( \{ 3, 0 \| \Gamma_T + \Gamma_R \} \), and the constraint representation \( \{ 90, 8 \} \) is changed by deletion of two edges \( \{-2, -2\} \) and addition of chords that exchange under reflection \( \{ \pm2, 0 \} \). Hence, the net mobility representation becomes \( \{ 0, 2 \} = \{ 1, 1 \} - \{ 1, -1 \} \), corresponding to reducible representation \( \Gamma(m) - \Gamma(s) = A - A' \). The prediction is of a mirror-symmetric breathing flex and an antisymmetric state of self-stress. Numerical computation of the rank of the Jacobian matrix of the corresponding constraint equations confirms that there are no additional states of stress and hence that the flex is indeed continuous (Kangwai & Guest, 1999; Guest & Fowler, 2007).

**Example (c):** In the configuration illustrated in Fig. 6c, two edge distance constraints have been replaced by a total of two point–line distance constraints, to give a configuration with point group \( C_{2v} \). We take the mirror planes preserving and exchanging the point–line constraints as \( \sigma_x \) and \( \sigma_y \), respectively. The calculation is then as follows:

| \( \tilde{\gamma} = C_{2v} \) | \( E \) | \( C_2 \) | \( \sigma_y \) | \( \sigma_x \) |
|-----------------|-----|-----|-----|-----|
| \( \Gamma(m) - \Gamma(s) \) | 96  | 0   | 8   | 8   |
| \( \Gamma(m) - \Gamma(s) \) | 90  | 0   | -6  | -8  |
| \( \Gamma(m) - \Gamma(s) \) | -6  | 2   | 0   | 0   |

and, as a reducible representation, \( \Gamma(m) - \Gamma(s) = \{ 1, 1, 1, 1 \} - \{ 1, -1, -1, 1 \} = A_1 - B_1 \), indicating a fully symmetric mechanism and a state of self-stress with the symmetry of a tangential vector parallel to the missing edges. Again, detection of a totally symmetric...
where we have used (13) to express the face freedoms of the polyhedron as $\Gamma(\text{S}) \times \Gamma(\text{T})$. The result in the final line is that $\Gamma(\text{m}) - \Gamma(\text{s})$ spans the null representation, which implies that the configuration is either rigid or has mechanisms masked by equisymmetric states of self-stress. The latter is in fact the case, as it is straightforward to show (see below) that this configuration has a set of three
symmetry-equivalent finite mechanisms in which pairs of opposite faces flex into congruent rhombi. In the initial high symmetry of the cube, this set corresponds to the representation $T_{2g}$ in $O_h$, with pure distortion along any one of the three orthogonal modes leading to $D_{3h}$ symmetry. An arbitrary mixture of all three modes reduces the point group to $C_i$, preserving only inversion symmetry.

The space of finite distortions can be characterized as follows. Label the points $p_1$, $p_2$, $p_3$, and $p_4$ in cyclic order on one face and corresponding points $p_5$, $p_6$, $p_7$, and $p_8$ on the opposite face, hence with pairs $\{p_i, p_{i+1}\}$ related by inversion. We assume that all edges have length 1. Let $\hat{b}(\theta, \phi) = \sin \theta (\cos \phi \hat{x} + \sin \phi \hat{y}) + \cos \phi \hat{z}$ be a unit vector, and $\hat{a}(\chi) = \cos \chi \hat{x} + \sin \chi \hat{y}$ be a unit vector in the xy plane. Let $p_1 = (0, 0, 0)$, $p_2 = (1, 0, 0)$, $p_3 = p_2 + \hat{a}(\chi)$, $p_4 = p_1 + \hat{a}(\chi)$, $p_5 = p_1 + \hat{b}(\theta, \phi)$, $p_6 = p_2 + \hat{b}(\theta, \phi)$, $p_7 = p_3 + \hat{b}(\theta, \phi)$, and $p_8 = p_4 + \hat{b}(\theta, \phi)$. Then, $|p_12| = |p_{34}| = |p_{45}| = |p_{56}| = |p_{67}| = |p_{78}| = |p_{81}| = 1$. Points $\{p_1, p_2, p_3, p_4\}$ are coplanar because $p_{12}$ is parallel to $p_{34}$. Similarly, sets $\{p_5, p_6, p_7, p_8\}$, $\{p_1, p_4, p_5, p_8\}$, $\{p_2, p_3, p_5, p_7\}$, $\{p_1, p_2, p_3, p_4\}$, and $\{p_1, p_3, p_4, p_8\}$ are each coplanar. The three infinitesimal flexes that correspond to changes in $\chi$, $\theta$, and $\phi$ are independent unless $\theta$ or $\phi$ are 0 or $\pi$, respectively. Hence, there are three independent finite mechanisms except at special points.

Figure 7 illustrates one realization of the set of mechanisms and their matching states of self-stress for the floppy cube. The degenerate irreducible representation $T_{2g}$ is spanned by the set of Cartesian spherical harmonic functions $\{xy, yz, zx\}$, and these can be used to project out a set of three independent flexes and corresponding states of self-stress. Flexes have displacements in the planes of the maxima in the Cartesian function; thus, four cube faces rotate and two flex to make a rhombus. Each pattern of stresses involves four $D_{3h}$ constraints on the edges normal to the nominated Cartesian plane, with an alternating cyclic pattern $\{+s, -s, +s, -s\}$.

Point groups $O_h$, $D_{3h}$, $D_{3d}$, $C_{2h}$, and $C_i$ are accessible in this distortion space (Fig. 8). (See Guest & Fowler, 2007 and also Jotham & Kettle, 1971, where the same descent in symmetry is discussed for a problem in chemistry related to Jahn–Teller distortion in octahedral complexes.)

8. Mobility Predictions for Convex Polyhedra

Simple counting of freedoms and constraints gives only the net mobility $m - s$ and no conditions on the individual values of $m$ and $s$. The symmetry-extended approach gives, in effect, a further set of necessary conditions on $m$ and $s$, through determination of the difference $\Gamma'(m) - \Gamma'(s)$, but again is not guaranteed to give full information on the separate representations $\Gamma'(m)$ and $\Gamma'(s)$. Symmetry often gives some added information through the pattern of signs in the reducible representation, giving partial but unambiguous contributions to $\Gamma'(m)$ and $\Gamma'(s)$. To take a concrete example, a count $m - s = 0$ shows that $m$ and $s$ are equal but not that they vanish individually. By the same token, if computation leads to the conclusion that $\Gamma'(m) - \Gamma'(s)$ is the null representation, we know only that $m - s$ and that the sets of mechanisms and states of self-stress are equisymmetric.
3D examples (a)–(d) include two cases where a convex polyhedron under the sole constraints of fixed edge lengths and planarity of faces turns out to have the null representation for \( \Gamma(m) - \Gamma(s) \). We can generalize this result to show that the mobility of any convex polyhedron under these constraints will span the null representation, \( \Gamma(m) - \Gamma(s) = 0 \).

The proof follows a method used for deriving the symmetry extension of the Euler theorem (Ceulemans & Fowler, 1991). Let \( \Gamma(F) \), \( \Gamma(E) \), and \( \Gamma(V) \) be the reducible representations for the permutations induced by symmetry elements of the group \( \mathcal{G} \) on the sets of vertices, edges, and faces of a polyhedron, with characters \( \chi_F(R), \chi_E(R), \) and \( \chi_V(R) \) equal to the numbers of components of each type that are unshifted under operation \( R \in \mathcal{G} \).

The case of the identity operation, \( R = e \), is that of pure scalar counting. Suppose the polyhedron has \( f \) faces, \( e \) edges, and \( v \) vertices, with face \( i \) having \( n_i \) vertices. There are \( n_i \) constraints of type \( DPF \) on the plane of face \( i \) and \( n_i \) edges on face \( i \). We have \( \Sigma n_i = 2e \), as each edge is on two faces. Hence, we have in total \( 3(f + v - e - 2) = F - C - 6 \) freedoms, \( 2e \) constraints \( DPF \), and \( e \) constraints \( DPF \) for our polyhedron with fixed-length edges and planar faces, and thus,

\[
F - C - 6 = 3(f + v - e - 2) = 0. \tag{24}
\]

by Euler's theorem.

Next, consider a rotation through a non-zero angle \( \phi \), i.e. the proper rotation \( C_\phi \). We have \( \Gamma_{\text{freedom}} = [\Gamma(F) + \Gamma(S)] \times \Gamma_T \). The axis of rotation must pass through two structural elements of the polyhedron (vertex + vertex, vertex + edge midpoint, vertex + face center, ..., face center + face center). It may pass through an edge center if \( \phi = \pi \). Let the number of faces, edges, and vertices intersected by a given \( C_\phi \) axis be \( f_\phi \), \( e_\phi \), and \( v_\phi \), respectively, with \( f_\phi + e_\phi + v_\phi = 2 \), and let the character of the rigid-body translations be \( \chi_T(C_\phi) \) [with explicit formula \( \chi_T(C_\phi) = 1 + 2\cos \phi \) and hence \( \chi_T(C_\phi) = -1 \) for \( \phi = \pi \)]. Then, for the character of the freedoms we have \( \chi_{TFS}(C_\phi) = (f_\phi + v_\phi)\chi_T(C_\phi) \). For the edge constraints, we have \( \chi_{FF}(C_\phi) = e_\phi = -e_\phi\chi_T(C_\phi) \) as either \( \phi = \pi \) or \( e_\phi = 0 \), or both. For the planarity constraints, we have \( \chi_{FS} = 0 \), as no vertex is at the center of a face. Rigid-body motions give \( \chi_T(C_\phi) + \chi_R(C_\phi) = 2\chi_T(C_\phi) \). Hence, in total we have

\[
\chi_{m - i}(C_\phi) = (f_\phi + v_\phi)\chi_T(C_\phi) + e_\phi \times \chi_T(C_\phi) - 2\chi_T(C_\phi) = 0. \tag{25}
\]

All faces, edges, and vertices are shifted by inversion and rotation–reflections, so it remains only to consider a pure reflection, \( R = \sigma \). A mirror plane intersects the shell in a loop of linked intersection elements of at most four types (Ceulemans & Fowler, 1991). These are shown as (i) to (iv) in Fig. 9. Counting contributions, using half-weighting for edges or vertices shared between two intersected elements, gives

| Character       | (i) | (ii) | (iii) | (iv) |
|-----------------|-----|------|-------|------|
| \( \chi_T(\sigma) \) | 1   | 1    | 1     | 0    |
| \( \chi_E(\sigma) \) | 0   | \( \frac{1}{2} \) | \( \frac{1}{2} + \frac{1}{2} \) | \( \frac{1}{2} + \frac{1}{2} \) |
| \( -\chi_{00}(\sigma) \) | \( -\frac{1}{2} + \frac{1}{2} \) | \( -\frac{1}{2} \) | 0 | -1 |
| \( -\chi_{00}(\sigma) \) | 0   | -1   | -2    | 0    |
| \( -\chi_T(\sigma) - \chi_R(\sigma) \) | 0   | 0    | 0     | 0    |

which shows that \( \chi_{m - i}(\sigma) \) sums to zero over the whole loop for every reflection plane. Hence, \( \Gamma(m) - \Gamma(s) = 0 \) for all polyhedra subject only to full edge-length and face-planarity constraints.

As we have seen, vanishing of the mobility representation does not exclude the possibility of undetected mechanisms. The example of the cube suggests an infinite family of polyhedra with such mechanisms: Any polyhedron constructed by extrusion of a polygon, i.e. formed by joining corresponding vertices of two parallel congruent copies of polygons, will have all those mechanisms that derive from in-phase combinations of the 2D mechanisms of the parallel polygons (if the polygons are of size greater than 3). Prisms form a subclass of extruded polyhedra. For the cube, the extrusion can be considered to have happened in any one of three independent directions, hence the threefold nature of the mechanism.
9. A Note on Freedoms of a Cubic Polyhedral Cage

The general symmetry-extended treatment can often be taken further in specific situations. One such specialization, which is of interest in several contexts, from CAD to structural mechanics and molecular force fields in physical chemistry, is based on the family of cubic polyhedral cages. In a CAD context, a polyhedron can be considered as an assemblage of point, line, and plane geometries. The polyhedral cages have rings of vertices and edges in place of solid faces, and these rings are not necessarily planar. Here, we concentrate on the freedoms of edges and vertices. From (3), (5), and (11), replacing $P$ and $L$ by $v$ and $e$, respectively, these are as follows:

- **vertices**: $\Gamma(v) \times \Gamma_T$.
- **edges**: $\Gamma(e) \times (\Gamma_T + \Gamma_k) - \Gamma_1(e) \times (\Gamma_0 + \Gamma_v)$.

For the object as a whole, rigid-body motions are accounted for by subtraction of one copy of $(\Gamma_T + \Gamma_k)$.

The sets of freedoms are not independent for a polyhedral object, as the Euler theorem and its symmetry-adapted counterpart apply, with significant consequences, as illustrated here for cubic polyhedra. We can give physical interpretations to the freedom equations and their differences. One such difference is relevant to the construction of force fields for cubic polyhedral molecular cages.

A set of edges, given full line freedoms, will simply drift apart. The representation $\Gamma_{\text{freedom}}(e)$ must be reduced by some constraint $\Gamma_e$ to produce $\Gamma(v) \times \Gamma_T$, the mechanical freedoms of the polyhedral cage. The constraint is that at each vertex of a cubic polyhedron the three incident edges must have a common intersection and therefore must satisfy three pairwise intersection constraints of type $D_{\perp\perp}$. We can show that these constraints correspond to a rotational triple at each vertex, and hence span

$$\Gamma_{ev} = \Gamma_{\text{freedom}}(e) - \Gamma_{\text{freedom}}(v) = \Gamma(v) \times \Gamma_R = \Gamma(v) \times \Gamma_T \times \Gamma_v.$$  \hspace{1cm} (28)

The proof is straightforward. Expansion of $\Gamma_{ev}$ using (5) and (3) gives

$$\Gamma_{ev} = \Gamma(e) \times (\Gamma_T + \Gamma_k) - \Gamma_1(e) \times (\Gamma_0 + \Gamma_v) - \Gamma(v) \times \Gamma_T,$$

and this can be simplified using the identities

- $\Gamma(e) \times \Gamma_T = \Gamma(e) + \Gamma_1(e) + \Gamma_0\Gamma_v$.
- $\Gamma_1(e) = \Gamma_\perp(e) \times \Gamma_T$.

and a specific consequence for cubic polyhedra of the symmetry-extended Euler theorem (Ceulemans & Fowler, 1991),

$$\Gamma(v) \times \Gamma_T = \Gamma(e) + \Gamma_1(e).$$

(32)

to give $\Gamma_{ev}$ as a function of edge representations, which collapses down to

$$\Gamma_{ev} = \Gamma(e) \times \Gamma_T + \Gamma_\perp(e) = \Gamma(v) \times \Gamma_T \times \Gamma_v = \Gamma(v) \times \Gamma_R.$$  \hspace{1cm} (33)

Hence, (28) is proved.

The combination $\Gamma(v) \times \Gamma_R$ also arises in other contexts. For example, in Guest et al. (2018) it appears as the symmetry spanned by extra states of self-stress in a panel–hinge description of the net mobility of a fully triangulated polyhedron (“extra” compared to the isostatic bar-and-joint model). As also in the present application, the key point is that hinge lines (edges) associated with panels (or faces) around each polyhedral vertex meet at that common vertex.

Identity (32) expresses the underappreciated fact that the vibrational motions of a cubic polyhedral molecular framework can be expressed as a combination of bond stretches and “bond slides” only, giving a basis for a novel type of force field for polyhedral cage molecules (Ceulemans et al., 2001).

10. Conclusions

A general symmetry extension of counting rules for CAD systems has been derived. Symmetry extensions of counting arguments are useful in many specific cases for deciding local redundancy of constraints, residual freedoms of the geometries, and the quadratic solvability of constraint equations. They also lead to general results for families of CAD systems, including, surprisingly, some where we can show that it is impossible to detect mechanisms by point-group-symmetry counting alone.

We finish with two further observations about the scope for the use of symmetry in CAD, as discussed here. The first is an intrinsic limitation of all symmetry-based treatments. Our premise for introducing symmetry was that a pure counting argument gives a necessary but not sufficient criterion for establishing residual freedoms in a CAD drawing. Consideration of symmetry gives a hierarchy of necessary but possibly still not sufficient criteria. Hidden flexes/mechanisms may be revealed, but sometimes cancellation hides further layers of detail, as e.g. when there is a geometric invariant of the kind found for the Stewart platform.

The second observation is more positive, in that although we have considered only point-group symmetry and rather simple objects here, there are further possibilities for symmetry-based explanations. The choice of objects can be made richer: spheres for points, cylinders for lines, and laminae for planes. More exotic surfaces can be considered, and the notion of symmetry can also be widened, to include other conserved invariants that generate an abstract group and hence open up possibilities for more symmetry-extended counting arguments (Olver, 1999).
Acknowledgements

The authors thank the referees for a number of useful comments on presentation for the CAD audience of the ideas in this paper.

Conflict of interest statement

None declared.

References

Altmann, S. L., & Herzig, P. (1994). Point-group theory tables. Clarendon Press.
Atkins, P. W., Child, M. S., & Phillips, C. S. G. (1970). Tables for group theory. Oxford University Press.
Barton, M., Shragai, N., & Elber, G. (2009). Kinematic simulation of planar and spatial mechanisms using a polynomial constraints solver. Computer-Aided Design and Applications, 6(1), 115–123.
Bishop, D. M. (1973). Group theory and chemistry. Clarendon Press.
Bouma, W., Fudos, I., Hoffmann, C., Cai, J., & Paige, R. (1995). Geometric constraint solver. Computer-Aided Design, 27(6), 487–501.
Calladine, C. R. (1978). Buckminster Fuller’s “Tensegity” structures and Clerk Maxwell’s rules for the construction of stiff frames. International Journal of Solids and Structures, 14, 161–172.
Ceulemans, A., & Fowler, P. W. (1991). Extension of Euler’s theorem to the symmetry properties of polyhedra. Nature, 353, 52–54.
Ceulemans, A., Titeca, B. C., Chibotaru, L. F., Vos, I., & Fowler, P. W. (2001). Complete bond force fields for trivalent and deltahedral cages: Group theory and applications to cubane, closo-dodecaborane and buckminsterfullerene. Journal of Physical Chemistry A, 105, 8284–8295.
Fowler, P. W., & Guest, S. D. (2000). A symmetry extension of Maxwell’s rule for rigidity of frames. International Journal of Solids and Structures, 37, 1793–1804.
Fowler, P. W., & Guest, S. D. (2005). A symmetry analysis of mechanisms in rotating rings of tetrahedra. Proceedings of the Royal Society: Mathematical, Physical & Engineering Sciences, 461, 1829–1846.
Fowler, P. W., & Quinn, C. M. (1986). σ, π and δ representations of the molecular point groups. Theoretica Chimica Acta, 70(5), 333–350.
Griffith, J. S. (2009). The Theory of transition-metal ions. Cambridge University Press.
Guest, S. D., & Fowler, P. W. (2005). A symmetry-extended mobility rule. Mechanism and Machine Theory, 40, 1002–1014.
Guest, S. D., & Fowler, P. W. (2007). Symmetry conditions and finite mechanisms. Journal of Mechanics of Materials and Structures, 2(2), 293–301.
Guest, S. D., Fowler, P. W., & Schulze, B. (2018). Mobility of symmetric block-and-hole polyhedra. International Journal of Solids and Structures, 150, 40–51.
Jackson, B., & Owen, J. C. (2016). When is a body-bar structure isostatic? International Journal of Solids and Structures, 47, 2745–2754.
Jackson, B., & Owen, J. C. (2019). Radically solvable graphs. Journal of Combinatorial Theory, Series B, 136, 135–153.
Jotham, R. W., & Kettle, S. F. A. (1971). Geometrical consequences of the Jahn-Teller effect. Inorganica Chimica Acta, 5, 183–187.
Kangwai, R. D., & Guest, S. D. (1999). Detection of finite mechanisms in symmetric structures. International Journal of Solids and Structures, 36(36), 5507–5527.
Kim, K., Lee, J. Y., & Kim, K. (1998). A 2-D geometric constraint solver using DOF-based graph reduction. Computer-Aided Design, 30(11), 883–896.
Klein, F. (1870). Zur Theorie der Liniencomplexe des ersten und zweiten Grades. Mathematische Annalen, 2, 198–226.
Maxwell, J. C. (1864). On the calculation of the equilibrium and stiffness of frames. Philosophical Magazine, 27, 294–299.
Müller, A. (2016). Recursive higher-order constraints for linkages with lower kinematic pairs. Mechanism and Machine Theory, 100, 33–43.
Nawratil, G. (2013). Types of self-motions of planar Stewart–Gough platforms. Meccanica, 48, 1177–1190.
Olver, P. J. (1999). Equivalence, invariants and symmetry(London Mathematical Society Lecture Notes). Cambridge University Press.
Owen, J. C. (1991). Algebraic solution for geometry from dimensional constraints. In Proceedings of the First ACM Symposium on Solid Modeling Foundations and CAD/CAM Applications, SMA ’91(pp. 397–407). ACM.
Owen, J. C., & Power, S. C. (2006). The non-solvability by radicals of generic 3-connected planar Laman graphs. Transactions of the American Mathematical Society, 359, 2269–2303.
Owen, J. C., & Power, S. C. (2010). Framework symmetry and rigidity. International Journal of Computational Geometry and Applications, 20, 723–750.
Plücker, J. (1865a). XVii. on a new geometry of space. Philosophical Transactions of the Royal Society of London, 155, 725–791.
Plücker, J. (1865b). I. on a new geometry of space. Proceedings of the Royal Society of London, 14, 53–58.
Quinn, C. M., McKiernan, J. G., & Redmond, D. B. (1984). Mollweide projections and symmetries on the spherical shell. Journal of Chemical Education, 61, 569–579.
Schulze, B. (2010). Symmetry as a sufficient condition for a finite flex. SIAM Journal on Discrete Mathematics, 24(4), 1291–1312.
Sitharam, M., A. St. John, & J. Sidman, eds., 2019. Handbook of geometric constraint systems. CRC Press.
Spivak, M. (1965). Calculus on manifolds. A modern approach to classical theorems of advanced calculus. W. A. Benjamin, Inc.