MATCH
Communications in Mathematical and in Computer Chemistry

MATCH Commun. Math. Comput. Chem. **76** (2016) 379-400

ISSN 0340 - 6253

Computer-Oriented Representations of Point Groups and Cycle Indices with Chirality Fittingness (CI-CFs) Calculated by the GAP System. Enumeration of Three-Dimensional Structures of Ligancy 4 by Fujita's Proligand Method

Shinsaku Fujita

Shonan Institute of Chemoinformatics and Mathematical Chemistry,
Kaneko 479-7 Ooimachi, Ashigara-Kami-Gun, Kanagawa-Ken,
258-0019 Japan

`shinsaku_fujita@nifty.com`

(Received April 15, 2016)

Abstract

Programs for calculating cycle indices with chirality fittingness (CI-CFs) have been developed as functions of the GAP (Groups, Algorithms, Programming) system in order to reinforce the practical usage of Fujita's proligand method (S. Fujita, *Combinatorial Enumeration of Graphs, Three-Dimensional Structures, and Chemical Compounds*, Mathematical Chemistry Monographs Series, Vol. 15, Kragujevac, 2013). After a mirror-permutation representation of a given point group is defined to differentiate between rotations and (roto)reflections, a combined-permutation representation is newly defined as a computer-oriented representation of the point group. Because such a combined-permutation representation can be regarded as a permutation group, it is generated from an appropriate generators by using the `Group` function of the GAP system. Thereby, the program (`CalcCICF`) for generating CI-CFs is developed as a function of the GAP system, which is used to calculate generating functions for combinatorial enumeration of 3D structures under point groups. The program (`CalcCICF_A`) for calculating the number of achiral 3D structures and the program (`CalcCICF_E`) for calculating the number of enantiomeric pairs of chiral 3D structures are also developed. A practical procedure for combinatorial enumeration of 3D structures is described on the basis of the GAP system. The source codes of these programs are stored in a file attached as an appendix.

1 Introduction

Pólya's theorem [1, 2, 3] has been widely used in gross enumeration of chemical compounds *as graphs*, as summarized in reviews [4, 5, 6, 7, 8] and books [9, 10, 11, 12]. Practical applications of Pólya's theorem have been supported by several computer systems (e.g., the GAP (Groups, Algorithms, Programming) system [13], the Maple system [14], and the Mathematica system [15]). For example, the GAP system supports a function (named `CycleIndex`) for calculating cycle indices (CIs), which are key polynomials for gross enumeration of graphs on the basis of Pólya's theorem.

On the other hand, Fujita's proligand method [16, 17, 18] has been adopted to accomplish gross enumeration of chemical compounds *as three-dimensional (3D) structures*, as summarized in a review [19] and books [20, 21]. As found by the comparison between the title of Pólya-Read's book [3] ("Combinatorial Enumeration of Groups, Graphs, and Chemical Compounds") and that of Fujita's book [20] ("Combinatorial Enumeration of Graphs, Three-Dimensional Structures, and Chemical Compounds"), Fujita's proligand method is different from Pólya's theorem in the applicability to stereochemical phenomena concerning 3D structures, where *cycle indices with chirality fittingness* (CI-CFs) are used in place of Pólya's CIs (without chirality fittingness (CF)). Such CI-CFs are defined by taking account of *sphericities of cycles*, in which Pólya's theorem is deficient, as discussed in a review [22].

In the process of calculating CI-CFs (cf. Chapter 7 of [20]), permutation representations for the substitution positions of a given skeleton are formulated in the form of a coset representation $\boldsymbol{G}(/\boldsymbol{G}_i)$, where the symbol $\boldsymbol{G}$ represents the global point-group symmetry of the skeleton and the symbol $\boldsymbol{G}_i$ represents the local symmetry (stabilizer) of each substitution position. When a permutation of $\boldsymbol{G}(/\boldsymbol{G}_i)$ corresponds to a (roto)reflection, it is attached by an overbar in order to emphasize ligand reflections, so that it is differentiated from a permutation corresponding to a (proper) rotation. Then, a product of sphericity indices (PSI) is assigned to a permutation in accord with the sphericities of cycles involved in the permutation. The PSIs for respective permutations are summed up and the resulting summation is divided by the order of $\boldsymbol{G}$, so as to give the corresponding CI-CF.

The process described above works well if we restrict our attention to combinatorial enumeration under point-group symmetry. However, the proligand method should be extended to support combinatorial enumeration under RS-stereoisomeric-group symmetry in Fujita's stereoisogram approach [21], where permutations of $\boldsymbol{G}(/\boldsymbol{G}_i)$ corresponding to (roto)reflections (with ligand reflections) should be differentiated from RS-permutations without ligand reflections. The present article is devoted to a new formulation of (roto)reflections, where ligand reflections are taken into explicit consideration, so as to develop computer-oriented representations of point groups.

2 Computer-Oriented Representations of Point Groups

2.1 Skeletons for the Proligand-Promolecule Model

Let us consider a set of n positions of a given skeleton belonging to a point group $\boldsymbol{G}$:

$$\mathbf{X} = \{1, 2, 3, \ldots, n\}, \tag{1}$$

where the positions are numbered sequentially. The point group $\boldsymbol{G}$ acts on the set $\mathbf{X}$, so that there appears a permutation representation. For example, Figure 1 shows an oxirane skeleton **1** belonging to the point group $\boldsymbol{C}_{2v}$, where the four positions are numbered sequentially as being $\{1, 2, 3, 4\}$.

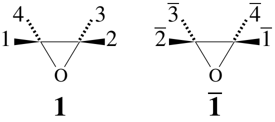

Figure 1. Numbered and mirror-numbered skeletons for oxirane derivatives.

When a (roto)reflection contained in the point group $\boldsymbol{G}$ acts on the set $\mathbf{X}$, the numbered skeleton is converted into a mirror-numbered skeleton, where the set of numbered positions $\mathbf{X}$ (Eq. 1) is converted into a set of mirror-numbered positions $\overline{\mathbf{X}}$ as follows:

$$\overline{\mathbf{X}} = \{\overline{1}, \overline{2}, \overline{3}, \ldots, \overline{n}\}. \tag{2}$$

Such a (roto)reflection brings about the mirror-transformation of the skeleton as well as the mirror-transformation of each position. For example, let us consider a mirror plane perpendicular to the oxirane ring **1** through the midpoints of the C—C bond and the oxygen atom, so as to generate a reflection represented by $\sigma_{v(1)} \sim \overline{(1\ 2)(3\ 4)}$. The action of a reflection $\sigma_{v(1)}$ on the numbered skeleton **1** produces the corresponding mirror-numbered skeleton $\overline{\mathbf{1}}$, where the product of cycles in the symbol $\overline{(1\ 2)(3\ 4)}$ represents the mirror-transformation of the skeleton, while its overline denotes the mirror-transformation of each position.

Let the symbol $\boldsymbol{P}_{\boldsymbol{G}}^{(\mathbf{X}\overline{\mathbf{X}})}$ be a permutation representation of $\boldsymbol{G}$ due to the sets of $\mathbf{X}$ (Eq. 1) and $\overline{\mathbf{X}}$ (Eq. 2), where it contains permutations without and with an overline. The mirror-transformation of each position (a permutation with an overline) is difficult to be treated by a computer system such as a GAP system. Hence, it is necessary to develop a computer-oriented representation for characterizing the effect of an overline.

2.2 Mirror-Permutation Representations

In order to develop such a computer-oriented representation for the mirror-transformation of each position, we consider the following set of local chiralities:

$$\chi = \{\mathbf{X}, \overline{\mathbf{X}}\} = \{n+1, n+2\}, \tag{3}$$

where the numbers $n+1$ and $n+2$ are given in accord with the number n of $\mathbf{X}$ (Eq. 1). If the set χ is regarded as an ordered set, it is transformed into itself $\{\mathbf{X}, \overline{\mathbf{X}}\}$ or the other ordered set $\{\overline{\mathbf{X}}, \mathbf{X}\}$ under the action of the point group $\boldsymbol{G}$. The action of $\boldsymbol{G}$ on the set χ gives the following permutations:

$$\boldsymbol{P}_{G}^{(\chi)} = \begin{pmatrix} \mathbf{X} & \overline{\mathbf{X}} \\ \mathbf{X} & \overline{\mathbf{X}} \end{pmatrix} = \begin{pmatrix} n+1 & n+2 \\ n+1 & n+2 \end{pmatrix} = (n+1)(n+2) \quad \text{for } G\text{: rotations} \tag{4}$$

$$\boldsymbol{P}_{G}^{(\chi)} = \begin{pmatrix} \mathbf{X} & \overline{\mathbf{X}} \\ \overline{\mathbf{X}} & \mathbf{X} \end{pmatrix} = \begin{pmatrix} n+1 & n+2 \\ n+2 & n+1 \end{pmatrix} = (n+1\ \ n+2) \quad \text{for } G\text{: (roto)reflections} \tag{5}$$

which bring about a permutation representation of degree 2 ($G \in \boldsymbol{G}$). Thereby, the effect of an overline is replaced by the permutation representation $\boldsymbol{P}_{\boldsymbol{G}}^{(\chi)}$, which is called *a mirror-permutation representation*:

$$\boldsymbol{P}_{\boldsymbol{G}}^{(\chi)} = \{\boldsymbol{P}_{G}^{(\chi)} \mid \forall G \in \boldsymbol{G}\}, \tag{6}$$

each permutation of which is represented by Eqs. 4 and 5.

Let the symbol $\boldsymbol{P}_{\boldsymbol{G}}^{(\mathbf{X})}$ be a permutation representation of $\boldsymbol{G}$ due to the set of $\mathbf{X}$ (Eq. 1), where it contains permutations without an overline. The permutation representation $\boldsymbol{P}_{\boldsymbol{G}}^{(\mathbf{X})}$ is combined with the permutation representation $\boldsymbol{P}_{\boldsymbol{G}}^{(\chi)}$ to give *a combined-permutation representation*:

$$\boldsymbol{P}_{\boldsymbol{G}}^{(\mathbf{X}\chi)} = \boldsymbol{P}_{\boldsymbol{G}}^{(\mathbf{X})} \oplus \boldsymbol{P}_{\boldsymbol{G}}^{(\chi)} = \{\boldsymbol{P}_{G}^{(\mathbf{X})} \oplus \boldsymbol{P}_{G}^{(\chi)} \mid \forall G \in \boldsymbol{G}\}, \tag{7}$$

where the symbol $\boldsymbol{P}_{\boldsymbol{G}}^{(\mathbf{X})} \oplus \boldsymbol{P}_{\boldsymbol{G}}^{(x)}$ is a combination of the two permutations at issue, e.g., $(1\ 3)(2\ 4) \oplus (5\ 6) = (1\ 3)(2\ 4)(5\ 6)$. The representation $\boldsymbol{P}_{\boldsymbol{G}}^{(\mathbf{X}x)}$ (Eq. 7) can be used in place of the permutation representation $\boldsymbol{P}_{\boldsymbol{G}}^{(\mathbf{X}\overline{\mathbf{X}})}$.

Figure 2 shows the comparison between $\boldsymbol{P}_{\boldsymbol{C}_{2v}}^{(\mathbf{X}\overline{\mathbf{X}})}$ and $\boldsymbol{P}_{\boldsymbol{C}_{2v}}^{(\mathbf{X}x)}$ ($\boldsymbol{G} = \boldsymbol{C}_{2v}$), where a rotation C_2 ($\in \boldsymbol{C}_{2v}$) is operated onto an oxirane skeleton **1** (or $\overline{\mathbf{1}}$) belonging to $\boldsymbol{C}_{2v}$. Because a permutation $\boldsymbol{P}_{G}^{(x)}$ in the $\boldsymbol{P}_{\boldsymbol{G}}^{(x)}$-part of $\boldsymbol{P}_{\boldsymbol{G}}^{(\mathbf{X}x)}$ (Eq. 7) for rotations is equal to a product of 1-cycles (cf. Eq. 4, e.g., $\boldsymbol{P}_{C_2}^{(x)} = (5)(6)$), the permutation $\boldsymbol{P}_{G}^{(\mathbf{X}x)}$ of $\boldsymbol{P}_{\boldsymbol{G}}^{(\mathbf{X}x)}$ (e.g., $\boldsymbol{P}_{C_2}^{(\mathbf{X}x)} = (1\ 3)(2\ 4)(5)(6) = (1\ 3)(2\ 4)$) is identical with the permutation $\boldsymbol{P}_{G}^{(\mathbf{X})}$ of $\boldsymbol{P}_{\boldsymbol{G}}^{(\mathbf{X})}$ (e.g., $\boldsymbol{P}_{C_2}^{(\mathbf{X})} = (1\ 3)(2\ 4)$).

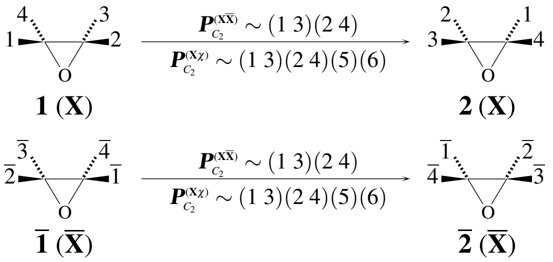
4
3
1
2
$P_{C_2}^{(X\overline{X})} \sim (1\ 3)(2\ 4)$
$P_{C_2}^{(X\chi)} \sim (1\ 3)(2\ 4)(5)(6)$
2
1
3
4
O
O
1 (X)
2 (X)
$\overline{3}$
$\overline{4}$
$\overline{2}$
$\overline{1}$
$P_{C_2}^{(X\overline{X})} \sim (1\ 3)(2\ 4)$
$P_{C_2}^{(X\chi)} \sim (1\ 3)(2\ 4)(5)(6)$
$\overline{1}$
$\overline{2}$
$\overline{4}$
$\overline{3}$
O
O
$\overline{1}$ ($\overline{X}$)
$\overline{2}$ ($\overline{X}$)

Figure 2. Rotations for an Oxirane Skeleton

On the other hand, Figure 3 shows the comparison between $\boldsymbol{P}_{\boldsymbol{C}_{2v}}^{(\mathbf{X}\overline{\mathbf{X}})}$ and $\boldsymbol{P}_{\boldsymbol{C}_{2v}}^{(\mathbf{X}x)}$ ($\boldsymbol{G} = \boldsymbol{C}_{2v}$), where a reflection $\sigma_{v(1)}$ ($\in \boldsymbol{C}_{2v}$) is operated onto an oxirane skeleton **1** (or $\overline{\mathbf{1}}$) belonging to $\boldsymbol{C}_{2v}$. Because a permutation $\boldsymbol{P}_{G}^{(x)}$ in the $\boldsymbol{P}_{\boldsymbol{G}}^{(x)}$-part of $\boldsymbol{P}_{\boldsymbol{G}}^{(\mathbf{X}x)}$ (Eq. 7) for reflections is equal to a 2-cycle (cf. Eq. 5, e.g., $\boldsymbol{P}_{\sigma_{v(1)}}^{(x)} = (5\ 6)$), the permutation $\boldsymbol{P}_{G}^{(\mathbf{X}x)}$ of $\boldsymbol{P}_{\boldsymbol{G}}^{(\mathbf{X}x)}$ (e.g., $\boldsymbol{P}_{C_2}^{(\mathbf{X}x)} = (1\ 3)(2\ 4)(5\ 6)$) is different from the permutation $\boldsymbol{P}_{G}^{(\mathbf{X})}$ of $\boldsymbol{P}_{\boldsymbol{G}}^{(\mathbf{X})}$ (e.g., $\boldsymbol{P}_{\sigma_{v(1)}}^{(\mathbf{X})} = \overline{(1\ 3)(2\ 4)}$). Note that the effect of the overline of $\overline{(1\ 3)(2\ 4)}$ is replaced by the 2-cycle $(5\ 6)$ in the permutation $(1\ 3)(2\ 4)(5\ 6)$.

The procedures for Figures 2 and 3 are repeated to cover all of the operations of the point group $\boldsymbol{C}_{2v}$, so as to give Table 1. Thereby, the permutations listed in the $\boldsymbol{P}_{\boldsymbol{C}_{2v}}^{(\mathbf{X}\overline{\mathbf{X}})}$-column are represented by the combined permutations listed in the $\boldsymbol{P}_{\boldsymbol{C}_{2v}}^{(\mathbf{X}x)}$-column of Table 1. It should be noted that each of rotations (I and C_2) is specified by a product of 1-cycles (i.e., $(5)(6)$), while each of reflections ($\sigma_{v(1)}$ and $\sigma_{v(2)}$) is specified by a 2-cycle (i.e., $(5\ 6)$), as listed in the $\boldsymbol{P}_{\boldsymbol{C}_{2v}}^{(x)}$-column.

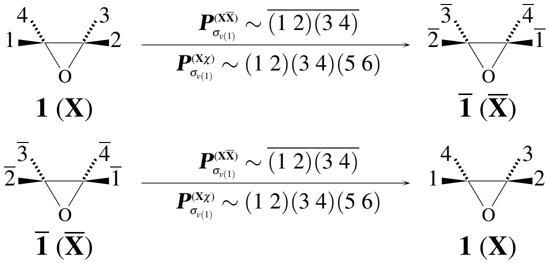
4
3
1
2
O
1 (X)
$P^{(X\overline{X})}_{\sigma_{v(1)}} \sim \overline{(1\ 2)(3\ 4)}$
$P^{(X\chi)}_{\sigma_{v(1)}} \sim (1\ 2)(3\ 4)(5\ 6)$
$\overline{3}$
$\overline{4}$
$\overline{2}$
$\overline{1}$
O
$\overline{1}$ ($\overline{X}$)
$\overline{3}$
$\overline{4}$
$\overline{2}$
$\overline{1}$
O
$\overline{1}$ ($\overline{X}$)
$P^{(X\overline{X})}_{\sigma_{v(1)}} \sim \overline{(1\ 2)(3\ 4)}$
$P^{(X\chi)}_{\sigma_{v(1)}} \sim (1\ 2)(3\ 4)(5\ 6)$
4
3
1
2
O
1 (X)

Figure 3. Reflections for an Oxirane Skeleton

Table 1. Symmetry Operations for an Oxirane Skeleton and Products of Sphericity Indices

C_{2v}	$P^{(X\overline{X})}_{C_{2v}}$	$P^{(\chi)}_{C_{2v}}$	$P^{(X\chi)}_{C_{2v}}$	PSI
symmetry operation	permutation representation	mirror-permutation representation	combined-permutation representation	(product of sphericity index)
I	(1)(2)(3)(4)	(5)(6)	(1)(2)(3)(4)(5)(6)	b_1^4
C_2	(1 3)(2 4)	(5)(6)	(1 3)(2 4)(5)(6)	b_2^2
$\sigma_{v(1)}$	$\overline{(1\ 2)(3\ 4)}$	(5 6)	(1 2)(3 4)(5 6)	c_2^2
$\sigma_{v(2)}$	$\overline{(1\ 4)(2\ 3)}$	(5 6)	(1 4)(2 3)(5 6)	c_2^2

2.3 Generation of Combined-Permutation Representations

Combined-permutation representations defined above are generated by the generator function of of the GAP system, where they are regarded as permutation groups. For example, a combined-permutation representation $\boldsymbol{P}^{(\mathbf{X}\chi)}_{\boldsymbol{C}_{2v}}$ of the point group $\boldsymbol{C}_{2v}$ (Table 1) is generated by inputting the following codes:

```
gap> C2v := Group((1,3)(2,4), (1,2)(3,4)(5,6));;
gap> element_C2v := Elements(C2v);;
gap> Display(element_C2v);
```

where the symbol `gap>` denotes the prompt of the GAP system operated on the command-prompt screen of the Windows system (or other operating systems). The function `Group` of the GAP system takes a list of generators as an argument. Note that each 1-cycle is omitted in the above codes. The result appears as follows:

```
[ (), (1,2)(3,4)(5,6), (1,3)(2,4), (1,4)(2,3)(5,6) ]
```

which is consistent with the list of operators of $\boldsymbol{C}_{2v}$ shown in the $\boldsymbol{P}^{(\mathbf{X}\chi)}_{\boldsymbol{C}_{2v}}$-column of Table 1 (each 1-cycle omitted). The elements containing a 2-cycle (5 6) correspond to reflections, which are alternatively represented by attaching an overline, as shown in the the $\boldsymbol{P}_{C_{2v}}^{(\mathbf{x}\overline{\mathbf{x}})}$-column of Table 1.

The procedure described above is applied to the skeletons of ligancy 4 listed in Figure 4, i.e., an allene skeleton **3** ($\boldsymbol{D}_{2d}$), an ethylene skeleton **4** ($\boldsymbol{D}_{2h}$), a tetrahedral skeleton **5** ($\boldsymbol{T}_{d}$), a square-planar (SP-4) skeleton **6** ($\boldsymbol{D}_{4h}$), and a square-pyramidal (SPY-4) skeleton **7** ($\boldsymbol{C}_{4v}$). A list of generators of the point group for characterizing each skeleton is shown in the first line of each point-group row of Table 2 by using the `Group` function of the GAP system. The order is calculated by using the `Size` function, e.g., `Size(C2v)` for the point groups $\boldsymbol{C}_{2v}$. The resulting list of elements due to the `Elements` function of the GAP system is shown in the next line of each point-group row of Table 2. The list of elements can be regarded as a combined-permutation representation, where a vacant pair of parentheses indicates an identity element, which may be represented by a product of 1-cycles.

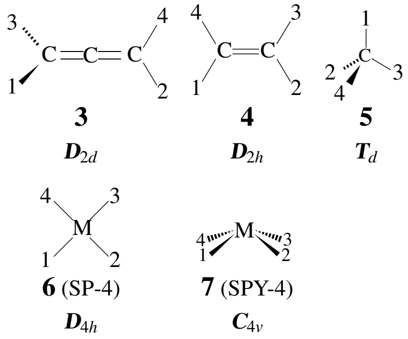

Figure 4. Stereoskeletons of ligancy 4. The point-group symmetry of each stereoskeleton is shown by using the Schönflies notation. The symbol M represents a central metal.

3 Combinatorial Enumeration

3.1 Cycle Indices with Chirality Fittingness (CI-CFs)

Fujita's proligand method for gross enumeration of 3D structures [16, 17, 18, 20] adopts the concept of *sphericities of cycles.* Thereby, a k-cycle contained in a given product of cycles for a rotation G (without an overline or $\boldsymbol{P}_{G}^{(x)} = (n+1)(n+2)$) is referred to as *a hemispheric cycle* (sphericity index: b_k), while a k-cycle contained in a given product of

Table 2. Combined-Permutation Representations of Point Groups for Characterizing the Skeletons of Ligancy 4

point group	a list of generators, order, a list of elements

$\boldsymbol{C}_{2v}$	C2v := Group([(1,3)(2,4), (1,2)(3,4)(5,6)])
	Order =4
	[(), (1,2)(3,4)(5,6), (1,3)(2,4), (1,4)(2,3)(5,6)]

$\boldsymbol{D}_{2d}$	D2d := Group([(1,3)(2,4), (1,2)(3,4), (2,4)(5,6)])
	Order =8
	[(), (2,4)(5,6), (1,2)(3,4), (1,2,3,4)(5,6), (1,3)(5,6),
	(1,3)(2,4), (1,4,3,2)(5,6), (1,4)(2,3)]

$\boldsymbol{D}_{2h}$	D2h := Group([(1,3)(2,4), (1,2)(3,4), (5,6)])
	Order =8
	[(), (5,6), (1,2)(3,4), (1,2)(3,4)(5,6), (1,3)(2,4),
	(1,3)(2,4)(5,6), (1,4)(2,3), (1,4)(2,3)(5,6)]

$\boldsymbol{T}_{d}$	Td := Group([(1,2)(3,4), (2,3,4), (3,4)(5,6)]) Order =24 [(), (3,4)(5,6), (2,3)(5,6), (2,3,4), (2,4,3), (2,4)(5,6), (1,2)(5,6), (1,2)(3,4), (1,2,3), (1,2,3,4)(5,6), (1,2,4,3)(5,6), (1,2,4), (1,3,2), (1,3,4,2)(5,6), (1,3)(5,6), (1,3,4), (1,3)(2,4), (1,3,2,4)(5,6), (1,4,3,2)(5,6), (1,4,2), (1,4,3), (1,4)(5,6), (1,4,2,3)(5,6), (1,4)(2,3)]
$\boldsymbol{D}_{4h}$	D4h := Group([(1,3)(2,4), (1,2)(3,4), (1,2,3,4), (5,6)]) Order =16 [(), (5,6), (2,4), (2,4)(5,6), (1,2)(3,4), (1,2)(3,4)(5,6), (1,2,3,4), (1,2,3,4)(5,6), (1,3), (1,3)(5,6), (1,3)(2,4), (1,3)(2,4)(5,6), (1,4,3,2), (1,4,3,2)(5,6), (1,4)(2,3), (1,4)(2,3)(5,6)]

$\boldsymbol{C}_{4v}$	C4v := Group([(1,3)(2,4), (1,2,3,4), (1,2)(3,4)(5,6)])
	Order =8
	[(), (2,4)(5,6), (1,2)(3,4)(5,6), (1,2,3,4), (1,3)(5,6),
	(1,3)(2,4), (1,4,3,2), (1,4)(2,3)(5,6)]

cycles for a (roto)reflection (with an overline or $\boldsymbol{P}_G^{(x)} = (n+1 \;\; n+2)$) is categorized into a homospheric cycle (sphericity index: a_k if k is odd) or an enantiospheric cycle (sphericity index: c_k if k is even).

Suppose that an element $\boldsymbol{P}_G^{(\mathbf{X})}$ (cf. Eq. 7) is a permutation of degree n, which is represented by a cycle decomposition involving the number $\nu_k(\boldsymbol{P}_G^{(\mathbf{X})})$ of k-cycles ($\sum_{k=1}^{n} k\nu_k(\boldsymbol{P}_G^{(\mathbf{X})})$). Then the element $\boldsymbol{P}_G^{(\mathbf{X}x)}$ corresponding to $\boldsymbol{P}_G^{(\mathbf{X})}$ (cf. Eq. 7) is specified by a product of sphericity indices (PSI):

$$\mathrm{PSI}_{\boldsymbol{P}_G^{(\mathbf{X}x)}} = \$_1^{\nu_1(\boldsymbol{P}_G^{(\mathbf{X})})} \$_2^{\nu_2(\boldsymbol{P}_G^{(\mathbf{X})})} \cdots \$_n^{\nu_n(\boldsymbol{P}_G^{(\mathbf{X})})}, \tag{8}$$

where $\$_k$ is a_k if $\boldsymbol{P}_G^{(x)} = (n+1 \;\; n+2)$ (one 2-cycle) and k is odd; $\$_k$ is c_k if $\boldsymbol{P}_G^{(x)} = (n+1 \;\; n+2)$ (one 2-cycle) and k is even; and $\$_k$ is b_k if $\boldsymbol{P}_G^{(x)} = (n+1)(n+2)$ (two 1-cycles). According to Def. 7.20 of [20], the cycle index with chirality fittingness (CI-CF) for $\boldsymbol{P}_{\boldsymbol{G}}^{(\mathbf{X}_\chi)}$ is calculated as follows by using the PSIs (Eq. 8):

$$\text{CI-CF}(\boldsymbol{P}_{\boldsymbol{G}}^{(\mathbf{X}_\chi)}; \$_k) = \frac{1}{|\boldsymbol{G}|} \sum_{G \,\in\, \boldsymbol{G}} \$_1^{\nu_1(\boldsymbol{P}_G^{(\mathbf{X})})} \$_2^{\nu_2(\boldsymbol{P}_G^{(\mathbf{X})})} \cdots \$_n^{\nu_n(\boldsymbol{P}_G^{(\mathbf{X})})}. \tag{9}$$

For example, the data of the $\boldsymbol{P}_{\boldsymbol{C}_{2v}}^{(\mathbf{X}_\chi)}$-column of Table 1 give the respective PCIs collected in the PSI-column of Table 1. These PCIs are summed up and the resulting sum is divided by the order of $\boldsymbol{C}_{2v}$ ($|\boldsymbol{C}_{2v}| = 4$) according to Eq. 9. Thereby, we obtain the following CI-CF:

$$\text{CI-CF}(\boldsymbol{P}_{\boldsymbol{C}_{2v}}^{(\mathbf{X}_\chi)}; \$_k) = \frac{1}{4}\left\{b_1^4 + b_2^2 + 2c_2^2\right\}. \tag{10}$$

According to Def. 7.25 of [20], the gross enumeration of achiral 3D structures is conducted by using the following CI-CF:

$$\text{CI-CF}^{(a)}(\boldsymbol{P}_{\boldsymbol{G}}^{(\mathbf{X}_\chi)}; \$_k) = \frac{2}{|\boldsymbol{G}|} \sum_{G \,\in\, \boldsymbol{G}^{(a)}} \$_1^{\nu_1(\boldsymbol{P}_G^{(\mathbf{X})})} \$_2^{\nu_2(\boldsymbol{P}_G^{(\mathbf{X})})} \cdots \$_n^{\nu_n(\boldsymbol{P}_G^{(\mathbf{X})})}. \tag{11}$$

The symbol $\boldsymbol{G}^{(a)}$ denotes the coset which contains all of the (roto)reflections of $\boldsymbol{G}$, so that the coset $\boldsymbol{G}^{(a)}$ satisfies the following coset decomposition:

$$\boldsymbol{G} = \boldsymbol{G}^{(m)} + \boldsymbol{G}^{(a)}, \tag{12}$$

where the symbol $\boldsymbol{G}^{(m)}$ denotes the maximum chiral subgroup of $\boldsymbol{G}$. Note that if the element G is contained in the coset $\boldsymbol{G}^{(a)}$, it exhibits $\boldsymbol{P}_G^{(\chi)} = (n+1\ \ n+2)$.

For example, the elements having a 2-cycle (5 6) in the $\boldsymbol{P}_{\boldsymbol{C}_{2v}}^{(\mathbf{X}_\chi)}$-column of Table 1 are contained in $\boldsymbol{C}_{2v}^{(a)}$. The corresponding PCIs are summed up and the resulting sum is multiplied by ($2/|\boldsymbol{C}_{2v}| = 2/4 = 1/2$) according to Eq. 11. Thereby, we obtain the following CI-CF:

$$\text{CI-CF}^{(a)}(\boldsymbol{P}_{\boldsymbol{C}_{2v}}^{(\mathbf{X}_\chi)}; \$_k) = \frac{1}{2} \times 2c_2^2 = c_2^2. \tag{13}$$

According to Def. 7.28 of [20], the gross enumeration of enantiomeric pairs of chiral 3D structures is conducted by using the following CI-CF:

$$\text{CI-CF}^{(e)}(\boldsymbol{P}_{\boldsymbol{G}}^{(\mathbf{X}_\chi)}; \$_k) = \\ \frac{1}{|\boldsymbol{G}|}\left\{\sum_{G \,\in\, \boldsymbol{G}^{(m)}} b_1^{\nu_1(\boldsymbol{P}_G^{(\mathbf{X})})} b_2^{\nu_2(\boldsymbol{P}_G^{(\mathbf{X})})} \cdots b_n^{\nu_n(\boldsymbol{P}_G^{(\mathbf{X})})} - \sum_{G \,\in\, \boldsymbol{G}^{(a)}} \$_1^{\nu_1(\boldsymbol{P}_G^{(\mathbf{X})})} \$_2^{\nu_2(\boldsymbol{P}_G^{(\mathbf{X})})} \cdots \$_n^{\nu_n(\boldsymbol{P}_G^{(\mathbf{X})})}\right\}, \tag{14}$$

where the symbols $G \in \boldsymbol{G}^{(m)}$ and $G \in \boldsymbol{G}^{(a)}$ are shown in Eq. 12.

For example, Eq. 14 is applied to the PCIs listed in the PCI-column of Table 1, so as to give the following CI-CF:

$$\text{CI-CF}^{(e)}(\boldsymbol{P}_{\boldsymbol{C}_{2v}}^{(\mathbf{X}_\chi)}; \$_k) = \frac{1}{4}\left\{b_1^4 + b_2^2 - 2c_2^2\right\}. \tag{15}$$

The CI-CFs for the point group $\boldsymbol{C}_{2v}$ (Eqs. 10, 13, and 15) satisfy the following equation:

$$\text{CI-CF}(\boldsymbol{P}^{(\mathbf{X}\chi)}_{\boldsymbol{C}_{2v}}; \$_k) = \text{CI-CF}^{(a)}(\boldsymbol{P}^{(\mathbf{X}\chi)}_{\boldsymbol{C}_{2v}}; \$_k) + \text{CI-CF}^{(e)}(\boldsymbol{P}^{(\mathbf{X}\chi)}_{\boldsymbol{C}_{2v}}; \$_k). \tag{16}$$

In general, the CI-CF for the point group $\boldsymbol{G}$ (Eq. 9) is equal to the sum of Eq. 11 and Eq. 14, i.e.,

$$\text{CI-CF}(\boldsymbol{P}^{(\mathbf{X}\chi)}_{\boldsymbol{G}}; \$_k) = \text{CI-CF}^{(a)}(\boldsymbol{P}^{(\mathbf{X}\chi)}_{\boldsymbol{G}}; \$_k) + \text{CI-CF}^{(e)}(\boldsymbol{P}^{(\mathbf{X}\chi)}_{\boldsymbol{G}}; \$_k). \tag{17}$$

Because the combined-permutation representations of other point groups have been calculated (Table 2), the procedures for calculating the CI-CFs for the point group $\boldsymbol{C}_{2v}$ (Eqs. 10, 13, and 15) are effective to calculate the CI-CFs for the other skeletons of ligancy 4 (Figure 4).

3.2 GAP Functions for Calculating CI-CFs

The next step is the development of GAP functions for calculating CI-CFs by starting from the combined-permutation representation of a point group.

The key is the differentiation between rotations and (roto)reflections, where the GAP function `CycleLengths` is used to detect permutations without and with $\boldsymbol{P}^{(\chi)}_{G} = (n+1)(n+2)$. For example, let us examine the combined-permutation representation $\boldsymbol{P}^{(\mathbf{X}\chi)}_{\boldsymbol{C}_{2v}}$ corresponding to the point group $\boldsymbol{C}_{2v}$:

```
gap> C2v := Group((1,4)(2,3), (1,2)(3,4)(5,6));;
gap> l_elements := Elements(C2v);
[ (), (1,2)(3,4)(5,6), (1,3)(2,4)(5,6), (1,4)(2,3) ]
gap> no_elements := Size(l_elements);
4
gap> temp_cycstrX := CycleLengths(l_elements[2], [1..6]);
[ 2, 2, 2 ]
gap> temp_cycstrX := CycleLengths(l_elements[4], [1..6]);
[ 2, 2, 1, 1 ]
```

The second permutation $(1,2)(3,4)(5,6)$ with $\boldsymbol{P}^{(\chi)}_{\sigma_{v(1)}} = (5\ 6)$, which is the second component of the list denoted as `l_elements[2]`, is composed of three 2-cycles, which are detected as `[ 2, 2, 2 ]` by means of the GAP function `CycleLengths`. On the other hand, the fourth permutation $(1,4)(2,3)$ denoted as `l_elements[4]`, which has an implicit set of 1-cycles (i.e., $\boldsymbol{P}^{(\chi)}_{C_2} = (5)(6)$), is composed of two 2-cycles and two implicit 1-cycles. Because these cycles are detected as `[ 2, 2, 1, 1 ]`, the 1-cycle due to $\boldsymbol{P}^{(\chi)}_{C_2} = (5)(6)$ is detected by the last cycle length '1' appearing in the output of the GAP function `CycleLengths`. Thereby, the fourth permutation $(1,4)(2,3)$ is detected as being a rotation by the GAP system.

The function `CalcCICF` for calculating the CI-CF of a given group $\boldsymbol{G}$ (Eq. 9) is developed by using the `function` keyword of the GAP system, where $\boldsymbol{G}$ is expressed in the form of a combined-permutation representation $\boldsymbol{P}_{G}^{(\mathbf{X}_\chi)}$. In a similar way, the function `CalcCICF_A` for calculating the achiral part CI-CF$^{(a)}$ (Eq. 11) and the function `CalcCICF_E` for calculating the chiral part CI-CF$^{(e)}$ (Eq. 14) are also developed:

```
CalcCICF(group, degree, degreefull)
CalcCICF_A(group, degree, degreefull)
CalcCICF_E(group, degree, degreefull)
```

where the first argument `group` denotes a given group $\boldsymbol{G}$ (as a combined-permutation representation $\boldsymbol{P}_{G}^{(\mathbf{X}_\chi)}$), the second argument `degree` denotes the degree of $\boldsymbol{P}_{G}^{(\mathbf{X})}$, and the third argument `degreefull` denotes the degree of $\boldsymbol{P}_{G}^{(\mathbf{X}_\chi)}$. The source lists of these functions are stored in a file named `CICFgen.gapfunc`, which is attached as Appendix A.

3.3 Calculation of CI-CFs

Let us illustrate a typical procedure for executing the above-developed functions (`CalcCICF`, `CalcCICF_A`, and `CalcCICF_E`), where we calculate the CI-CFs (CI-CF, CI-CF$^{(a)}$, and CI-CF$^{(e)}$) of an oxirane skeleton of $\boldsymbol{C}_{2v}$ (cf. Table 1). We first make a working directory with an appropriate name (e.g., `c:/fujita0/calcCICF`), in which we beforehand place the file `CICFgen.gapfunc` containing the functions for generating CI-CFs (cf. Appendix A). This directory also contains a work file with an appropriate name (e.g., `C2v-CICF.gap`), which contains the following codes:

```
#Read("c:/fujita0/calcCICF/C2v-CICF.gap");
                                    #Input this line to the gap> prompt
LogTo("c:/fujita0/calcCICF/C2v-CICFlog.txt");

Read("c:/fujita0/calcCICF/CICFgen.gapfunc"); #Loading of CICFgen.gapfunc

C2v := Group((1,3)(2,4), (1,2)(3,4)(5,6)); #C2v
Print("C2v␣:=", C2v, "\n");
Print("Order␣=", Size(C2v), "\n", Elements(C2v), "\n");
Print("CICF_C2v␣:=␣", CalcCICF(C2v, 4, 6), "\n");
Print("CICF_C2v_A␣:=␣", CalcCICF_A(C2v, 4, 6), "\n");
Print("CICF_C2v_E␣:=␣", CalcCICF_E(C2v, 4, 6), "\n");

LogTo();
```

To execute these codes, the first line commented out by the top symbol # is copied and pasted after the gap> prompt of the command-prompt window of the Windows system. Thereby, the above codes contained in the file C2v-CICF.gap are successively executed after the loading of CICFgen.gapfunc. The calculated CI-CFs are written down into the log file named C2v-CICFlog.txt (an appropriate name) as follows:

```
C2v :=Group( [ (1,3)(2,4), (1,2)(3,4)(5,6) ] )
Order =4
[ (), (1,2)(3,4)(5,6), (1,3)(2,4), (1,4)(2,3)(5,6) ]
CICF_C2v := 1/4*b_1^4+1/2*c_2^2+1/4*b_2^2
CICF_C2v_A := c_2^2
CICF_C2v_E := 1/4*b_1^4-1/2*c_2^2+1/4*b_2^2
```

where each multiplication appearing in the CI-CFs is represented by an asterisk and sphericity indices are represented by b_1 (for b_1), c_2 (for c_2), and b_2 (for b_2). The CI-CFs are consistent with Eqs. 10, 13, and 15.

In a similar way, the respective combined-permutation representations listed in Table 2 generate the corresponding CI-CFs, which are collected in Table 3.

Table 3. CI-CFs of Point Groups for Characterizing the Skeletons of Ligancy 4

point group	CICFs for total as well as for achiral (_A) and chiral (_C) parts

$\boldsymbol{C}_{2v}$	`CICF_C2v := 1/4*b_1^4+1/2*c_2^2+1/4*b_2^2`
	`CICF_C2v_A := c_2^2`
	`CICF_C2v_E := 1/4*b_1^4-1/2*c_2^2+1/4*b_2^2`

$\boldsymbol{D}_{2d}$	`CICF_D2d := 1/8*b_1^4+1/4*c_2*a_1^2+3/8*b_2^2+1/4*c_4`
	`CICF_D2d_A := 1/2*c_2*a_1^2+1/2*c_4`
	`CICF_D2d_E := 1/8*b_1^4-1/4*c_2*a_1^2+3/8*b_2^2-1/4*c_4`

$\boldsymbol{D}_{2h}$	`CICF_D2h := 1/8*b_1^4+1/8*a_1^4+3/8*c_2^2+3/8*b_2^2`
	`CICF_D2h_A := 1/4*a_1^4+3/4*c_2^2`
	`CICF_D2h_E := 1/8*b_1^4-1/8*a_1^4-3/8*c_2^2+3/8*b_2^2`

$\boldsymbol{T}_{d}$	`CICF_Td := 1/24*b_1^4+1/4*c_2*a_1^2+1/3*b_1*b_3+1/8*b_2^2+1/4*c_4`
	`CICF_Td_A := 1/2*c_2*a_1^2+1/2*c_4`
	`CICF_Td_E := 1/24*b_1^4-1/4*c_2*a_1^2+1/3*b_1*b_3+1/8*b_2^2-1/4*c_4`

$\boldsymbol{D}_{4h}$	`CICF_D4h := 1/16*b_1^4+1/16*a_1^4+1/8*b_1^2*b_2+1/8*c_2*a_1^2`
	`+3/16*c_2^2+3/16*b_2^2+1/8*c_4+1/8*b_4`
	`CICF_D4h_A := 1/8*a_1^4+1/4*c_2*a_1^2+3/8*c_2^2+1/4*c_4`
	`CICF_D4h_E := 1/16*b_1^4-1/16*a_1^4+1/8*b_1^2*b_2-1/8*c_2*a_1^2`
	`-3/16*c_2^2+3/16*b_2^2-1/8*c_4+1/8*b_4`

$\boldsymbol{C}_{4v}$	`CICF_C4v := 1/8*b_1^4+1/4*c_2*a_1^2+1/4*c_2^2+1/8*b_2^2+1/4*b_4`
	`CICF_C4v_A := 1/2*c_2*a_1^2+1/2*c_2^2`
	`CICF_C4v_E := 1/8*b_1^4-1/4*c_2*a_1^2-1/4*c_2^2+1/8*b_2^2+1/4*b_4`

These CI-CFs have been alternatively calculated by using permutation representations $\boldsymbol{P}_{\boldsymbol{G}}^{(\mathbf{X}\overline{\mathbf{X}})}$ (cf. Table 1 for $\boldsymbol{G} = \boldsymbol{C}_{2v}$) in place of combined-permutation representations $\boldsymbol{P}_{\boldsymbol{G}}^{(\mathbf{X}_X)}$

(cf. Table 1 for $\boldsymbol{G} = \boldsymbol{C}_{2v}$) [23]. Thus, CICF_C2v of Table 3 is identical with Eq. 51 of [23]; CICF_D2d with Eq. 8 of [16] and with Eq. 48 of [23]; CICF_D2h with Eq. 49 of [23]; CICF_Td with Eq. 46 of [23]; CICF_D4h with Eq. 47 of [23]; and CICF_C4v with Eq. 50 of [23].

3.4 Introduction of Ligand-Inventory Functions into CI-CFs

Suppose that the n positions of a given skeleton accommodate a set of n proligands selected from a given ligand inventory:

$$\boldsymbol{L} = \{\mathrm{L}_1, \mathrm{L}_2, \ldots, \mathrm{L}_\nu; \mathrm{p}_1, \mathrm{p}_2, \ldots, \mathrm{p}_\nu; \overline{\mathrm{p}}_1, \overline{\mathrm{p}}_2, \ldots, \overline{\mathrm{p}}_\nu\}, \tag{18}$$

where each uppercase symbol (e.g., L_1) denotes an achiral proligands and each pair of lowercase symbols without and with an overbar (e.g., p_1 and $\overline{\mathrm{p}}_1$) denotes an enantiomeric pair of chiral proligands. The number ν is selected appropriately according to our discussions. According to Theorem 7.14 of [20], the sphericity indices ($\$_k$: a_k, c_k, and b_k) control the modes of proligand packing in the form of ligand-inventory functions as follows:

$$a_k = \mathrm{L}_1^k + \mathrm{L}_2^k + \cdots + \mathrm{L}_\nu^k \tag{19}$$

$$c_k = \mathrm{L}_1^k + \mathrm{L}_2^k + \cdots + \mathrm{L}_\nu^k + 2\mathrm{p}_1^{k/2}\overline{\mathrm{p}}_1^{k/2} + 2\mathrm{p}_2^{k/2}\overline{\mathrm{p}}_2^{k/2} + \cdots + 2\mathrm{p}_\nu^{k/2}\overline{\mathrm{p}}_\nu^{k/2} \tag{20}$$

$$b_k = \mathrm{L}_1^k + \mathrm{L}_2^k + \cdots + \mathrm{L}_\nu^k + \mathrm{p}_1^k + \mathrm{p}_2^k + \ldots + \mathrm{p}_\nu^k + \overline{\mathrm{p}}_1^k + \overline{\mathrm{p}}_2^k + \cdots + \overline{\mathrm{p}}_\nu^k. \tag{21}$$

The ligand-inventory functions (Eqs. 19–21) are introduced into the CI-CF represented by Eq. 9 according to Theorem 7.21 of [20]. The expansion of the resulting equation give a generating function, in which the coefficient of each monomial represents the number of isomers with the corresponding composition, where each pair of (self-)enantiomers is counted once. Note that a pair of self-enantiomers means an achiral 3D structure.

In a similar way, the introduction of the ligand-inventory functions (Eqs. 19–21) into the CI-CF represented by Eq. 11 gives a generating function, in which the coefficient of each monomial represents the number of achiral 3D structures with the corresponding composition (Theorem 7.27 of [20]). On the other hand, the introduction of the ligand-inventory functions (Eqs. 19–21) into the CI-CF represented by Eq. 14 gives a generating function, in which the coefficient of each monomial represents the number of enantiomeric pairs of chiral 3D structures (Theorem 7.29 of [20]).

As a typical example, let us calculate the numbers of oxirane 3D structures by using the following ligand inventory:

$$\boldsymbol{L} = \{\mathrm{A}, \mathrm{B}, \mathrm{C}, \mathrm{D}; \mathrm{p}, \mathrm{q}, \mathrm{r}, \mathrm{s}; \overline{\mathrm{p}}, \overline{\mathrm{q}}, \overline{\mathrm{r}}, \overline{\mathrm{s}}\}. \tag{22}$$

The ligand-inventory functions (Eqs. 19–21) are obtained from this inventory, where a pair of an lowercase letter and the corresponding uppercase letter (e.g., p/P) is used in place of a pair of lowercase letters without and with an overbar (e.g., p/$\overline{\mathrm{p}}$). The following codes are stored in a work file named C2v-Enum.gap, which is placed in the above-mentioned directory c:/fujita0/calcCICF.

```
#Read("c:/fujita0/calcCICF/C2v-Enum.gap");
LogTo("c:/fujita0/calcCICF/C2v-Enumlog.txt");

Read("c:/fujita0/calcCICF/CICFgen.gapfunc");

A := Indeterminate(Rationals, "A"); B := Indeterminate(Rationals, "B");
C := Indeterminate(Rationals, "C"); D := Indeterminate(Rationals, "D");
p := Indeterminate(Rationals, "p"); P := Indeterminate(Rationals, "P");
q := Indeterminate(Rationals, "q"); Q := Indeterminate(Rationals, "Q");
r := Indeterminate(Rationals, "r"); R := Indeterminate(Rationals, "R");
s := Indeterminate(Rationals, "s"); S := Indeterminate(Rationals, "S");

a_1 := Indeterminate(Rationals, "a_1");
a_2 := Indeterminate(Rationals, "a_2");
a_3 := Indeterminate(Rationals, "a_3");
a_4 := Indeterminate(Rationals, "a_4");
c_2 := Indeterminate(Rationals, "c_2");
c_4 := Indeterminate(Rationals, "c_4");
b_1 := Indeterminate(Rationals, "b_1");
b_2 := Indeterminate(Rationals, "b_2");
b_3 := Indeterminate(Rationals, "b_3");
b_4 := Indeterminate(Rationals, "b_4");

C2v := Group((1,3)(2,4), (1,2)(3,4)(5,6)); #C2v

CICF_C2v := CalcCICF(C2v, 4, 6);
Print("CICF_C2v␣:=␣", CICF_C2v, "\n");

 aa_1 := A + B + C + D; aa_2 := A^2 + B^2 + C^2 + D^2;
 aa_3 := A^3 + B^3 + C^3 + D^3; aa_4 := A^4 + B^4 + C^4 + D^4;
 bb_1 := A + B + C + D + p + q + r + s + P + Q + R + S;
 bb_2 := A^2 + B^2 + C^2 + D^2
        + p^2 + q^2 + r^2 + s^2 + P^2 + Q^2 + R^2 + S^2;
 bb_3 := A^3 + B^3 + C^3 + D^3
        + p^3 + q^3 + r^3 + s^3 + P^3 + Q^3 + R^3 + S^3;
 bb_4 := A^4 + B^4 + C^4 + D^4
        + p^4 + q^4 + r^4 + s^4 + P^4 + Q^4 + R^4 + S^4;
 cc_2 := A^2 + B^2 + C^2 + D^2 + 2*p*P + 2*q*Q + 2*r*R + 2*s*S;
 cc_4 := A^4 + B^4 + C^4 + D^4
        + 2*p^2*P^2 + 2*q^2*Q^2 + 2*r^2*R^2 + 2*s^2*S^2;
```

```
f_C2v := Value(CICF_C2v,
[a_1, a_2, a_3, a_4, b_1, b_2, b_3, b_4, c_2, c_4],
[aa_1, aa_2, aa_3, aa_4, bb_1, bb_2, bb_3, bb_4, cc_2, cc_4]);
Print("f_C2v␣:=␣", f_C2v, "\n");
LogTo();
```

The first line is copied and pasted after the gap> prompt of the command-prompt window of the Windows system. Thereby, the above codes contained in the file C2v-Enum.gap are successively executed after the loading of CICFgen.gapfunc. The set of ligand-inventory functions (aa_1 etc.) is introduced into the sphericity indices (a_1 etc.) of the calculated CI-CF for $\boldsymbol{C}_{2v}$ (CICF_C2v). The resulting generating function after expansion (f_C2v) is written down into the log file named C2v-Enumlog.txt (an appropriate name) as follows:

```
CICF_C2v := 1/4*b_1^4+1/2*c_2^2+1/4*b_2^2
f_C2v := A^4+A^3*B+A^3*C+A^3*D+A^3*p+A^3*P+A^3*q+A^3*Q+A^3*r+A^3*R+A^3*s
+A^3*S+3*A^2*B^2+3*A^2*B*C+3*A^2*B*D+3*A^2*B*p+3*A^2*B*P+3*A^2*B*q
+3*A^2*B*Q+3*A^2*B*r+3*A^2*B*R+3*A^2*B*s+3*A^2*B*S+3*A^2*C^2+3*A^2*C*D
+3*A^2*C*p+3*A^2*C*P+3*A^2*C*q+3*A^2*C*Q+3*A^2*C*r+3*A^2*C*R+3*A^2*C*s
+3*A^2*C*S+3*A^2*D^2+3*A^2*D*p+3*A^2*D*P+3*A^2*D*q+3*A^2*D*Q+3*A^2*D*r
+3*A^2*D*R+3*A^2*D*s+3*A^2*D*S+2*A^2*p^2+5*A^2*p*P+3*A^2*p*q+3*A^2*p*Q
+3*A^2*p*r+3*A^2*p*R+3*A^2*p*s+3*A^2*p*S+2*A^2*P^2+3*A^2*P*q+3*A^2*P*Q
+3*A^2*P*r+3*A^2*P*R+3*A^2*P*s+3*A^2*P*S+2*A^2*q^2+5*A^2*q*Q+3*A^2*q*r
+3*A^2*q*R+3*A^2*q*s+3*A^2*q*S+2*A^2*Q^2+3*A^2*Q*r+3*A^2*Q*R+3*A^2*Q*s
+3*A^2*Q*S+2*A^2*r^2+5*A^2*r*R+3*A^2*r*s+3*A^2*r*S+2*A^2*R^2+3*A^2*R*s
+3*A^2*R*S+2*A^2*s^2+5*A^2*s*S+2*A^2*S^2+A*B^3+3*A*B^2*C+3*A*B^2*D
+3*A*B^2*p+3*A*B^2*P+3*A*B^2*q+3*A*B^2*Q+3*A*B^2*r+3*A*B^2*R+3*A*B^2*s
+3*A*B^2*S+3*A*B*C^2+6*A*B*C*D+6*A*B*C*p+6*A*B*C*P+6*A*B*C*q+6*A*B*C*Q
+6*A*B*C*r+6*A*B*C*R+6*A*B*C*s+6*A*B*C*S+3*A*B*D^2+6*A*B*D*p+6*A*B*D*P
+6*A*B*D*q+6*A*B*D*Q+6*A*B*D*r+6*A*B*D*R+6*A*B*D*s+6*A*B*D*S+3*A*B*p^2
+6*A*B*p*P+6*A*B*p*q+6*A*B*p*Q+6*A*B*p*r+6*A*B*p*R
(omitted)
+1/2*p^4+p^3*P+p^3*q+p^3*Q+p^3*r+p^3*R+p^3*s+p^3*S+4*p^2*P^2+3*p^2*P*q
+3*p^2*P*Q+3*p^2*P*r+3*p^2*P*R+3*p^2*P*s+3*p^2*P*S+2*p^2*q^2+3*p^2*q*Q
+3*p^2*q*r+3*p^2*q*R+3*p^2*q*s+3*p^2*q*S+2*p^2*Q^2+3*p^2*Q*r+3*p^2*Q*R
+3*p^2*Q*s+3*p^2*Q*S+2*p^2*r^2+3*p^2*r*R+3*p^2*r*s+3*p^2*r*S+2*p^2*R^2
+3*p^2*R*s+3*p^2*R*S+2*p^2*s^2+3*p^2*s*S+2*p^2*S^2+p*P^3+3*p*P^2*q
+3*p*P^2*Q+3*p*P^2*r+3*p*P^2*R+3*p*P^2*s+3*p*P^2*S+3*p*P*q^2+10*p*P*q*Q
+6*p*P*q*r+6*p*P*q*R+6*p*P*q*s+6*p*P*q*S+3*p*P*Q^2
(omitted)
```

This result is consistent with the previous result which has been reported as a tabular form (Table 5 of [23]). For example, the coefficient of the term A^4 is equal to 1, which indicates that there appears one oxirane as a 3D structure with the composition A^4. The term 6*A*B*C*D indicates that there appear six pairs of enantiomeric oxiranes with the composition ABCD (cf. Figure 8 of [24]). The term 1/2*p^4 corresponds to 1 ×

$\frac{1}{2}(\mathrm{p}^4 + \overline{\mathrm{p}}^4)$, which means the presence of one pair of enantiomeric oxiranes with the compositions p^4 and $\overline{\mathrm{p}}^4$. The term `10*p*P*q*Q` ($10\mathrm{p}\overline{\mathrm{p}}\mathrm{q}\overline{\mathrm{q}}$) indicates the presence of 10 pairs of (self-)enantiomers, which are categorized into two pairs of enantiomers and 8 achiral oxiranes (cf. Figure 16 of [24]).

The codes stored in the work file `C2v-Enum.gap` are effective to calculated the generating functions of other groups, if the definition of the group $\boldsymbol{C}_{2v}$ (due to the GAP function `Group`) and the descriptions of the remaining codes are replaced by those of the other groups to be examined (e.g., Table 2). Thereby, the corresponding CI-CFs are calculated as found in Table 3, so that they are used to calculate the respective generating functions.

3.5 Cycle Indices Without Chirality Fittingness (Pólya's CIs)

The GAP system originally supports a function (named `CycleIndex`) for calculating Pólya's cycle indices (CIs). Pólya's cycle indices (CIs) can be regarded as degenerate cases of Fujita's CI-CFs [22], i.e., cycle indices *without* chirality fittingness, so that they are derived from Fujita's CI-CFs by putting $s_k = a_k = c_k = b_k$. This derivation is confirmed by the following source code (stored in the file named `Td-Polya4.gap`), where $\boldsymbol{T}_{\widetilde{\sigma}}$ (denoted by the symbol `Ts` in this source code) is isomorphic to the symmetric group of degree 4 ($\boldsymbol{S}^{[4]}$).

```
#Read("c:/fujita0/calcCICF/Td-Polya4.gap");
LogTo("c:/fujita0/calcCICF/Td-Polya4log.txt");

s_1 := Indeterminate(Rationals, "s_1");
s_2 := Indeterminate(Rationals, "s_2");
s_3 := Indeterminate(Rationals, "s_3");
s_4 := Indeterminate(Rationals, "s_4");

Ts := Group((1,2)(3,4), (2,3,4), (3,4)); #Ts
CI_Ts := CycleIndex(Ts); #Polya's CI
Print("CI_Ts␣:=␣␣␣", CI_Ts, "\n");

Read("c:/fujita0/calcCICF/CICFgen.gapfunc");

Td := Group((1,2)(3,4), (2,3,4), (3,4)(5,6)); #Td
CICF_Td := CalcCICF(Td, 4, 6); #Fujita's CI-CF
Print("CICF_Td␣:=␣", CICF_Td, "\n");

a_1 := Indeterminate(Rationals, "a_1");
a_2 := Indeterminate(Rationals, "a_2");
a_3 := Indeterminate(Rationals, "a_3");
a_4 := Indeterminate(Rationals, "a_4");
```

```
c_2 := Indeterminate(Rationals, "c_2");
c_4 := Indeterminate(Rationals, "c_4");
b_1 := Indeterminate(Rationals, "b_1");
b_2 := Indeterminate(Rationals, "b_2");
b_3 := Indeterminate(Rationals, "b_3");
b_4 := Indeterminate(Rationals, "b_4");

CICF_Td_s := Value(CICF_Td,
[a_1, a_2, a_3, a_4, b_1, b_2, b_3, b_4, c_2, c_4],
[s_1, s_2, s_3, s_4, s_1, s_2, s_3, s_4, s_2, s_4]);
                                        #degeneration of CI-CF to CI
Print("CICF_Td_s:=", CICF_Td_s, "\n");

Print("validity␣check:␣", CICF_Td_s = CI_Ts, "\n");

LogTo();
```

The degenerate CI-CF denoted by the symbol `CICF_Td_s` is calculated by putting $s_k = a_k = c_k = b_k$ in Fujita's CI-CF denoted by the symbol `CICF_Td`, where the GAP function `Value` is used to accomplish the substitution of sphericity indices (a_k, c_k, and b_k) by dummy variables s_k. The resulting degenerate CI-CF (`CICF_Td_s`) is identical with the Pólya's CI denoted by the symbol `CI_Ts`, which is calculated by the function `CycleIndex`. It follows that the validity check is printed out as being true, as follows:

```
CI_Ts := 1/24*s_1^4+1/4*s_1^2*s_2+1/3*s_1*s_3+1/8*s_2^2+1/4*s_4
CICF_Td := 1/24*b_1^4+1/4*a_1^2*c_2+1/3*b_1*b_3+1/8*b_2^2+1/4*c_4
CICF_Td_s:=1/24*s_1^4+1/4*s_1^2*s_2+1/3*s_1*s_3+1/8*s_2^2+1/4*s_4
validity check: true
```

which is output into the log file named `Td-Polya4log.txt`.

The group $\boldsymbol{T}_{\widetilde{\sigma}}$ (denoted by the symbol `Ts`) can be regarded as an RS-permutation group according to Fujita's stereoisogram approach [21]. The CI-CF of the group $\boldsymbol{T}_{\widetilde{\sigma}}$ is calculated by using the function `CalcCICF` as follows:

```
gap> Read("c:/fujita0/calcCICF/CICFgen.gapfunc");
gap> Ts := Group((1,2)(3,4), (2,3,4), (3,4));
Group([ (1,2)(3,4), (2,3,4), (3,4) ])
gap> CICF_Ts := CalcCICF(Ts, 4, 4);
1/24*b_1^4+1/4*b_1^2*b_2+1/3*b_1*b_3+1/8*b_2^2+1/4*b_4
```

The resulting CI-CF (`CICF_Ts`) contains the sphericity indices of one kind (b_k) and becomes identical with Pólya's CI if we place $b_k = s_k$. It should be noted, however, that the CI-CF (`CICF_Ts` as a polynomial in indeterminates b_k) takes sphericities of cycles into consideration, whereas Pólya's CI (`CI_Ts` a polynomial in indeterminates s_k) ignores sphericities of cycles [22].

As a result, the ligand-inventory function for b_k (Eq. 21 for CICF_Ts) differentiates between p_d and $\overline{\mathrm{p}}_d$ $(d = 1, 2, \ldots, \nu)$, while the ligand-inventory function for s_k (for CI_Ts or CICF_Td_s) does not differentiate between them:

$$s_k = \mathrm{L}_1^k + \mathrm{L}_2^k + \cdots + \mathrm{L}_\nu^k + \ddot{\mathrm{p}}_1^k + \ddot{\mathrm{p}}_2^k + \ldots + \ddot{\mathrm{p}}_\nu^k, \tag{23}$$

where the symbol $\ddot{\mathrm{p}}_d$ represents a graph corresponding to a pair of enantiomeric proligands p_d and $\overline{\mathrm{p}}_d$ $(d = 1, 2, \ldots, \nu)$. Thereby, the former CI-CF (CICF_Ts) is concerned with 3D structures, while the latter CI (CI_Ts or CICF_Td_s) is concerned with graphs (or 2D structures).

4 Conclusion

After a mirror-permutation representation of a given point group is defined to differentiate between rotations and (roto)reflections, a combined-permutation representation is developed as a computer-oriented representation of the point group. Such a combined-permutation representation is regarded as a permutation group, which is generated from an appropriate generators by using the Group function of the GAP system. Thereby, the program (CalcCICF) for generating cycle indices with chirality fittingness (CI-CFs) is developed as a function of the GAP system. This program is used to calculate generating functions for combinatorial enumeration of 3D structures under point groups. The program (CalcCICF_A) for calculating the number of achiral 3D structures and the program (CalcCICF_E) for calculating the number of enantiomeric pairs of chiral 3D structures are also developed as functions of the GAP system. A practical procedure for combinatorial enumeration of 3D structures is described on the basis of the GAP system.

Appendix A. Source Code of CICFgen.gapfunc for Calculating CI-CFs

The following codes are stored in the file named CICFgen.gapfunc, which is loaded during the calculation of CI-CFs.

```
##CICFgen.gapfunc
##for calculating a CI-CF for Fujita's Proligand Method

##############
# CICF for G #
##############
```

```
CalcCICF := function(G, degree, degreefull)
local i, j, Orig_Grp, l_elements, no_elements, DegGr, DegCGr, CICF,
temp_cycstr, temp_cycstrX, AchOrCh, tempSI, tempSIX, tempSIY, tempCICF;
Orig_Grp := G; #group to be examined
l_elements := Elements(Orig_Grp); #elements of Orig_Grp
#Display(l_elements); #for debug
no_elements := Size(l_elements); #order of Orig_Grp
#Display(no_elements); #for debug
#Degrees of Permutations
DegGr := degreefull; #degree of chiral and achiral parts,
                     #e.g. 6: [1,2,3,4,5,6]
DegCGr := degree; #degree of chiral parts, e.g. 4: [1,2,3,4]
#Chirality/achirality is determined by the final digit of temp_cycstrX
CICF := 0;
for i in [1..no_elements] do
temp_cycstr := CycleLengths(l_elements[i], [1..DegCGr]); #degree
temp_cycstrX := CycleLengths(l_elements[i], [1..DegGr]); #full degree
#Display(temp_cycstr); #for debug
#Display(temp_cycstrX); #for debug
AchOrCh := temp_cycstrX[Size(temp_cycstrX)]; #chiral 1; achiral 2
#Print("AchOrCh=", AchOrCh, "\n"); #for debug
#Determination of sphericity indices (SIs) of cycles
tempCICF := 1;
for j in [1..Size(temp_cycstr)] do
if DegGr = DegCGr then
 tempSI := ["b_", temp_cycstr[j]]; #hemispheric cycle
else
 if AchOrCh = 1 then
   tempSI := ["b_", temp_cycstr[j]]; #hemispheric cycle
 else
  if IsOddInt(temp_cycstr[j]) then
    tempSI := ["a_", temp_cycstr[j]]; #homospheric cycle
   else
    tempSI := ["c_", temp_cycstr[j]]; #enantiospheric cycle
  fi;
 fi;
fi;
#Calculation of products of sphericity indices (PSIs)
tempSIX := JoinStringsWithSeparator(tempSI, "");
tempSIY := Indeterminate(Rationals, tempSIX);
tempCICF := tempCICF*tempSIY;
od;
#Display(tempCICF); #for debug
#Calculation of CI-CF (cycle index with chirality fittingness)
CICF := CICF + (1/no_elements)*tempCICF;
od;
#Display(CICF); #for debug
return CICF;
end; # end of function CalcCICF

##############################
# Achiral Part of CICF for G #
##############################
```

```
CalcCICF_A := function(G, degree, degreefull)
local i, j, Orig_Grp, l_elements, no_elements, DegGr, DegCGr, CICF_A,
temp_cycstr, temp_cycstrX, AchOrCh, tempSI,
tempSIX, tempSIY, tempCICF;
Orig_Grp := G; #group to be examined
l_elements := Elements(Orig_Grp); #elements of Orig_Grp
#Display(l_elements); #for debug
no_elements := Size(l_elements); #order of Orig_Grp
#Display(no_elements); #for debug
#Degrees of Permutations
DegGr := degreefull; #degree of chiral and achiral parts,
                     #e.g. 6: [1,2,3,4,5,6]
DegCGr := degree; #degree of chiral parts, e.g. 4: [1,2,3,4]
if DegGr = DegCGr then
return false;
break;
fi;
#Chirality/achirality is determined by the final digit of temp_cycstrX
CICF_A := 0;
for i in [1..no_elements] do
temp_cycstr := CycleLengths(l_elements[i], [1..DegCGr]); #degree
temp_cycstrX := CycleLengths(l_elements[i], [1..DegGr]); #full degree
#Display(temp_cycstr); #for debug
#Display(temp_cycstrX); #for debug
#Display(Size(temp_cycstrX)); #for debug

AchOrCh := temp_cycstrX[Size(temp_cycstrX)]; #chiral 1; achiral 2
#Print("AchOrCh=", AchOrCh, "\n"); #for debug

#Determination of sphericity indices (SIs) of cycles
if AchOrCh = 1 then
tempCICF := 0;
else
tempCICF := 1;
for j in [1..Size(temp_cycstr)] do
if DegGr = DegCGr then
 else
  if IsOddInt(temp_cycstr[j]) then
    tempSI := ["a_", temp_cycstr[j]]; #homospheric cycle
   else
    tempSI := ["c_", temp_cycstr[j]]; #enantiospheric cycle
  fi;
#Calculation of products of sphericity indices (PSIs)
tempSIX := JoinStringsWithSeparator(tempSI, "");
tempSIY := Indeterminate(Rationals, tempSIX);
tempCICF := tempCICF*tempSIY;
fi;
od;
fi;
#Display(tempCICF); #for debug
#Calculation of CI-CF (cycle index with chirality fittingness)
CICF_A := CICF_A + (2/no_elements)*tempCICF;
od;
#Display(CICF_A); #for debug
```

```
return CICF_A;
end; # end of function CalcCICF_A

##############################
# Chiral Part of CICF for G #
##############################
CalcCICF_E := function(G, degree, degreefull)
local CICF, CICF_A, CICF_E;
if degree = degreefull then
return false;
break;
fi;
CICF := CalcCICF(G, degree, degreefull);
CICF_A := CalcCICF_A(G, degree, degreefull);
CICF_E := CICF - CICF_A;
return CICF_E;
end; # end of function CalcCICF_E
```

References

[1] G. Pólya, Un problème combinatoire général sur les groupes de permutations et le calcul du nombre des isomères des composés organiques, *Compt. Rend. Acad. Sci. Paris* **201** (1935) 1167–1169.

[2] G. Pólya, Kombinatorische Anzahlbestimmungen für Gruppen, Graphen und chemische Verbindungen, *Acta Math.* **68** (1937) 145–254.

[3] G. Pólya, R. C. Read, *Combinatorial Enumeration of Groups, Graphs, and Chemical Compounds*, Springer, New York, 1987.

[4] H. Hosoya, Kagaku to topology no deai (When chemistry meats topology), *Kagaku no Ryoiki* **26** (1972) 989–1001.

[5] D. H. McDaniel, A restatement of Pólya's theorem, *Inorg. Chem.* **11** (1972) 2678–2682.

[6] D. H. Rouvray, Isomer enumeration methods, *Chem. Soc. Rev.* **3** (1974) 355–372.

[7] O. E. Polansky, Pólya's method for the enumeration of isomers, *MATCH Commun. Math. Comput. Chem.* **1** (1975) 11–31.

[8] K. Balasubramanian, Application of combinatorics and graph theory to spectroscopy and quantum chemistry, *Chem. Rev.* **85** (1985) 599–618.

[9] N. L. Biggs, E. K. Lloyd, R. J. Wilson, *Graph Theory 1736–1936*, Oxford Univ. Press, Oxford, 1976.

[10] A. T. Balaban (Ed.), *Chemical Applications of Graph Theory*, Academic Press, London, 1976.

[11] G. Pólya, R. E. Tarjan, D. R. Woods, *Notes on Introductory Combinatorics*, Birkhäuser, Boston, 1983.

[12] N. Trinajstić, *Chemical Graph Theory*, CRC Press, Boca Raton, 1992.

[13] http://www.gap-system.org/Manuals/doc/ref/chap41.html.

[14] http://www.maplesoft.com/support/help/Maple/view.aspx?path=GroupTheory %2FCycleIndexPolynomial.

[15] https://reference.wolfram.com/language/ref/CycleIndexPolynomial.html.

[16] S. Fujita, Graphs to chemical structures 1. Sphericity indices of cycles for stereochemical extension of Pólya's theorem, *Theor. Chem. Acc.* **113** (2005) 73–79.

[17] S. Fujita, Graphs to chemical structures 2. Extended sphericity indices of cycles for stereochemical extension of Pólya's coronas, *Theor. Chem. Acc.* **113** (2005) 80–86.

[18] S. Fujita, Graphs to chemical structures 3. General theorems with the use of different sets of sphericity indices for combinatorial enumeration of nonrigid stereoisomers, *Theor. Chem. Acc.* **115** (2006) 37–53.

[19] S. Fujita, Numbers of alkanes and monosubstituted alkanes. A long–standing interdisciplinary problem over 130 years, *Bull. Chem. Soc. Jpn.* **83** (2010) 1–18.

[20] S. Fujita, *Combinatorial Enumeration of Graphs, Three–Dimensional Structures, and Chemical Compounds*, Univ. Kragujevac, Kragujevac, 2013.

[21] S. Fujita, *Mathematical Stereochemistry*, De Gruyter, Berlin, 2015.

[22] S. Fujita, Sphericities of cycles. What Pólya's theorem is deficient in for stereoisomer enumeration, *Croat. Chem. Acta* **79** (2006) 411–427.

[23] S. Fujita, Combinatorial approach to group hierarchy for stereoskeletons of ligancy 4, *J. Math. Chem.* **53** (2015) 1010–1053.

[24] S. Fujita, Stereoisograms for three-membered heterocycles. II. Chirality, *RS*-stereogenicity, and *ortho*-stereogenicity on the basis of the proligand–promolecule model, *J. Math. Chem.* **53** (2015) 305–352.