MATCH
Communications in Mathematical and in Computer Chemistry

MATCH Commun. Math. Comput. Chem. **77** (2017) 409-442

ISSN 0340 - 6253

Computer–Oriented Representations of $\boldsymbol{O_h}$-Skeletons for Supporting Combinatorial Enumeration by Fujita's Proligand Method. GAP Calculation of Cycle Indices with Chirality Fittingness (CI-CFs)

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 May 14, 2016)

Abstract

In order to enhance the applicability 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), functions for calculating cycle indices with chirality fittingness (CI-CFs) have been developed on the basis of the GAP (Groups, Algorithms, Programming) system. After a mirror-permutation representations is defined to characterize a (roto)reflection, a combined-permutation representation is defined as a computer-oriented representation of a point group. Such a computer-oriented representation has been used to develop the GAP functions for calculating CI-CFs, where conjugacy classes are taken into consideration for the purpose of simplifying calculation processes. The source program containing these GAP functions is attached as an appendix. The resulting CI-CFs have been applied to combinatorial enumeration of promolecules derived from $\boldsymbol{O_h}$-skeletons (an octahedron, a cube, a cuboctahedron, a truncated octahedron, and a truncated hexahedron). A GAP program for calculating the numbers of cubane derivatives is attached as another appendix in order to illustrate a straightforward procedure of Fujita's proligand method.

1 Introduction

Gross enumeration of chemical compounds has been widely conducted by using Pólya's theorem, as summarized in reviews [1–5] and books [6–9]. As the title of the English translation [10] of Pólya's original article [11] ("Combinatorial Enumeration of Groups, Graphs, and Chemical Compounds") indicates, Pólya's theorem aims at the enumeration of chemical compounds *as graphs*, where it is based on cycle indices (CIs) as key polynomials for enumerating graphs.

On the other hand, the title of a book on Fujita's proligand method [12] ("Combinatorial Enumeration of Graphs, Three-Dimensional Structures, and Chemical Compounds") indicates that its main target is the enumeration of chemical compounds *as three-dimensional (3D) structures*, where it is based on cycle indices with chirality fittingness (CI-CFs) as key polynomials for enumerating 3D structures.

Computer systems for supporting group theory (e.g., the GAP (Groups, Algorithms, Programming) system [13], the Maple system [14], and the Mathematica system [15]) have laid stress on enumeration of graphs without considering ligand reflections, where Pólya's theorem [10, 11] has been mainly used as an enumeration device. As a result, they support a function of calculating CIs *without chirality fittingness* (e.g., the function `CycleIndex` of the GAP system).

As a mirror of the stress laid on graphs, such computer systems have not yet equipped with functions of calculating CI-CFs, which are used to enumerate 3D structures with considering ligand reflections in Fujita's proligand method [12, 16]. For the purpose of enumerating chemical compounds as 3D structures by such computer systems, the effects of ligand reflections for characterizing chirality fittingness have recently been formulated on the basis of the concept of *sphericities*, where ligand reflections are specified by mirror-permutation representations. Thereby, a set of GAP functions for calculating CI-CFs has been developed after formulating computer-oriented representations of point groups, as reported recently in this journal [17].

To pursue further applications based on the GAP system, the set of GAP functions should be linked to group-theoretical properties such as conjugacy classes and homomorphism related to the computer-oriented representations. The purpose of this article is to develop another set of GAP functions for calculating CI-CFs after examining conjugacy classes and homomorphism by using several $\boldsymbol{O}_h$-skeletons as probes. Practical proce-

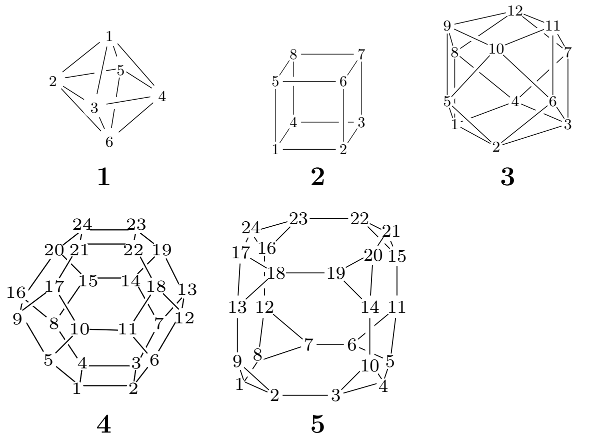


Figure 1. $\boldsymbol{O}_h$-Skeletons: Octahedron **1** (*OC*-6), cube **2** (*CU*-8), cuboctahedron **3**, truncated octahedron **4**, and truncated hexahedron (truncated cube) **5**.

dures for calculating CI-CFs will be demonstrated to enhance the applicability of Fujita's proligand method.

2 Computer-Oriented Representations for Enumeration Under Point Groups

For the sake of self-containment, several items formulated in a recent article [17] are adopted after appropriate modifications to treat $\boldsymbol{O}_h$-skeletons listed in Figure 1.

2.1 Coset Representations Assigned to $\boldsymbol{O}_h$-Skeletons

Let us first examine an octahedral skeleton **1** as a typical example of an $\boldsymbol{O}_h$-skeleton, where the six positions construct an orbit governed by a coset representation $\boldsymbol{O}_h(/\boldsymbol{C}_{4v})$. The concrete form of $\boldsymbol{O}_h(/\boldsymbol{C}_{4v})$ can be obtained algebraically by starting from its multiplication table and the corresponding coset decomposition of the point group $\boldsymbol{O}_h$ (order: $|\boldsymbol{O}_h| = 48$) by the subgroup $\boldsymbol{C}_{4v}$ (order: $|\boldsymbol{C}_{4v}| = 8$) [18–20]. The coset representation $\boldsymbol{O}_h(/\boldsymbol{C}_{4v})$ is shown as a set of products of cycles in Figure 2, which is a modification of Figure 2 of [20] (the numbering of the two-fold axes has been changed).

Geometrically speaking, the six positions of **1** are equivalent under the action of the point group $\boldsymbol{O}_h$, so that they belong to an orbit (equivalence class) governed by the

		$\boldsymbol{O}_h(/\boldsymbol{C}_{4v})$, $\boldsymbol{P}^{(\mathbf{x}\overline{\mathbf{x}})}_{O_h}$	$\boldsymbol{P}^{(\chi)}_{O_h}$	
operation		product of cycles		PSI
	I	(1)(2)(3)(4)(5)(6)	(7)(8)	b_1^6
	$C_{2(3)}$	(1)(2 4)(3 5)(6)	(7)(8)	$b_1^2b_2^2$
	$C_{2(1)}$	(1 6)(3 5)(2)(4)	(7)(8)	$b_1^2b_2^2$
	$C_{2(2)}$	(1 6)(2 4)(3)(5)	(7)(8)	$b_1^2b_2^2$
	$C_{3(1)}$	(1 3 2)(4 6 5)	(7)(8)	b_3^2
	$C_{3(3)}$	(1 4 5)(2 3 6)	(7)(8)	b_3^2
	$C_{3(2)}$	(1 4 3)(2 5 6)	(7)(8)	b_3^2
	$C_{3(4)}$	(1 2 5)(3 6 4)	(7)(8)	b_3^2
	$C^2_{3(1)}$	(1 2 3)(4 5 6)	(7)(8)	b_3^2
	$C^2_{3(3)}$	(1 5 4)(2 6 3)	(7)(8)	b_3^2
	$C^2_{3(2)}$	(1 3 4)(2 6 5)	(7)(8)	b_3^2
$\boldsymbol{O}$	$C^2_{3(4)}$	(1 5 2)(3 4 6)	(7)(8)	b_3^2
	$C'_{2(6)}$	(1 6)(2 5)(3 4)	(7)(8)	b_2^3
	$C'_{2(1)}$	(1 6)(2 3)(4 5)	(7)(8)	b_2^3
	$C'_{2(4)}$	(1 2)(3 5)(4 6)	(7)(8)	b_2^3
	$C'_{2(2)}$	(1 5)(2 4)(3 6)	(7)(8)	b_2^3
	$C'_{2(5)}$	(1 4)(2 6)(3 5)	(7)(8)	b_2^3
	$C'_{2(3)}$	(1 3)(2 4)(5 6)	(7)(8)	b_2^3
	$C^3_{4(3)}$	(1)(2 3 4 5)(6)	(7)(8)	$b_1^2b_4$
	$C_{4(3)}$	(1)(2 5 4 3)(6)	(7)(8)	$b_1^2b_4$
	$C^3_{4(1)}$	(1 5 6 3)(2)(4)	(7)(8)	$b_1^2b_4$
	$C_{4(1)}$	(1 3 6 5)(2)(4)	(7)(8)	$b_1^2b_4$
	$C_{4(2)}$	(1 4 6 2)(3)(5)	(7)(8)	$b_1^2b_4$
	$C^3_{4(2)}$	(1 2 6 4)(3)(5)	(7)(8)	$b_1^2b_4$

		$\boldsymbol{O}_h(/\boldsymbol{C}_{4v})$, $\boldsymbol{P}^{(\mathbf{x}\overline{\mathbf{x}})}_{O_h}$	$\boldsymbol{P}^{(\chi)}_{O_h}$	
operation		product of cycles		PSI
	i	$\overline{(1\ 6)(2\ 4)(3\ 5)}$	(7 8)	c_2^3
	$\sigma_{h(3)}$	$\overline{(1)(2\ 4)(3)(5)(6)}$	(7 8)	$a_1^4c_2$
	$\sigma_{h(2)}$	$\overline{(1)(2)(3\ 5)(4)(6)}$	(7 8)	$a_1^4c_2$
	$\sigma_{h(1)}$	$\overline{(1\ 6)(2)(3)(4)(5)}$	(7 8)	$a_1^4c_2$
	$S^5_{6(1)}$	$\overline{(1\ 4\ 3\ 6\ 2\ 5)}$	(7 8)	c_6
	$S^5_{6(3)}$	$\overline{(1\ 3\ 4\ 6\ 5\ 2)}$	(7 8)	c_6
	$S^5_{6(2)}$	$\overline{(1\ 5\ 4\ 6\ 3\ 2)}$	(7 8)	c_6
	$S^5_{6(4)}$	$\overline{(1\ 3\ 2\ 6\ 5\ 4)}$	(7 8)	c_6
	$S_{6(1)}$	$\overline{(1\ 5\ 2\ 6\ 3\ 4)}$	(7 8)	c_6
	$S_{6(4)}$	$\overline{(1\ 4\ 5\ 6\ 2\ 3)}$	(7 8)	c_6
	$S_{6(3)}$	$\overline{(1\ 2\ 5\ 6\ 4\ 3)}$	(7 8)	c_6
$\boldsymbol{O}i$	$S_{6(2)}$	$\overline{(1\ 2\ 3\ 6\ 4\ 5)}$	(7 8)	c_6
	$\sigma_{d(1)}$	$\overline{(1)(2\ 3)(4\ 5)(6)}$	(7 8)	$a_1^2c_2^2$
	$\sigma_{d(6)}$	$\overline{(1)(2\ 5)(3\ 4)(6)}$	(7 8)	$a_1^2c_2^2$
	$\sigma_{d(2)}$	$\overline{(1\ 3)(2)(4)(5\ 6)}$	(7 8)	$a_1^2c_2^2$
	$\sigma_{d(4)}$	$\overline{(1\ 5)(2)(4)(3\ 6)}$	(7 8)	$a_1^2c_2^2$
	$\sigma_{d(3)}$	$\overline{(1\ 4)(2\ 6)(3)(5)}$	(7 8)	$a_1^2c_2^2$
	$\sigma_{d(5)}$	$\overline{(1\ 2)(4\ 6)(3)(5)}$	(7 8)	$a_1^2c_2^2$
	$S_{4(3)}$	$\overline{(1\ 6)(2\ 3\ 4\ 5)}$	(7 8)	c_2c_4
	$S^3_{4(3)}$	$\overline{(1\ 6)(2\ 5\ 4\ 3)}$	(7 8)	c_2c_4
	$S_{4(1)}$	$\overline{(1\ 3\ 6\ 5)(2\ 4)}$	(7 8)	c_2c_4
	$S^3_{4(1)}$	$\overline{(1\ 5\ 6\ 3)(2\ 4)}$	(7 8)	c_2c_4
	$S_{4(2)}$	$\overline{(1\ 2\ 6\ 4)(3\ 5)}$	(7 8)	c_2c_4
	$S^3_{4(2)}$	$\overline{(1\ 4\ 6\ 2)(3\ 5)}$	(7 8)	c_2c_4

Figure 2. Operations of the point group $\boldsymbol{O}_h$, a coset representation $\boldsymbol{O}_h(/\boldsymbol{C}_{4v})$ as a set of products of cycles, and the corresponding set of products of sphericity indices (PSIs).

coset representation $\boldsymbol{O}_h(/\boldsymbol{C}_{4v})$. The degree of $\boldsymbol{O}_h(/\boldsymbol{C}_{4v})$ indicates the size of the orbit ($|\boldsymbol{O}_h|/|\boldsymbol{C}_{4v}| = 48/8 = 6$), which shows the number of equivalent positions in **1**. The symbol $\boldsymbol{O}_h(/\boldsymbol{C}_{4v})$ indicates that the global symmetry of **1** is the point group $\boldsymbol{O}_h$ and that the local symmetry assigned to each position of the $\boldsymbol{O}_h(/\boldsymbol{C}_{4v})$-orbit of **1** is the subgroup $\boldsymbol{C}_{4v}$. The local symmetry $\boldsymbol{C}_{4v}$ (or its conjugate subgroup in $\boldsymbol{O}_h$) is the stabilizer of one of the six positions, so that the action of $\boldsymbol{C}_{4v}$ fixes (stabilizes) the position at issue.

For example, suppose that the four-fold axis is selected to run through the positions 1 and 6 of **1**. Then, the point group $\boldsymbol{C}_{4v}$ is determined to be:

$$\boldsymbol{C}_{4v} = \{I, C_{2(3)}, C_{4(3)}, C^3_{4(3)}, \sigma_{d(1)}, \sigma_{d(6)}, \sigma_{h(2)}, \sigma_{h(3)}\}, \tag{1}$$

where the corresponding products of cycles are shown in the $\boldsymbol{O}_h(/\boldsymbol{C}_{4v})$-column of Figure 2. The point group $\boldsymbol{C}_{4v}$ (Eq. 1) fixes the position 1 (and 6) of **1**. In fact, the elements corresponding to $\boldsymbol{C}_{4v}$ (Eq. 1) commonly contain two 1-cycles, i.e., (1)(6) (or $\overline{(1)(6)}$), as listed in the $\boldsymbol{O}_h(/\boldsymbol{C}_{4v})$-column of Figure 2. These two 1-cycles show that the positions 1 and 6 are fixed under the action of $\boldsymbol{C}_{4v}$ (Eq. 1).

Each operation of $\boldsymbol{O}_h(/\boldsymbol{C}_{4v})$ is classified into a proper operation (a rotation) or an improper operation (a reflection or a rotoreflection) according to the following coset decomposition:

$$\boldsymbol{O}_h = \boldsymbol{O} + \boldsymbol{O}i = \boldsymbol{O} + \boldsymbol{O}\sigma_{h(1)}, \tag{2}$$

where the first coset $\boldsymbol{O}$ $(= \boldsymbol{O}I)$ contains rotations (the left part of Figure 2), while the second coset $\boldsymbol{O}i$ $(= \boldsymbol{O}\sigma_{h(1)})$ contains reflections and rotoreflections (the right part of Figure 2). Because each (roto)reflection is accompanied by a ligand reflection, the corresponding product of cycles is overlined in order to treat the sphericities of cycles, as shown in the right part of Figure 2.

The properties mentioned above by using an octahedral skeleton **1** can be discussed also in case of the other $\boldsymbol{O}_h$-skeletons listed in Figure 1. The eight positions of a cubic skeleton **2** construct an orbit governed by the coset representation $\boldsymbol{O}_h(/\boldsymbol{C}_{3v})$. The concrete form of $\boldsymbol{O}_h(/\boldsymbol{C}_{3v})$ has been reported in a previous report in this journal [21]. The degree of the coset representation $\boldsymbol{O}_h(/\boldsymbol{C}_{3v})$ is calculated to be $|\boldsymbol{O}_h|/|\boldsymbol{C}_{3v}| = 48/6 = 8$. The coset representation $\boldsymbol{O}_h(/\boldsymbol{C}_{3v})$ also governs the eight triangular faces of an octahedral skeleton **1**.

Suppose that the three-fold axis is selected to run through the positions 1 and 7 of **2**. Then, the subgroup $\boldsymbol{C}_{3v}$ is determined to be:

$$\boldsymbol{C}_{3v} = \{I, C_{3(1)}, C_{3(1)}^2, \sigma_{d(1)}, \sigma_{d(2)}, \sigma_{d(5)}\}. \tag{3}$$

The subgroup $\boldsymbol{C}_{3v}$ fixes the position 1 (or 7) of **2**, so that the local symmetry of the position 1 (or 7) is determined to be $\boldsymbol{C}_{3v}$. For the product of cycles for the three-fold rotation $C_{3(1)}$, i.e., $(2, 5, 4)(3, 6, 8)$ (implicitly, two 1-cycles, (1)(7)), see the correspondence listed in Table 2 described later.

The twelve positions of a cuboctahedral skeleton **3** construct an orbit governed by the coset representation $\boldsymbol{O}_h(/\boldsymbol{C}''_{2v})$. The degree of the coset representation $\boldsymbol{O}_h(/\boldsymbol{C}''_{2v})$ is calculated to be $|\boldsymbol{O}_h|/|\boldsymbol{C}''_{2v}| = 48/4 = 12$. The coset representation $\boldsymbol{O}_h(/\boldsymbol{C}''_{2v})$ also governs the twelve edges of an octahedral skeleton **1** [22] as well as the the twelve edges of a cubic skeleton **2**. Note that there are the following three kinds of subgroups:

$$\boldsymbol{C}_{2v} = \{I, C_{2(3)}, \sigma_{h(2)}, \sigma_{h(3)}\} \tag{4}$$

$$\boldsymbol{C}'_{2v} = \{I, C_{2(3)}, \sigma_{d(1)}, \sigma_{d(6)}\} \tag{5}$$

$$\boldsymbol{C}''_{2v} = \{I, C'_{2(1)}, \sigma_{h(1)}, \sigma_{d(1)}\}, \tag{6}$$

which are not conjugate to one another under the action of $\boldsymbol{O}_h$. Among them, the last one $\boldsymbol{C}''_{2v}$ (Eq. 6) is selected as the local symmetry of the cuboctahedral skeleton **3**, because $\boldsymbol{C}''_{2v}$ (Eq. 6) fixes the position 6 (or 8) of **3**. For the product of cycles for the two-fold rotation $C'_{2(1)}$, i.e., $(1,12)(2,11)(3,10)(4,9)(5,7)$ (implicitly, two 1-cycles, $(6)(8)$), see the correspondence listed in Table 2 described later.

The twenty-four positions of a truncated octahedral skeleton **4** construct an orbit governed by the coset representation $\boldsymbol{O}_h(/\boldsymbol{C}_s)$. The degree of the coset representation $\boldsymbol{O}_h(/\boldsymbol{C}_s)$ is calculated to be $|\boldsymbol{O}_h|/|\boldsymbol{C}_s| = 48/2 = 24$. The $\sigma_{h(2)}$-mirror plane of **4** is selected to contain the positions 2, 6, 18, 22, 24, 20, 8, and 4, so as to generate the following subgroup:

$$\boldsymbol{C}_s = \{I, \sigma_{h(2)}\}. \tag{7}$$

This subgroup can be adopted as the local symmetry of the $\boldsymbol{O}_h(/\boldsymbol{C}_s)$-orbit, because anyone of the positions contained in the the $\sigma_{h(2)}$-mirror plane (2, 6, 18, 22, 24, 20, 8, and 4) is fixed under the action of $\boldsymbol{C}_s$.

The twenty-four positions of a truncated hexahedral skeleton **5** construct an orbit governed by the coset representation $\boldsymbol{O}_h(/\boldsymbol{C}'_s)$. The degree of the coset representation $\boldsymbol{O}_h(/\boldsymbol{C}'_s)$ is calculated to be $|\boldsymbol{O}_h|/|\boldsymbol{C}'_s| = 48/2 = 24$. The $\sigma_{d(1)}$-mirror plane of **5** is selected to contain the positions 9, 13, 15, and 11. Thereby, the local symmetry of the $\boldsymbol{O}_h(/\boldsymbol{C}'_s)$-orbit is determined as follows:

$$\boldsymbol{C}'_s = \{I, \sigma_{d(1)}\}, \tag{8}$$

which fixes anyone of the positions contained in the the $\sigma_{d(1)}$-mirror plane (9, 13, 15, and 11).

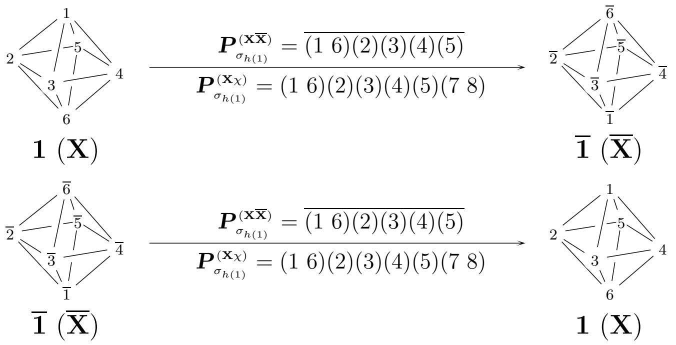
According to Fujita's unit-subduced-cycle-index (USCI) approach [18], the derivation of a coset representation $\boldsymbol{G}(/\boldsymbol{G}_i)$ is based on a coset decomposition of a given group $\boldsymbol{G}$ by its subgroup $\boldsymbol{G}_i$, where the group $\boldsymbol{G}$ is algebraically constructed by starting from the multiplication table of $\boldsymbol{G}$. As a result, the subgroup $\boldsymbol{G}_i$ runs to cover all of the subgroups of $\boldsymbol{G}$, so that the resulting coset representations $\boldsymbol{G}(/\boldsymbol{G}_i)$ are linked with one another in accord with the group-subgroup relationship of $\boldsymbol{G}$. Thus, the coset representations for characterizing $\boldsymbol{O}_h$-skeletons, i.e., $\boldsymbol{O}_h(/\boldsymbol{C}_{4v})$ for **1**, $\boldsymbol{O}_h(/\boldsymbol{C}_{3v})$ for **2**, $\boldsymbol{O}_h(/\boldsymbol{C}''_{2v})$ for **3**, $\boldsymbol{O}_h(/\boldsymbol{C}_s)$ for **4**, and $\boldsymbol{O}_h(/\boldsymbol{C}'_s)$ for **5**, are linked with one another in accord with the group-subgroup relationship of $\boldsymbol{O}_h$, so long as we obey the methodology of Fujita's USCI approach [18].

It should be noted that the subgroups represented by Eqs. 1, 3, 6, 7, and 8 can be commonly applied to all of the $\boldsymbol{O}_h$-skeletons listed in Figure 1, because they are the subgroups of the point group $\boldsymbol{O}_h$. For example, the subgroup $\boldsymbol{C}_{4v}$ (Eq. 1 or its conjugate subgroup) fixes one object (vertex or face) selected from the 6 vertices of **1**, the 6 square faces of **2**, the 6 square faces of **3**, the 8 square faces of **4**, or the 8 octagonal faces of **5**, so that the respective orbits are governed commonly by the coset representation $\boldsymbol{O}_h(/\boldsymbol{C}_{4v})$ (degree: $|\boldsymbol{O}_h| = |\boldsymbol{C}_{4v}| = 48/8 = 6$). Similarly, the subgroup $\boldsymbol{C}_{3v}$ (Eq. 3 or its conjugate subgroup) fixes one object (vertex or face) selected from the 8 triangular faces of **1**, the 8 vertices of **2**, the 8 triangular faces of **3**, the 8 hexagonal faces of **4**, or the 8 triangular faces of **5**, so that the respective orbits are governed commonly by the coset representation $\boldsymbol{O}_h(/\boldsymbol{C}_{3v})$ (degree: $|\boldsymbol{O}_h| = |\boldsymbol{C}_{3v}| = 48/6 = 8$).

On the other hand, the present approach described below adopts a different way on the basis of the GAP function `Group`, where each coset representation is regarded as a permutation group generated from an appropriate set of generators. So long as a permutation group is constructed by using the GAP function `Group`, such a permutation group as corresponding to a coset representation $\boldsymbol{G}(/\boldsymbol{G}_i)$ (e.g., $\boldsymbol{O}_h(/\boldsymbol{C}_{4v})$) is not linked initially with another permutation group corresponding to $\boldsymbol{G}(/\boldsymbol{G}'_i)$ (e.g., $\boldsymbol{O}_h(/\boldsymbol{C}_{3v})$). Conceptually speaking, the two permutation groups constructed by using the GAP function `Group` are independent of each other. Hence, it is necessary to clarify the correspondence between one permutation group for $\boldsymbol{G}(/\boldsymbol{G}_i)$ and the other for $\boldsymbol{G}(/\boldsymbol{G}'_i)$ in a subsequent step (cf. Tables 2 and 3).

2.2 Mirror-Permutation Representations

Figure 3 illustrates the effects of the reflection $\sigma_{h(1)}$ ($\sim \boldsymbol{P}^{(\mathbf{x}\overline{\mathbf{x}})}_{\sigma_{h(1)}} = \overline{(1\ 6)(2)(3)(4)(5)}$), where an octahedral skeleton **1** is converted into its enantiomer $\overline{\mathbf{1}}$ and vice versa. When we focus



$\boldsymbol{P}^{(\mathbf{X}\overline{\mathbf{X}})}_{\sigma_{h(1)}} = \overline{(1\ 6)(2)(3)(4)(5)}$
$\boldsymbol{P}^{(\mathbf{X}\chi)}_{\sigma_{h(1)}} = (1\ 6)(2)(3)(4)(5)(7\ 8)$
$\mathbf{1}$ ($\mathbf{X}$)
$\overline{\mathbf{1}}$ ($\overline{\mathbf{X}}$)

Figure 3. Reflections for an Octahedral Skeleton

our attention on the six positions numbered sequentially, such a (roto)reflection as $\sigma_{h(1)}$ implies a mirror transformation, which converts a numbered set of positions:

$$\mathbf{X} = \{1, 2, \ldots, 6\} \tag{9}$$

into a mirror-numbered set of positions:

$$\overline{\mathbf{X}} = \{\overline{1}, \overline{2}, \ldots, \overline{6}\} \tag{10}$$

and vice versa. It follows that $\mathbf{X}$ (or $\overline{\mathbf{X}}$) represents local chirality.

In order to formulate the effects of ligand reflections implied by (roto)reflections, we consider a set of local chiralities:

$$\chi = \{\mathbf{X}, \overline{\mathbf{X}}\} = \{7, 8\}, \tag{11}$$

where the selection of the numbers (7 and 8) depends on the size of $\mathbf{X}$. The action of $\boldsymbol{O}_h$ on the set χ is determined by a permutation representation $\boldsymbol{P}^{(\chi)}_{\boldsymbol{O}_h}$, the elements of which are represented as follows:

$$\boldsymbol{P}^{(\chi)}_{\mathrm{G}} = \begin{pmatrix} \mathbf{X} & \overline{\mathbf{X}} \\ \mathbf{X} & \overline{\mathbf{X}} \end{pmatrix} = \begin{pmatrix} 7 & 8 \\ 7 & 8 \end{pmatrix} = (7)(8) \quad \text{for } G \in \boldsymbol{O}\text{: rotations} \tag{12}$$

$$\boldsymbol{P}^{(\chi)}_{\mathrm{G}} = \begin{pmatrix} \mathbf{X} & \overline{\mathbf{X}} \\ \overline{\mathbf{X}} & \mathbf{X} \end{pmatrix} = \begin{pmatrix} 7 & 8 \\ 8 & 7 \end{pmatrix} = (7\ 8) \quad \text{for } G \in \boldsymbol{O}i\text{: (roto)reflections} \tag{13}$$

The permutation representation $\boldsymbol{P}^{(\chi)}_{\boldsymbol{O}_h}$ is called *a mirror-permutation representation* [17], which can be used to evaluate the effects of ligand reflections. Note that such a mirror-permutation representation corresponds to a coset representation $\boldsymbol{O}_h(/\boldsymbol{O})$ of degree 2.

2.3 Groups Corresponding to Combined Representations

By omitting overline symbols from the respective permutations of $\boldsymbol{P}_{\boldsymbol{O}_h}^{(\mathbf{x}\overline{\mathbf{x}})}$ $(= \boldsymbol{O}_h(/\boldsymbol{C}_{4v}))$, we construct another permutation representation $\boldsymbol{P}_{\boldsymbol{O}_h}^{(\mathbf{x})}$. Then, the resulting permutation representation $\boldsymbol{P}_{\boldsymbol{O}_h}^{(\mathbf{x})}$ is combined with the mirror-permutation representation $\boldsymbol{P}_{\boldsymbol{O}_h}^{(\chi)}$ to give *a combined-permutation representation*:

$$\boldsymbol{P}_{\boldsymbol{O}_h}^{(\mathbf{x}\chi)} = \boldsymbol{P}_{\boldsymbol{O}_h}^{(\mathbf{x})} \oplus \boldsymbol{P}_{\boldsymbol{O}_h}^{(\chi)} = \{\boldsymbol{P}_{G}^{(\mathbf{x})} \oplus \boldsymbol{P}_{G}^{(\chi)} \mid \forall G \in \boldsymbol{O}_h\}, \tag{14}$$

where the symbol $\boldsymbol{P}_{G}^{(\mathbf{x})} \oplus \boldsymbol{P}_{G}^{(\chi)}$ is a combination of the two permutations at issue, e.g., $(1\ 6)(2)(3)(4)(5) \oplus (7\ 8) = (1\ 6)(2)(3)(4)(5)(7\ 8)$. The representation $\boldsymbol{P}_{\boldsymbol{O}_h}^{(\mathbf{x}\chi)}$ (Eq. 14) can be used in place of the permutation representation $\boldsymbol{P}_{\boldsymbol{O}_h}^{(\mathbf{x}\overline{\mathbf{x}})}$ [17].

If a given skeleton (e.g., **1**) is faithfully characterized by a permutation representation (e.g., $\boldsymbol{P}_{\boldsymbol{O}_h}^{(\mathbf{x}\overline{\mathbf{x}})}$), the permutation representation is presumed to be a permutation group. Thereby, the corresponding combined representation (e.g., $\boldsymbol{P}_{\boldsymbol{O}_h}^{(\mathbf{x}\chi)}$) is regarded as a permutation group isomorphic to the original point group (e.g., $\boldsymbol{O}_h$).

To construct a permutation group corresponding to $\boldsymbol{P}_{\boldsymbol{O}_h}^{(\mathbf{x}\chi)}$, for example, the following set of generators is taken into consideration:

$$C_{4(3)}^{3} \sim (1)(2\ 3\ 4\ 5)(6)(7)(8) = (2\ 3\ 4\ 5) \tag{15}$$

$$C_{3(1)}^{2} \sim (1\ 2\ 3)(4\ 5\ 6)(7)(8) = (1\ 2\ 3)(4\ 5\ 6) \tag{16}$$

$$\sigma_{h(1)} \sim (1\ 6)(2)(3)(4)(5)(7\ 8) = (1\ 6)(7\ 8), \tag{17}$$

where a four-fold rotation ($C_{4(3)}^{3}$), a three-fold rotation ($C_{3(1)}^{2}$), and a reflection ($\sigma_{h(1)}$) are appropriately selected as generators. These generators are selected from Figure 2 by referring to the numbering of **1** (Figure 1). Note that every 1-cycles are omitted according to the convention of the GAP system. These generators are stored as a list `gen1`, which is placed as an argument of the GAP function `Group` as follows:

```
gap> gen1 := [(2,3,4,5), (1,2,3)(4,5,6), (1,6)(7,8)];; #generators
gap> Oh_octa := Group(gen1); #octahedral skeleton
Group([ (2,3,4,5), (1,2,3)(4,5,6), (1,6)(7,8) ])
gap> Size(Oh_octa);
48
gap> Elements(Oh_octa);
[ (), (3,5)(7,8), (2,3)(4,5)(7,8), (2,3,4,5), (2,4)(7,8), (2,4)(3,5), (2,5,4,3), (2,5)(3,4)(7,8),
  (1,2)(4,6)(7,8),
(omitted)
  (1,6)(2,5,4,3)(7,8), (1,6)(2,5)(3,4) ]
```

Thereby, the permutation group named `Oh_octa` (order 48) is generated in accord with the combined-permutation representation $\boldsymbol{P}_{\boldsymbol{O}_h}^{(\mathbf{x}\chi)}$, where its order is calculated by the GAP function `Size` and its elements are obtained by means of the GAP function `Elements`.

2.4 Partition into Conjugacy Classes

To calculate CI-CFs, we should determine cycle structures of permutations. Because such a cycle structure is common to permutations of the same conjugacy class, it is necessary to develop a function for calculating the partition of permutations into conjugacy classes. To do this task, the GAP function `ConjugacyClasses` and related functions are used as follows, where the permutation group `Oh_octa` has been obtained from the above-mentioned set of generators (`gen1`).

```
gap> conj_class := ConjugacyClasses(Oh_octa);;
gap> Size(conj_class); #number of conjugacy classes
10
gap> Elements(conj_class[2]);
[ (3,5)(7,8), (2,4)(7,8), (1,6)(7,8) ]
gap> r_conj_class := Representative(conj_class[2]);
(3,5)(7,8)
gap> CycleLengths(r_conj_class, [1..8]); #full degree
[ 1, 1, 2, 1, 1, 2 ]
gap> CycleLengths(r_conj_class, [1..6]); #net degree
[ 1, 1, 2, 1, 1 ]
```

As a result, the list of conjugacy classes (`conj_class`) obtained by the GAP function `ConjugacyClasses` contains 10 inner lists of permutations. Among the inner lists of permutations, the 2nd inner list of permutations, for example, is obtained by inputting `Elements(conj_class[2])` so as to show three permutations for specifying horizontal reflection operations ($\sigma_{h(2)}$, $\sigma_{h(3)}$, and $\sigma_{h(1)}$), which is referred to as a conjugacy class $\mathrm{Cl}(\sigma_{h(2)})$. The representative (`r_conj_class`) of the 2nd inner list of permutations is determined to be $(3,5)(7,8)$ ($\sim \sigma_{h(2)}$), which has a cycle structure $1^4 2^2$ as found in the data `[ 1, 1, 2, 1, 1, 2 ]`. The net cycle structure for calculating a CI-CF is $1^4 2$, as found in the data `[ 1, 1, 2, 1, 1 ]`. Note that anyone of the permutations (for $\sigma_{h(2)}$, $\sigma_{h(3)}$, and $\sigma_{h(1)}$) can be selected as a representative, so that $\mathrm{Cl}(\sigma_{h(1)})$ or $\mathrm{Cl}(\sigma_{h(3)})$ denotes the same conjugacy class as $\mathrm{Cl}(\sigma_{h(2)})$.

To calculate CI-CFs, the list of conjugacy classes (`conj_class`) should be intermediately divided into the list of rotations and the list of (roto)reflections. By comparing `[ 1, 1, 2, 1, 1, 2 ]` with `[ 1, 1, 2, 1, 1 ]`, the last digit 2 of the former indicates the presence of the permutation (7 8), which shows that the permutations of the 2nd inner list correspond to (roto)reflections. On the basis of this information, a function `divideConjClasses` for obtaining such an intermediately divided list of conjugacy classes is developed:

`divideConjClasses(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 list of this function is stored in a file named `CICFgenCC.gapfunc`, which is attached as Appendix A. The return value of the function `divideConjClasses` is a doubly nested list having the following format:

```
[ [ [ () ], ..., [...] ],
  [ [...], ..., [...] ] ]
```

where the first row indicates the intermediate list containing conjugacy classes of rotations and the second row indicates the intermediate list containing conjugacy classes of (roto)reflections. Each of the most inner pairs of square brackets indicates a conjugacy class Cl(G) for an appropriate representative G ($\in$ Cl(G) $\subset$ $\boldsymbol{G}$).

For example, the intermediately divided list of conjugacy classes for characterizing the group `Oh_octa` is obtained as follows after reading the file `CICFgenCC.gapfunc` stored in an appropriate work directory (e.g., `c:/fujita0/calcCICF/`). Note that this output has been realigned in order to ensure visibility.

```
gap> Read("c:/fujita0/calcCICF/CICFgenCC.gapfunc");
gap> l_conjclass := divideConjClasses(Oh_octa, 6, 8);
[ [ [ () ], [ (2,3,4,5), (2,5,4,3), (1,2,6,4), (1,3,6,5), (1,4,6,2), (1,5,6,3) ],
    [ (2,4)(3,5), (1,6)(3,5), (1,6)(2,4) ], [ (1,2)(3,5)(4,6), (1,3)(2,4)(5,6), (1,4)(2,6)(3,5),
      (1,5)(2,4)(3,6), (1,6)(2,3)(4,5), (1,6)(2,5)(3,4) ], [ (1,2,3)(4,5,6), (1,2,5)(3,6,4),
      (1,3,2)(4,6,5), (1,3,4)(2,6,5), (1,4,5)(2,3,6), (1,4,3)(2,5,6),
      (1,5,2)(3,4,6), (1,5,4)(2,6,3) ] ],
  [ [ (3,5)(7,8), (2,4)(7,8), (1,6)(7,8) ], [ (2,3)(4,5)(7,8), (2,5)(3,4)(7,8), (1,2)(4,6)(7,8),
      (1,3)(5,6)(7,8), (1,4)(2,6)(7,8), (1,5)(3,6)(7,8) ],
    [ (1,2,3,6,4,5)(7,8), (1,2,5,6,4,3)(7,8), (1,3,4,6,5,2)(7,8), (1,3,2,6,5,4)(7,8), (1,4,5,6,2,3)(7,8),
      (1,4,3,6,2,5)(7,8), (1,5,4,6,3,2)(7,8), (1,5,2,6,3,4)(7,8) ],
    [ (1,2,6,4)(3,5)(7,8), (1,3,6,5)(2,4)(7,8), (1,4,6,2)(3,5)(7,8), (1,5,6,3)(2,4)(7,8),
      (1,6)(2,3,4,5)(7,8), (1,6)(2,5,4,3)(7,8) ], [ (1,6)(2,4)(3,5)(7,8) ] ] ]
```

The resulting list `l_conjclass` contains the first nested list of five conjugacy classes of rotations and the second nested list of five conjugacy classes of (roto)reflections. The latter list is characterized by the presence of the mirror-permutation representation (7 8). This result is consistent with the data collected in Figure 2.

3 Correspondence Between $\boldsymbol{O}_h$-Skeletons

3.1 Permutation Groups for $\boldsymbol{O}_h$-Skeletons

The above-mentioned procedure for an octahedral skeleton **1** can be applied to the other $\boldsymbol{O}_h$-skeletons **2**–**5** listed in Figure 1. The sets of of generators for these skeletons are listed in Table 1, where the generators of each set correspond to $C_{4(3)}^{3}$ (cf. Eq. 15 for **1**), $C_{3(1)}^{2}$ (cf. Eq. 16 for **1**), and $\sigma_{h(1)}$ (cf. Eq. 17 for **1**). In particular, the last generator of each set corresponds to the reflection $\sigma_{h(1)}$, where a mirror-permutation representation is represented by (9 10) for a cube **2**, (13 14) for a cuboctahedron **3**, as well as (25 26) for a truncated octahedron **4** and for a truncated hexahedron (a truncated cube) **5**.

Table 1. Group Generators for Specifying $\boldsymbol{O}_h$-Skeletons

$\boldsymbol{O}_h$-skeleton	Generators: `gen1` — `gen5`, Groups: `Group(gen1)` — `Group(gen5)`

octahedron **1**	`gen1 := [(2,3,4,5), (1,2,3)(4,5,6), (1,6)(7,8)];`
	`Oh_octa := Group(gen1); #octahedron`

cube **2**	`gen2 := [(1,2,3,4)(5,6,7,8), (2,4,5)(3,8,6), (1,5)(2,6)(3,7)(4,8)(9,10)];`
	`Oh_cube := Group(gen2); #cube`

cuboctahedron **3**	`gen3 := [(1,2,3,4)(5,6,7,8)(9,10,11,12), (1,5,2)(3,8,10)(4,9,6)(7,12,11),`
	`(1,9)(2,10)(3,11)(4,12)(13,14)];`
	`Oh_cuboct := Group(gen3); #cuboctahedron`

truncated octahedron **4**	`gen4 := [(1,2,3,4)(5,6,7,8)(9,11,13,15)(10,12,14,16)(17,18,19,20)(21,22,23,24),`
	`(1,8,9)(4,16,5)(2,15,17)(3,20,10)(6,14,21)(7,24,11)(12,19,22)(13,23,18),`
	`(1,21)(2,22)(3,23)(4,24)(5,17)(6,18)(7,19)(8,20)(25,26)];`
	`Oh_troct := Group(gen4); #truncated octahedron`

truncated hexahedron **5**	`gen5 := [(1,3,5,7)(2,4,6,8)(9,10,11,12)(13,14,15,16)(17,19,21,23)(18,20,22,24),`
	`(1,9,2)(8,13,3)(4,12,18)(5,16,19)(6,24,14)(7,17,10)(11,23,20)(15,22,21),`
	`(1,17)(2,18)(3,19)(4,20)(5,21)(6,22)(7,23)(8,24)(9,13)(10,14)(11,15)(12,16)(25,26)];`
	`Oh_trhex := Group(gen5); #truncated hexahedron`

3.2 Isomorphism between Permutation Groups for $\boldsymbol{O}_h$-Skeletons

The resulting permutation groups, i.e., `Oh_octa` for **1**, `Oh_cube` for **2**, `Oh_cuboct` for **3**, `Oh_troct` for **4**, and `Oh_trhex` for **5**, are isomorphic to the point group $\boldsymbol{O}_h$, which is originally generated by a multiplication table of 48 operations. These permutation groups are independently constructed, so that the correspondence between them should be determined. For the sake of convenience, the permutation group `Oh_octa` for **1** is selected as a standard group in place of the original point group $\boldsymbol{O}_h$.

The following source list written in a file named `Oh-Polyons4.gap` evaluates the correspondence between `Oh_octa` (**1**) and `Oh_cube` (**2**) as well as between `Oh_octa` (**1**) and `Oh_cuboct` (**3**). The file `Oh-Polyons4.gap` is stored in the work directory named `c:/fujita0/calcCICF/`, so that the first line commented out by the symbol `#` is copied and pasted after the command prompt `gap>` to start the execution of the evaluation process. The result is written down in a log file named `Oh-Polyons4log.txt`, which is stored in the work directory `c:/fujita0/calcCICF/`.

Sample Program (`Oh-Polyons4.gap`) for Group Homomorphism:

```
#Read("c:/fujita0/calcCICF/Oh-Polyons4.gap");
LogTo("c:/fujita0/calcCICF/Oh-Polyons4log.txt");

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

gen1 := [(2,3,4,5), (1,2,3)(4,5,6), (1,6)(7,8)];
Oh_octa := Group(gen1); #octahedron
gen2 := [(1,2,3,4)(5,6,7,8), (2,4,5)(3,8,6), (1,5)(2,6)(3,7)(4,8)(9,10)];
Oh_cube := Group(gen2); #cube
gen3 := [(1,2,3,4)(5,6,7,8)(9,10,11,12), (1,5,2)(3,8,10)(4,9,6)(7,12,11),
(1,9)(2,10)(3,11)(4,12)(13,14)];
Oh_cuboct := Group(gen3); #cuboctahedron

l_conjclass := divideConjClasses(Oh_octa, 6, 8);
hom1 := GroupHomomorphismByImages(Oh_octa, Oh_cube, gen1, gen2);
hom2 := GroupHomomorphismByImages(Oh_octa, Oh_cuboct, gen1, gen3);

l1_conjclass := l_conjclass[1];
l2_conjclass := l_conjclass[2];

for j in [1..Size(l1_conjclass)] do
ll1 := l1_conjclass[j];
for i in [1..Size(ll1)] do
Print(ll1[i], "&", Image(hom1, ll1[i]), "&", Image(hom2, ll1[i]), "\\\\␣\n");
od;
Print("\\hline␣\n");
od;
Print("\\hline␣\n");
for j in [1..Size(l2_conjclass)] do
ll2 := l2_conjclass[j];
for i in [1..Size(ll2)] do
Print(ll2[i], "&", Image(hom1, ll2[i]), "&", Image(hom2, ll2[i]), "\\\\␣\n");
od;
Print("\\hline␣\n");
od;

LogTo();
```

During the execution, the file `CICFgenCC.gapfunc` (Appendix A) is first loaded to use the function `divideConjClasses` defined above. Then, the homomorphism (in this case, isomorphism) between `Oh_octa` (**1**) and `Oh_cube` (**2**) as well as between `Oh_octa` (**1**) and `Oh_cuboct` (**3**) is obtained by using the GAP function `GroupHomomorphismByImages`. The resulting homomorphism is stored in a list named `homo1` (or `homo2`), which is written down into the log file by using the GAP function `Image`. The result written in the log file `Oh-Polyons4log.txt` is converted into a tabular form, as shown in Table 2.

In Table 2, the sets of conjugacy classes are divided by horizontal lines. This mode of division is based on the list `l_conjclass` (for `Oh_octa`), which has been determined by the function `divideConjClasses`, as described above. For example, the rows concerning $(2,3,4,5)$, $(1,2,3)(4,5,6)$, and $(1,6)(7,8)$ (appearing at the first column for **1** in Table 2) demonstrate the correspondence concerning the sets of generators (`gen1`, `gen2`, and `gen3`), where they are accompanied by respective conjugacy classes. The upper part over a horizontal double line lists the permutations of rotations (stored in the list `l_conjclass[1]` for `Oh_octa`) while the lower part below the horizontal double line lists the permutations

Table 2. Correspondence between an octahedron **1**, a cube **2**, and a cuboctahedron **3** as $\boldsymbol{O}_h$-skeletons

octahedron **1**	cube **2**	cuboctahedron **3**
`Oh_octa := Group(gen1)`	`Oh_cube := Group(gen2)`	`Oh_cuboct := Group(gen3)`
$\boldsymbol{P}^{(\mathbf{X}\chi)}_{\boldsymbol{O}_h\text{-octa}}$	$\boldsymbol{P}^{(\mathbf{X}\chi)}_{\boldsymbol{O}_h\text{-cube}}$	$\boldsymbol{P}^{(\mathbf{X}\chi)}_{\boldsymbol{O}_h\text{-cuboct}}$
()	()	()
(2,3,4,5)	(1, 2, 3, 4)(5, 6, 7, 8)	(1, 2, 3, 4)(5, 6, 7, 8)(9,10,11,12)
(2,5,4,3)	(1, 4, 3, 2)(5, 8, 7, 6)	(1, 4, 3, 2)(5, 8, 7, 6)(9,12,11,10)
(1,2,6,4)	(1, 5, 6, 2)(3, 4, 8, 7)	(1, 9,11, 3)(2, 5,10, 6)(4, 8,12, 7)
(1,3,6,5)	(1, 5, 8, 4)(2, 6, 7, 3)	(1, 5, 9, 8)(2,10,12, 4)(3, 6,11, 7)
(1,4,6,2)	(1, 2, 6, 5)(3, 7, 8, 4)	(1, 3,11, 9)(2, 6,10, 5)(4, 7,12, 8)
(1,5,6,3)	(1, 4, 8, 5)(2, 3, 7, 6)	(1, 8, 9, 5)(2, 4,12,10)(3, 7,11, 6)
(2,4)(3,5)	(1, 3)(2, 4)(5, 7)(6, 8)	(1, 3)(2, 4)(5, 7)(6, 8)(9,11)(10,12)
(1,6)(3,5)	(1, 8)(2, 7)(3, 6)(4, 5)	(1, 9)(2,12)(3,11)(4,10)(5, 8)(6, 7)
(1,6)(2,4)	(1, 6)(2, 5)(3, 8)(4, 7)	(1,11)(2,10)(3, 9)(4,12)(5, 6)(7, 8)
(1,2)(3,5)(4,6)	(1, 4)(2, 8)(3, 5)(6, 7)	(2, 8)(3, 9)(4, 5)(6,12)(7,10)
(1,3)(2,4)(5,6)	(1, 2)(3, 5)(4, 6)(7, 8)	(1, 6)(3, 5)(4,10)(7, 9)(8,11)
(1,4)(2,6)(3,5)	(1, 7)(2, 3)(4, 6)(5, 8)	(1,11)(2, 7)(4, 6)(5,12)(8,10)
(1,5)(2,4)(3,6)	(1, 7)(2, 8)(3, 4)(5, 6)	(1, 7)(2,12)(3, 8)(5,11)(6, 9)
(1,6)(2,3)(4,5)	(1, 5)(2, 8)(3, 7)(4, 6)	(1,10)(2, 9)(3,12)(4,11)(6, 8)
(1,6)(2,5)(3,4)	(1, 7)(2, 6)(3, 5)(4, 8)	(1,12)(2,11)(3,10)(4, 9)(5, 7)
(1,2,3)(4,5,6)	(2, 4, 5)(3, 8, 6)	(1, 5, 2)(3, 8,10)(4, 9, 6)(7,12,11)
(1,2,5)(3,6,4)	(1, 8, 3)(2, 5, 7)	(1, 8, 4)(2, 9, 7)(3, 5,12)(6,10,11)
(1,3,2)(4,6,5)	(2, 5, 4)(3, 6, 8)	(1, 2, 5)(3,10, 8)(4, 6, 9)(7,11,12)
(1,3,4)(2,6,5)	(1, 6, 3)(4, 5, 7)	(1,10, 7)(2, 6, 3)(4, 5,11)(8, 9,12)
(1,4,5)(2,3,6)	(1, 6, 8)(2, 7, 4)	(1, 6,12)(2,11, 8)(3, 7, 4)(5,10, 9)
(1,4,3)(2,5,6)	(1, 3, 6)(4, 7, 5)	(1, 7,10)(2, 3, 6)(4,11, 5)(8,12, 9)
(1,5,2)(3,4,6)	(1, 3, 8)(2, 7, 5)	(1, 4, 8)(2, 7, 9)(3,12, 5)(6,11,10)
(1,5,4)(2,6,3)	(1, 8, 6)(2, 4, 7)	(1,12, 6)(2, 8,11)(3, 4, 7)(5, 9,10)
(3,5)(7,8)	(1, 4)(2, 3)(5, 8)(6, 7)(9,10)	(2, 4)(5, 8)(6, 7)(10,12)(13,14)
(2,4)(7,8)	(1, 2)(3, 4)(5, 6)(7, 8)(9,10)	(1, 3)(5, 6)(7, 8)(9,11)(13,14)
(1,6)(7,8)	(1, 5)(2, 6)(3, 7)(4, 8)(9,10)	(1, 9)(2,10)(3,11)(4,12)(13,14)
(2,3)(4,5)(7,8)	(2, 4)(6, 8)(9,10)	(1, 2)(3, 4)(6, 8)(9,10)(11,12)(13,14)
(2,5)(3,4)(7,8)	(1, 3)(5, 7)(9,10)	(1, 4)(2, 3)(5, 7)(9,12)(10,11)(13,14)
(1,2)(4,6)(7,8)	(2, 5)(3, 8)(9,10)	(2, 5)(3, 9)(4, 8)(6,10)(7,12)(13,14)
(1,3)(5,6)(7,8)	(3, 6)(4, 5)(9,10)	(1, 5)(3, 6)(4,10)(7,11)(8, 9)(13,14)
(1,4)(2,6)(7,8)	(1, 6)(4, 7)(9,10)	(1,11)(2, 6)(4, 7)(5,10)(8,12)(13,14)
(1,5)(3,6)(7,8)	(1, 8)(2, 7)(9,10)	(1, 8)(2,12)(3, 7)(5, 9)(6,11)(13,14)
(1,2,3,6,4,5)(7,8)	(1, 5, 6, 7, 3, 4)(2, 8)(9,10)	(1, 5,10,11, 7, 4)(2, 9, 6,12, 3, 8)(13,14)
(1,2,5,6,4,3)(7,8)	(1, 4, 8, 7, 6, 2)(3, 5)(9,10)	(1, 8,12,11, 6, 2)(3, 5, 4, 9, 7,10)(13,14)
(1,3,4,6,5,2)(7,8)	(1, 2, 6, 7, 8, 4)(3, 5)(9,10)	(1, 2, 6,11,12, 8)(3,10, 7, 9, 4, 5)(13,14)
(1,3,2,6,5,4)(7,8)	(1, 5, 8, 7, 3, 2)(4, 6)(9,10)	(1,10, 8,11, 4, 6)(2, 5, 9,12, 7, 3)(13,14)
(1,4,5,6,2,3)(7,8)	(1, 2, 3, 7, 8, 5)(4, 6)(9,10)	(1, 6, 4,11, 8,10)(2, 3, 7,12, 9, 5)(13,14)
(1,4,3,6,2,5)(7,8)	(1, 7)(2, 6, 5, 8, 4, 3)(9,10)	(1, 7, 2,11, 5,12)(3, 6,10, 9, 8, 4)(13,14)
(1,5,4,6,3,2)(7,8)	(1, 4, 3, 7, 6, 5)(2, 8)(9,10)	(1, 4, 7,11,10, 5)(2, 8, 3,12, 6, 9)(13,14)
(1,5,2,6,3,4)(7,8)	(1, 7)(2, 3, 4, 8, 5, 6)(9,10)	(1,12, 5,11, 2, 7)(3, 4, 8, 9,10, 6)(13,14)
(1,2,6,4)(3,5)(7,8)	(1, 8, 6, 3)(2, 4, 5, 7)(9,10)	(1, 9,11, 3)(2, 8,10, 7)(4, 5,12, 6)(13,14)
(1,3,6,5)(2,4)(7,8)	(1, 6, 8, 3)(2, 5, 7, 4)(9,10)	(1, 6, 9, 7)(2,10,12, 4)(3, 5,11, 8)(13,14)
(1,4,6,2)(3,5)(7,8)	(1, 3, 6, 8)(2, 7, 5, 4)(9,10)	(1, 3,11, 9)(2, 7,10, 8)(4, 6,12, 5)(13,14)
(1,5,6,3)(2,4)(7,8)	(1, 3, 8, 6)(2, 4, 7, 5)(9,10)	(1, 7, 9, 6)(2, 4,12,10)(3, 8,11, 5)(13,14)
(1,6)(2,3,4,5)(7,8)	(1, 6, 3, 8)(2, 7, 4, 5)(9,10)	(1,10, 3,12)(2,11, 4, 9)(5, 6, 7, 8)(13,14)
(1,6)(2,5,4,3)(7,8)	(1, 8, 3, 6)(2, 5, 4, 7)(9,10)	(1,12, 3,10)(2, 9, 4,11)(5, 8, 7, 6)(13,14)
(1,6)(2,4)(3,5)(7,8)	(1, 7)(2, 8)(3, 5)(4, 6)(9,10)	(1,11)(2,12)(3, 9)(4,10)(5, 7)(6, 8)(13,14)

Table 3. Correspondence between an octahedron **1**, a truncated octahedron **4**, and a truncated hexahedron **5** as $\boldsymbol{O}_h$-skeletons

octahedron **1** `Oh_octa := Group(gen1)` $\boldsymbol{P}^{(\mathbf{X}\chi)}_{\boldsymbol{O}_h\text{-octa}}$	truncated octahedron **4** `Oh_troct := Group(gen4)` $\boldsymbol{P}^{(\mathbf{X}\chi)}_{\boldsymbol{O}_h\text{-troct}}$	truncated hexahedron **5** `Oh_trhex := Group(gen5)` $\boldsymbol{P}^{(\mathbf{X}\chi)}_{\boldsymbol{O}_h\text{-trhex}}$
()	()	()
(2,3,4,5)	(1, 2, 3, 4)(5, 6, 7, 8)(9,11,13,15)(10,12,14,16)(17,18,19,20)(21,22,23,24)	(1, 3, 5, 7)(2, 4, 6, 8) (9,10,11,12)(13,14,15,16)(17,19,21,23)(18,20,22,24)
(2,5,4,3)	(1, 4, 3, 2)(5, 8, 7, 6)(9,15,13,11)(10,16,14,12)(17,20,19,18)(21,24,23,22)	(1, 7, 5, 3)(2, 8, 6, 4) (9,12,11,10)(13,16,15,14)(17,23,21,19)(18,24,22,20)
(1,2,6,4)	(1,16,21,11)(2, 8,24,18)(3,15,23,12)(4,20,22, 6)(5, 9,17,10)(7,14,19,13)	(1,17,20, 4)(2,13,19,10) (3, 9,18,14)(5, 8,24,21)(6,12,23,15)(7,16,22,11)
(1,3,6,5)	(1,17,23, 7)(2,10,22,13)(3, 5,21,19)(4, 9,24,14)(6,11,18,12)(8,16,20,15)	(1,13,24,12)(2,18,23, 7) (3,19,22, 6)(4,14,21,11)(5,10,20,15)(8, 9,17,16)
(1,4,6,2)	(1,11,21,16)(2,18,24, 8)(3,12,23,15)(4, 6,22,20)(5,10,17, 9)(7,13,19,14)	(1, 4,20,17)(2,10,19,13) (3,14,18, 9)(5,21,24, 8)(6,15,23,12)(7,11,22,16)
(1,5,6,3)	(1, 7,23,17)(2,13,22,10)(3,19,21, 5)(4,14,24, 9)(6,12,18,11)(8,15,20,16)	(1,12,24,13)(2, 7,23,18) (3, 6,22,19)(4,11,21,14)(5,15,20,10)(8,16,17, 9)
(2,4)(3,5)	(1, 3)(2, 4)(5, 7)(6, 8)(9,13)(10,14)(11,15)(12,16)(17,19)(18,20)(21,23)(22,24)	(1, 5)(2, 6)(3, 7) (4, 8)(9,11)(10,12)(13,15)(14,16)(17,21)(18,22)(19,23)(20,24)
(1,6)(3,5)	(1,23)(2,22)(3,21)(4,24)(5,19)(6,18)(7,17)(8,20)(9,14)(10,13)(11,12)(15,16)	(1,24)(2,23)(3,22) (4,21)(5,20)(6,19)(7,18)(8,17)(9,16)(10,15)(11,14)(12,13)
(1,6)(2,4)	(1,21)(2,24)(3,23)(4,22)(5,17)(6,20)(7,19)(8,18)(9,10)(11,16)(12,15)(13,14)	(1,20)(2,19)(3,18) (4,17)(5,24)(6,23)(7,22)(8,21)(9,14)(10,13)(11,16)(12,15)
(1,2)(3,5)(4,6)	(1,15)(2,20)(3,16)(4, 8)(5,14)(6,24)(7, 9)(10,19)(11,23)(12,21)(13,17)(18,22)	(1, 8)(2,12) (3,16)(4,24)(5,17)(6,13)(7, 9)(10,23)(11,18)(14,22)(15,19)(20,21)
(1,3)(2,4)(5,6)	(1, 5)(2, 9)(3,17)(4,10)(6,16)(7,21)(8,11)(12,20)(13,24)(14,22)(15,18)(19,23)	(1,10)(2, 3) (4, 9)(5,13)(6,18)(7,19)(8,14)(11,17)(12,20)(15,24)(16,21)(22,23)
(1,4)(2,6)(3,5)	(1,12)(2, 6)(3,11)(4,18)(5,13)(7,10)(8,22)(9,19)(14,17)(15,21)(16,23)(20,24)	(1,21)(2,15) (3,11)(4, 5)(6,10)(7,14)(8,20)(9,22)(12,19)(13,23)(16,18)(17,24)
(1,5)(2,4)(3,6)	(1,19)(2,14)(3, 7)(4,13)(5,23)(6,15)(8,12)(9,22)(10,24)(11,20)(16,18)(17,21)	(1,15)(2,22) (3,23)(4,16)(5,12)(6, 7)(8,11)(9,21)(10,24)(13,20)(14,17)(18,19)
(1,6)(2,3)(4,5)	(1,24)(2,23)(3,22)(4,21)(5,20)(6,19)(7,18)(8,17)(9,16)(10,15)(11,14)(12,13)	(1,18)(2,17) (3,24)(4,23)(5,22)(6,21)(7,20)(8,19)(9,13)(10,16)(11,15)(12,14)
(1,6)(2,5)(3,4)	(1,22)(2,21)(3,24)(4,23)(5,18)(6,17)(7,20)(8,19)(9,12)(10,11)(13,16)(14,15)	(1,22)(2,21) (3,20)(4,19)(5,18)(6,17)(7,24)(8,23)(9,15)(10,14)(11,13)(12,16)
(1,2,3)(4,5,6)	(1, 8, 9)(2,15,17)(3,20,10)(4,16, 5)(6,14,21)(7,24,11)(12,19,22)(13,23,18)	(1, 9, 2)(3, 8,13) (4,12,18)(5,16,19)(6,24,14)(7,17,10)(11,23,20)(15,22,21)
(1,2,5)(3,6,4)	(1,20,13)(2,16,19)(3, 8,14)(4,15, 7)(5,24,12)(6, 9,23)(10,21,18)(11,17,22)	(1,16, 6)(2,24,11) (3,17,15)(4,13,22)(5, 9,23)(7, 8,12)(10,18,21)(14,19,20)
(1,3,2)(4,6,5)	(1, 9, 8)(2,17,15)(3,10,20)(4, 5,16)(6,21,14)(7,11,24)(12,22,19)(13,18,23)	(1, 2, 9)(3,13, 8) (4,18,12)(5,19,16)(6,14,24)(7,10,17)(11,20,23)(15,21,22)
(1,3,4)(2,6,5)	(1,10, 6)(2, 5,11)(3, 9,18)(4,17,12)(7,16,22)(8,21,13)(14,20,23)(15,24,19)	(1,19,11)(2,14, 5) (3,10, 4)(6, 9,20)(7,13,21)(8,18,15)(12,17,22)(16,24,23)
(1,4,5)(2,3,6)	(1,18,14)(2,12, 7)(3, 6,13)(4,11,19)(5,22,15)(8,10,23)(9,21,20)(16,17,24)	(1,14,23)(2,20,16) (3,21,12)(4,15, 7)(5,11, 6)(8,10,22)(9,19,24)(13,18,17)
(1,4,3)(2,5,6)	(1, 6,10)(2,11, 5)(3,18, 9)(4,12,17)(7,22,16)(8,13,21)(14,23,20)(15,19,24)	(1,11,19)(2, 5,14) (3, 4,10)(6,20, 9)(7,21,13)(8,15,18)(12,22,17)(16,23,24)
(1,5,2)(3,4,6)	(1,13,20)(2,19,16)(3,14, 8)(4, 7,15)(5,12,24)(6,23, 9)(10,18,21)(11,22,17)	(1, 6,16)(2,11,24) (3,15,17)(4,22,13)(5,23, 9)(7,12, 8)(10,21,18)(14,20,19)
(1,5,4)(2,6,3)	(1,14,18)(2, 7,12)(3,13, 6)(4,19,11)(5,15,22)(8,23,10)(9,20,21)(16,24,17)	(1,23,14)(2,16,20) (3,12,21)(4, 7,15)(5, 6,11)(8,22,10)(9,24,19)(13,17,18)
(3,5)(7,8)	(1, 3)(5, 7)(9,14)(10,13)(11,12)(15,16)(17,19)(21,23)(25,26)	(1, 8)(2, 7)(3, 6)(4, 5)(9,12)(10,11) (13,16)(14,15)(17,24)(18,23)(19,22)(20,21)(25,26)
(2,4)(7,8)	(2, 4)(6, 8)(9,10)(11,16)(12,15)(13,14)(18,20)(22,24)(25,26)	(1, 4)(2, 3)(5, 8)(6, 7)(9,10)(11,12) (13,14)(15,16)(17,20)(18,19)(21,24)(22,23)(25,26)
(1,6)(7,8)	(1,21)(2,22)(3,23)(4,24)(5,17)(6,18)(7,19)(8,20)(25,26)	(1,17)(2,18)(3,19)(4,20)(5,21)(6,22) (7,23)(8,24)(9,13)(10,14)(11,15)(12,16)(25,26)
(2,3)(4,5)(7,8)	(1, 4)(2, 3)(5, 8)(6, 7)(9,16)(10,15)(11,14)(12,13)(17,20)(18,19)(21,24)(22,23)(25,26)	(1, 2) (3, 8)(4, 7)(5, 6)(10,12)(14,16)(17,18)(19,24)(20,23)(21,22)(25,26)
(2,5)(3,4)(7,8)	(1, 2)(3, 4)(5, 6)(7, 8)(9,12)(10,11)(13,16)(14,15)(17,18)(19,20)(21,22)(23,24)(25,26)	(1, 6) (2, 5)(3, 4)(7, 8)(9,11)(13,15)(17,22)(18,21)(19,20)(23,24)(25,26)
(1,2)(4,6)(7,8)	(1,16)(2,20)(3,15)(4, 8)(5, 9)(6,24)(7,14)(10,17)(11,21)(12,23)(13,19)(18,22)(25,26)	(2, 9) (3,13)(4,17)(5,24)(6,16)(7,12)(10,18)(11,23)(14,19)(15,22)(25,26)
(1,3)(5,6)(7,8)	(1, 5)(2,10)(3,17)(4, 9)(6,11)(7,21)(8,16)(12,18)(13,22)(14,24)(15,20)(19,23)(25,26)	(1, 9) (4,10)(5,14)(6,19)(7,18)(8,13)(11,20)(12,17)(15,21)(16,24)(25,26)
(1,4)(2,6)(7,8)	(1,11)(2, 6)(3,12)(4,18)(5,10)(7,13)(8,22)(9,17)(14,19)(15,23)(16,21)(20,24)(25,26)	(1,20) (2,14)(3,10)(6,11)(7,15)(8,21)(9,19)(12,22)(13,18)(16,23)(25,26)
(1,5)(3,6)(7,8)	(1,19)(2,13)(3, 7)(4,14)(5,23)(6,12)(8,15)(9,24)(10,22)(11,18)(16,20)(17,21)(25,26)	(1,16) (2,23)(3,22)(4,15)(5,11)(8,12)(9,24)(10,21)(13,17)(14,20)(25,26)
(1,2,3,6,4,5)(7,8)	(1,20,10,23, 6,14)(2,15, 5,24,11,19)(3, 8, 9,21,18,13)(4,16,17,22,12, 7)(25,26)	(1,13,19,21,11, 7)(2,17,14,22, 5,12)(3,24,10,23, 4,16)(6, 8, 9,18,20,15)(25,26)
(1,2,5,6,4,3)(7,8)	(1, 8,14,23,18,10)(2,16, 7,24,12,17)(3,20,13,21, 6, 9)(4,15,19,22,11, 5)(25,26)	(1,12,23,21,14, 3)(2, 8,16,22,20,10)(4, 9, 7,24,15,19)(5,13, 6,17,11,18)(25,26)
(1,3,4,6,5,2)(7,8)	(1,10,18,23,14, 8)(2,17,12,24, 7,16)(3, 9, 6,21,13,20)(4, 5,11,22,19,15)(25,26)	(1, 3,14,21,23,12)(2,10,20,22,16, 8)(4,19,15,24, 7, 9)(5,18,11,17, 6,13)(25,26)
(1,3,2,6,5,4)(7,8)	(1, 9,20,23,13, 6)(2, 5,16,24,19,12)(3,10, 8,21,14,18)(4,17,15,22, 7,11)(25,26)	(1,18,16,21, 6,10)(2,13,24,22,11, 4)(3, 9,17,23,15, 5)(7,14, 8,19,12,20)(25,26)
(1,4,5,6,2,3)(7,8)	(1, 6,13,23,20, 9)(2,12,19,24,16, 5)(3,18,14,21, 8,10)(4,11, 7,22,15,17)(25,26)	(1,10, 6,21,16,18)(2, 4,11,22,24,13)(3, 5,15,23,17, 9)(7,20,12,19, 8,14)(25,26)
(1,4,3,6,2,5)(7,8)	(1,18, 9,23, 8,13)(2,11,17,24,15, 7)(3, 6,10,21,20,14)(4,12, 5,22,16,19)(25,26)	(1,15, 2,21, 9,22)(3,20,13,23, 8,11)(4,14,18,24,12, 6)(5,10,19,17,16, 7)(25,26)
(1,5,4,6,3,2)(7,8)	(1,14, 6,23,10,20)(2,19,11,24, 5,15)(3,13,18,21, 9, 8)(4, 7,12,22,17,16)(25,26)	(1, 7,11,21,19,13)(2,12, 5,22,14,17)(3,16, 4,23,10,24)(6,15,20,18, 9, 8)(25,26)
(1,5,2,6,3,4)(7,8)	(1,13, 8,23, 9,18)(2, 7,15,24,17,11)(3,14,20,21,10, 6)(4,19,16,22, 5,12)(25,26)	(1,22, 9,21, 2,15)(3,11, 8,23,13,20)(4, 6,12,24,18,14)(5, 7,16,17,19,10)(25,26)
(1,2,6,4)(3,5)(7,8)	(1,15,21,12)(2, 8,24,18)(3,16,23,11)(4,20,22, 6)(5,14,17,13)(7, 9,19,10)(25,26)	(1,24,20, 5)(2,16,19,11)(3,12,18,15)(4, 8,17,21)(6, 9,23,14)(7,13,22,10)(25,26)
(1,3,6,5)(2,4)(7,8)	(1,17,23, 7)(2, 9,22,14)(3, 5,21,19)(4,10,24,13)(6,16,18,15)(8,11,20,12)(25,26)	(1,14,24,11)(2,19,23, 6)(3,18,22, 7)(4,13,21,12)(5, 9,20,16)(8,10,17,15)(25,26)
(1,4,6,2)(3,5)(7,8)	(1,12,21,15)(2,18,24, 8)(3,11,23,16)(4, 6,22,20)(5,13,17,14)(7,10,19, 9)(25,26)	(1, 5,20,24)(2,11,19,16)(3,15,18,12)(4,21,17, 8)(6,14,23, 9)(7,10,22,13)(25,26)
(1,5,6,3)(2,4)(7,8)	(1, 7,23,17)(2,14,22, 9)(3,19,21, 5)(4,13,24,10)(6,15,18,16)(8,12,20,11)(25,26)	(1,11,24,14)(2, 6,23,19)(3, 7,22,18)(4,12,21,13)(5,16,20, 9)(8,15,17,10)(25,26)
(1,6)(2,3,4,5)(7,8)	(1,22, 3,24)(2,23, 4,21)(5,18, 7,20)(6,19, 8,17)(9,11,13,15)(10,12,14,16)(25,26)	(1,19, 5,23)(2,20, 6,24)(3,21, 7,17)(4,22, 8,18)(9,14,11,16)(10,15,12,13)(25,26)
(1,6)(2,5,4,3)(7,8)	(1,24, 3,22)(2,21, 4,23)(5,20, 7,18)(6,17, 8,19)(9,15,13,11)(10,16,14,12)(25,26)	(1,23, 5,19)(2,24, 6,20)(3,17, 7,21)(4,18, 8,22)(9,16,11,14)(10,13,12,15)(25,26)
(1,6)(2,4)(3,5)(7,8)	(1,23)(2,24)(3,21)(4,22)(5,19)(6,20)(7,17)(8,18)(9,13)(10,14)(11,15)(12,16)(25,26)	(1,21)(2,22)(3,23)(4,24)(5,17)(6,18)(7,19)(8,20)(9,15)(10,16)(11,13)(12,14)(25,26)

of (roto)reflections (stored in the list `l_conjclass[2]` for `Oh_octa`).

Th correspondence between `Oh_octa` (**1**) and `Oh_troct` (**4**) or between `Oh_octa` (**1**) and `Oh_trhex` (**5**) can be obtained in a similar way, where the set of generators (`gen2` or `gen3`) in the above source list is replaced by the set of generators (`gen4` or `gen5`) listed in Table 1. The resulting correspondence is listed in Table 3.

4 Calculation of CI-CFs

4.1 Products of Sphericity Indices

When an element G ($\in \boldsymbol{G}$ or $\boldsymbol{O}_h$ in particular) belongs to a conjugacy class $\mathrm{Cl}(G)$, let the symbol $\boldsymbol{P}^{(\mathbf{X})}_{\mathrm{Cl}(G)}$ represent a representative permutation (degree n) of the conjugacy class. Suppose that the representative permutation $\boldsymbol{P}^{(\mathbf{X})}_{\mathrm{Cl}(G)}$ is represented by a cycle decomposition involving the number $\nu_k(\boldsymbol{P}^{(\mathbf{X})}_{\mathrm{Cl}(G)})$ of k-cycles ($n = \sum_{k=1}^{n} k\nu_k(\boldsymbol{P}^{(\mathbf{X})}_{\mathrm{Cl}(G)})$), so that the corresponding cycle structure is represented as follows:

$$1^{\nu_1(\boldsymbol{P}^{(\mathbf{X})}_{\mathrm{Cl}(G)})}2^{\nu_2(\boldsymbol{P}^{(\mathbf{X})}_{\mathrm{Cl}(G)})}\cdots n^{\nu_n(\boldsymbol{P}^{(\mathbf{X})}_{\mathrm{Cl}(G)})}. \tag{18}$$

Note that any permutation of the conjugacy class $\mathrm{Cl}(G)$ has the same mode of cycle decomposition, or equivalently the same cycle structure. Then, the element $\boldsymbol{P}^{(\mathbf{X}\chi)}_{\mathrm{Cl}(G)}$ corresponding to $\boldsymbol{P}^{(\mathbf{X})}_{\mathrm{Cl}(G)}$ is specified by a product of sphericity indices (PSI):

$$\mathrm{PSI}_{\boldsymbol{P}^{(\mathbf{X}\chi)}_{\mathrm{Cl}(G)}} = \$_1^{\nu_1(\boldsymbol{P}^{(\mathbf{X})}_{\mathrm{Cl}(G)})}\$_2^{\nu_2(\boldsymbol{P}^{(\mathbf{X})}_{\mathrm{Cl}(G)})}\cdots\$_n^{\nu_n(\boldsymbol{P}^{(\mathbf{X})}_{\mathrm{Cl}(G)})}, \tag{19}$$

where $\$_k$ is a_k if $\boldsymbol{P}^{(\chi)}_{\mathrm{Cl}(G)} = (n+1 \quad n+2)$ (one 2-cycle) and k is odd; $\$_k$ is c_k if $\boldsymbol{P}^{(\chi)}_{\mathrm{Cl}(G)} = (n+1 \quad n+2)$ (one 2-cycle) and k is even; and $\$_k$ is b_k if $\boldsymbol{P}^{(\chi)}_{\mathrm{Cl}(G)} = (n+1)(n+2)$ (two 1-cycles). The PSI (Eq. 19) is common to all of the permutations belonging to the conjugacy class $\mathrm{Cl}(G)$.

For example, the list `l_conjclass`, which has been calculated above to reveal the sets of conjugacy classes for the group `Oh_octa` for **1**, is used to calculate the cycle decomposition of the permutation `(2,3,4,5)`, i.e., (1)(2 3 4 5)(6)(7)(8) for a rotation $C^3_{4(3)}$ (Eq. 15), by means of the GAP system:

```
gap> gen1 := [(2,3,4,5), (1,2,3)(4,5,6), (1,6)(7,8)];; #generators
gap> Oh_octa := Group(gen1); #octahedral skeleton
Group([ (2,3,4,5), (1,2,3)(4,5,6), (1,6)(7,8) ])
gap> Read("c:/fujita0/calcCICF/CICFgenCC.gapfunc");
gap> l_conjclass := divideConjClasses(Oh_octa, 6, 8);;
gap> l_1_2_1 := l_conjclass[1][2][1]; #rotation
(2,3,4,5)
gap> CycleLengths(l_1_2_1, [1..8]); #full degree
[ 1, 4, 1, 1, 1 ]
gap> CycleLengths(l_1_2_1, [1..6]); #net degree
[ 1, 4, 1 ]
```

The output of `#net degree` indicates that the corresponding cycle structure of the permutation $\boldsymbol{P}^{(\mathbf{X})}_{\mathrm{Cl}(C^3_{4(3)})}$ for the conjugacy class $\mathrm{Cl}(C^3_{4(3)})$ is calculated to be $1^2 4^1$. The last two digits `... 1, 1 ]` of the output of `#full degree` indicates a mirror-permutation representation (7)(8), which shows that this permutation is a rotation. Thereby, the corresponding PSI is calculated to be $b_1^2 b_4$ by means of Eq. 19.

On the other hand, the cycle decomposition of the permutation `(1,6)(7,8)`, i.e., (1 6)(2)(3)(4)(5)(7 8) for a reflection $\sigma_{h(1)}$ (Eq. 17), is calculated as follows:

```
gap> l_2_1_3 := l_conjclass[2][1][3]; #reflection
(1,6)(7,8)
gap> CycleLengths(l_2_1_3, [1..8]); #full degree
[ 2, 1, 1, 1, 1, 2 ]
gap> CycleLengths(l_2_1_3, [1..6]); #net degree
[ 2, 1, 1, 1, 1 ]
```

The output of `#net degree` indicates that the corresponding cycle structure of the permutation $\boldsymbol{P}^{(\mathbf{X})}_{\mathrm{Cl}(\sigma_{h(1)})}$ for the conjugacy class $\mathrm{Cl}(\sigma_{h(1)})$ is calculated to be $1^4 2^1$. The last digit `... 2 ]` of the output of `#full degree` indicates a mirror-permutation representation (7 8), which shows that this permutation is a (roto)reflection. Thereby, the corresponding PSI is calculated to be $a_1^4 c_2$ by means of Eq. 19.

4.2 Definition of CI-CFs

According to Def. 7.20 of [12], the cycle index with chirality fittingness (CI-CF) for $\boldsymbol{P}^{(\mathbf{X}_X)}_{\boldsymbol{G}}$ is calculated as follows by using the PSIs (Eq. 19):

$$\text{CI-CF}(\boldsymbol{P}^{(\mathbf{X}_X)}_{\boldsymbol{G}}; \$_k) = \frac{1}{|\boldsymbol{G}|} \sum_{\mathrm{Cl}(G)} |\mathrm{Cl}(G)| \$_1^{\nu_1(\boldsymbol{P}^{(\mathbf{X})}_{\mathrm{Cl}(G)})} \$_2^{\nu_2(\boldsymbol{P}^{(\mathbf{X})}_{\mathrm{Cl}(G)})} \cdots \$_n^{\nu_n(\boldsymbol{P}^{(\mathbf{X})}_{\mathrm{Cl}(G)})}, \tag{20}$$

where the summation concerning $\mathrm{Cl}(G)$ runs to cover the representatives of the conjugacy classes contained in $\boldsymbol{G}$ and the symbol $|\mathrm{Cl}(G)|$ represents the size of the corresponding conjugacy class $\mathrm{Cl}(G)$. The CI-CF (Eq. 20) is a modification of Eq. 9 of [17].

For example, the data of Figure 2 give the respective PCIs collected in the PSI-column. Thereby, we obtain the following CI-CF for characterizing the octahedral skeleton **1**:

$$\begin{aligned} &\text{CI-CF}(\boldsymbol{P}^{(\mathbf{X}_X)}_{\boldsymbol{O}_h\text{-octa}}; \$_k) \\ &= \frac{1}{48}\left\{b_1^6 + 3b_1^2 b_2^2 + 8b_3^2 + 6b_2^3 + 6b_1^2 b_4 + c_2^3 + 3a_1^4 c_2 + 8c_6 + 6a_1^2 c_2^2 + 6c_2 c_4\right\}. \end{aligned} \tag{21}$$

According to Def. 7.25 of [12], the gross enumeration of achiral 3D structures is conducted by using the following CI-CF:

$$\text{CI-CF}^{(a)}(\boldsymbol{P}^{(\mathbf{X}_X)}_{\boldsymbol{G}\text{-octa}}; \$_k) = \frac{2}{|\boldsymbol{G}|} \sum_{\mathrm{Cl}^{(a)}(G)} |\mathrm{Cl}^{(a)}(G)| \$_1^{\nu_1(\boldsymbol{P}^{(\mathbf{X})}_{\mathrm{Cl}^{(a)}(G)})} \$_2^{\nu_2(\boldsymbol{P}^{(\mathbf{X})}_{\mathrm{Cl}^{(a)}(G)})} \cdots \$_n^{\nu_n(\boldsymbol{P}^{(\mathbf{X})}_{\mathrm{Cl}^{(a)}(G)})}, \tag{22}$$

where the summation concerning $\mathrm{Cl}^{(a)}(G)$ runs to cover the representatives of the conjugacy classes contained in $\boldsymbol{G}^{(a)}$, which denotes the coset which contains all of the (roto)reflections of $\boldsymbol{G}$. Note that the coset $\boldsymbol{G}^{(a)}$ satisfies the following coset decomposition:

$$\boldsymbol{G} = \boldsymbol{G}^{(m)} + \boldsymbol{G}^{(a)}, \tag{23}$$

where the symbol $\boldsymbol{G}^{(m)}$ denotes the maximum chiral subgroup of $\boldsymbol{G}$. If the element G is contained in $\mathrm{Cl}^{(a)}(G)$ ($\subset \boldsymbol{G}^{(a)}$), it exhibits $\boldsymbol{P}^{(\mathrm{x})}_{\mathrm{Cl}^{(a)}(G)} = (n+1 \;\; n+2)$. The CI-CF (Eq. 22) is a modification of Eq. 11 of [17].

For example, the right part of Figure 2 is concerned with $\boldsymbol{O}_h^{(a)}$ $(= \boldsymbol{O}i)$. The corresponding PCIs are summed up and the resulting sum is multiplied by $2/|\boldsymbol{O}_h|$ $(= 2/48 = 1/24)$ according to Eq. 22. Thereby, we obtain the following CI-CF:

$$\text{CI-CF}^{(a)}(\boldsymbol{P}^{(\mathbf{X}\chi)}_{\boldsymbol{O}_h\text{-octa}}; \$_k) = \frac{1}{24}\left\{c_2^3 + 3a_1^4c_2 + 8c_6 + 6a_1^2c_2^2 + 6c_2c_4\right\}. \qquad (24)$$

According to Def. 7.28 of [12], the gross enumeration of enantiomeric pairs of chiral 3D structures is conducted by using the following CI-CF:

$$\begin{aligned}
&\text{CI-CF}^{(e)}(\boldsymbol{P}^{(\mathbf{X}\chi)}_{\boldsymbol{G}\text{-octa}}; \$_k) = \\
&\frac{1}{|\boldsymbol{G}|}\left\{\sum_{\text{Cl}^{(m)}(G)} |\text{Cl}^{(m)}(G)| b_1^{\nu_1(\boldsymbol{P}^{(\mathbf{X})}_{\text{Cl}^{(m)}(G)})} b_2^{\nu_2(\boldsymbol{P}^{(\mathbf{X})}_{\text{Cl}^{(m)}(G)})} \cdots b_n^{\nu_n(\boldsymbol{P}^{(\mathbf{X})}_{\text{Cl}^{(m)}(G)})}\right. \\
&\left. - \sum_{\text{Cl}^{(a)}(G)} |\text{Cl}^{(a)}(G)| \$_1^{\nu_1(\boldsymbol{P}^{(\mathbf{X})}_{\text{Cl}^{(a)}(G)})} \$_2^{\nu_2(\boldsymbol{P}^{(\mathbf{X})}_{\text{Cl}^{(a)}(G)})} \cdots \$_n^{\nu_n(\boldsymbol{P}^{(\mathbf{X})}_{\text{Cl}^{(a)}(G)})}\right\}, \qquad (25)
\end{aligned}$$

where the summation concerning $\text{Cl}^{(m)}(G)$ runs to cover the representatives of the conjugacy classes contained in $\boldsymbol{G}^{(m)}$, which denotes the coset which contains all of the rotations of $\boldsymbol{G}$ (cf. Eq. 23). The CI-CF (Eq. 25) is a modification of Eq. 14 of [17].

For example, the data of Figure 2 give the respective PCIs collected in the PSI-column. Thereby, we obtain the following CI-CF$^{(e)}$ for characterizing the octahedral skeleton **1**:

$$\begin{aligned}
&\text{CI-CF}^{(e)}(\boldsymbol{P}^{(\mathbf{X}\chi)}_{\boldsymbol{O}_h\text{-octa}}; \$_k) \\
&= \frac{1}{48}\left\{b_1^6 + 3b_1^2b_2^2 + 8b_3^2 + 6b_2^3 + 6b_1^2b_4 - c_2^3 - 3a_1^4c_2 - 8c_6 - 6a_1^2c_2^2 - 6c_2c_4\right\}. \qquad (26)
\end{aligned}$$

4.3 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. On the basis of the unit procedures mentioned above, the function `CalcConjClassCICF` is developed to calculate the CI-CF of a given group $\boldsymbol{G}$ (Eq. 20). In a similar way, the function `CalcConjClassCICF_A` for calculating the achiral part CI-CF$^{(a)}$ (Eq. 22) and the function `CalcConjClassCICF_E` for calculating the chiral part CI-CF$^{(e)}$ (Eq. 25) are also developed:

```
CalcConjClassCICF(group, degree, degreefull)
CalcConjClassCICF_A(group, degree, degreefull)
CalcConjClassCICF_E(group, degree, degreefull)
```

where the first argument `group` denotes a given group $\boldsymbol{G}$ (as a combined-permutation representation $\boldsymbol{P}^{(\mathbf{X}\chi)}_{\boldsymbol{G}}$), the second argument `degree` denotes the degree of $\boldsymbol{P}^{(\mathbf{X})}_{\boldsymbol{G}}$, 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 `CICFgenCC.gapfunc`, which is attached as Appendix A.

4.4 Practices of Calculation of CI-CFs

A typical procedure for executing the above-developed functions (`CalcConjClassCICF`, `CalcConjClassCICF_A`, and `CalcConjClassCICF_E`) is illustrated by using an octahedral skeleton **1** of $\boldsymbol{O}_h$ (`Oh_octa`). The file `CICFgenCC.gapfunc` containing the functions for generating CI-CFs (cf. Appendix A) is beforehand placed in an appropriate work directory named `c:/fujita0/calcCICF/`. This directory also contains a work file with an appropriate name (e.g., `Oh-octa-CICF1.gap`), which contains the following codes:

Sample Program (`Oh-octa-CICF1.gap`) for Calculating CI-CFs:

```
#Read("c:/fujita0/calcCICF/Oh-octa-CICF1.gap");
LogTo("c:/fujita0/calcCICF/Oh-octa-CICF1log.txt");

Read("c:/fujita0/calcCICF/CICFgenCC.gapfunc"); #Loading of CICFgenCC.gapfunc
gen1 := [(2,3,4,5), (1,2,3)(4,5,6), (1,6)(7,8)];; #generators
Oh_octa := Group(gen1); #octahedral skeleton

Print("CICF_Oh_octa␣:=␣", CalcConjClassCICF(Oh_octa, 6, 8), "\n");
Print("CICF_Oh_octa_A␣:=␣", CalcConjClassCICF_A(Oh_octa, 6, 8), "\n");
Print("CICF_Oh_octa_E␣:=␣", CalcConjClassCICF_E(Oh_octa, 6, 8), "\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 commands contained in the file `Oh-octa-CICF1.gap` are successively executed after the loading of `CICFgenCC.gapfunc`. The calculated CI-CFs are written down into a log file named `Oh-octa-CICF1log.txt` (an appropriate name) as follows:

```
CICF_Oh_octa := 1/48*b_1^6+1/16*a_1^4*c_2+1/16*b_1^2*b_2^2+1/8*a_1^2*c_2^2+1/8*b_1^2*b_4
  +1/48*c_2^3+1/8*b_2^3+1/8*c_2*c_4+1/6*b_3^2+1/6*c_6
CICF_Oh_octa_A := 1/8*a_1^4*c_2+1/4*a_1^2*c_2^2+1/24*c_2^3+1/4*c_2*c_4+1/3*c_6
CICF_Oh_octa_E := 1/48*b_1^6-1/16*a_1^4*c_2+1/16*b_1^2*b_2^2-1/8*a_1^2*c_2^2+1/8*b_1^2*b_4
  -1/48*c_2^3+1/8*b_2^3-1/8*c_2*c_4+1/6*b_3^2-1/6*c_6
```

Each multiplication appearing in these CI-CFs is represented by an asterisk and sphericity indices are represented by `b_1` (for b_1), `a_1` (for a_1), `c_2` (for c_2), and so on. The CI-CFs are consistent with Eqs. 21, 24, and 26. The CI-CF `CICF_Oh_octa` is identical with Eqs. 8 and 9 of [23].

In a similar way, the respective groups for the combined-permutation representations listed in Table 1 (or Tables 2 and 3) generate the corresponding CI-CFs, which are collected in Table 4.

The CI-CFs of the cube **2** have been calculated during the enumeration by Fujita's

Table 4. CI-CFs for Characterizing $\boldsymbol{O}_h$-Skeletons

$\boldsymbol{O}_h$-skeleton	CICFs for total as well as for achiral (_A) and chiral (_C) parts

1, $\boldsymbol{P}^{(\mathbf{X}\chi)}_{\boldsymbol{O}_h\text{-octa}}$
`Oh_octa`
`:= Group(gen1)`
	`CICF_Oh_octa := 1/48*b_1^6+1/16*a_1^4*c_2+1/16*b_1^2*b_2^2+1/8*a_1^2*c_2^2`
	`+1/8*b_1^2*b_4+1/48*c_2^3+1/8*b_2^3+1/8*c_2*c_4+1/6*b_3^2+1/6*c_6`
	`CICF_Oh_octa_A := 1/8*a_1^4*c_2+1/4*a_1^2*c_2^2+1/24*c_2^3+1/4*c_2*c_4+1/3*c_6`
	`CICF_Oh_octa_E := 1/48*b_1^6-1/16*a_1^4*c_2+1/16*b_1^2*b_2^2-1/8*a_1^2*c_2^2`
	`+1/8*b_1^2*b_4-1/48*c_2^3+1/8*b_2^3-1/8*c_2*c_4+1/6*b_3^2-1/6*c_6`

2, $\boldsymbol{P}^{(\mathbf{X}\chi)}_{\boldsymbol{O}_h\text{-cube}}$
`Oh_cube`
`:= Group(gen2)`
	`CICF_Oh_cube := 1/48*b_1^8+1/8*a_1^4*c_2^2+1/6*b_1^2*b_3^2+1/12*c_2^4+3/16*b_2^4`
	`+1/6*c_2*c_6+1/8*b_4^2+1/8*c_4^2`
	`CICF_Oh_cube_A := 1/4*a_1^4*c_2^2+1/6*c_2^4+1/3*c_2*c_6+1/4*c_4^2`
	`CICF_Oh_cube_E := 1/48*b_1^8-1/8*a_1^4*c_2^2+1/6*b_1^2*b_3^2-1/12*c_2^4+3/16*b_2^4`
	`-1/6*c_2*c_6+1/8*b_4^2-1/8*c_4^2`

3, $\boldsymbol{P}^{(\mathbf{X}\chi)}_{\boldsymbol{O}_h\text{-cuboct}}$
`Oh_cuboct`
`:= Group(gen3)`
	`CICF_Oh_cuboct := 1/48*b_1^12+1/16*a_1^4*c_2^4+1/8*b_1^2*b_2^5+1/8*a_1^2*c_2^5`
	`+1/48*c_2^6+1/16*b_2^6+1/6*b_3^4+1/8*b_4^3+1/8*c_4^3+1/6*c_6^2`
	`CICF_Oh_cuboct_A := 1/8*a_1^4*c_2^4+1/4*a_1^2*c_2^5+1/24*c_2^6+1/4*c_4^3+1/3*c_6^2`
	`CICF_Oh_cuboct_E := 1/48*b_1^12-1/16*a_1^4*c_2^4+1/8*b_1^2*b_2^5-1/8*a_1^2*c_2^5`
	`-1/48*c_2^6+1/16*b_2^6+1/6*b_3^4+1/8*b_4^3-1/8*c_4^3-1/6*c_6^2`

4, $\boldsymbol{P}^{(\mathbf{X}\chi)}_{\boldsymbol{O}_h\text{-troct}}$
`Oh_troct`
`:= Group(gen4)`
	`CICF_Oh_troct := 1/48*b_1^24+1/16*a_1^8*c_2^8+7/48*c_2^12+3/16*b_2^12+1/6*b_3^8`
	`+1/8*b_4^6+1/8*c_4^6+1/6*c_6^4`
	`CICF_Oh_troct_A := 1/8*a_1^8*c_2^8+7/24*c_2^12+1/4*c_4^6+1/3*c_6^4`
	`CICF_Oh_troct_E := 1/48*b_1^24-1/16*a_1^8*c_2^8-7/48*c_2^12+3/16*b_2^12+1/6*b_3^8`
	`+1/8*b_4^6-1/8*c_4^6-1/6*c_6^4`

5, $\boldsymbol{P}^{(\mathbf{X}\chi)}_{\boldsymbol{O}_h\text{-trhex}}$
`Oh_trhex`
`:= Group(gen5)`
	`CICF_Oh_trhex := 1/48*b_1^24+1/8*a_1^4*c_2^10+1/12*c_2^12+3/16*b_2^12+1/6*b_3^8`
	`+1/8*b_4^6+1/8*c_4^6+1/6*c_6^4`
	`CICF_Oh_trhex_A := 1/4*a_1^4*c_2^10+1/6*c_2^12+1/4*c_4^6+1/3*c_6^4`
	`CICF_Oh_trhex_E := 1/48*b_1^24-1/8*a_1^4*c_2^10-1/12*c_2^12+3/16*b_2^12+1/6*b_3^8`
	`+1/8*b_4^6-1/8*c_4^6-1/6*c_6^4`

proligand method [12]. Thus, the CI-CF `CICF_Oh_cube` is identical with Eq. 2 of [21], the CI-CF `CICF_Oh_cube_A` is identical with Eq. 17 of [21], and the CI-CF `CICF_Oh_cube_E` is identical with Eq. 20 of [21]. They are also identical with the corresponding CI-CFs calculated by the markaracter method (Eqs. 40, 49, and 50 of [24]) and with those calculated by the characteristic-monomial method (Eqs. 24, 32, and 33 of [25]).

5 Combinatorial Enumeration

5.1 Generating Functions Derived From CI-CFs

According to Theorem 7.14 of [12], 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 a typical example, let us calculate the numbers of octahedral 3D structures by using the following ligand inventory:

$$\boldsymbol{L} = \{\mathrm{H}, \mathrm{A}, \mathrm{B}, \mathrm{C}, \mathrm{D}, \mathrm{W}; \mathrm{p}, \overline{\mathrm{p}}, \mathrm{q}, \overline{\mathrm{q}}\}, \tag{27}$$

where the uppercase letters represent achiral proligands, while a pair of lowercase letters without and with an overbar (p/$\overline{\mathrm{p}}$ or q/$\overline{\mathrm{q}}$) represents a pair of enantiomeric proligands. The corresponding ligand-inventory functions (aa_1, bb_1, etc.) 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/$\overline{\mathrm{p}}$), as found in the following source list:

Sample Program (Oh-Enum-octa2.gap) for Calculating Generating Functions:

```
#Read("c:/fujita0/calcCICF/Oh-Enum-octa2.gap");
LogTo("c:/fujita0/calcCICF/Oh-Enum-octa2log.txt");

Read("c:/fujita0/fujita2016/RSstereoOcta/calcOh/CICFgenCC.gapfunc"); #Loading of CICFgenCC.gapfunc
gen1 := [(2,3,4,5), (1,2,3)(4,5,6), (1,6)(7,8)];; #generators
Oh_octa := Group(gen1); #octahedral skeleton

CICF_Oh_octa := CalcConjClassCICF(Oh_octa, 6, 8);

H := Indeterminate(Rationals, "H"); A := Indeterminate(Rationals, "A");
B := Indeterminate(Rationals, "B"); C := Indeterminate(Rationals, "C");
D := Indeterminate(Rationals, "D"); W := Indeterminate(Rationals, "W");
p := Indeterminate(Rationals, "p"); P := Indeterminate(Rationals, "P");
q := Indeterminate(Rationals, "q"); Q := Indeterminate(Rationals, "Q");

b_1 := Indeterminate(Rationals, "b_1"); b_2 := Indeterminate(Rationals, "b_2");
b_3 := Indeterminate(Rationals, "b_3"); b_4 := Indeterminate(Rationals, "b_4");
a_1 := Indeterminate(Rationals, "a_1"); c_2 := Indeterminate(Rationals, "c_2");
c_4 := Indeterminate(Rationals, "c_4"); c_6 := Indeterminate(Rationals, "c_6");

 aa_1 := H + A + B + C + D + W;
 bb_1 := H + A + B + C + D + W + p + q + P + Q;
 bb_2 := H^2 + A^2 + B^2 + C^2 + D^2 + W^2 + p^2 + q^2 + P^2 + Q^2;
 bb_3 := H^3 + A^3 + B^3 + C^3 + D^3 + W^3 + p^3 + q^3 + P^3 + Q^3;
 bb_4 := H^4 + A^4 + B^4 + C^4 + D^4 + W^4 + p^4 + q^4 + P^4 + Q^4;
 cc_2 := H^2 + A^2 + B^2 + C^2 + D^2 + W^2 + 2*p*P + 2*q*Q;
 cc_4 := H^4 + A^4 + B^4 + C^4 + D^4 + W^4 + 2*p^2*P^2 + 2*q^2*Q^2;
 cc_6 := H^6 + A^6 + B^6 + C^6 + D^6 + W^6 + 2*p^3*P^3 + 2*q^3*Q^3;

f_Oh_octa := Value(CICF_Oh_octa,
[a_1, b_1, b_2, b_3, b_4, c_2, c_4, c_6],
[aa_1, bb_1, bb_2, bb_3, bb_4, cc_2, cc_4, cc_6]);;

Print("f_Oh_octa␣:=␣", f_Oh_octa, "\n");

LogTo();
```

This source list is stored in a work file named Oh-Enum-octa2.gap, which is placed in the above-mentioned directory c:/fujita0/calcCICF. To execute the codes stored in the file Oh-Enum-octa2.gap, 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 commands contained in the file Oh-Enum-octa2.gap are successively executed after the loading of CICFgenCC.gapfunc. The ligand-inventory functions (aa_1, bb_1, etc.) are introduced into the indeterminates (a_1, b_1, etc.)

contained in the CI-CF `CICF_Oh_octa` (Table 4, Eq. 21) according to Theorem 7.21 of [12]. The resulting generating function `f_Oh_octa` is written down into a log file named `Oh-Enum-octa2log.txt` (an appropriate name) as follows:

```
f_Oh_octa := H^6+H^5*A+H^5*B+H^5*C+H^5*D+H^5*W+1/2*H^5*p+1/2*H^5*P+1/2*H^5*q+1/2*H^5*Q+2*H^4*A^2
+2*H^4*A*B+2*H^4*A*C+2*H^4*A*D+2*H^4*A*W+H^4*A*p+H^4*A*P+H^4*A*q+H^4*A*Q+2*H^4*B^2+2*H^4*B*C
+2*H^4*B*D+2*H^4*B*W+H^4*B*p+H^4*B*P+H^4*B*q
(omitted)
+15*H*A*B*C*D*W+15*H*A*B*C*D*p+15*H*A*B*C*D*P+15*H*A*B*C*D*q+15*H*A*B*C*D*Q+9*H*A*B*C*W^2
+15*H*A*B*C*W*p+15*H*A*B*C*W*P+15*H*A*B*C*W*q+15*H*A*B*C*W*Q+15/2*H*A*B*C*p^2+18*H*A*B*C*p*P
+15*H*A*B*C*p*q+15*H*A*B*C*p*Q+15/2*H*A*B*C*P^2+15*H*A*B*C*P*q+15*H*A*B*C*P*Q+15/2*H*A*B*C*q^2
+18*H*A*B*C*q*Q+15/2*H*A*B*C*Q^2+4*H*A*B*D^3+9*H*A*B*D^2*W+15/2*H*A*B*D^2\
(omitted)
+3/2*P*q^2*Q^3+P*q*Q^4+1/2*P*Q^5+1/2*q^6+1/2*q^5*Q+q^4*Q^2+2*q^3*Q^3+q^2*Q^4+1/2*q*Q^5+1/2*Q^6
```

The coefficient of each term $\mathrm{H}^h\mathrm{A}^a\mathrm{B}^b\mathrm{C}^c\mathrm{D}^d\mathrm{W}^w\mathrm{p}^p\overline{\mathrm{p}}^{\overline{p}}\mathrm{q}^q\overline{\mathrm{q}}^{\overline{q}}$ $(h + a + b + c + d + w + p + \overline{p} + q + \overline{q} = 6)$ represents the number of pairs of (self-)enantiomeric octahedral derivatives with the respective composition, where a pair of enantiomers is counted once and a pair of self-enantiomers represents one achiral derivative. For example, the term `15*H*A*B*C*D*W` indicates the presence of 15 pairs of enantiomers with the composition HABCDW. On the other hand, the term `18*H*A*B*C*p*P` indicates the presence of 18 pairs of (self-)enantiomers with the composition HABCp$\overline{\mathrm{p}}$, which are found to be composed of 6 achiral derivatives and 12 pairs of enantiomers. This is confirmed by the fact that the generating function calculated from `Oh_octa_A` (Table 4, Eq. 24) is determined to contain `6*H*A*B*C*p*P`, while the generating function calculated from `Oh_octa_E` (Table 4, Eq. 26) is determined to contain `12*H*A*B*C*p*P`. It is to be noted that a pair of fractional coefficients such as `1/2*H^5*p+1/2*H^5*P` should be regarded as $1 \times \frac{1}{2}(\mathrm{H}^5\mathrm{p} + \mathrm{H}^5\overline{\mathrm{p}})$, which indicates the presence of one pair of enantiomers.

5.2 Selective Calculation of Coefficients of Generating Functions

Because such generating functions as calculated above contain huge numbers of terms, it is convenient to calculate the coefficient of a specific term which is necessary to our discussions. For example, the coefficient 15 of the term `15*H*A*B*C*D*W` appearing in the generating function `f_Oh_octa` is obtained selectively, where the GAP function `PolynomialCoefficientsOfPolynomial` is used in a nested fashion as follows:

```
gap> coeff_H := PolynomialCoefficientsOfPolynomial(f_Oh_octa, H);;
gap> coeff_HA := PolynomialCoefficientsOfPolynomial(coeff_H[2], A);;
gap> coeff_HAB := PolynomialCoefficientsOfPolynomial(coeff_HA[2], B);;
gap> coeff_HABC := PolynomialCoefficientsOfPolynomial(coeff_HAB[2], C);;
gap> coeff_HABCD := PolynomialCoefficientsOfPolynomial(coeff_HABC[2], D);;
gap> coeff_HABCDW := PolynomialCoefficientsOfPolynomial(coeff_HABCD[2], W);
[ 15*p+15*P+15*q+15*Q, 15 ]
gap> Print("coeff␣HABCDW␣:=", coeff_HABCDW[2], "\n");
coeff HABCDW :=15
```

The GAP function PolynomialCoefficientsOfPolynomial is applied to the generating function f_Oh_octa in a similar way, so that the coefficient 18 of the term 18*H*A*B*C*p*P is obtained selectively as follows:

```
gap> coeff_HABCp := PolynomialCoefficientsOfPolynomial(coeff_HABC[2], p);;
gap> coeff_HABCpP := PolynomialCoefficientsOfPolynomial(coeff_HABCp[2], P);
[ 15*D+15*W+15*q+15*Q, 18 ]
gap> Print("coeff␣HABCpP␣:=", coeff_HABCpP[2], "\n");
coeff HABCpP :=18
```

A function calcCoeffGen is developed in order to simplify the above-mentioned process of calculating coefficients of a generating function genfunc:

calcCoeffGen(genfunc, list_ligand, list_partition)

For the sake of convenience, such a mode of substitution as $\mathrm{H}^h\mathrm{A}^a\mathrm{B}^b\mathrm{C}^c\mathrm{D}^d\mathrm{W}^w\mathrm{p}^p\overline{\mathrm{p}}^{\overline{p}}\mathrm{q}^q\overline{\mathrm{q}}^{\overline{q}}$ is represented by a substitution pattern $[h, a, b, c, d, w; p, \overline{p}, q, \overline{q}]$ $(h+a+b+c+d+w+p+\overline{p}+q+\overline{q}=6)$, where we can presume $h \geq a \geq b \geq c \geq d \geq w$; $p \geq q$, $p \geq \overline{p}$, and $q \geq \overline{q}$ without losing generality. For example, the substitution pattern $[1, 1, 1, 1, 1, 1; 0, 0, 0, 0]$ corresponds to the term represented by HABCDW, while $[1, 1, 1, 1, 0, 0; 1, 1, 0, 0]$ corresponds to the terms represented by $\mathrm{HABCp\overline{p}}$, $\mathrm{HABDp\overline{p}}$, and so on. Such a substitution pattern is stored in a list list_partition, which collects respective exponents according to the appearance of proligands in a ligand inventory list_ligand, e.g., [H,A,B,C,D,W,p,P,q,Q] (cf. Eq. 27).

The source list of the function calcCoeffGen is stored in the above-mentioned file CICFgenCC.gapfunc, which is attached as Appendix A. Respective coefficients are calculated by using the function calcCoeffGen after loading the file CICFgenCC.gapfunc. For example, the coefficients of the terms 15*H*A*B*C*D*W and 18*H*A*B*C*p*P in the generating function f_Oh_octa (cf. the above-mentioned program Oh-Enum-octa2.gap) are calculated as follows:

```
gap> calcCoeffGen(f_Oh_octa, [H,A,B,C,D,W,p,P,q,Q], [1,1,1,1,1,1,0,0,0,0]);
15
gap> calcCoeffGen(f_Oh_octa, [H,A,B,C,D,W,p,P,q,Q], [1,1,1,1,0,0,1,1,0,0]);
18
```

The function calcCoeffGen is designed to be also applicable to generating functions for calculating achiral promolecules (e.g., f_Oh_octa_A) and for calculating pairs of enantiomers (e.g., f_Oh_octa_E) as follows:

```
gap> calcCoeffGen(f_Oh_octa, [H,A,B,C,D,W,p,P,q,Q], [1,1,1,1,1,1,0,0,0,0]);
15
gap> calcCoeffGen(f_Oh_octa_A, [H,A,B,C,D,W,p,P,q,Q], [1,1,1,1,1,1,0,0,0,0]);
0
gap> calcCoeffGen(f_Oh_octa_E, [H,A,B,C,D,W,p,P,q,Q], [1,1,1,1,1,1,0,0,0,0]);
15
```

Table 5. Numbers of Octahedral Derivatives with Achiral Proligands

partition	3D	A	E	partition	3D	A	E
$[6,0,0,0,0,0,0,0,0,0]$	1	1	0	$[5,1,0,0,0,0,0,0,0,0]$	1	1	0
$[4,2,0,0,0,0,0,0,0,0]$	2	2	0	$[4,1,1,0,0,0,0,0,0,0]$	2	2	0
$[3,3,0,0,0,0,0,0,0,0]$	2	2	0	$[3,2,1,0,0,0,0,0,0,0]$	3	3	0
$[3,1,1,1,0,0,0,0,0,0]$	4	3	1	$[2,2,2,0,0,0,0,0,0,0]$	5	4	1
$[2,2,1,1,0,0,0,0,0,0]$	6	4	2	$[2,1,1,1,1,0,0,0,0,0]$	9	3	6
$[1,1,1,1,1,1,0,0,0,0]$	15	0	15				

The function `calcCoeffGen` is convenient to transform generating functions into the corresponding tabular forms of coefficients, as shown in Table 5. Table 5 collects the numbers of octahedral derivatives with achiral proligands. The 3D-column of Table 5 lists the coefficients of the generating function `f_Oh_octa`, the A-column lists the coefficients of the generating function `f_Oh_octa_A`, and the E-column lists the coefficients of the generating function `f_Oh_octa_E`.

The data of Table 5 are consistent with Table 5 of [19], which reported the symmetry-itemized enumeration of octahedral derivatives on the basis of Fujita's USCI approach [18]. For example, the values 9, 3, and 6 appearing in the $[2,1,1,1,1,0,0,0,0,0]$-row of Table 5 (the composition H^2ABCD) correspond to the value 9 of the Total-column, the value 3 of the $\boldsymbol{C}_s$-column (achiral), and the value 6 of the $\boldsymbol{C}_1$-column (chiral), which appear at the [21111]-row of Table 5 of [19].

Table 6 collects the numbers of octahedral derivatives with achiral and chiral proligands. The coefficients listed in each partition with an asterisk should be duplicated to obtain the numbers of promolecules. For example, the fraction 1/2 appearing in the $[5,0,0,0,0,0,1,0,0,0]$*-row corresponds to the term $1 \times \frac{1}{2}(H^5p + H^5\overline{p})$, because the partition is implicitly accompanied by the $[5,0,0,0,0,0,0,1,0,0]$*. It follows that this value indicates the presence of one pair of enantiomers. The value 1 appearing in the $[4,1,0,0,0,0,1,0,0,0]$*-row corresponds to the term $2 \times \frac{1}{2}(H^4Ap + H^4A\overline{p})$, so that it indicates the presence of two pairs of enantiomers (*cis* and *trans*).

The data of Table 6 are consistent with Table 7 of [19]. For example, the values 1/2, 0, 1/2 appearing in the $[5,0,0,0,0,0,1,0,0,0]$*-row of Table 6 (1 for 3D (total promolecules), 0 for A (achiral promolecules), and 1 for E (enantiomeric pairs)) corresponds to the value 1 at the intersection of the [5; 1]-row and the $\boldsymbol{C}_4$-column (chiral) of Table 7 [19], where

Table 6. Numbers of Octahedral Derivatives with Achiral and Chiral Proligands

partition	3D	A	E	partition	3D	A	E
[5, 0, 0, 0, 0, 0, 1, 0, 0, 0]*	1/2	0	1/2	[4, 1, 0, 0, 0, 0, 1, 0, 0, 0]*	1	0	1
[4, 0, 0, 0, 0, 0, 2, 0, 0, 0]*	1	0	1	[4, 0, 0, 0, 0, 0, 1, 1, 0, 0]	2	2	0
[4, 0, 0, 0, 0, 0, 1, 0, 1, 0]*	1	0	1	[3, 2, 0, 0, 0, 0, 1, 0, 0, 0]*	3/2	0	3/2
[3, 1, 1, 0, 0, 0, 1, 0, 0, 0]*	5/2	0	5/2	[3, 1, 0, 0, 0, 0, 2, 0, 0, 0]*	3/2	0	3/2
[3, 1, 0, 0, 0, 0, 1, 1, 0, 0]	4	3	1	[3, 1, 0, 0, 0, 0, 1, 0, 1, 0]*	5/2	0	5/2
[3, 0, 0, 0, 0, 0, 3, 0, 0, 0]*	1	0	1	[3, 0, 0, 0, 0, 0, 2, 1, 0, 0]*	3/2	0	3/2
[3, 0, 0, 0, 0, 0, 2, 0, 1, 0]*	3/2	0	3/2	[3, 0, 0, 0, 0, 0, 1, 1, 1, 0]*	5/2	0	5/2
[2, 2, 1, 0, 0, 0, 1, 0, 0, 0]*	4	0	4	[2, 2, 0, 0, 0, 0, 2, 0, 0, 0]*	3	0	3
[2, 2, 0, 0, 0, 0, 1, 1, 0, 0]	6	4	2	[2, 2, 0, 0, 0, 0, 1, 0, 1, 0]*	4	0	4
[2, 1, 1, 0, 0, 0, 2, 0, 0, 0]*	4	0	4	[2, 1, 1, 0, 0, 0, 1, 1, 0, 0]	10	5	5
[2, 1, 1, 0, 0, 0, 1, 0, 1, 0]*	15/2	0	15/2	[2, 1, 0, 0, 0, 0, 3, 0, 0, 0]*	3/2	0	3/2
[2, 1, 0, 0, 0, 0, 2, 1, 0, 0]*	4	0	4	[2, 1, 0, 0, 0, 0, 2, 0, 1, 0]*	4	0	4
[2, 1, 0, 0, 0, 0, 1, 1, 1, 0]*	15/2	0	15/2	[2, 0, 0, 0, 0, 0, 4, 0, 0, 0]*	1	0	1
[2, 0, 0, 0, 0, 0, 3, 1, 0, 0]*	3/2	0	3/2	[2, 0, 0, 0, 0, 0, 3, 0, 1, 0]*	3/2	0	3/2
[2, 0, 0, 0, 0, 0, 2, 2, 0, 0]	4	2	2	[2, 0, 0, 0, 0, 0, 2, 1, 1, 0]*	4	0	4
[2, 0, 0, 0, 0, 0, 2, 0, 1, 1]*	4	0	4	[2, 0, 0, 0, 0, 0, 1, 1, 1, 1]	9	3	6
[1, 1, 1, 1, 1, 0, 1, 0, 0, 0]*	15	0	15	[1, 1, 1, 1, 0, 0, 2, 0, 0, 0]*	15/2	0	15/2
[1, 1, 1, 1, 0, 0, 1, 1, 0, 0]	18	6	12	[1, 1, 1, 1, 0, 0, 1, 0, 1, 0]*	15	0	15
[1, 1, 1, 0, 0, 0, 3, 0, 0, 0]*	5/2	0	5/2	[1, 1, 1, 0, 0, 0, 2, 1, 0, 0]*	15/2	0	15/2
[1, 1, 1, 0, 0, 0, 2, 0, 1, 0]*	15/2	0	15/2	[1, 1, 1, 0, 0, 0, 1, 1, 1, 0]*	15	0	15
[1, 1, 0, 0, 0, 0, 4, 0, 0, 0]*	1	0	1	[1, 1, 0, 0, 0, 0, 3, 1, 0, 0]*	5/2	0	5/2
[1, 1, 0, 0, 0, 0, 3, 0, 1, 0]*	5/2	0	5/2	[1, 1, 0, 0, 0, 0, 2, 2, 0, 0]	5	2	3
[1, 1, 0, 0, 0, 0, 2, 1, 1, 0]*	15/2	0	15/2	[1, 1, 0, 0, 0, 0, 2, 0, 1, 1]*	15/2	0	15/2
[1, 1, 0, 0, 0, 0, 1, 1, 1, 1]	17	4	13	[1, 0, 0, 0, 0, 0, 5, 0, 0, 0]*	1/2	0	1/2
[1, 0, 0, 0, 0, 0, 4, 1, 0, 0]*	1	0	1	[1, 0, 0, 0, 0, 0, 4, 0, 1, 0]*	1	0	1
[1, 0, 0, 0, 0, 0, 3, 2, 0, 0]*	3/2	0	3/2	[1, 0, 0, 0, 0, 0, 3, 0, 2, 0]*	3/2	0	3/2
[1, 0, 0, 0, 0, 0, 3, 1, 1, 0]*	5/2	0	5/2	[1, 0, 0, 0, 0, 0, 3, 0, 1, 1]*	5/2	0	5/2
[1, 0, 0, 0, 0, 0, 2, 2, 1, 0]*	4	0	4	[1, 0, 0, 0, 0, 0, 2, 1, 2, 0]*	4	0	4
[1, 0, 0, 0, 0, 0, 2, 1, 1, 1]*	15/2	0	15/2	[0, 0, 0, 0, 0, 0, 6, 0, 0, 0]*	1/2	0	1/2
[0, 0, 0, 0, 0, 0, 5, 1, 0, 0]*	1/2	0	1/2	[0, 0, 0, 0, 0, 0, 5, 0, 1, 0]*	1/2	0	1/2
[0, 0, 0, 0, 0, 0, 4, 2, 0, 0]*	1	0	1	[0, 0, 0, 0, 0, 0, 4, 1, 1, 0]*	1	0	1
[0, 0, 0, 0, 0, 0, 4, 0, 2, 0]*	1	0	1	[0, 0, 0, 0, 0, 0, 4, 0, 1, 1]*	1	0	1
[0, 0, 0, 0, 0, 0, 3, 3, 0, 0]	2	2	0	[0, 0, 0, 0, 0, 0, 3, 2, 1, 0]*	3/2	0	3/2
[0, 0, 0, 0, 0, 0, 3, 1, 1, 1]	5/2	0	5/2	[0, 0, 0, 0, 0, 0, 3, 0, 3, 0]*	1	0	1
[0, 0, 0, 0, 0, 0, 3, 0, 2, 1]*	3/2	0	3/2				

this value means 1 for 3D, 0 for A, and 1 for E in the present context. The set of values 4 for 3D (total), 3 for A (achiral), and 1 for E (enantiomeric pairs) appearing in the $[3, 1, 0, 0, 0, 0, 1, 1, 0, 0]$-row of Table 6 corresponds to the set of values 1 for $\boldsymbol{C}_1$ (chiral), 1 for $\boldsymbol{C}_s$ (achiral), and 2 for $\boldsymbol{C}'_s$ (achiral) appearing in the $[31; 11]$-row of Table 7 [19], where this set of values means 4 ($= 1 + 1 + 2$) for 3D, 3 ($= 1 + 2$) for A, and 1 for E in the present context.

5.3 Merits of the Present Procedure in Enumeration by Fujita's Proligand Method

In a previous procedure, the concrete form of a coset representation (e.g., $\boldsymbol{O}_h(/\boldsymbol{C}_{4v})$ of Figure 2) has been first calculated by applying a FORTRAN program (unpublished) to a coset decomposition [18]. Then, it has been used to calculate the corresponding CI-CF, which in turn has been applied to combinatorial enumeration by using such a computer system as the Maple system [12]. The source list of a Maple program for calculating the numbers of cubane derivatives, for example, has been reported as an appendix in an article appearing in this journal [21].

On the other hand, the present procedure does not explicitly require the concrete form of a coset representation during the calculation of a CI-CF, but instead, it treats a permutation group defined from an appropriate set of generators. By using the newly-defined GAP functions (`CalcConjClassCICF`, `calcCoeffGen`, and so on stored in the file `CICFgenCC.gapfunc` of Appendix A), a straightforward procedure for enumeration can be developed by using the GAP system.

As a typical example of such a straightforward procedure, Appendix B shows a source list of calculating the numbers of cubane derivatives, which is stored in a work file named `Oh-Enum-cube.gap`. In this procedure, the CI-CFs for **2**, i.e., `CICF_Oh_cube`, `CICF_Oh_cube_A`, and `CICF_Oh_cube_E` (Table 4), are calculated respectively by using `CalcConjClassCICF`, `CalcConjClassCICF_A`, and `CalcConjClassCICF_E` after loading the file `CICFgenCC.gapfunc` (Appendix A). Then, the coefficients of the corresponding generating functions are calculated by means of `calcCoeffGen`, which is also effective to this case of treating eight achiral proligands (and two pairs of enantiomeric proligands). The data obtained by the execution of `Oh-Enum-cube.gap` are concerned with the numbers of cubane derivatives with achiral proligands. They are formatted into a tabular form so as to give Table 7. Note that the 8 digits of each partition are concerned with eight achi-

Table 7. Numbers of Cubane Derivatives with Achiral Proligands

partition	3D	A	E	partition	3D	A	E
$[8,0,0,0,0,0,0,0,0,0,0,0]$	1	1	0	$[7,1,0,0,0,0,0,0,0,0,0,0]$	1	1	0
$[6,2,0,0,0,0,0,0,0,0,0,0]$	3	3	0	$[6,1,1,0,0,0,0,0,0,0,0,0]$	3	3	0
$[5,3,0,0,0,0,0,0,0,0,0,0]$	3	3	0	$[5,2,1,0,0,0,0,0,0,0,0,0]$	6	5	1
$[5,1,1,1,0,0,0,0,0,0,0,0]$	10	6	4	$[4,4,0,0,0,0,0,0,0,0,0,0]$	6	5	1
$[4,3,1,0,0,0,0,0,0,0,0,0]$	10	7	3	$[4,2,2,0,0,0,0,0,0,0,0,0]$	16	10	6
$[4,2,1,1,0,0,0,0,0,0,0,0]$	22	9	13	$[4,1,1,1,1,0,0,0,0,0,0,0]$	38	6	32
$[3,3,2,0,0,0,0,0,0,0,0,0]$	17	10	7	$[3,3,1,1,0,0,0,0,0,0,0,0]$	30	12	18
$[3,2,2,1,0,0,0,0,0,0,0,0]$	42	14	28	$[3,2,1,1,1,0,0,0,0,0,0,0]$	76	12	64
$[3,1,1,1,1,1,0,0,0,0,0,0]$	140	0	140	$[2,2,2,2,0,0,0,0,0,0,0,0]$	68	22	46
$[2,2,2,1,1,0,0,0,0,0,0,0]$	114	18	96	$[2,2,1,1,1,1,0,0,0,0,0,0]$	216	12	204
$[2,1,1,1,1,1,1,0,0,0,0,0]$	420	0	420	$[1,1,1,1,1,1,1,1,0,0,0,0]$	840	0	840

ral proligands and the remaining four digits are concerned with two pairs of enantiomeric proligands. For example, the values 6, 5, and 1 in the $[5,2,1,0,0,0,0,0,0,0,0,0]$-row corresponding to the composition H^5A^2B (without no chiral proligands) indicates the total number 6, which consists of five achiral promolecules and one pair of enantiomers. The data resulted by the GAP system (Table 7) are consistent with those collected in Table 2 of [21], which have been obtained by the combination of the FORTRAN program and the Maple program.

The numbers of cubane derivatives with achiral and chiral proligands are obtained in a similar way. Several selected data are collected in Table 8. The numbers of each row with an asterisk should be duplicated to give the numbers of enantiomeric pairs. For example, the values 1/2, 0, and 1/2 in the $[7,0,0,0,0,0,0,0,1,0,0,0]$*-row (corresponding to the composition $\frac{1}{2}(H^7p + H^7\overline{p})$) indicates the total number 1, which consists of no achiral promolecule and one pair of enantiomers. The data of Table 8 are consistent with those collected in Table 3 of [21].

6 Conclusion

The GAP functions `CalcConjClassCICF`, `..._A`, and `..._E` have been developed to generate respective CI-CFs, which are used to calculate generating functions for giving the total numbers of 3D structures, those of achiral 3D structures, and those of enantiomeric pairs of chiral 3D structures. During the calculations of CI-CFs, conjugacy classes are

Table 8. Numbers of Cubane Derivatives with Achiral and Chiral Proligands

partition	3D	A	E	partition	3D	A	E
$[7,0,0,0,0,0,0,0,1,0,0,0]^*$	1/2	0	1/2	$[6,1,0,0,0,0,0,0,1,0,0,0]^*$	3/2	0	3/2
$[6,0,0,0,0,0,0,0,2,0,0,0]^*$	3/2	0	3/2	$[6,0,0,0,0,0,0,0,1,1,0,0]$	3	3	0
$[6,0,0,0,0,0,0,0,1,0,1,0]^*$	3/2	0	3/2	$[5,2,0,0,0,0,0,0,1,0,0,0]^*$	7/2	0	7/2
$[5,1,0,0,0,0,0,0,2,0,0,0]^*$	7/2	0	7/2	$[5,1,0,0,0,0,0,0,1,1,0,0]$	9	4	5
$[5,1,0,0,0,0,0,0,1,0,1,0]^*$	7	0	7	$[5,0,0,0,0,0,0,0,3,0,0,0]^*$	3/2	0	3/2
$[5,0,0,0,0,0,0,0,2,1,0,0]^*$	7/2	0	7/2	$[5,0,0,0,0,0,0,0,2,0,1,0]^*$	7/2	0	7/2
$[5,0,0,0,0,0,0,0,1,1,1,0]^*$	7	0	7	$[4,3,0,0,0,0,0,0,1,0,0,0]^*$	13/2	0	13/2
$[4,2,0,0,0,0,0,0,2,0,0,0]^*$	11	0	11	$[4,2,0,0,0,0,0,0,1,1,0,0]$	23	11	12
$[4,2,0,0,0,0,0,0,1,0,1,0]^*$	35/2	0	35/2	$[4,1,1,0,0,0,0,0,2,0,0,0]^*$	35/2	0	35/2
$[4,1,1,0,0,0,0,0,1,1,0,0]$	41	12	29	$[4,1,1,0,0,0,0,0,1,0,1,0]^*$	35	0	35
$[4,1,0,0,0,0,0,0,3,0,0,0]^*$	13/2	0	13/2	$[4,1,0,0,0,0,0,0,2,1,0,0]^*$	35/2	0	35/2
$[4,1,0,0,0,0,0,0,2,0,1,0]^*$	35/2	0	35/2	$[4,1,0,0,0,0,0,0,1,1,1,0]^*$	35	0	35
$[4,0,0,0,0,0,0,0,4,0,0,0]^*$	7/2	0	7/2	$[4,0,0,0,0,0,0,0,3,1,0,0]^*$	13/2	0	13/2
$[4,0,0,0,0,0,0,0,3,0,1,0]^*$	13/2	0	13/2	$[4,0,0,0,0,0,0,0,2,2,0,0]$	14	6	8
$[4,0,0,0,0,0,0,0,2,0,2,0]^*$	11	0	11	$[4,0,0,0,0,0,0,0,2,1,1,0]^*$	35/2	0	35/2
$[4,0,0,0,0,0,0,0,2,0,1,1]^*$	35/2	0	35/2	$[4,0,0,0,0,0,0,0,1,1,1,1]$	40	10	30

taken into consideration after the elements of a point group are classified into rotations and (roto)reflections by means of a mirror-permutation representation. The CI-CFs have been calculated to characterize $\boldsymbol{O}_h$-skeletons (an octahedron, a cube, a cuboctahedron, a truncated octahedron, and a truncated hexahedron) and applied to combinatorial enumeration of promolecules derived from these skeletons.

Appendix A. Source List of `CICFgenCC.gapfunc` for Calculating CI-CFs

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

```
#CICFgenCC.gapfunc
####################################
# Division into Conjugacy Classes #
####################################
divideConjClasses := function(G, degree, degreefull)
local i, j, Orig_Grp, l_chiral_elements, l_achiral_elements, l_chiralachiral,
conj_class, e_conj_class, r_conj_class,
DegGr, DegCGr, temp_cycstrX, AchOrCh, l_conjelem, l_chiral_achiral;

Orig_Grp := G; #group to be examined
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]
conj_class := ConjugacyClasses(Orig_Grp);
l_chiral_elements := [];
l_achiral_elements := [];
l_chiralachiral := [];
for j in [1..Size(conj_class)] do
```

```
e_conj_class := Elements(conj_class[j]);
r_conj_class := Representative(conj_class[j]);
temp_cycstrX := CycleLengths(r_conj_class, [1..DegGr]); #full degree
AchOrCh := temp_cycstrX[Size(temp_cycstrX)]; #chiral 1; achiral 2
if AchOrCh = 1 then
l_conjelem := [];
for i in [1..Size(e_conj_class)] do
Add(l_conjelem, e_conj_class[i]);
od;
Add(l_chiral_elements, l_conjelem);
else
l_conjelem := [];
for i in [1..Size(e_conj_class)] do
Add(l_conjelem, e_conj_class[i]);
od;
Add(l_achiral_elements, l_conjelem);
fi;
od;
#Display(l_chiral_elements); #for debug
#Display(l_achiral_elements); #for debug
l_chiral_achiral := [l_chiral_elements, l_achiral_elements];
return l_chiral_achiral;
end; #end of function divideConjClass

################################################
# Calaculation of CICF based Conjugacy Classes #
################################################
CalcConjClassCICF := function(G, degree, degreefull)
local i, j, k, Orig_Grp,
order_Orig_Grp, conj_class, n_conj_class, CICF,
r_conj_class, s_conj_class,
DegGr, DegCGr, temp_cycstr, temp_cycstrX, AchOrCh,
tempSI,tempSIX,tempSIY,tempCICF;

Orig_Grp := G; #group to be examined
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]

order_Orig_Grp := Size(Orig_Grp);
conj_class := ConjugacyClasses(Orig_Grp);
n_conj_class := Size(conj_class);
CICF := 0;
for j in [1..n_conj_class] do
r_conj_class := Representative(conj_class[j]);
s_conj_class := Size(conj_class[j]);
temp_cycstr := CycleLengths(r_conj_class, [1..DegCGr]); #degree
temp_cycstrX := CycleLengths(r_conj_class, [1..DegGr]); #full degree
AchOrCh := temp_cycstrX[Size(temp_cycstrX)]; #chiral 1; achiral 2
#Determination of sphericity indices (SIs) of cycles
tempCICF := s_conj_class; # initial value in place of 1
for k in [1..Size(temp_cycstr)] do
if DegGr = DegCGr then
 tempSI := ["b_", temp_cycstr[k]]; #hemispheric cycle
else
 if AchOrCh = 1 then
   tempSI := ["b_", temp_cycstr[k]]; #hemispheric cycle
 else
  if IsOddInt(temp_cycstr[k]) then
    tempSI := ["a_", temp_cycstr[k]]; #homospheric cycle
   else
    tempSI := ["c_", temp_cycstr[k]]; #enantiospheric cycle
  fi;
 fi;
fi;
#Calculation of products of sphericity indices (PSIs)
tempSIX := JoinStringsWithSeparator(tempSI, "");
tempSIY := Indeterminate(Rationals, tempSIX);
tempCICF := tempCICF*tempSIY;
od;
CICF := CICF + (1/order_Orig_Grp)*tempCICF;
od;
#Display(CICF); #for debug
return CICF;
```

```
end; #end of the function CalcConjClassCICF

##############################
# Achiral Part of CICF for G #
##############################
CalcConjClassCICF_A := function(G, degree, degreefull)
local i, j, k, Orig_Grp,
order_Orig_Grp, conj_class, n_conj_class, CICF_A,
r_conj_class, s_conj_class,
DegGr, DegCGr, temp_cycstr, temp_cycstrX, AchOrCh,
tempSI,tempSIX,tempSIY,tempCICF;

Orig_Grp := G; #group to be examined
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]

order_Orig_Grp := Size(Orig_Grp);
conj_class := ConjugacyClasses(Orig_Grp);
n_conj_class := Size(conj_class);
CICF_A := 0;
for j in [1..n_conj_class] do
r_conj_class := Representative(conj_class[j]);
s_conj_class := Size(conj_class[j]);
temp_cycstr := CycleLengths(r_conj_class, [1..DegCGr]); #degree
temp_cycstrX := CycleLengths(r_conj_class, [1..DegGr]); #full degree
AchOrCh := temp_cycstrX[Size(temp_cycstrX)]; #chiral 1; achiral 2
#Determination of sphericity indices (SIs) of cycles
if AchOrCh = 1 then
   tempCICF := 0;
 else
   tempCICF := s_conj_class; # initial value in place of 1
   for k in [1..Size(temp_cycstr)] do
    if DegGr = DegCGr then
      else
        if IsOddInt(temp_cycstr[k]) then
          tempSI := ["a_", temp_cycstr[k]]; #homospheric cycle
        else
          tempSI := ["c_", temp_cycstr[k]]; #enantiospheric cycle
        fi;
      #Calculation of products of sphericity indices (PSIs)
      tempSIX := JoinStringsWithSeparator(tempSI, "");
      tempSIY := Indeterminate(Rationals, tempSIX);
      tempCICF := tempCICF*tempSIY;
    fi;
   od;
fi;
  CICF_A := CICF_A + (2/order_Orig_Grp)*tempCICF;
od;
#Display(CICF_A); #for debug
return CICF_A;
end; #end of the function CalcConjClassCICF_A

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

##############################################################
# Coefficients of a Generating Function Derived from a CI-CF #
##############################################################
list_ligand := [];
list_partition := [];
calcCoeffGen := function(genfunc, list_ligand, list_partition)
```

```
local i, j, l_p, l_l, temppt, coeff, tempcoeff;
l_l := list_ligand;
l_p := list_partition;
coeff := 99999999; #tentative assignment
if Size(l_l) = Size(l_p) then
temppt := PolynomialCoefficientsOfPolynomial(genfunc, l_l[1]);
for i in [1..Size(l_l)-1] do
   if Size(temppt) < l_p[i]+1 then
     coeff := 0;
     break;
   else
     temppt := PolynomialCoefficientsOfPolynomial(temppt[l_p[i]+1], l_l[i+1]);
   fi;
od;
if coeff = 0 then
  else
  coeff := temppt[l_p[Size(l_l)]+1];
fi;
return coeff;
else
return false;
fi;
end; #end of the function calcCoeffGen
```

Appendix B. Source List of `Oh-Enum-cube.gap` for Enumerating Cubane Derivatives

The following program for combinatorial enumeration of cubane derivatives is stored in a file named `Oh-Enum-cube.gap` (an arbitrary name), which is placed in a work directory named `c:/fujita0/calcCICF/Oh-Enum-cube` (an arbitrary name). To execute this file, the first line commented out by the # symbol is copied and paste after the `gap>` prompt in the command prompt of the Windows operating system. The output is stored in the log file named `Oh-Enum-cubelog.txt` (an arbitrary name), which contains the data for constructing Table 7.

```
#Read("c:/fujita0/calcCICF/Oh-Enum-cube.gap");
LogTo("c:/fujita0/calcCICF/Oh-Enum-cubelog.txt");

Read("c:/fujita0/fujita2016/RSstereoOcta/calcOh/CICFgenCC.gapfunc"); #Loading of CICFgenCC.gapfunc
gen2 := [(1,3)(2,4)(5,7)(6,8), (1,2,3,4)(5,6,7,8), (2,4,5)(3,8,6),
(1,5)(2,6)(3,7)(4,8)(9,10)];
Oh_cube := Group(gen2); #cube

CICF_Oh_cube := CalcConjClassCICF(Oh_cube, 8, 10);
CICF_Oh_cube_A := CalcConjClassCICF_A(Oh_cube, 8, 10);
CICF_Oh_cube_E := CalcConjClassCICF_E(Oh_cube, 8, 10);

H := Indeterminate(Rationals, "H"); A := Indeterminate(Rationals, "A");
B := Indeterminate(Rationals, "B"); C := Indeterminate(Rationals, "C");
D := Indeterminate(Rationals, "D"); U := Indeterminate(Rationals, "U");
V := Indeterminate(Rationals, "V"); W := Indeterminate(Rationals, "W");
p := Indeterminate(Rationals, "p"); P := Indeterminate(Rationals, "P");
q := Indeterminate(Rationals, "q"); Q := Indeterminate(Rationals, "Q");

b_1 := Indeterminate(Rationals, "b_1"); b_2 := Indeterminate(Rationals, "b_2");
b_3 := Indeterminate(Rationals, "b_3"); b_4 := Indeterminate(Rationals, "b_4");
a_1 := Indeterminate(Rationals, "a_1"); c_2 := Indeterminate(Rationals, "c_2");
c_4 := Indeterminate(Rationals, "c_4"); c_6 := Indeterminate(Rationals, "c_6");
 aa_1 := H + A + B + C + D + U + V + W;
 bb_1 := H + A + B + C + D + U + V + W + p + q + P + Q;
 bb_2 := H^2 + A^2 + B^2 + C^2 + D^2 + U^2 + V^2 + W^2 + p^2 + q^2 + P^2 + Q^2;
 bb_3 := H^3 + A^3 + B^3 + C^3 + D^3 + U^3 + V^3 + W^3 + p^3 + q^3 + P^3 + Q^3;
 bb_4 := H^4 + A^4 + B^4 + C^4 + D^4 + U^4 + V^4 + W^4 + p^4 + q^4 + P^4 + Q^4;
 cc_2 := H^2 + A^2 + B^2 + C^2 + D^2 + U^2 + V^2 + W^2 + 2*p*P + 2*q*Q;
 cc_4 := H^4 + A^4 + B^4 + C^4 + D^4 + U^4 + V^4 + W^4 + 2*p^2*P^2 + 2*q^2*Q^2;
 cc_6 := H^6 + A^6 + B^6 + C^6 + D^6 + U^6 + V^6 + W^6 + 2*p^3*P^3 + 2*q^3*Q^3;

f_Oh_cube := Value(CICF_Oh_cube,
[a_1, b_1, b_2, b_3, b_4, c_2, c_4, c_6],
[aa_1, bb_1, bb_2, bb_3, bb_4, cc_2, cc_4, cc_6]);;

f_Oh_cube_A := Value(CICF_Oh_cube_A,
[a_1, b_1, b_2, b_3, b_4, c_2, c_4, c_6],
[aa_1, bb_1, bb_2, bb_3, bb_4, cc_2, cc_4, cc_6]);;

f_Oh_cube_E := Value(CICF_Oh_cube_E,
[a_1, b_1, b_2, b_3, b_4, c_2, c_4, c_6],
[aa_1, bb_1, bb_2, bb_3, bb_4, cc_2, cc_4, cc_6]);;

list_partitions :=[];
calcCoeffGenTAE := function(list_partitions)
local list_ligand_L, l_pp;
list_ligand_L := [H,A,B,C,D,U,V,W,p,P,q,Q];
l_pp := list_partitions;
Print("$", l_pp, "$␣&␣",
calcCoeffGen(f_Oh_cube, list_ligand_L, list_partitions), "␣&␣",
calcCoeffGen(f_Oh_cube_A, list_ligand_L, list_partitions), "␣&␣",
calcCoeffGen(f_Oh_cube_E, list_ligand_L, list_partitions), "␣\\\\␣\n");
end;

calcCoeffGenTAE([8,0,0,0,0,0,0,0,0,0,0,0]); calcCoeffGenTAE([7,1,0,0,0,0,0,0,0,0,0,0]);
calcCoeffGenTAE([6,2,0,0,0,0,0,0,0,0,0,0]); calcCoeffGenTAE([6,1,1,0,0,0,0,0,0,0,0,0]);
calcCoeffGenTAE([5,3,0,0,0,0,0,0,0,0,0,0]); calcCoeffGenTAE([5,2,1,0,0,0,0,0,0,0,0,0]);
calcCoeffGenTAE([5,1,1,1,0,0,0,0,0,0,0,0]); calcCoeffGenTAE([4,4,0,0,0,0,0,0,0,0,0,0]);
calcCoeffGenTAE([4,3,1,0,0,0,0,0,0,0,0,0]); calcCoeffGenTAE([4,2,2,0,0,0,0,0,0,0,0,0]);
calcCoeffGenTAE([4,2,1,1,0,0,0,0,0,0,0,0]); calcCoeffGenTAE([4,1,1,1,1,0,0,0,0,0,0,0]);
calcCoeffGenTAE([3,3,2,0,0,0,0,0,0,0,0,0]); calcCoeffGenTAE([3,3,1,1,0,0,0,0,0,0,0,0]);
calcCoeffGenTAE([3,2,2,1,0,0,0,0,0,0,0,0]); calcCoeffGenTAE([3,2,1,1,1,0,0,0,0,0,0,0]);
calcCoeffGenTAE([3,1,1,1,1,1,0,0,0,0,0,0]); calcCoeffGenTAE([2,2,2,2,0,0,0,0,0,0,0,0]);
calcCoeffGenTAE([2,2,2,1,1,0,0,0,0,0,0,0]); calcCoeffGenTAE([2,2,1,1,1,1,0,0,0,0,0,0]);
calcCoeffGenTAE([2,1,1,1,1,1,1,0,0,0,0,0]); calcCoeffGenTAE([1,1,1,1,1,1,1,1,0,0,0,0]);

LogTo();
```

References

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

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

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

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

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

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

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

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

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

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

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

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

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

[14] http://www.maplesoft.com/support/help/Maple/view.aspx?sid=2539.

[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, 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, *MATCH Commun. Math. Comput. Chem.* **76** (2016) 379–400.

[18] S. Fujita, *Symmetry and Combinatorial Enumeration in Chemistry*, Springer-Verlag, Berlin-Heidelberg, 1991.

[19] S. Fujita, Promolecules with a subsymmetry of $\boldsymbol{O}_h$. Combinatorial enumeration and stereochemical properties, *Polyhedron* **12** (1993) 95–110.

[20] S. Fujita, Stereoisograms of octahedral complexes. I. Chirality and *RS*-stereogenicity, *MATCH Commun. Math. Comput. Chem.* **71** (2014) 511–536.

[21] S. Fujita, Combinatorial enumeration of cubane derivatives as three-dimensional entities. I. Gross enumeration by the proligand method, *MATCH Commun. Math. Comput. Chem.* **67** (2012) 5–24.

[22] S. Fujita, N. Matsubara, Edge configurations on a regular octahedron. Their exhaustive enumeration and examination with respect to edge numbers and point-group symmetries, *Int. El. J. Mol. Des.* **2** (2003) 224–241.

[23] S. Fujita, Type-itemized enumeration of *RS*-stereoisomers of octahedral complexes, *Iran. J. Math. Chem.* **7** (2016) 113–153.

[24] S. Fujita, Combinatorial enumeration of cubane derivatives as three-dimensional entities. II. Gross enumeration by the markaracter method, *MATCH Commun. Math. Comput. Chem.* **67** (2012) 25–54.

[25] S. Fujita, Combinatorial enumeration of cubane derivatives as three-dimensional entities. III. Gross enumeration by the characteristic-monomial method, *MATCH Commun. Math. Comput. Chem.* **67** (2012) 649–668.