Algorithm for Advanced Canonical Coding of Planar Chemical Structures That Considers Stereochemical and Symmetric Information

Shungo Koichi,[†] Satoru Iwata,*[,‡] Takeaki Uno,[§] Hiroyuki Koshino,[||] and Hiroko Satoh*[,§]

Graduate School of Information Science and Technology, University of Tokyo, Tokyo 113-8656, Japan, Research Institute for Mathematical Sciences, Kyoto University, Kyoto 606-8502, Japan, National Institute of Informatics, Tokyo 101-8430, Japan, and RIKEN (The Institute of Physical and Chemical Research), Saitama 351-0198, Japan

Received June 14, 2006

We describe a rigorous and fast algorithm for advanced canonical coding of planar chemical structures based on the algorithm of Faulon et al. (*J. Chem. Inf. Comput. Sci.* **2004**, *44*, 427−436). Our algorithm works well even for highly symmetric structures; moreover, an advantage of our algorithm includes providing a rigorous canonical numbering of atoms with a consideration of stereochemistry and recognizing symmetric moieties. The planar structural line notation with the canonical numbering is also fit for use with stereochemical line notation. These capabilities are usable for general purposes in chemical structural coding and are particularly essential for detecting equivalent atoms in NMR studies. This algorithm was implemented on a ^{13}C NMR chemical shift prediction system CAST/CNMR. Applications of the algorithm to several organic compounds demonstrate the practical efficiency of the rigorous coding.

1. INTRODUCTION

Coding of chemical structures is a fundamental issue in the computer-processing of chemical information. Many kinds of coding algorithms have been developed and utilized in chemical software.[1−21] A style of representation mainly used in retrieving chemical structures is line notation. Line notation reduces the estimation of similarities in chemical structures to a simple comparison of compact ASCII-coded strings[1,2,3,5,7−21] and makes possible a binary search with the strings.

An essential issue to resolve for encoding chemical structures is canonicalization, namely, giving a unique line notation independent of the input order of atoms. Since a molecule with *n* atoms has *n*! different numberings of atoms, a naïve enumeration of the numberings becomes practically impossible. We need a sensible way for generating a canonical line notation. Although the Morgan algorithm[4] is the most widely used algorithm in chemical databases and software such as the Chemical Abstracts Service, it needs improvement since it causes oscillatory behavior for certain structures.[13]

As a problem on graphs, canonicalization is equivalent to finding the canonical labeling of a graph, which is closely related to the graph isomorphism problem. The graph isomorphism problem can be solved at least as fast as the canonical labeling problem, but an efficient algorithm has yet to be reported for the graph isomorphism problem. Therefore, we have no polynomial-time algorithm for the canonical labeling of general graphs. However, as molecules can be regarded as graphs of bounded valence, the canonical labeling problem for chemical structures can be solved in polynomial time with the aid of Babai and Luks' polynomial-time algorithm for graphs of bounded valence.[22] Nevertheless, coding methods actually have been developed after the theoretical result. This is because no practical implementation of the Babai and Luks' algorithm exists at present. Computer chemists have continued to develop heuristic algorithms for canonicalization. A polynomial-time algorithm for planar graphs was developed by Faulon.[23] Non-polynomial-time algorithms for all graphs corresponding to molecules were devised by Weininger et al.,[7−10] Ouyang et al.,[13] Wong et al.,[14] Faulon et al.,[15] and Plavšić et al.[24]

Since a line notation's usability depends on the system using it, it is important to design a coding method so that line notations are suitable for the intended purpose. Furthermore, various criteria may be required for retrieving chemical structures in practice. Certain practical cases such as NMR analysis require more various criteria to distinguish chemical structures. These structures, for example, include ones that are equivalent in the planar phase but nonequivalent in stereochemistry, two atoms that are designated by *R* and *S*, or *E* and *Z*, respectively, because of structural differences between structures far apart from the atoms, but are considered equivalent in a NMR spectroscopy, and partial structures that are equivalent even considering stereochemistry, which are also useful information in structural analyses with NMR data.

However, most of the established algorithms were designed to generate a simple line notation for representing a planar structure. With only these line notations, it is difficult to consider additional criteria such as stereochemistry. Although some line notations such as SMILES[10] represent stereochemistry by adding stereochemical labels concerning

* Corresponding authors. (S.I.) Tel.: +81-75-753-7236; fax: +81-75-753-7272; e-mail: iwata@kurims.kyoto-u.ac.jp. (H.S.) Tel.: +81-3-4212-2501; fax: +81-3-556-1916; e-mail: cheminfo@nii.ac.jp.
[†] University of Tokyo.
[‡] Kyoto University
[§] National Institute of Informatics
[||] RIKEN.

10.1021/ci600238j CCC: $37.00
Published on Web 07/18/2007

E/*Z* and *R*/*S* to the corresponding atoms or codes in the planar structural line notation, they do not have enough descriptive power for dealing with stereochemistry in NMR analysis.

Some of the authors developed a **ca**nonical coding of **st**ereochemistry (CAST) method, which provides a canonical stereochemical line notation based on dihedral angles,[16−18] and by using it made it possible to take into account stereochemistry for highly accurate prediction of ^{13}C NMR chemical shifts in a CAST/CNMR system.[25] The CAST notation uses the numbering of atoms in its planar notation. In the first reports of CAST, a CANOST[5] line notation was used as the basis of the planar-CAST notation;[16−18] however, it was not robust enough for dealing with the several criteria. We therefore have developed a new algorithm for the advanced canonical coding of planar structures in order to increase the versatility of structural searching. The algorithm not only generates a canonical line notation for a planar structure but also takes into account stereochemistry to provide a canonical numbering of atoms. With the use of the new canonical numbering, the CAST line notations have been improved to more rigorously represent stereochemical structures. In addition, the algorithm was extended to generate a line notation for a partial structure and to recognize symmetric moieties. Using the partial structural line notation, we can search partial structures equivalent in both planar and stereochemical phases. The algorithm was implemented on the CAST/CNMR system.[25,26]

Our new algorithm is based on the algorithm introduced by Faulon et al.[15] with several extensions and modifications to facilitate to usage of the CAST line notations. The line notation depicts the hierarchy of topological structure environments from a selected atom. Like Faulon et al.'s algorithm, our algorithm performs well even for highly symmetric structures. In this paper, we omit the parts of our algorithm that are in common with Faulon et al. to avoid duplication, and we focus on novel features: the ordering rule that is based on three-dimensional structures and recognition of several sizes of symmetric moieties in a chemical structure.

2. CANONICAL CODING METHOD

2.1. Generation of a Canonical Line Notation. As mentioned in the Introduction, the basis of our algorithm is the same as that of Faulon et al.;[15] the description will be reduced to only the differences between the two algorithms. The maximum number of bonds incident to an atom is set at four because our targets are organic compounds.

Figure 1 shows the block diagram of our algorithm to generate a canonical line notation for an input chemical structure, which is regarded as a molecular graph. The atoms are assigned partial structure codes defined in the CAST method,[16] called planar-CAST codes. Some of the planar-CAST codes are listed in Table 1, where they are ordered on the basis of the system-predefined priority.

For each atom in the molecular graph, we conduct the following processes in the same way as the algorithm of Faulon et al. An atom is designated as a *root* and the input chemical structure is converted into a rooted directed acyclic graph (DAG) starting from the atom with the aid of a breadth-first search. The breadth-first search classifies atoms on the basis of the minimum number of bonds from the root to each of the atoms. The class is called a *layer*. The first layer is defined as the highest layer. If there are two connected atoms on the same layer, a pseudo-atom is inserted between them and is located at one layer lower, as shown in Figure 2. This arrangement can be done simultaneously with the construction of a rooted DAG, without a loss of efficiency. Every pair of connected atoms in a rooted DAG now has a hierarchy; that is, one is in a layer higher than that of the other. The higher and lower atoms are called the *parent* and *child* of the other, respectively.

Each of the atoms in the rooted DAG has an invariant called *rank*, which is an integer ranging from one to the number of atoms, and a vector invariant, which is a collection of invariants. Since a rooted DAG consists of layers, the atoms are ordered layer-by-layer from the lowest to highest and then from the highest to lowest according to their vector invariants. This process is called layer-based classification (LBC).[15] The layer-by-layer ordering of atoms from the lowest to highest is called a bottom-up sort, and the one from the highest to lowest is called a top-down sort. In our algorithm, the initial rank for each atom is the predefined order of its planar-CAST code. The vector invariant of our algorithm is different from that of Faulon et al., so that its line notations are coordinated with CAST line notations.

To define the vector invariant, we consider three consecutive layers, which are a target layer, its next-higher layer, and its next-lower layer. Let the kth layer be the target layer. We denote the rank of an atom a in the $(k-1)$th layer by $\alpha(a)$, that of an atom b in the kth layer by $\beta(b)$, and that of an atom c in the $(k+1)$th layer by $\gamma(c)$.

The vector invariant $\Delta(b)$ of the atom b consists of nine elements. Each element of $\Delta(b)$ is denoted by $\Delta_j(b)$, $j = 1, 2, ..., 9$. From the assumption that every atom has at most four neighboring atoms, every atom has at most four parents and children in total. Let p_1, p_2, p_3, and p_4 be the parents and let c_1, c_2, c_3, and c_4 be the children of atom b. First, we collect the ranks of the parents, and their order is $\alpha(p_1) \le \alpha(p_2) \le \alpha(p_3) \le \alpha(p_4)$, where the relation $x \le y$ means that x precedes or equals y in order. When the number of parents is less than four, for example, when the number is two, $\alpha(p_3)$ and $\alpha(p_4)$ are substituted by infinity "∞" in Δ. Next, as with the parents, the ranks of the children are collected and ordered as $\gamma(c_1) \le \gamma(c_2) \le \gamma(c_3) \le \gamma(c_4)$. The substitution of infinity is also applied to the children. The vector invariant $\Delta(b)$ of b is given by

$$\Delta(b) = [\alpha(p_1), \beta(b), \gamma(c_1), \gamma(c_2), \gamma(c_3), \gamma(c_4), \alpha(p_2), \alpha(p_3), \alpha(p_4)]$$

By repeatedly applying the LBC process to a rooted DAG with the vector invariants, we order its atoms. In the LBC process, for each layer, (1) the vector invariants of the atoms in the layer are updated, (2) the vector invariants are lexicographically ordered, and (3) the rank of the atoms is renewed according to the order of their vector invariants. In the first bottom-up sort, we disregard the ranks of the parents; that is, we put infinity "∞" in $\Delta_j(a)$, $j = 1, 7, 8$, and 9, for all atoms. We call a set of atoms with the same rank a *tie class*. The LBC process iterates until the number of tie classes remains unchanged. Note that the number of tie classes does not decrease because of $\Delta_2(b) = \beta(b)$.

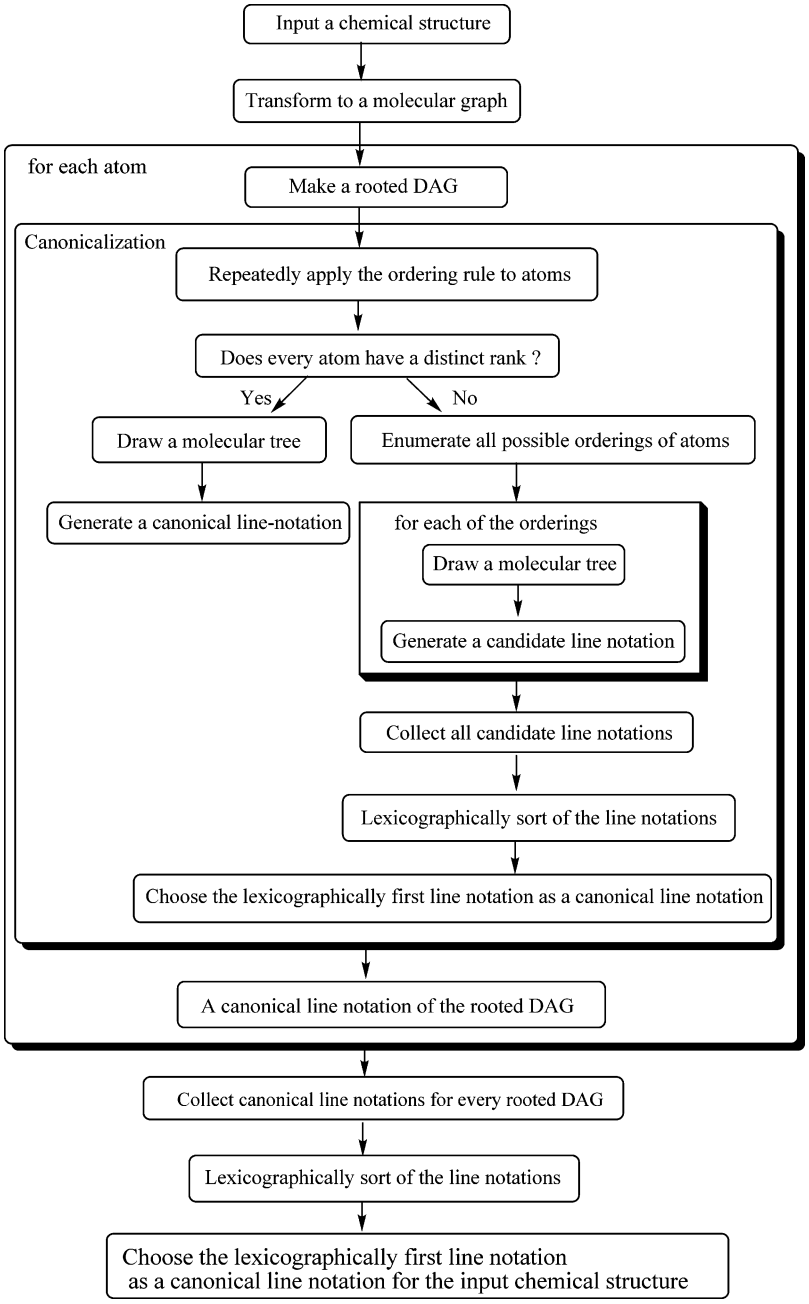


Figure 1. Block diagram of our algorithm to generate a canonical line notation for a chemical structure.

Table 1. Part of the Planar-CAST Codes[16]

no.	code	substructure	no.	code	substructure
1 [a)]	&nc	code for ring closing	5	C3	
2	C		6	Q	
3	C1		7	Q1	
4	C2		8	H	

[a] &nc is an identification number of the closing site.

The LBC process is demonstrated with hexopyranose (**1**) in Figure 3. The molecular graph with planar-CAST codes is shown in Figure 3i, and the rooted DAG starting from the carbon designated by the arrow is in Figure 3ii. Although **1** is an asymmetric structure, we use it because it is convenient for the description and comprehension of the LBC process. Figure 3iii illustrates the ordering of the atoms in the third layer after the first execution of a bottom-up sort, and Figure 3iv shows the ordering of the atoms in the third layer after the following top-down sort.

If the ordering is successfully completed, we obtain an order in which each atom has an unambiguous rank. In this case, the rooted DAG is transformed into a molecular tree depending on the order, which differs from the signature tree of Faulon et al. in the treatment of ring closure sites. The transformation of a rooted DAG into a molecular tree is described in section 2.3. The molecular tree is codified in a line-styled character string, which is the canonical line notation of the rooted DAG.

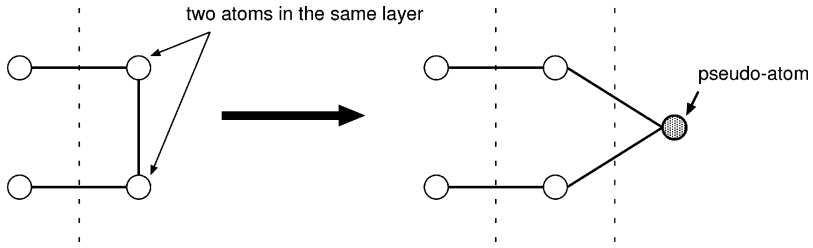


Figure 2. A pseudo-atom is inserted between two connected atoms in the same layer and is located at one layer lower.

However, in some cases, the ordering is not completed; that is, there can remain a tie class of at least two atoms. Arbitrary ordering for the atoms may induce different line notations. Therefore, as Faulon et al. enumerate all possible colors, we enumerate all possible orderings which preserve in an appropriate manner the order decided already and in which each atom has an individual rank. The details of this enumeration are described in section 2.2. For each of the orderings, a rooted DAG is transformed into a molecular tree, from which a line notation, called a *candidate*, is generated. Hence, in these cases, we potentially obtain a number of candidates.

From the candidates, the lexicographically first one is selected as the canonical line notation for the rooted DAG without ambiguity. The line notation represents the structural environment in a hierarchical order starting from the root.

After the above processes are applied to each atom in the input chemical structure, we obtain the canonical line notations starting from each atom. Finally, the lexicographically first line notation in the canonical line notations is chosen to be the canonical line notation for the input chemical structure.

2.2. Enumeration of All Possible Line Notations. As mentioned in section 2.1, after the LBC iteration, tie classes having more than one atom can remain. We enumerate the orderings for the atoms in such a tie class and generate a line notation for each of them because in some cases different line notations can be generated from a rooted DAG that includes such a tie class. If each atom has an individual rank after the LBC iteration, we skip this enumeration process and go to the process described in section 2.3.

Before we describe the enumeration process, we describe a case in which arbitrary orderings induce different line notations from a rooted DAG. The rooted DAG shown in Figure 4I is constructed from **2**, where the arrow indicates the root. The hydrogen atoms are omitted for simplicity. A pseudo-atom is represented by "•", and some carbon atoms are labeled with **a**, **b** **c**, **d**, **e**, and **f**, which are referred to below. The tie class containing **a** and **b** remains in the second layer after the LBC iteration. The line notations to be generated depend on the ranks of **a** and **b**. Some of the line notations are shown in Figure 4i and ii, where the codes with labels **a**, **b**, **c**, **d**, **e**, and **f** denote the corresponding carbon atoms in I. According to the process described in section 2.3, line notation i is generated when **a** is assigned a higher order than **b**, and line notation ii is generated when **b** is assigned a higher order than **a**. In the two line notations, there are distinctions of the numbers with "&" codes, denoting ring closure sites, as shown with frames in i and ii. On the other hand, we obtain the same line notations even if we exchange the order of **c** and **d** or **e** and **f**. Note that this is confirmed only after we generate line notations from these different orderings. Therefore, to confirm whether distinct line notations can be generated or not, we must enumerate all assignments of the ranks. The enumeration is necessary to make an exhaustive structure search.

If we naively enumerate all assignments, the number will exponentially increase. Therefore, we carefully enumerate all necessary assignments as follows. First, we choose the tie class of more than one atom whose ranks are the minimum in such tie classes and which are in the highest layer of such tie classes. Second, we select one atom in the chosen tie class and adjust the rank of the atom so that the atom has the highest rank in the tie class. Third, we again iterate the LBC process until the number of tie classes remains unchanged. We repeat these three steps until there is no tie class of more than one atom, at which point we obtain one assignment of the ranks that each atom has an individual rank. Finally, we generate a line notation, which is a *candidate*, by using the process described in section 2.3. The candidates are not necessarily distinct from each other as character strings but are distinct as the order of atoms. To obtain all the necessary assignments, we check all possible choices for the atoms to be selected in the second step.

We illustrate this enumeration process with cubane (**3**; Figure 5), which possesses highly symmetric structures. Starting from one of the carbon atoms, we obtain the rooted DAG shown in Figure 5i, where the hydrogen atoms are omitted for simplification, and the ranks of atoms are denoted in the parentheses beside the planar-CAST codes of the carbon atoms. After the first LBC iteration, two tie classes having three carbon atoms are obtained. Hence, we would have $3! \times 3! = 36$ assignments of the ranks of the atoms if we were to naively enumerate them, but our algorithm reduces the number of assignments to six as follows: (phase I) We choose one of the carbon atoms of the tie class in the second layer and assign a higher order to the atom than the other two carbon atoms. And then, the LBC iteration is performed, again, and new tie classes are obtained, as shown in Figure 5ii. In the second layer, there still remains a tie class having two carbon atoms. (phase II) We choose one of the carbon atoms of the tie class in the second layer, assign a higher order to the atom than the other, and then iterate LBC, again. As a result, we obtain an assignment of the ranks such that each atom has an individual rank, as shown in Figure 5iii. (phase III) To obtain other possible assignments, we choose a carbon other than the atom chosen in phase II as the atom which we assign a higher order. As a result of iterating LBC, we obtain another assignment. Then, the enumeration process goes back to phase I. Another atom is chosen, and the process goes to phase II.

Recall that we have three choices of carbon atoms in phase I, and we must try all three choices to enumerate all the

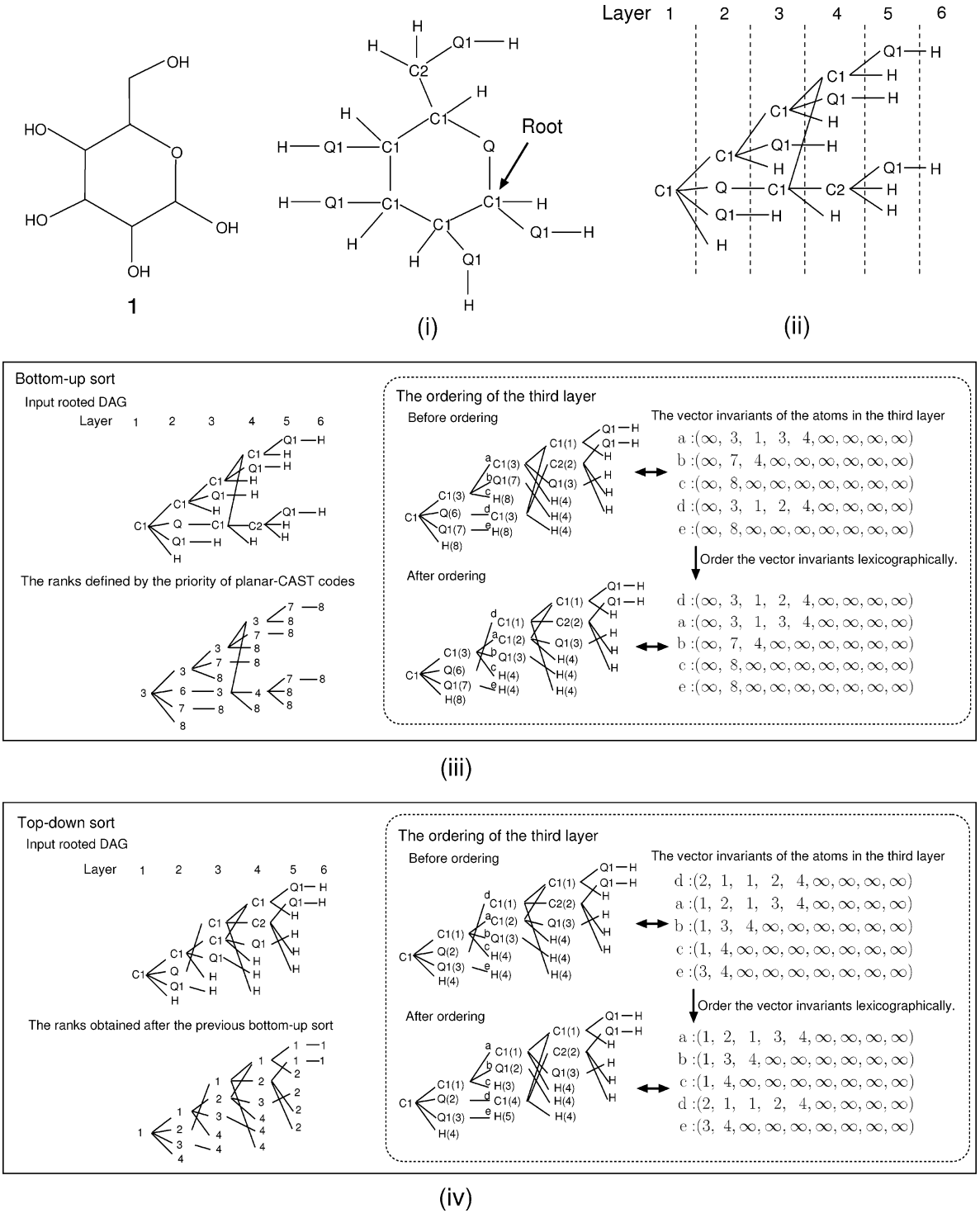


Figure 3. The structure of hexopyranose (**1**), (i) its molecular graph with planar-CAST codes, and (ii) its rooted DAG starting from the anomeric carbon atom. (iii) The ordering of the atoms in the third layer after the first execution of a bottom-up sort. The input initial rooted DAG and the ranks of the atoms are drawn on the left-hand side. For illustration purposes, the atoms in the third layer are labeled with **a**, **b**, **c**, **d**, and **e**. The numbers in parentheses beside the atoms in the second, third, and fourth layers are their ranks. The atoms **a**, **b**, **c**, **d**, and **e** are sorted on the basis of the lexicographical order of their vector invariants and ordered as $\beta(\mathbf{d}) < \beta(\mathbf{a}) < \beta(\mathbf{b}) < \beta(\mathbf{c}) = \beta(\mathbf{e})$. (iv) The ordering of the atoms in the third layer at the top-down sort following the bottom-up sort in iii. The ranks of the atoms **a**, **b**, **c**, **d**, and **e** are determined as $\beta(\mathbf{a}) < \beta(\mathbf{b}) < \beta(\mathbf{c}) < \beta(\mathbf{d}) < \beta(\mathbf{e})$ according to the lexicographical orders of their vector invariants.

necessary assignments. Altogether, there are three choices in phase I and two alternatives in phase II, so that we enumerate six assignments. Thus, the number of candidates from a carbon atom is six, and the sum of the candidates from all of the carbon atoms is $8 \times 6 = 48$. When the hydrogen atoms are taken into account, the number of the candidates from all 16 atoms is $16 \times 6 = 96$.

The enumeration is the computation-time-determining process in our algorithm, and thus we adopted an efficient and practical method to dramatically reduce the enumeration number; we describe this method together with the process for recognizing symmetric moieties in section 3.

2.3. Transformation from a Rooted DAG to a Line Notation. From each of the enumerated assignments of the

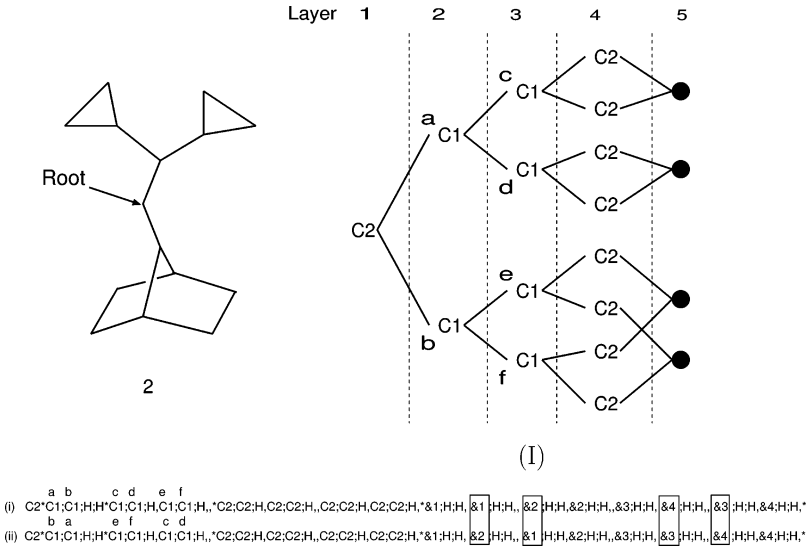


Figure 4. Example in which different assignments of ranks to atoms induce distinct line notations. A rooted DAG (I) is constructed from **2**, where the root is indicated by the arrow. C1 with **a** and C1 with **b** in I compose a tie class in the second layer. The line notations depend on the order of the two carbon atoms. According to the rules described in section 2.3, line notation i is generated when **a** is assigned a higher order than **b**, and line notation ii is generated when **b** is assigned a higher order than **a**. In the line notations, the corresponding planar-CAST codes are labeled with the same letters used in I, and the differences of the identification numbers with "&" codes between the line notations are framed.

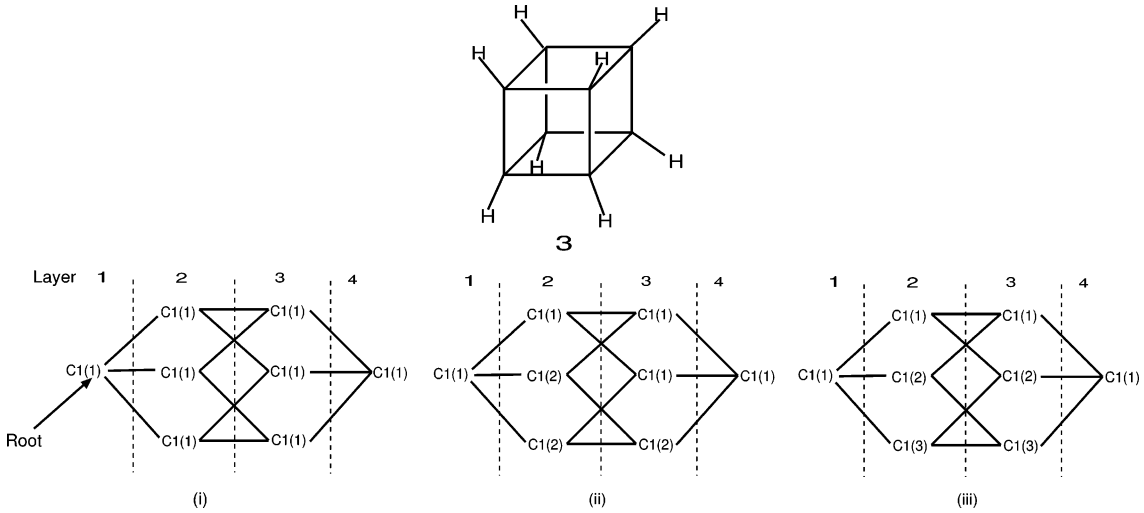


Figure 5. Example of enumerating the assignments of the ranks of the carbon atoms for cubane (**3**). Hydrogen atoms and their ranks are omitted for simplicity. (i) A rooted DAG starting from one of the carbon atoms. The ranks are denoted in the parentheses beside the planar-CAST codes. (ii) A rooted DAG obtained by performing the LBC iteration on i after assigning a higher order to one of the carbon atoms in the second layer. (iii) The final rooted DAG, in which every layer has discretely ordered atoms.

ranks, we generate a line notation in the same manner as that of the CAST-coding method.[16]

The first step is the transformation from a rooted DAG into a molecular tree. Here, the ring closure site is chosen in accordance with ring closure types 1 and 2, which correspond to even- and odd-membered rings, respectively.

Type 1: See Figure 6. For every non-pseudo-atom having more than one parent, we cut all of the bonds connecting to its parents except for the highest ranked one, and all of the cut sites are coded with a new end-point code represented by "&*nc*", where *nc* is its identification number.

Type 2: Each of the pseudo-atoms denoted by "●" in Figure 6 is divided into two end points, both of which are also coded with "&*nc*".

Two end points coded with the same identification number *nc* indicate that there is a bond between them in an original molecular structure; namely, a ring is closed at this point. The identification numbers of the pairs of end points to share the same number are sequentially assigned from the pair which is in the highest layer and has the highest ranked parent in those pairs. Finally, each end point having the "&*nc*" code is treated as the highest ranked child of the children. When there are more than one "&*nc*" codes having the same parent, they are arranged in ascending order of *nc*.

We generate a line notation by arranging the planar-CAST codes in the order of the atoms and the end points in each layer from the highest to lowest layer. The arranged codes are, in addition, divided by a semicolon (;), a comma (,), and an asterisk (*), which are code, group, and layer separators, respectively, where the group consists of atoms that connect to the same parent, and the layer is the same concept as the level used in CAST coding.[16−18]

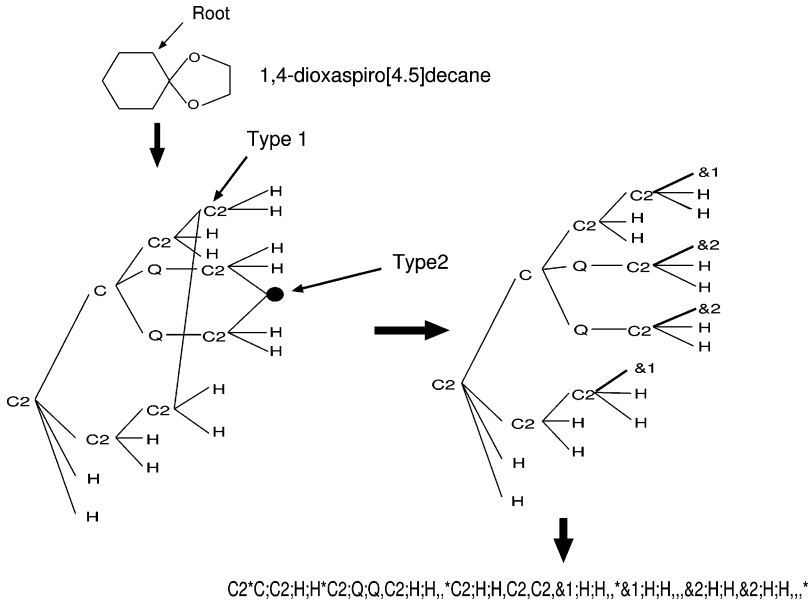


Figure 6. Transformation from a rooted DAG into a line notation via a molecular tree. The order of atoms in the rooted DAG or molecular tree is represented with a top-to-bottom arrangement in each layer.

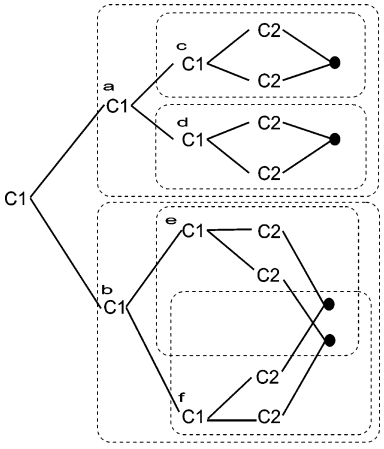


Figure 7. A rooted DAG and some of its rooted-DA subgraphs. The rooted-DA subgraphs starting from **a**, **b**, **c**, **d**, **e**, and **f** are denoted by dotted-line frames.

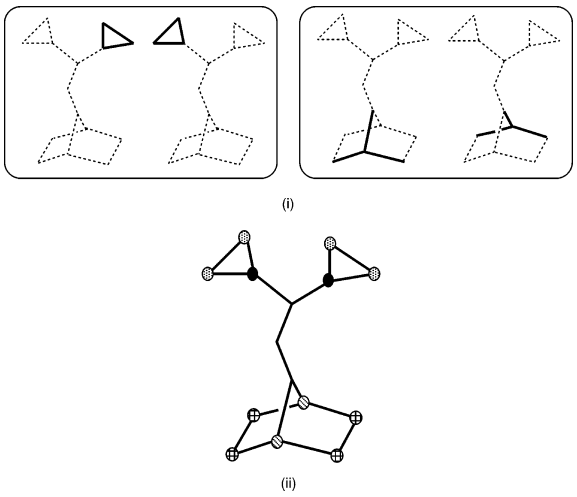


Figure 8. Two sets of recognized symmetric moieties and equivalent atoms for the planar structure of **2**. (i) Recognized symmetric moieties, drawn in bold. (ii) Groups of equivalent carbon atoms. The groups are indicated by texture.

3. RECOGNITION OF SYMMETRIC MOIETIES

3.1. Recognition of Equivalent Atoms. Equivalent atoms in the planar structure of a molecule can be recognized by comparing canonical line notations for each atom in the molecule. Concretely, two atoms are recognized as planar structurally equivalent atoms if the two roots have the same canonical line notation.

3.2. Recognition of Symmetric Moieties. Our algorithm recognizes some types of symmetric moieties in a chemical structure. Symmetric moieties are recognized as rooted-DA subgraphs in a rooted DAG. The rooted-DA subgraph starting from an atom, called the *subroot*, is defined to be composed of all atoms and bonds reachable from the subroot by tracing bonds from parents to children. Each atom in a rooted DAG induces the rooted-DA subgraph starting from the atom. The rooted DAG shown in Figure 7 is the same as the one in Figure 4I. Some of its rooted-DA subgraphs are shown in Figure 7, where they are enclosed by dotted-line frames, and their subroots are **a**, **b**, **c**, **d**, **e**, and **f**.

A rooted-DA subgraph is said to be a branch rooted-DA subgraph if it does not contain any ring structures and is not included in any ring structures. For example, an alkyl group may alter to a branch rooted-DA subgraph. A ring structure as a set of edges is called independent if it has no intersection with other rings. A rooted-DA subgraph is said to be a pendant rooted-DA subgraph if it contains only branch rooted-DA subgraphs or independent rings. In Figure 7, the rooted-DA subgraph starting from **c** contains an independent ring, and the rooted-DA subgraph starting from **a** is a pendant rooted-DA subgraph.

Before we describe how to find symmetric moieties, we explain an important property of our algorithm in association with pendant rooted-DA subgraphs. If an input molecule has no ring structure, the rooted DAGs that we obtain from the molecule are rooted directed trees. Then, as with the algorithm of Faulon et al., our algorithm canonicalizes the trees as well as Hopfcroft and Tarjan's algorithm.[27] As an extension of this property, it can be confirmed that the pendant rooted-DA subgraphs starting from each of the atoms in a tie class of at least two atoms are exactly the same, that

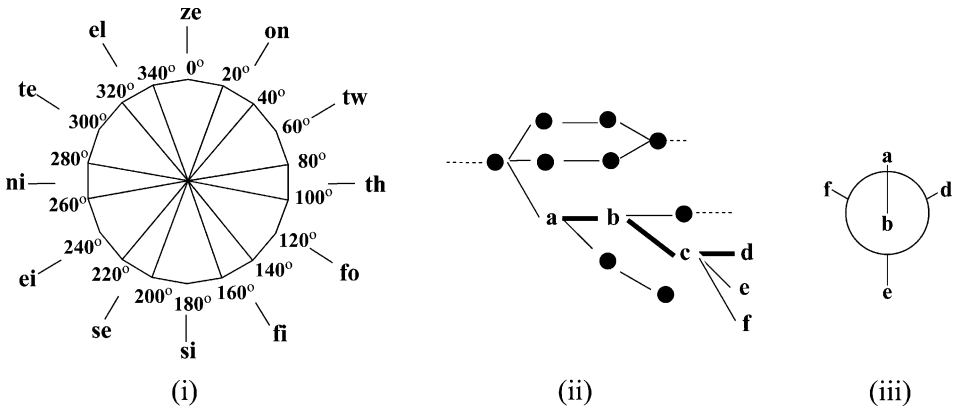


Figure 9. (i) Twelve types of CAST codes **ze**, **on**, **tw**, **th**, **fo**, **fi**, **si**, **se**, **ei**, **ni**, **te**, and **el** are defined for 12 areas of the dihedral angle. (ii) Atoms **a**, **b**, **c**, and **d** are four consecutive atoms in a molecular tree. The CAST code for **d** is assigned according to the dihedral angle between the plane defined by **a**, **b**, and **c** and the plane defined by **b**, **c**, and **d**. (iii) An example of the configuration of atom **d**. In this case, the CAST code of **d** is **tw**.

is, isomorphic. Hence, those subroots are equivalent. In other words, for a tie class of which all atoms can be the subroots of pendant rooted-DA subgraphs, one need not enumerate all the assignments of their ranks; generating just one candidate is sufficiently exhaustive. This can remarkably reduce the computation time because, in general, a molecule has a number of such structures, and they can be easily detected by a depth-first search and additional processing. This improvement is one of the important features of our algorithm.

To return to the subject in this section, we find symmetric moieties in a chemical structure by using the set L of orderings that induce the canonical line notation and the tie classes obtained after the first LBC iteration. Note that the canonical line notation is unique, but there might be some orderings that induce the canonical line notation.

First, atoms are grouped as follows. If there are two orderings L_1 and L_2 in L, and an atom u in L_1 has the same rank as an atom v in L_2, the atoms u and v are assembled in a group. In fact, groups consisting of more than one atom are classes of equivalent atoms. For example, in the case of the rooted DAG in Figure 4 (or 7), atom **a** is higher than **b** in any orderings which induce the canonical line notation for the rooted DAG, so **a** and **b** are not grouped, whereas the atoms **e** and **f** are grouped.

Next, in the tie classes obtained after the first LBC iteration, a tie class of which all atoms can be the subroots of pendant rooted-DA subgraphs is taken to be a group. As described, such a tie class consists of equivalent atoms and the orderings of their ranks are not enumerated. Therefore, we need this grouping. In the case of the rooted DAG in Figure 4, since atoms **c** and **d** induce pendant rooted-DA subgraphs, the orderings for the tie class of **c** and **d** is not enumerated. We can see that atoms **c** and **d** are equivalent. The tie class of **c** and **d** is chosen as a group.

Symmetric moieties are recognized as follows. The order of those groups is defined to be the order of their own atoms. To make moieties as large as possible, we choose one from the first group in the order of groups and compose rooted-DA subgraphs starting from each atom in the group. We define the substructures corresponding to those rooted-DA subgraphs as symmetric moieties. If a group that is not contained in the rooted-DA subgraphs still remains, we construct rooted-DA subgraphs as symmetric moieties from the group.

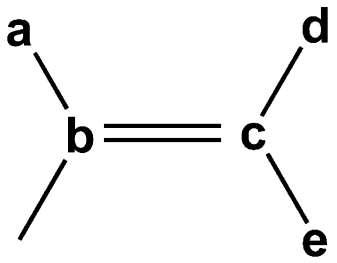


Figure 10. Example of the configuration of atoms **d** and **e**. Atoms **d** and **e** have the same parent **c** and the same rank. The CAST codes of **d** and **e** are determined by atoms **a**, **b**, and **c** where **a**, **b**, and **c** belong to layers three, two, and one higher than the layer containing **d** and **e**, respectively. If the bond between **b** and **c** is a double or aromatic bond, **d** or **e**, that is in cis for **a** has a higher order. In this case, **d** is in cis for **a**; hence, **d** has a higher order.

For example, with reference to **2** in Figure 4, our algorithm recognizes symmetric moieties starting from **c** and **d**, which are shown on the left-hand side of Figure 8i, and symmetric moieties starting from **e** and **f**, which are shown on the right-hand side of Figure 8i, where the moieties are drawn as bold lines. They are symmetric, and corresponding atoms are equivalent in the planar phase.

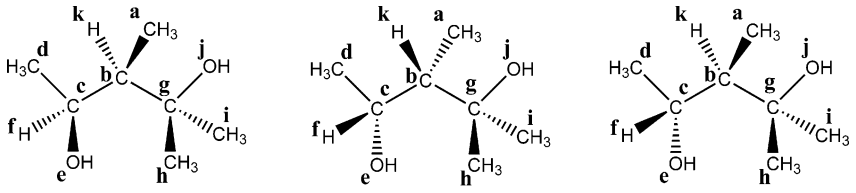
Our algorithm also recognizes equivalent atoms in **2**. There are 14 groups of equivalent atoms, of which five groups consist of one atom. Figure 8ii shows the groups consisting of more than one carbon.

4. ORDERING RULES BASED ON STEREOCHEMISTRY

One of the most important features of our algorithm is the ordering rule considering stereochemistry. This ordering rule is applied to pendant rooted-DA subgraphs. Before we describe the ordering rule, we introduce a canonical line notation for a partial structure.

4.1. Canonical Line Notation for a Partial Structure. For a rooted DAG, by neglecting the layers below a designated level, we obtain a restricted rooted DAG from the root to the specified layer. By applying all of the processes described above, such as the LBC iteration and transformation to line notations, to the restricted rooted DAG, we get the canonical line notation for the restricted rooted DAG as a partial structure.

A rooted DAG must be restricted before the LBC especially in retrieving partial structures, because the order



3a **3b** **3c**

Figure 11. Molecules **3a**, **3b**, and **3c** have the same planar structures; **3a** and **3b** are enantiomers, and **3c** is a diastereomer for **3a** and **3b**.

of atoms is influenced by the differences far apart from the root even though the atoms are in the same partial structures.

4.2. Ordering Rule Based on Dihedral Angles. *1. The CAST Method.* We briefly describe the coding process of CAST, which generates line notations of a three-dimensional structure based on dihedral angles represented with 12 types of CAST codes shown in Figure 9i.[16,18]

Let atom **d** be in or below the fourth layer of a molecular tree. The CAST code of atom **d** is determined by the dihedral angle made by four uniquely selected atoms **a**, **b**, **c,** and **d** in a molecular tree where **c** is the parent of **d**, **b** is the parent of **c**, and **a** is the parent of **b**; see Figure 9ii. The dihedral angle is the angle between the plane defined by **a**, **b**, and **c** and the plane defined by **b**, **c**, and **d**. The treatment for the atoms above the fourth layer and the case that **a**, **b**, and **c**, or **b**, **c**, and **d**, do not determine a plane are described in the papers of Satoh et al.[16,17,18] For example, if the Newman projection for **a**, **b**, **c**, and **d** is represented as shown in Figure 9iii, the CAST code of **d** is **tw**.

All atoms are assigned the CAST codes and are arranged in the same order as that in its planar line notation to generate a stereochemical CAST line notation; namely, the order of the CAST codes depends on the canonical numbering in the planar phase.[16−18]

4.2.2. Ranks Based on the CAST Codes. Before transforming a rooted DAG into a molecular tree, each of the atoms has an individual rank except for the atoms in pendant rooted-DA subgraphs. We determine the ranks of the atoms in pendant rooted-DA subgraphs from the highest layer using CAST codes.

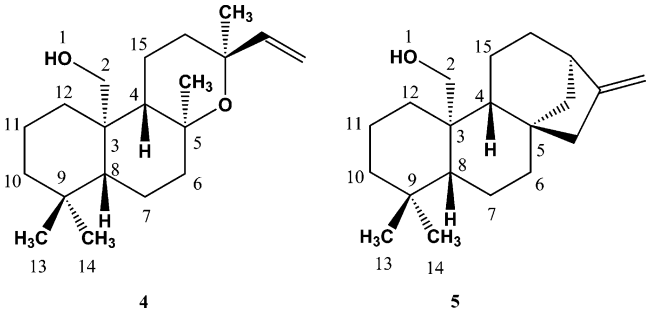
Case 1: In the case shown in Figure 9ii, if two children of **c**, say **d** and **e**, have the same rank, and the rest of the children **f** have a rank distinct from **d** and **e**, we determine the order of **d** and **e** as follows. The structure is oriented so that atom **b** is located at the closer position to the observer, as shown in the Newman projection of Figure 9iii, and the order of **d** and **e** is determined according to the anticlockwise order of **d** and **e** by starting from **f**. The order of **d** and **e** is determined independent of the priority of **f**, and **f** is ordered as the highest among the three children if **f** is the highest and is ordered as the lowest otherwise. In Figure 9iii, the atom **e** comes before **d** in the anticlockwise order from **f**; thus, the atom **e** has a higher order than **d**.

Case 2: In the case shown in Figure 9ii, if atoms **d**, **e**, and **f** have the same rank, we arbitrarily choose an atom, say **f**, and as with case 1, determine the order of **d** and **e** according to the anticlockwise order of **d** and **e** from **f**. The atom **f** is treated as the highest-ordered atom among the three children.

Case 3: Suppose that atoms **d** and **e** have the same parent and the same rank. According to the manner described in section 4.2, the CAST codes of **d** and **e** are determined by

```
      a  b     g c k   i  h  j  d  e  f
3a-(i) C3*C1;H;H;H*C;C1;H,,,*C3;C3;Q1,C3;Q1;H,*H;H;H,H;H;H,H,H;H;H,H,*

         g c k  g c k  g c k
3a-(ii) #,,*si;te;tw,,*si;te;tw,,*si;te;tw,,#%H;H;H*H;H;H*H;H;H%
      A*B;H;H;H*C;C;C,,,*si;tw;te,si;tw;te,*H;H;H,H;H;H,HO,H;H;H,HO,*
      a  b     g c k   i  h  j  d  e  f

      a  b     g c k   i  h  j  d  e  f
3b-(i) C3*C1;H;H;H*C;C1;H,,,*C3;C3;Q1,C3;Q1;H,*H;H;H,H;H;H,H,H;H;H,H,*

         g c k  g c k  g c k
3b-(ii) #,,*si;tw;te,,*si;tw;te,,*si;tw;te,,#%H;H;H*H;H;H*H;H;H%
      A*B;H;H;H*C;C;C,,,*si;tw;te,si;te;tw,*H;H;H,H;H;H,HO,H;H;H,HO,*
      a  b     g c k   i  h  j  d  e  f

      a  b     g c k   i  h  j  d  e  f
3c-(i) C3*C1;H;H;H*C;C1;H,,,*C3;C3;Q1,C3;Q1;H,*H;H;H,H;H;H,H,H;H;H,H,*

         g c k  g c k  g c k
3c-(ii) #,,*si;te;tw,,*si;te;tw,,*si;te;tw,,#%H;H;H*H;H;H*H;H;H%
      A*B;H;H;H*C;C;C,,,*si;tw;te,si;te;tw,*H;H;H,H;H;H,HO,H;H;H,HO,*
      a  b     g c k   i  h  j  d  e  f
```

Figure 12. Planar line notations. **3a**-(i), **3b**-(i), and **3c**-(i) are the canonical planar line notations for the molecules **3a**, **3b**, and **3c** in Figure 11, respectively. The three canonical line notations coincide with each other because **3a**, **3b**, and **3c** have the same planar structure. **3a**-(ii), **3b**-(ii), and **3c**-(ii) are configurational CAST line notations for the molecules **3a**, **3b**, and **3c** in Figure 11, respectively. The planar-CAST codes and configurational CAST codes corresponding to the labeled atoms in Figure 11 are also labeled with the same letters used in Figure 11. **3a**-(ii) is different from **3b**-(ii) in the sections underlined with bold lines because of the asymmetric carbon centers **b** and **c**. **3a**-(ii) is distinct from **3c**-(ii) in the sections underlined with broken lines because of the asymmetric carbon center **c**. **3b**-(ii) and **3c**-(ii) are distinct from each other in the sections underlined with dotted lines because of the asymmetric carbon center **b**.



4 **5**

Figure 13. Molecules **4** and **5**. These have the same partial structure around oxygen-1 within five levels.

atoms **a**, **b**, and **c** where **a**, **b**, and **c** belong to layers three, two, and one higher than the layer containing **d** and **e**, respectively. If the bond between **b** and **c** is a double or aromatic bond, then **d** or **e**, that is in cis for **a**, has a higher order. For example, atom **d** in Figure 10 is in cis for **a**; hence, **d** has a higher order than **e**.

Applying these ordering rules to the molecules **3a**, **3b**, and **3c** in Figure 11, we obtain their canonical planar line notations and configurational CAST line notations shown in Figure 12. In Figure 11, some atoms are labeled with **a**,

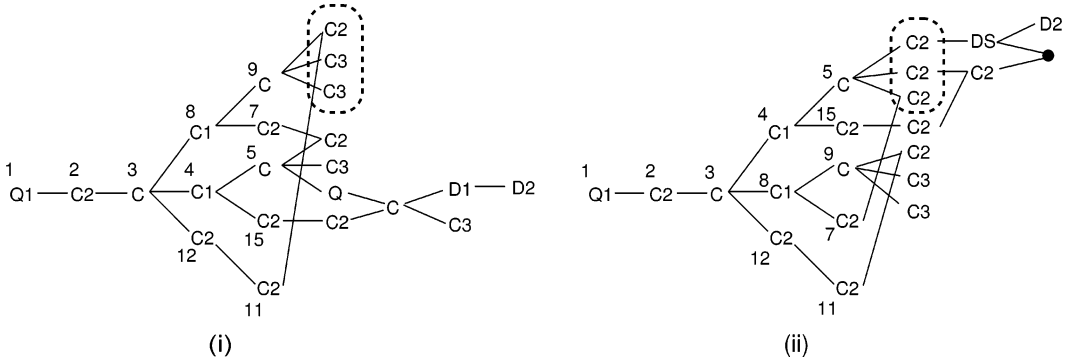
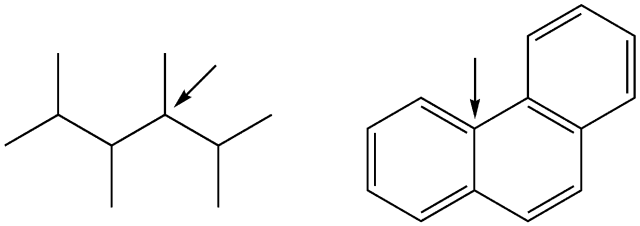


Figure 14. (i) The rooted DAG of **4**. (ii) The rooted DAG of **5**. These two rooted DAGs are the same within five levels. However, the order of carbon-4 and carbon-8 in i is different from that in ii because of the structures denoted by the dotted-line frames.



7 **8**

(i) C1*C1;C1;C1;H*C1;C1;H,&1;C1;H,&2;&3;H,*&1;C1;H,&2;&4;H,,,&3; &5;H,,,,*,&4;&5;H,,,,,,,*

(ii) DS*DS;DS;DS*DS;DS,DS;DS,DS;DS*DS;DS,&1;DS,DS;DS,&1;DS,&2;DS,&3;DS*DS; DS,&2;DS,,&4;DS,DS; DS,&3;DS,,&5;DS,,&6;DS,,&6;DS*&7;DS,&4;DS,,&7;DS,,&8;DS, &9;DS,&5;DS,,&9;DS,,&8;DS,,&10;DS,,&11 ;DS*,DS;DS,,&12;&13,,&10;DS,,&13;DS,,DS;DS,,&14;&15,,&11 ;DS,,&15;DS,,&16;DS,,&16;DS*&17;DS,&12; DS,,,,&17;&18,,&19;DS,&20;DS,&14;DS,,,,&20;&21,,&19; DS,,&18;DS,,&21;DS*,&22;DS,,&22;&23,,,,&23;DS ,,&24;DS,,&24;&25,,,,&25;&26,, &27;&28,,&28;&29*,&27;DS,,,,&26;&30,,&29;&31,,,,,,,*,&30;&31,,,,*

(iii) C1*C1;C1;C3;H*C1;C3;H,C3;C3;H,H;H;H,*C3;C3;H,H;H;H,, H;H;H,H;H;H,,,,*H;H;H,H;H;H,,,,,,,,,*

(iv) DS*DS;DS;D1*DS;D1,D1;D1,D1;H*D1;D1,D1;H,&1;H,D1;H, &2;H,*D1;H,&1;H,&3;H,,,,&2;H,,,*&3;H,,,,,,,*

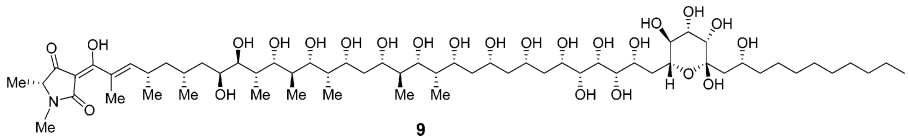
Figure 15. The canonical planar line notations for (i) cubane (**3**), (ii) fullerene C_{60} (**6**), (iii) **7**, and (iv) **8**. Any carbon atom can be the root of the canonical line notation for cubane or fullerene. The line notation for **7** or **8** starts from the carbon atom designated by the arrow or its equivalent atom.

b, **c**, **d**, **e**, **f**, **g**, **h**, **i**, **j**, and **k**. In Figure 12, the planar-CAST codes and configurational CAST codes corresponding to the labeled atoms are also labeled with the same letters used in Figure 11. The canonical line notations **3a**-(i), **3b**-(i), and **3c**-(i) for **3a**, **3b**, and **3c**, respectively, coincide with each other because they have the same planar structure. The configurational CAST line notation **3a**-(ii) for **3a** is different from that of **3b**-(ii) for **3b** in the sections underlined with bold lines because of the asymmetric carbon centers **b** and **c**. Note that the sections correspond with the children of **b** and **c**. The CAST line notation **3a**-(ii) is distinct from the CAST line notation **3c**-(ii) for **3c** in the sections underlined with broken lines because of the asymmetric carbon center **c**. The CAST line notations **3b**-(ii) and **3c**-(ii) are distinct from each other in the sections underlined with dotted lines because of the asymmetric carbon center **b**.

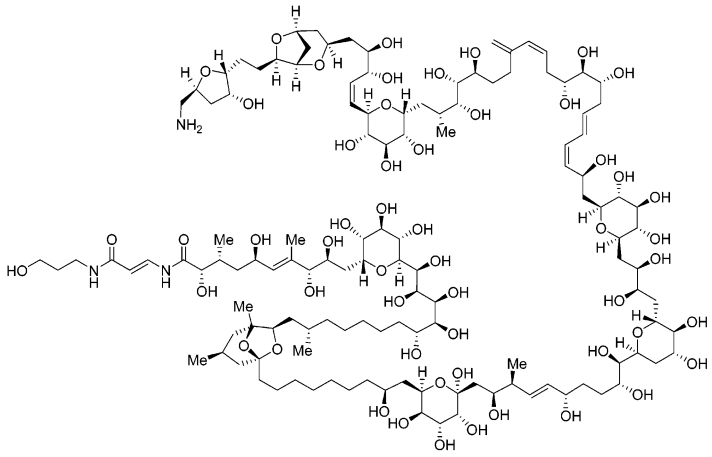
If a carbon atom is an asymmetric center, its children have different vector invariants from each other and thus have distinct ranks; namely, no matter what configuration the children have, the order depends on only the planar structures. Hence, the order of atoms **c**, **g**, and **k** is **g**, **c**, and **k**, and by assigning the configurational CAST codes on the basis of the configuration to the atoms in the order, we obtain a configurational CAST line notation, si;te;tw or si;tw;te.

On the other hand, if a carbon atom is not an asymmetric center, the same canonical line notation for a planer structure may be generated even if the order of its children is different. However, the configurational CAST line notation depends on the order. For example, the configurational CAST line notation of **h**, **i**, and **j** is si;tw;te in Figure 12 because their order in the planar phase is **i**, **h**, and **j** on the basis of the input order of the atoms, but if the order is **h**, **i**, and **j**, the configurational line notation is si;te;tw. It is necessary to avoid this ambiguity because the atom **g** is not an asymmetric carbon center. By considering stereochemistry in the generation of a canonical line notation in the planar phase, we could eliminate the ambiguity.

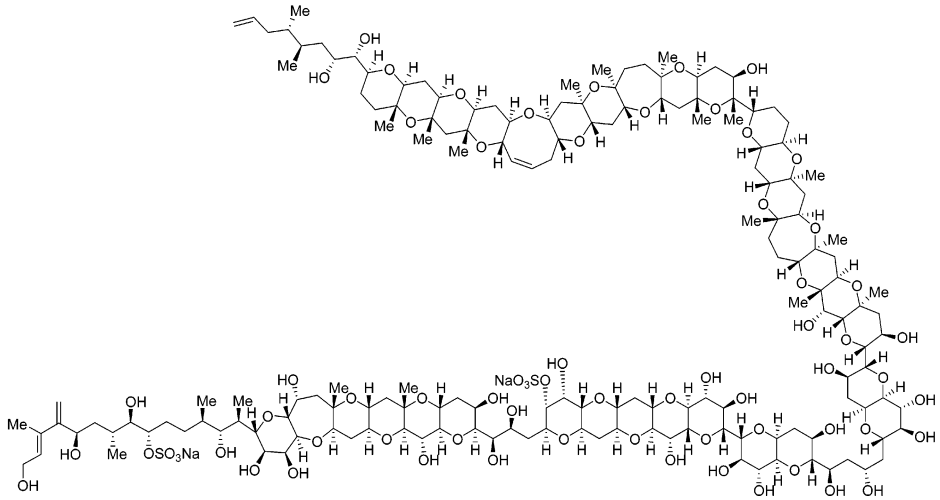
In addition, combined with these ordering rules, level restrictions are sometimes important for comparing canonical line notations of partial structure, such as for a partial structure search for NMR chemical shift prediction in the CAST/CNMR system. For example, compounds **4** and **5** have the same partial structure around oxygen-1 within five levels (Figure 13). If we construct the rooted DAGs of the whole structures from the oxygen atoms and apply the LBC iteration to the rooted DAGs, we obtain the rooted DAGs shown in Figure 14, where the hydrogen atoms are omitted for simplicity and the order of atoms is represented by their



9



10



11

(i) DS*DS;DS;Q1*D1;C3,V;V,H*C1;H,H;H;H,C1;QD,QD;N,*C2;C3;H,,,,,&1;C3; H,,,&1;C3*C1;H;H,H;H; H,,,H;H;H,,,H;H;H*C2;C3;H,,,,,,,,,,*C1;H;H,H; H;H,*C1;Q1;H,,,,,*C1;Q1;H,H,*C1;C3;H,H,,*C1;Q1;H, H;H;H,,*C1;C3;H,H,,,,*C1;Q1;H,H;H;H,,*C1;C3;H, H,,,,*C2;Q1;H,H;H;H,,*C1;H;H,H,,,,*C1;Q1;H,,,* C1;C3;H,H,*C1;Q1;H,H;H;H,,*C1;C3;H,H,,,,*C2;Q1;H,H;H;H,,*C1;H;H, H,,,,*C2;Q1;H,,,*C1;H;H,H,* C2;Q1;H,,,*C1;H;H,H,*C1;Q1;H,,,*C1;Q1; H,H,*C1;Q1;H,H,,*C1;Q1;H,H,,*C2;Q1;H,H,,*C1;H;H,H,,* C1;Q;H,,,*C1;Q1;H,C,*C1;Q1;H,H,,&2;C2;Q1*&2;Q1; H,H,,,,C1;H;H,H*,H,,,C2;Q1;H,,,*,C2;H;H,H,*C 2;H;H,,,*C2;H;H,,*C2;H;H,,*C2;H;H,,*C2;H;H,,*C2;H;H,,*C2;H;H,,*C3;H; H,,*H;H;H,,*

(ii) DS*D1;D2;C2*D1;H,H;H,C2;H;H*C2;H,,,,C1;H;H,,*C1;H;H,,C1;Q1;H,,*C1;Q1; H,,,C1;Q1;H,H,*C1;Q1 ;H,H,,C1;Q1;H,H,,*C2;Q1;H,H,,,C2;C3;H,H,,*D1;H;H,H,,,C1;H;H, H;H;H,,*D1;H,,,,C1;Q;H,,,,,*D1;H,, C1;Q1;H,C1,*D1;H,,C1;Q1;H,H,,&1;D1;H*C1;H,,&1;Q1;H,H,,,,D1;H,* C2;Q1;H,,,H,,,C1;H,*C1;H;H,H ,,,C1;Q1;H,*C1;Q;H,,,,C2;Q1;H,H,*C1;Q1;H,C1,,C1;H;H,H,,*C1;Q1;H, H,,&2;C2;H,C2;Q;H,,,*&2;Q1; H,H,,,,C1;H;H,,C1;H;H,C1,*,H,,,C1;Q1;H,,,C2;Q;H,,,&3;C1;H*,C2;Q1; H,H,,&3;H;H,&4,,,&4;C2;H,*C1 ;H;H,H,,,,,,,,C2;H;H,*C1;Q;H,,,,C1;H;H,,*C1;Q1;H,C1,,C1;Q;H,,*C2;Q1 ;H,H,,&5;C1;H,C2;Q1;H,C1,*& 5;H;H,H,,,,C1;Q1;H,,&6;H;H,H,,&6;C2;H*,,,,C2;Q1;H,H,,,,,,,N2;H;H, *C2;H;H,H,,,H;H,,*C1;H;H,,,,,*D1 ;Q1;H,,*D1;H,H,*C1;H,,*C1;C3;H,*C2;Q1;H,H;H;H,*C;H;H,H,,,,*C1;Q;Q1 ,,,*C1;Q1;H,C1,H*C1;Q1;H, H,,&7;C2;H,*&7;Q1;H,H,,,,C1;H;H,*,H,,,C2;Q1;H,,*,C2;H;H,H,*C2;H;H ,,,*C2;H;H,,*C2;H;H,,*C2;H;H, ,*C2;H;H,,*C;H;H,,*C2;Q;Q,,*C1;H;H,C,C1*C2;C3;H,,,&8;&9;C3,&9;C2; H*&8;H;H,H;H;H,,,,H;H;H,,C 1;H;H,*,,,,,,,,C2;C3;H,,*C2;H;H,H;H;H,*C2;H;H,,,,,*C2;H;H,,*C2;H;H,, *C1;H;H,,*C1;Q1;H,,*C1;Q1;H, H,*C1;Q1;H,H,,*C1;Q1;H,H,,*C1;Q1;H,H,,*C1;Q;H,H,,*C1;Q1;H,C1,,*C1;Q1; H,H,,&10;C2;H*&10;Q1; H,H,,,,C1;H;H,*,H,,,C1;Q1;H,,*,DS;Q1;H,H,*D1;C3,H,,*C1;H,H;H;H, *C2;Q1;H,,,,*C1;H;H,H,*C1;C3;H ,,,*V;Q1;H,H;H;H,*QD;N1,H,,,,*,D1;H,*D1;H,*V;H,*QD;N1,*,C2;H*C2;H;H, *C2;H;H,,*Q1;H;H,,*H,,*

(iii) DS*DS;D1;C3*D2;C1,C2;H,H;H;H*H;H,C2;Q1;H,Q1;H;H,,,,*,,C1;H;H,H,,H,,*C1; C3;H,,,,*C1;Q1;H,H;
H;H,*C2;Q;H,H,,,,*C2;H;H,SW,,*C1;H;H,,,QD;QD;Q*C1;C3;H,,,,,NA*C1;Q1;H, H;H;H,,*C1;C3;H,H,,
,,*C1;Q;H,H;H;H,,*C1;Q1;H,C1,,,,*C1;Q1;H,H,,&1;C1;H*&1;Q;H,H,,,,C2; Q1;H,*,C1,,,C;H;H,H,*&2;C
2;H,&2;C3;Q,,,*,C1;H;H,,,H;H;H,C1*&3;Q;H,,,,,,&3;C2;H*,C1,,,C;H;H,* &4;C1;H,&4;C3;Q,,*,C1;Q1;H
,,,H;H;H,C1*&5;Q;H,H,,,,,&5;C2;H*,C1,,,,C1;H;H,*&6;C1;H,&6;Q1;H,,*, C1;Q1;H,,,H,*C2;Q1;H,H,,*C1
;H;H,H,,*C1;Q;H,,,*C1;Q;H,C1,*C1;Q1;H,SW,,&7;C2;H*&7;Q;H,H,,QD;QD;Q,,C1; H;H,*,C1,,,,,NA,&8
;Q;H,,*&8;C2;H,,,C1,*,C1;H;H,,&9;C1;H*&9;Q;H,,,,C1;Q1;H,*,C1,,&10;Q; H,H,*&10;C1;H,,C1,,*,C1;Q
1;H,,&11;C1;H*&11;Q1;H,H,,,C1;Q;H,*,H,,,C1;Q1;H,C1,*,C1;Q1;H,H,,&12; C2;H*&12;Q;H,H,,,,C1;H;H,
,C1,,,&13;Q1;H,,&13;C1;H,,H,*,C2;Q1;H,,*C1;H;H,H,*C2;Q1;H,,,*C1;H;H, H,*C1;Q;H,,,*C1;Q1;H,C1,
C1;Q1;H,H,,&14;C2;H&14;Q;H,H,,,,C1;H;H,*,C1,,,&15;Q1;H,,*&15;C1; H,,H,*,C1;Q;H,,*C2;Q1;H,C1,
C;H;H,H,,&16;C1;H&16;C3;Q,,,,,C;Q1;H,*,H;H;H,C1,&17;C3;Q,H,*,,, &17;C2;H,,H;H;H,C1,*,C;H;H,,,
,,&18;C2;H*&18;C3;Q,,,,C2;H;H,*,H;H;H,C1,C;H;H,,*,,,&19;C2;H, &19;C3;Q,,*,C;H;H,,,H;H;H,C1*&20
;C3;Q,,,,,,&20;C2;H*,H;H;H,C1,,C1;H;H,*,,,&21;C2;H,&21;Q;H,,*,C2;H;H,,, C1,*&22;H;H,,,&22;C;H*,,,,
C1;C3;Q,*C2;Q1;H,H;H;H,C*C1;H;H,H,,,,,&23;C2;C3*&23;Q;H,,,,,C1;H;H,H;H;H*,C,, &24;Q;H,,,,,*&24;
C2;C3,,C1,*,C2;H;H,H;H;H,C;C2;H*&25;H;H,,,,,&25;C3;Q,C1;H;H,*,,,H;H;H, C,&26;Q;H,,*,,,&26;C2;C
3,,C1,*,C1;H;H,H;H;H,&27;C2;H*&27;Q;H,,,,,,D1;H;H,*,C1,,D1;H,,*C1;C2;H, &28;H,*&28;Q;H,C1;H;H,
,,*,C,,&29;Q;H,,*&29;C2;C3,,C1,*,C;H;H,H;H;H,&30;C2;H*&30;C3;Q,,,,,,, C1;H;H,*,H;H;H,C,&31;Q;H,,
,,,&31;C2;C3,,C1,,C2;H,H,H;H,&32;C1;H*&32;H;H,,,,,,C1;Q1;H,*,,,C2; Q1;H,H,*C1;H;H,H,,*C1;C3
;H,,,*C2;C3;H,H;H;H,*D1;H;H,H;H;H,,,,*D2;H,,,,,*H;H,*

Figure 16. The canonical planar line notations for (i) aflastatine A (**9**), (ii) palytoxin (**10**), and (iii) maitotoxin (**11**).

top-to-bottom arrangement. It is evident that the order of carbon-4 and carbon-8 in i is different from that in ii. We can avoid this problem as follows. By restricting the layered networks under the fifth level, we obtain exactly the same rooted DAGs. (Note that, only by this restriction, the canonical line notations for the partial structures become the same since the restricted rooted DAGs have no ring.) Moreover, according to the ordering rule with the planar-CAST codes, the order of the carbon-4 and carbon-8 in i becomes the same as that in ii. Hence, not only the canonical planar line notations but also the configurational CAST line notations become the same.

5. APPLICATIONS

5.1. Canonical Coding. We applied the algorithm to several organic compounds that include highly symmetric structures or a number of atoms. The resulting canonical planar line notations for highly symmetric molecules cubane (**3**), fullerene C_{60} (**6**), **7**, and **8** are shown in i, ii, iii, and iv of Figure 15, respectively. The canonical planar line notations for aflastatin A (**9**), palytoxin (**10**), and maitotoxin (**11**), which are organic compounds having more than 200 atoms, are shown in i, ii, and iii of Figure 16, respectively. They include some planar-CAST codes that are not listed in Table 1, such as QD and D1, which mean a carbonyl oxygen atom and a carbon atom of olefin having one hydrogen atom, respectively. Detailed planar-CAST codes are available in the articles on the CAST coding method.[16]

The canonical notation for fullerene C_{60} induces canonical numbering of the 60 carbon atoms as shown in Figure 17. The numbering is also independent of any input order of the atoms.

In the application to fullerene, we obtained a tie class consisting of two carbons in the second layer after the LBC iteration. It is notable that the order of those two carbons determined the order of the other atoms; if we gave the two carbons a distinct rank, the order of the other atoms was automatically fixed by repeating the LBC process. Therefore, we obtained two candidates, which had a mirror image relationship, for each root of 60 carbon atoms and $60 \times 2 = 120$ candidates in total. After generating the canonical notation for each of the 60 carbon atoms and comparing them, our algorithm concluded that all the carbons are equivalent.

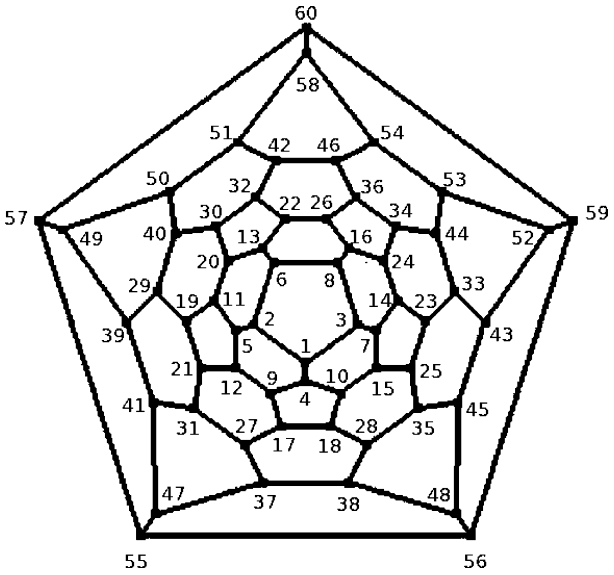


Figure 17. The canonical numbering for fullerene C_{60} (**6**). Its mirror image numbering was also obtained.

Table 2. Computation Time of Compounds Possessing Several Sizes and Structures

compound names	no. of atoms	no. of bonds	computation time(s)	no. of candidates
cubane (**3**)	16	20	0.01	96
fullerene C_{60} (**6**)	60	90	0.11	120
aflastatin A (**9**)	202	203	0.74	202
palytoxin (**10**)	409	418	4.55	409
maitotoxin (**11**)	492	523	7.53	492

5.2. Computation Time. Table 2 shows the computation times for several compounds. The computations were performed on an IBM Think Pad T42 with a 1.7 GHz Intel Pentium M CPU, with 512 MB of memory and running Vine Linux 3.2 (kernel 2.4.31). The listed computation times were measured for generating the canonical line notation for a molecule; this demonstrates the average performance of our algorithm.

Table 2 lists the number of atoms and bonds as well as the total number of candidates. The reasons for there being 96 candidates for **3** and 120 for **6**, respectively, are described in sections 2.2 and 5.1, respectively. The number of candidates for **9**, **10**, and **11** were the same as the number of atoms. This means that one candidate was obtained for every

atom in these compounds. The results show that the algorithm runs fast in practice.

The practical applicability of the algorithm was also confirmed by implementing it on a ^{13}C NMR chemical shift prediction system CAST/CNMR.[25,26] The algorithm works with a stereochemical coding CAST in searching for topologically and/or stereochemically equivalent partial structures.

6. CONCLUSION

We developed a rigorous and fast algorithm based on the algorithm of Faulon et al. for generating a canonical line notation of the planar structure of a chemical structure. Moreover, the algorithm efficiently detects symmetric moieties in the chemical structure and effectively utilizes information on stereochemistry to generate a line notation to be used together with the CAST line notation. Concrete applications to several organic compounds that include highly symmetric structures or consist of a number of atoms demonstrated the applicability of the algorithm for practical use in the computer processing of chemical structures.

ACKNOWLEDGMENT

This work was supported in part by a Grant for Joint Research from the National Institute of Informatics.

REFERENCES AND NOTES

(1) Wiswesser, W. J. *A Line-Formula Chemical Notation*; Y. Thomas Crowell: New York, 1954.
(2) Wiswesser, W. J. How the WLN Began in 1949 and How It Might Be in 1999. *J. Chem. Inf. Comput. Sci.* **1982**, *22*, 88−93.
(3) Wiswesser, W. J. Historic Development of Chemical Notations. *J. Chem. Inf. Comput. Sci.* **1985**, *25*, 258−263.
(4) Morgan, H. L. The Generation of a Unique Machine Description for Chemical Structures − A Technique Developed at Chemical Abstracts Service. *J. Chem. Doc.* **1965**, *5*, 107−113.
(5) Abe, H.; Kudo, Y.; Yamasaki, T.; Tanaka, K.; Sasaki, M.; Sasaki, S. A Convenient Notation System for Organic Structure on the Basis of Connectivity Stack. *J. Chem. Inf. Comput. Sci.* **1984**, *24*, 212−216.
(6) Bremser, W. Structure Elucidation and Artificial Intelligence. *Angew. Chem., Int. Ed. Engl.* **1988**, *27*, 247−260.
(7) Weininger, D. SMILES, a Chemical Language and Information System. 1. Introduction to Methodology and Encoding Rules. *J. Chem. Inf. Comput. Sci.* **1988**, *28*, 31−36.
(8) Weininger, D.; Weininger, A.; Weininger, J. L. SMILES2. Algorithms for Generation of Unique SMILES. *J. Chem. Inf. Comput. Sci.* **1989**, *29*, 97−101.
(9) Weininger, D. Smiles 3. Depict. Graphical Depiction of Chemical Structures. *J. Chem. Inf. Comput. Sci*. **1990**, *30*, 237−243.
(10) Weininger, D. II.3. SMILES − A Language for Molecules and Reactions. In *Handbook of Chemoinformatics*; Gasteiger, J., Ed.; Wiley-VCH GmbH & Co. KGaA: Weinheim, Germany, 2003; pp 80−102.
(11) Barnard, J. M.; Jochum, C. J.; Welford, S. M. ROSDAL: A Universal Structure/Substructure Representation for PC-host Communication. In *Chemical Structure Information: Interfaces, Communication and Standards. ACM Symposium Series No 400*; Warr, W. A., Ed.; American Chemical Society: Washington, DC, 1989; pp 76−81.
(12) Ash, S.; Cline, M. A.; Homer, R. W.; Hurst, T.; Smith, G. B. Sybyl Line Notation (SLN): A Versatile Language for Chemical Structure Representation. *J. Chem. Inf. Comput. Sci*. **1997**, *37*, 71−79.
(13) Ouyang, Z.; Yuan, S.; Brandt, J.; Zheng, C. An Effective Topological Symmetry Perception and Unique Numbering Algorithm. *J. Chem. Inf. Comput. Sci*. **1999**, *39*, 299−303.
(14) Wong, H.-W.; Li, X.; Swihart, M. T.; Broadbelt, L. J. Encoding of Polycyclic Si-Containing Molecules for Determining Species Uniqueness in Automated Mechanism Generation. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 735−742.
(15) Faulon, J.-L.; Collins, M. J.; Carr, R. D. The Signature Molecular Descriptor. 4. Canonizing Molecules Using Extended Valence Sequences. *J. Chem. Inf. Comput. Sci*. **2004**, *44*, 427−436.
(16) Satoh, H.; Koshino, H.; Funatsu, K.; Nakata, T. Novel Canonical Coding Method for Representation of Three-Dimensional Structures. *J. Chem. Inf. Comput. Sci.* **2000**, *40*, 622−630.
(17) Satoh, H.; Koshino, H.; Funatsu, K.; Nakata, T. Representation of Molecular Configurations by CAST Coding Method. *J. Chem. Inf. Comput. Sci.* **2001**, *41*, 1106−1112.
(18) Satoh, H.; Koshino, H.; Nakata, T. Extended CAST Coding Method for Exact Search of Stereochemical Structures. *J. Comput.-Aided Chem.* **2002**, *3*, *48*−*55*.
(19) InChi. http://iupac.org/inchi/ (accessed Jun 2007).
(20) MOLGEN-CID Canonicalizer. http://www.mathe2.uni-bayreuth.de/molgen4/ (accessed Jun 2007).
(21) More detailed descriptions on representation of chemical structures are found at: Engel, T. 2. Representation of Chemical Compounds. In *Chemoinformatics A Textbook*; Gasteiger, J., Engel, T., Eds.; Wiley-VCH GmbH & Co. KGaA: Weinheim, Germany, 2003; pp 15−168.
(22) Babai, L.; Luks, E. M. Canonical Labeling of Graphs. *Proceedings of the Fifteenth Annual ACM Symposium on Theory of Computing*; ACM Press: New York, 1983; pp 171−183.
(23) Faulon, J. Isomorphism, Automorphism Partitioning, and Canonical Labeling Can Be Solved in Polynomial-Time for Molecular Graphs. *J. Chem. Inf. Comput. Sci.* **1998**, *38*, 432−444.
(24) Plavšić, D.; Vukičević, D.; Randić, M. On Canonical Numbering of Carbon Atoms in Fullerenes: C_{60} Buckminsterfullerene. *Croat. Chem. Acta* **2005**, *78*, 493−502.
(25) Satoh, H.; Koshino, H.; Uzawa, J.; Nakata, T. CAST/CNMR: Highly Accurate ^{13}C NMR Chemical Shift Prediction System Considering Stereochemistry. *Tetrahedron* **2003**, *59*, 4539−4547.
(26) Satoh, H.; Koshino, H.; Uno, T.; Koichi, S.; Iwata, S.; Nakata, T. Effective Consideration of Ring Structures in CAST/CNMR for Highly Accurate ^{13}C NMR Chemical Shift Prediction. *Tetrahedron* **2005**, *61*, 7431−7437.
(27) Hopcroft, J. E.; Tarjan, R. E. Isomorphism of Planar Graphs. In *Complexity of Computer Computations*; Miller, R. E., Thatcher, J. W., Eds.; Plenum Press: New York, 1972; pp 131−150.

CI600238J