Encoding of Polycyclic Si-Containing Molecules for Determining Species Uniqueness in Automated Mechanism Generation

Hsi-Wu Wong,[†] Xuegeng Li,[‡] Mark T. Swihart,[‡] and Linda J. Broadbelt*,[†]

Department of Chemical Engineering, Northwestern University, 2145 Sheridan Road, Evanston, Illinois 60208, and Department of Chemical Engineering, University at Buffalo, 907 Furnas Hall, Buffalo, New York 14260

Received October 5, 2002

Automated mechanism generation is an attractive way to understand the fundamental kinetics of complex reaction systems such as silicon hydride clustering chemistry. It relies on being able to tell molecules apart as they are generated. The graph theoretic foundation allows molecules to be identified using unique notations created from their connectivity. To apply this technique to silicon hydride clustering chemistry, a molecule canonicalization and encoding algorithm was developed to handle complex polycyclic, nonplanar species. The algorithm combines the concepts of extended connectivity and the idea of breaking ties to encode highly symmetric molecules. The connected components in the molecules are encoded separately and reassembled using a depth-first search method to obtain the correct string codes. A revised cycle-finding algorithm was also developed to properly select the cycles used for ring corrections when thermodynamic properties were calculated using group additivity. In this algorithm, the molecules are expressed explicitly as trees, and all linearly independent cycles of every size in the molecule are found. The cycles are then sorted according to their size and functionality, and the cycles with higher priorities will be used to include ring corrections. Applying this algorithm, more appropriate cycle selection and more accurate estimation of thermochemical properties of the molecules can be obtained.

INTRODUCTION

Particulate contamination is the leading source of yield loss during semiconductor processing, and particles formed by homogeneous clustering reactions within process equipment are important and growing sources of this contamination. Kinetic modeling can play a critical role in developing a fundamental understanding of the particle clustering chemistry, which is important if we need to control the formation of silicon hydride particles. However, due to the complexity and the explosive growth of the reaction mechanism in this system, it is impractical to construct the whole reaction mechanism by hand. Therefore, implementation of automated mechanism generation by computers is very attractive.

Determination of the uniqueness of the species generated is one of the necessary tasks in automated mechanism generation. A newly generated species must be compared to all of the structures previously generated and only labeled as a new species to be reacted at a later time if it is unique. The connectivity of each molecule is the most detailed information that can distinguish uniqueness within a set of species. Using graph theory, we can express a molecule as a bond and electron (BE) matrix,[1] and thus store all the connectivity information of the molecule in the matrix. However, for a given molecule, there are as many as $n!$ permutations of the BE matrix, where n is the number of atoms in the molecule. Therefore, a direct brute force search and comparison of all BE matrices of a molecule are impractical, and it is more effective to seek an alternative way to represent the uniqueness of the chemical species generated. Encoding chemical species into a unique notation is the most widely used technique to overcome this problem. Each molecule in the system is translated into an unambiguous code by following a series of encoding algorithms. The differences among the species can be easily distinguished by comparing their codes, and thus the computational efficiency of uniqueness determination can be greatly enhanced.

Detecting the cycles in a given molecule is also an important step in automated mechanism generation. In most cases, the reactions generated do not have sufficient rate constant information available from experiment. Therefore, we need to evaluate the rate constants using kinetic correlations that relate reactivity to thermochemical properties, such as heat of formation, entropy and heat capacity, of the reactants and the products in the reaction. One strategy for obtaining these properties is to use a group additivity scheme.[2] In this approach, ring corrections need to be applied for cyclic compounds, and it is therefore critical to find and pick the correct cycles in the molecules to approximate the thermochemical properties for a given molecule correctly. An algorithm for comprehensively identifying and selecting all the proper cycles in a molecule is thus important in the implementation of automated mechanism generation.

To obtain detailed mechanistic information of silicon hydride clustering chemistry, automated generation software available in our group, developed originally by Broadbelt et

* Corresponding author phone: (847)491-5351; fax: (847)491-3728; e-mail: broadbelt@northwestern.edu.
[†] Northwestern University.
[‡] University at Buffalo.

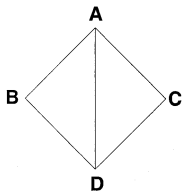
10.1021/ci020343b CCC: $25.00
Published on Web 04/04/2003

al.,[3] will be used. One of the biggest challenges in applying this program to silicon hydride clustering chemistry is encoding unique string codes and selecting the correct cycles to calculate thermochemical properties using a group additivity scheme for the polycyclic dehydrogenated species generated. Unlike most carbon−hydrogen or other organic reaction systems, the molecules produced in silicon hydride clustering reactions usually consist of multiple, condensed cycles. Our former encoding and cycle-finding algorithms, which were constructed based on carbon−hydrogen and planar aromatic systems,[4] were not able to distinguish the isomers of or determine the uniqueness of polycyclic, Si-containing species. Furthermore, not all of the cycles making up these complex molecules could be comprehensively identified. In the present work, algorithms for encoding polycyclic species and identifying appropriate cycles for evaluating thermochemical properties were developed. The differences between our current algorithms and algorithms developed by other researchers will also be discussed.

MOLECULE CANONICALIZATION AND ENCODING ALGORITHM

Graph isomorphism and canonical labeling problems have been studied extensively in the mathematics and computer science literature using graph theory as a foundation. One of the earliest works to address these problems was performed by Weinberg, who developed an algorithm for determining the isomorphism of planar, triply connected graphs.[5] In his work, an Euler's path of a graph is found, and the string code for the graph can then be established. Hopcroft and Tarjan[6] later developed a technique based on the Weinberg algorithm to construct unique notations for graphs using a depth-first search (dfs) method.[7] The algorithm divides the graphs into connected components by identifying articulation points and further divides them into biconnected components then triply connected components. Each graph can then be expressed as a tree-like structure. Several other authors, including Luks,[8] Hoffmann,[9] and Babai and Luks,[10] further investigated isomorphism and canonical labeling algorithms of bounded graphs, where the degree of each vertex is bound by a constant. This is particularly relevant to chemical species since atoms (vertices) have fixed valences. Although all of the algorithms mentioned above provide excellent insight into how to solve graph isomorphism and canonical labeling problems, implementation of these approaches can be challenging, particularly for triply connected components. As a result, they have not been adopted broadly by the communities interested in manipulating chemical species.

The NAUTY (**N**o **AUT**omorphisms, **Y**es?) code developed by McKay[11] is widely considered to be the most powerful program available to deal with graph isomorphism and canonical labeling problems. It not only provides automorphism information for a graph but also produces a canonically labeled isomorph. An example of a canonically labeled graph from NAUTY is illustrated in Figure 1. The algorithm rearranges the order in which a graph is numbered, and a connectivity table of this uniquely numbered graph is output. However, the algorithm was designed for generic use and would need to be modified further to apply it to systems involving comparison of chemical species. In addition, a line notation or string code is a more user-friendly, intuitive unique designation for chemical species, and NAUTY does not provide this information. Therefore, a more straightforward algorithm that is designed for chemical species and can be easily implemented into automatic mechanism generation is desired.



B C A D

B 0: 2 3
C 1: 2 3
A 2: 0 1 3
D 3: 0 1 2

Figure 1. A canonically labeled graph using NAUTY.[11] The graph is renumbered and a connectivity table of the graph is provided.

Determining isomorphism of chemical species is also a well-investigated problem in chemical information science. Since each molecule can be viewed as a graph, the isomorphism and canonical labeling problems of chemical species are essentially the same as the problems of bounded graphs mentioned above. Recognition of topological symmetry, i.e., partitioning of the atoms in a given molecule, is shown to be the same problem as canonical labeling of the molecule.[12] Several methods have been proposed in the context of chemical computation for automorphism partitioning, such as evaluation of the higher order of the BE matrices[13] or determination of the BE matrix eigenvalues.[14] However, the concept of extended connectivity (EC), first developed by Morgan,[15] is the most widely used technique and suggested to be the most viable method to deal with nonplanar components.[16] In the Morgan algorithm, each atom is assigned an EC value, which is the sum of the connectivity of its adjacent atoms. The number of different classes, *K*, can be determined from the number of different EC values in a molecule. The EC values are iteratively updated until the number of classes does not increase. Finally, the atoms can be partitioned according to their penultimate EC values. Figure 2(a) shows the procedure of the Morgan algorithm. Although the Morgan algorithm works efficiently for many molecular structures, several authors have indicated that the algorithm often leads to convergence problems and oscillatory behavior and fails even for some simple molecules.[17,18] Figure 2(b) gives one example where the Morgan algorithm cannot discriminate among the atoms. In this case, there are two different kinds of atoms, ***A, D, F, G*** and ***B, C, E, H***, respectively, in the molecule. However, the same EC value for all atoms is given by the Morgan algorithm after two iterations. The algorithm is not able to differentiate among the atoms, and the identification of topological symmetry fails.

To resolve the shortcomings of the Morgan algorithm, several authors have advanced different methods to increase the applicability of the Morgan-type algorithms. Shelly and Munk developed an extended Morgan algorithm approach for atom partitioning of chemical species.[19] In their modified

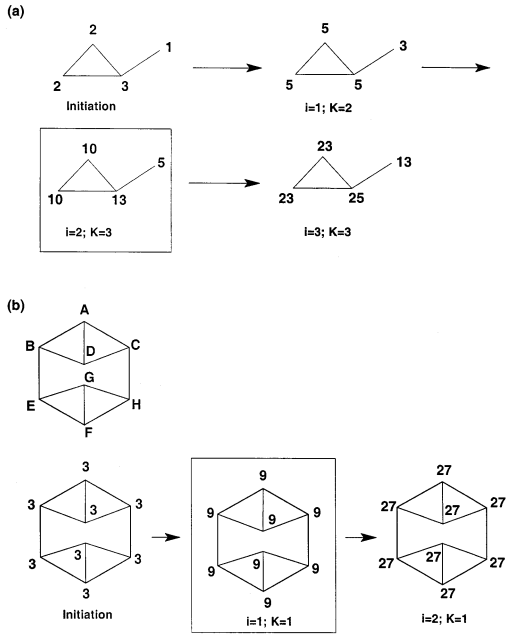
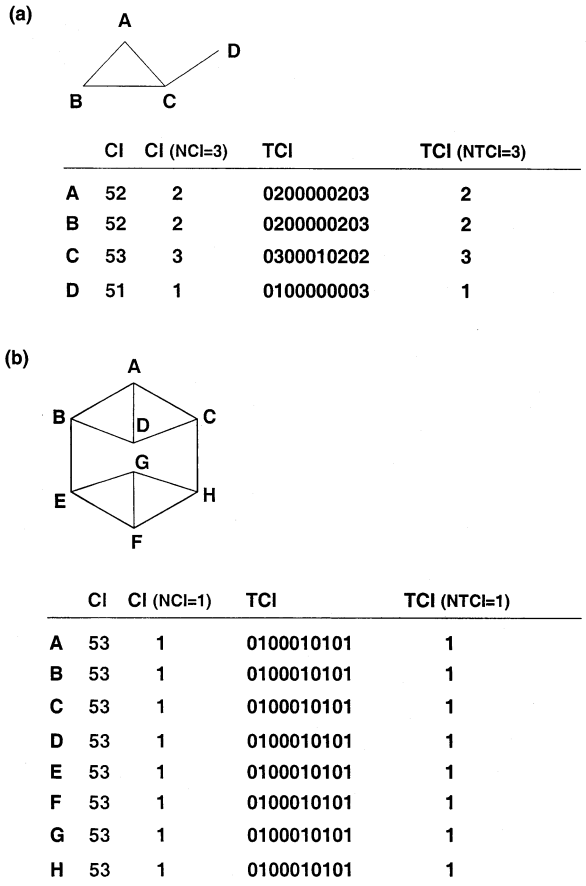


Figure 2. An illustration of the Morgan algorithm:[15] (a) The algorithm successfully partitions the molecule into three classes ($K = 3$) of atoms after three iterations ($i = 3$). (b) The Morgan algorithm fails to determine the topological symmetry of the molecule when all of the atoms have the same connectivity.

algorithm, extended connectivity including each atom's element type was introduced, and the way to evaluate the EC values was redefined. They also proposed a new topological symmetry algorithm (TSA), which combined the concepts from the Ugi algorithm[20] and the modified Morgan algorithm. In the TSA, a class identifier (CI) is initially assigned to each non-hydrogen atom according to its atom type and its connectivity information. Figure 3 illustrates the procedure employed to carry out the algorithm. Two digits of the CI consist of the atoms' elemental indexes (first digit) and number of non-hydrogen bonds attached (second digit). In the example here, the element index is defined as Si = 5, since 2, 3, 4 are assigned arbitrarily to C, N, and O by Shelly and Munk.[19] The number of unique CI values (NCI) is counted and new CI values are assigned accordingly. A trial class identifier (TCI), which consists of five two-digit integers, is then assigned to each atom. The first two integers represent the CI of the atom itself, and the next four fields contain an ordered ascending list of CI values of the atom's neighbors. The number of different TCI (NTCI) values is calculated, and iteration is stopped if NTCI is less than or equal to NCI or if NTCI is equal to the total number of atoms in the molecule. Otherwise, new TCI values are assigned between 1 and NTCI, and iteration continues with setting the new CI value of each atom to its TCI value and setting the new NCI value to the previous NTCI value. In Figure 3(a), the algorithm successfully partitions the molecule into three classes of atoms. However, in Figure 3(b), the TSA is not able to identify the topological symmetry of the same molecule for which the Morgan algorithm fails (Figure 2(b)). To resolve this, a revised algorithm by Shelly and Munk using a permutation method was proposed.[21] In this algorithm, a comparison of $\Pi(C_i)$ permutations is constructed, where C_i is the number of atoms of class i partitioned by the previous algorithm. Although the revised algorithm gives



	CI	CI (NCI=3)	TCI	TCI (NTCI=3)
A	52	2	0200000203	2
B	52	2	0200000203	2
C	53	3	0300010202	3
D	51	1	0100000003	1

	CI	CI (NCI=1)	TCI	TCI (NTCI=1)
A	53	1	0100010101	1
B	53	1	0100010101	1
C	53	1	0100010101	1
D	53	1	0100010101	1
E	53	1	0100010101	1
F	53	1	0100010101	1
G	53	1	0100010101	1
H	53	1	0100010101	1

Figure 3. An illustration of the topological symmetry algorithm (TSA) by Shelly and Munk.[19] (a) Three classes of atoms (NTCI = 3) are found after constructing the trial class identifier (TCI). (b) TSA fails to encode the same molecule shown in Figure 2(b), where adding elemental types is not sufficient to distinguish two different kinds of atoms.

more rigorous results, carrying out all of the permutations inside the same class of atoms is computationally inefficient.

Our previous canonicalization algorithm, developed by Broadbelt et al.,[4] was carried out by constructing the structurally explicit decomposition tree from which the unique string code of the molecule was obtained. In this algorithm, the first step is to determine biconnected components (bicomps) in the molecule via identification of graph articulation points.[22] The decomposition tree is then constructed by reassembling the bicomps with proper connectivity using a depth first search (dfs) method.[7] The unique string code is obtained by iteratively encoding and ordering the subtrees of the decomposition tree. Finally, the string code of the species newly generated is compared to the string codes of the species generated previously, and thus the uniqueness of the species can be determined.

While encoding the decomposition subtree of each bicomp, the atom connecting the current bicomp to its parent bicomp is chosen as the root atom. If the bicomp is the root bicomp itself, all the atoms that have two cycles and are able to initiate lexicographically minimum codes are chosen as the root atom candidates. After the root atom is determined, two branches are constructed by traversing the whole bicomp in a prescribed order, and the decomposition subtree of this bicomp is obtained by assembling the two branches afterward. Last, the interior atoms that are not visited during this branch traversal are appended to the end of the decomposition

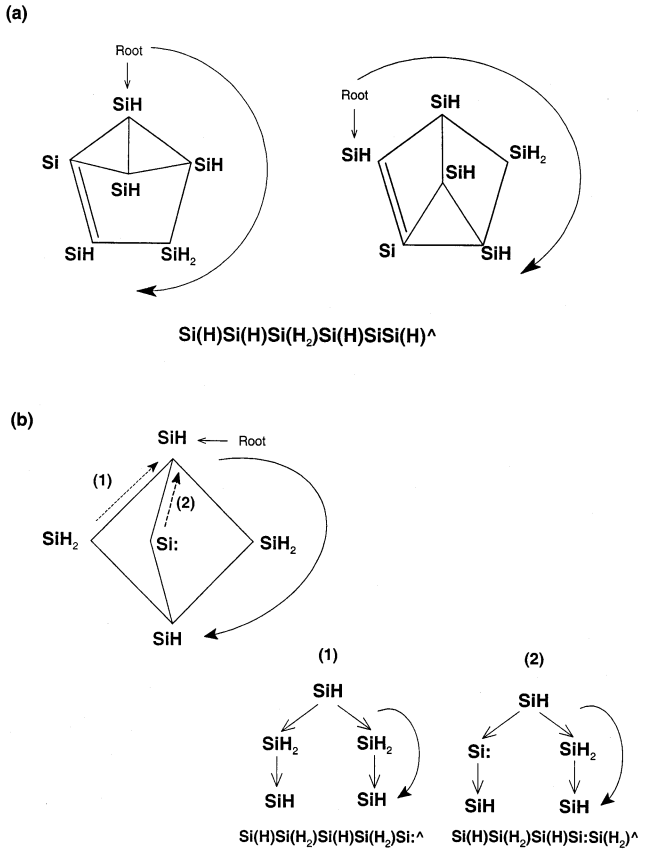


Figure 4. Failure of our previous algorithm as applied to silicon hydride clustering chemistry: (a) Two different molecules have the same traversal path, which results in one string code. (b) One molecule has two different kinds of traversal paths depending on the selection of the cycles, which results in more than one string code.

subtree of the bicomp with a special symbol ("∧" in this work). If there is more than one root atom candidate, all of the possible decomposition subtrees are constructed, and the lexicographically minimum code is chosen by comparing the results from all candidates.

Although our algorithm works efficiently for conventional hydrocarbon systems, it is not able to encode nonplanar molecules we may encounter in the silicon hydride clustering chemistry. In addition, it fails when dealing with some planar species such as polycyclic aromatic hydrocarbons that have the same number of atoms in their interior.[16] Two modes of failure were observed: the same molecule was labeled with multiple string codes or multiple molecules shared the same string code. In both cases, determination of the uniqueness of the species failed. Figure 4 shows two examples when the algorithm was unable to determine the species uniquely. In Figure 4(a), two different molecules had the same string code after applying the algorithm. Both molecules selected one of the Si(H) groups, which had two cycles and could initiate lexicographically minimum codes, as their root atoms. Two traversal directions were considered for each root atom, and the clockwise direction was chosen to obtain the lexicographically minimum traversal order. Since the same traversal order was identified for both molecules and they are composed of the same groups, their string codes are identical. This implies that it is not sufficient to determine the uniqueness of a molecule by simply looking at its traversal order. In Figure 4(b), on the other hand, one molecule generated two different string codes when it was encountered at two different times during mechanism generation. Three four-member rings are present in the molecule, but only two of them are stored. Both Si(H) groups were chosen as candidates as the root atom since they initiate lexicographically minimum codes and belong to two cycles. A subset of the other three groups, however, could be designated as interior atoms depending on how the algorithm selected the cycles. For example, two cycles containing the Si: group were chosen in path (1) of Figure 4(b), and thus Si: became an interior atom. On the other hand, the $Si(H_2)$ group was chosen as an interior group in path (2) because only one cycle containing Si: was selected. This revealed that two cycles were being chosen arbitrarily during traversal of the molecule, and the selection differed based on how the atoms were numbered. This resulted in different string codes generated from the same molecule.

Prickett and Mavrovouniotis[23] proposed a revised Morgan-type algorithm which constructs all of the extended connectivity values in ordered lists of integers. In this algorithm, all of the atoms in the molecule are ranked initially based on the primary index (PI), which consists of a set of properties of the atoms, including atomic number, number of double bonds, number of non-hydrogen neighbors, etc. The ranking of the PI is stored in the class index (CI), and another atom identifier, the secondary index (SI), is calculated based on the atom's CI value. The SI explicitly stores all of the CI information of each atom in a list of integers. For each atom, its SI consists of its CI as the first element, followed by the CI of its neighboring atoms in ascending order. After the SI is calculated for each atom, the CI for each atom is recalculated based on its SI. Then the same iterations are carried out until the number of classes between iterations remains constant. Finally, the string code of the molecule is constructed by finding a path with the minimum possible rank, starting from the atom with the lowest value of CI. Figure 5 illustrates an example of the Prickett and Mavrovouniotis algorithm. Three classes of atoms were identified in this molecule. The string code was then constructed starting from atom ***A***, the atom with lowest rank. However, two possible paths with lowest rank, ***ABFDECG*** (or equivalently, ***ACGEDBF***) and ***ABFGCED*** (or equivalently, ***ACGFBDE***), could be found in this case, and two different possible string codes could thus be constructed. Note that the properties in the primary index used here are different from those used by Prickett and Mavrovouniotis in order to provide more detail required by our silicon hydride reaction system. However, the total classes should be independent of the properties used in the primary index as long as they are selected well enough to distinguish the atoms. Because there are two different paths with equivalent lowest rank, two string codes can still be found for the molecule shown in Figure 5, and determination of the uniqueness of the molecule will thus fail.

The failure of the Prickett and Mavrovouniotis algorithm was derived from its inability to handle molecules which are more symmetric than those that have simple bilateral symmetries. One example is the molecule shown in Figure 5, where four atoms ***D***, ***E***, ***F***, and ***G***, are totally identical, with ***B*** and ***C*** being symmetric. However, once the first atom of one of the two diverging traversal paths has been decided, ***B*** and ***C*** will no longer be the same, and ***D***, ***E***, ***F***, and ***G*** can

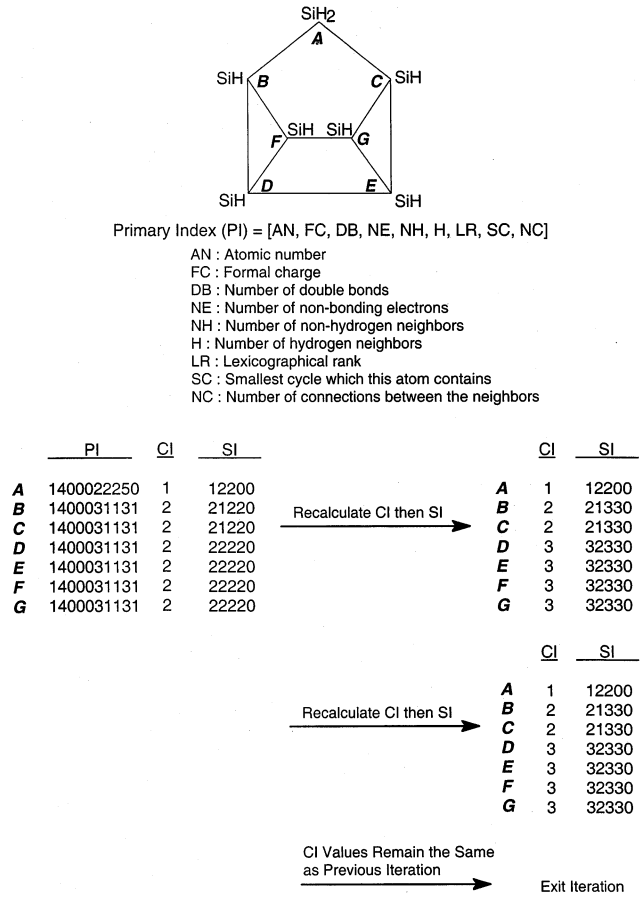


Figure 5. An illustration of the Prickett and Mavrovouniotis[23] algorithm.

be split into two groups: ***D***, ***F***, and ***E***, ***G***, respectively. We can further differentiate ***D*** from ***F*** (or ***E*** from ***G***) by choosing the path of the second traversal route. Weininger[24] proposed the concept of "breaking ties" for more symmetric molecules for encoding unique SMILES codes (USMILES). The algorithm attempts to distinguish the "tie atoms" by doubling all ranks and reducing the value of the lowest valued atom which is tied by one. Similar concepts can also be found in the algorithms developed by Schubert and Ugi[25] and Ouyang et al.[18] By implementing this idea, there will not be any atoms with the same rank, and the ambiguity that causes multiple string codes can then be resolved.

We therefore propose an encoding algorithm that combines the concepts of breaking ties with the Prickett and Mavrovouniotis algorithm. The algorithm also encodes individual connected components separately, which are delineated from the identification of articulation points, instead of traversing the whole molecule. Figure 6 shows an example of this modified encoding algorithm. Starting with the final set of class indices in Figure 5, the ties are manually broken by adding one to all of the values in the CI (underlined bold numbers) except the first atom with the lowest tied value and atoms with CI values below it. The SI values and the new CI values based on these new SI values are then recalculated. Iteration continues until the CI values remain the same as those for the previous iteration. If the maximum rank reaches the number of atoms in the connected component, iteration ends. Otherwise, tie-breaking and iteration continues until the maximum rank is equal to the number of atoms of the connected component. At this point, the path of the minimum possible rank can be constructed. If there are any atoms that are not visited by this path, they will be appended to the end of the path in ascending order of their CI values with a special symbol "∧" tacked onto their lexicographical code. Figure 7 shows the result of this encoding algorithm for the molecule in the previous figures. The path of the minimum possible rank was unambiguously determined according to the final CI values in Figure 6, and the unique string code was constructed based on this path. In addition, the connectivity information for each atom was appended using several numbers inside a curly brace, which represents the other atoms it connects to along the path of minimum possible rank. For example, the code corresponding to the starting point, ***A***, is written as $Si(H_2)\{1,5\}$, which indicates that it connects to the second (***B***) and the sixth (***C***) elements in the path. Similarly, the code for the third element, ***D***, is Si(H){1,3,6}, which can be verified by the connection between the second (***B***), fourth (***F***), and the seventh (***E***) elements. This approach is similar to the concept of ring closure bond notations used in SMILES.[26]

The biggest difference between our algorithm and the algorithms developed by others is that we encode the whole molecule by breaking it down into connected components first. The algorithm works more efficiently, since the number of iterations, which is related to the number of atoms that are encoded, can be greatly reduced. To achieve this, we also have added lexicographical comparison as a component of the primary index of the atoms. Figure 8 shows an example where lexicographical comparison is critical. There are two equivalent paths of minimum possible rank, ***ABCDE*** and ***AEDCB***, if we do not consider the lexicographical code of each atom. The differences outside the connected component, i.e., two branches starting from ***C*** and ***D***, respectively, cannot be distinguished if no lexicographical comparison is made, and atoms ***C*** and ***D*** are considered to be the same erroneously. It is possible to generate two different string codes, and the algorithm will thus fail in this case. By including a lexicographical comparison of the atom codes into the primary index, atom ***C***, with the code of Si(HSi(H2Si(H3))), would have a different rank from atom ***D***, with the code of Si(HSi(H2Si:(H))). The ambiguity is thus resolved, and the unique string code for the molecule results.

We also introduced a new element into our primary index. We defined the last component of our primary index as the number of connections between the neighbors of the atom of interest. Given the set of an atom's neighbors, the number of bonds connecting atoms within that set is tallied. This additional entry was shown to be critical when differentiating atoms in some highly symmetric silicon hydride molecules. Figure 9 illustrates an example where atoms could not be distinguished correctly without using this concept. The molecule shown in the graph consists of eight Si(H) groups. Each of them connects to three other Si(H) groups. It can be clearly seen that the first eight components of the primary index are identical for each Si atom comprising the core of the molecule. However, it is also obvious that there are two different kinds of Si atoms, ***A***, ***D***, ***F***, ***G*** and ***B***, ***C***, ***E***, ***H***, respectively. Therefore, it is impossible to distinguish those two sets of Si atoms by the conventional eight properties used in the first part of the primary index. By adding the last element, i.e., the number of "bridges" between their neighboring atoms, it is possible to differentiate between the

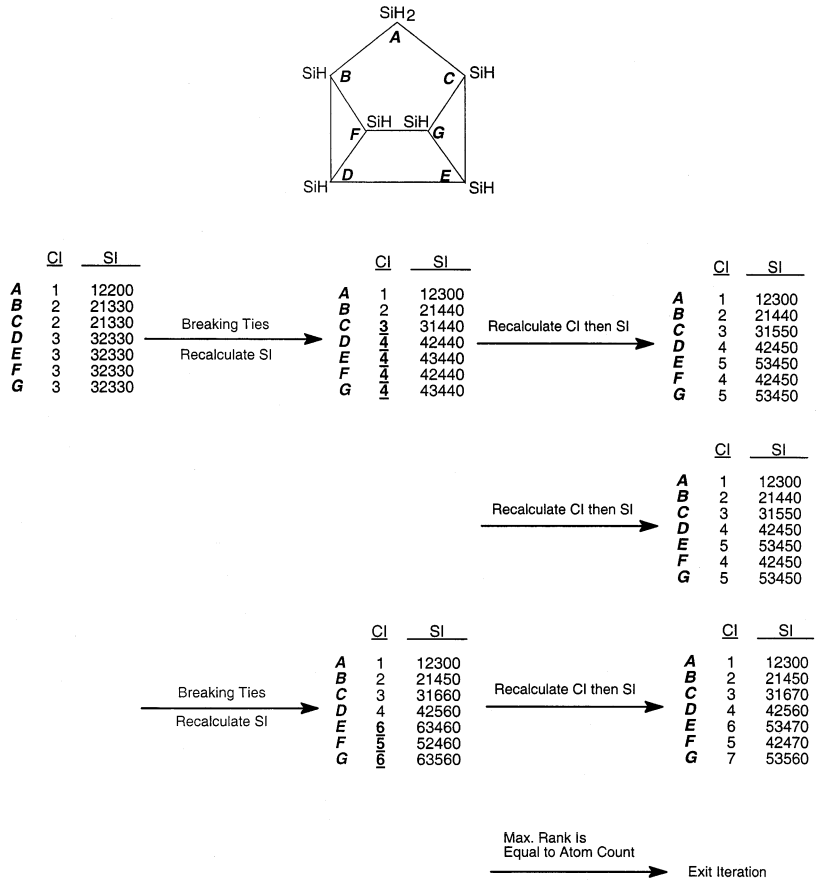
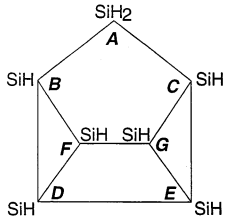


Figure 6. A modified version of the Prickett and Mavrovouniotis algorithm[23] that is able to solve the problem of ambiguous string codes from choosing arbitrary paths of tied atoms.



Path of Minimum Possible Rank : ***ABDFGCE***

String Code : [Si(H2){1,5} Si(H){0,2,3} Si(H){1,3,6} Si(H){1,2,4} Si(H){3,5,6} Si(H){0,4,6} Si(H){2,4,5}]
A B D F G C E

Figure 7. The path of minimum possible rank and the string code constructed from our improved algorithm.

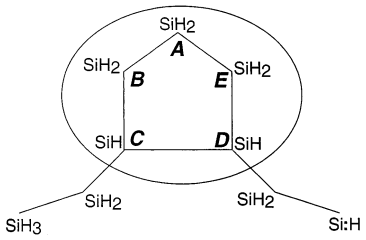


Figure 8. Lexicographical comparison of the code for each atom is critical when encoding connected components separately.

two sets of atoms, and highly symmetric molecules such as the one shown in Figure 9 can be encoded uniquely.

CYCLE-FINDING ALGORITHM

Ring perception algorithms have been well studied in the literature, and a summary of different approaches can be found in the review paper by Downs et al.[27] The cycle-finding algorithm we used previously in our automatic mechanism generation program was based on the algorithm of Prickett and Mavrovouniotis,[23] which is a modified version of the algorithm developed by Gasteiger and Jochum.[28] In this algorithm, a spanning tree of the molecule is created beginning with an arbitrary root atom. All ring closure bonds, which are the bonds not in the spanning tree, are identified and stored in a set. For each ring closure bond, two chains are generated toward the root atom from both ends of the bond until a common atom or bond is found. A cycle can be constructed by combining the ring closure bond with the two chains and the common bond or atom just found. The algorithm is called three times using three different root atoms that are randomly chosen. All of the cycles are stored in a list during iteration, and a check for the linear

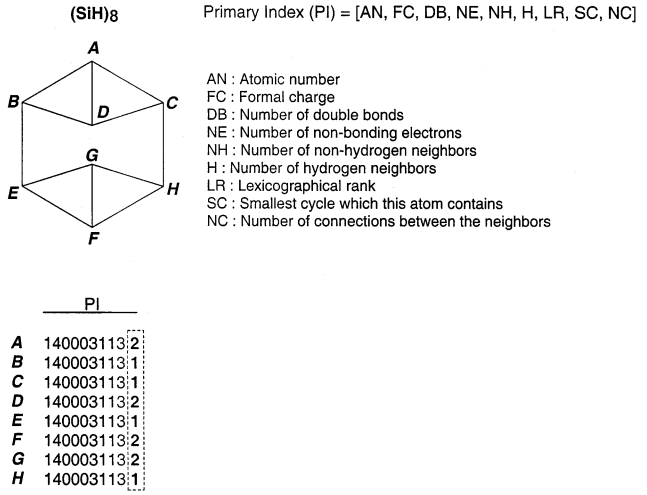


Figure 9. Number of connections between the neighbors is added as one of the elements of the primary index to differentiate atoms in highly symmetric silicon hydride molecules.

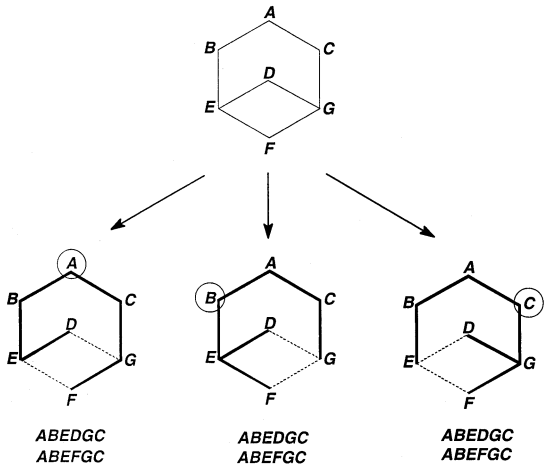


Figure 10. The cycle-finding algorithm by Prickett and Mavrovouniotis.[23] Three different root atoms (***A***, ***B***, ***C***) are chosen to construct the spanning trees (in bold lines) and the ring closure bonds (in dashed lines). None of the three iterations can identify the four-member ring ***DEFG***.

independence of each cycle is carried out to pick the appropriate cycles after the iterations are completed. This algorithm was developed to find the smallest set of smallest linearly independent cycles (SSSC) in the molecule,[29] where the number of cycles in this set, ***N***, is equal to

$$N = (\text{number of bonds}) - (\text{number of atoms}) + 1 \qquad (1)$$

For example, there are three cycles in a naphthalene molecule, two six-member rings and a cycle consisting of 10 atoms. However, the number of cycles in the SSSC is two ($N = 11 - 10 + 1 = 2$). Therefore, only the six-member rings will be selected, because they are linearly independent of each other and are the smallest two in the system. This algorithm, however, does not guarantee that all of the cycles needed will be found. Figure 10 shows one example where some cycles might not be found, depending on the ways in which the root atoms are chosen and the spanning trees are constructed. As mentioned in the review paper by Downs et al.,[27] one should select the cycle-finding algorithm depending on the system of interest. To resolve the problems we encountered in silicon hydride clustering chemistry and find

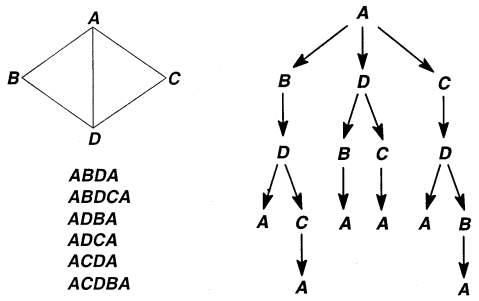


Figure 11. A molecule is expressed as a tree using ***A*** as the root atom. Six cycles are found during this iteration.

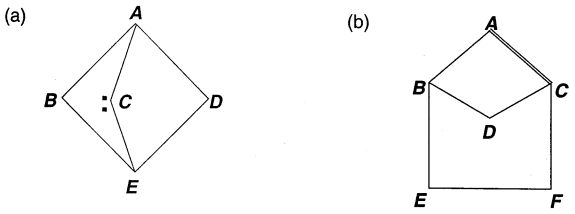


Figure 12. Arbitrarily choosing cycles with the same size results in ambiguity because there are more than ***N*** cycles that are smaller than or equal in size to the largest size needed to complete the SSSC.

the comprehensive set of cycles in the molecule, we propose an alternative cycle-finding algorithm based on the earliest and the simplest "walking" algorithms.[29,30] In this algorithm, a molecule can be expressed as a tree as shown in Figure 11. The branches of the tree are formed by searching all of the atoms to which a given node connects. If an atom is visited by the same branch twice, the branch will end with that atom. A cycle is identified if the terminal atom of a branch is the root atom. The same procedure is carried out using each of the heavy atoms in the molecule as the root atom, and all of the cycles found are stored in a list. After all of the iterations are completed, all of the duplicate and linearly dependent cycles in the list are deleted. The cycles needed for ring corrections using a group additivity scheme can then be selected from this list of linearly independent cycles with all possible ring sizes available.

During the selection of the cycles, the SSSC concept mentioned above is used to determine the numbers of cycles to be chosen, which is equal to ***N*** calculated in eq 1. However, a critical issue was found for polycyclic species such as silicon hydride clusters: there might be more than ***N*** cycles that are smaller than or equal in size to the largest size needed to complete the SSSC. Two examples are shown in Figure 12(a) and Figure 12(b). In Figure 12(a), three four-member rings, ***ABEC***, ***ADEC***, and ***ABED***, can be found, while $N = 2$ in this case. Therefore, two of the three cycles will be randomly chosen. Similarly, in Figure 12(b), there are one four-member ring (***ABDC***) and two five-member rings (***ABEFC*** and ***DBEFC***) in the molecule. Since $N = 2$ for this molecule, the four-member ring (smallest cycle) and one of the five member rings will be arbitrarily selected.

To avoid ambiguity and arbitrariness in choosing cycles and to choose the cycles that represent the molecule most appropriately, we have developed a hierarchy to select the cycles for polycyclic molecules. First, the cycles are categorized according to their size and the functional groups they contain. In the silicon hydride clustering reactions, we keep track of two kinds of functional groups so far: double

bonds and silylene atoms (atoms with two nonbonded electrons). Since we do not allow molecules with multiple functionalities at this point, all of the cycles can be divided into three categories: cycles with a double bond (labeled as ***A***), cycles with a silylene atom (labeled as ***B***), and cycles without any functional group (no labels). All of the cycles can be sorted based on their size, from smallest to biggest, and their functionality, for which we dictated that the cycles labeled as ***A*** have higher priority than the ones labeled as ***B***, and the cycles without any labels have the lowest priority. All of the sorted cycles are then stored in a list, and the ***N*** cycles can be chosen from the list.

Following this hierarchy, the selection of the cycles of the molecules in Figure 12 can be easily done. In Figure 12(a), two cycles with the silylene atom ***C*** (***ABEC*** and ***ADEC***) will be chosen since the third four-member ring does not have any functional groups and thus has lower priority. Similarly, for the molecule shown in Figure 12(b), the cycle ***ABDC*** will be chosen first because it is the smallest ring in the graph. Between the two five-member rings, ***ABEFC*** will be selected rather than ***DBEFC*** since the former ring has a double bond while the latter ring does not. After the proper cycles are chosen, the ring corrections for these polycyclic species can be applied when the group additivity scheme is used, and more accurate estimation can be thus obtained since the hierarchy is consistent with the fitting scheme used to obtain the groups in the first place.

CONCLUSION

A string code encoding algorithm for polycyclic species generated from silicon hydride clustering reactions was developed. This algorithm combines the concepts of extended connectivity and breaking ties, and polycyclic, nonplanar molecules that fail to be encoded using our former algorithm can now be uniquely identified. The algorithm encodes the connected components in the molecules separately and reassembles them using a depth-first search method in order to enhance the efficiency of the algorithm. A cycle-finding algorithm was also developed in this work. In this algorithm, all of the cycles in the molecule of each size are found by expressing the molecule explicitly as a tree. All of the cycles are sorted according to their size and functionality in a list, and the cycles are selected according to their priorities. By applying this algorithm, more consistent cycle selection results. In addition, the appropriate thermochemical properties of the molecules as estimated from a group additivity scheme can be obtained.

ACKNOWLEDGMENT

The authors are grateful for financial support from the National Science Foundation (NSF-CTS0087315).

REFERENCES AND NOTES

(1) Ugi, I.; Bauer, J.; Brandt, J.; Friedrich, J.; Gasteiger, J.; Jochum, C.; Schubert, W. New Applications of Computers in Chemistry. *Angew. Chem., Int. Ed. Engl.* **1979**, *18*, 111−123.

(2) Benson, S. W. *Thermochemical Kinetics − Methods for the Estimation of Thermochemical Data and Rate Parameters*; John Wiley & Sons: New York, 1968.

(3) Broadbelt, L. J.; Stark, S. M.; Klein, M. T. Computer Generated Pyrolysis Modeling: On-the-Fly Generation of Species, Reactions, and Rates. *Ind. Eng. Chem. Res.* **1994**, *33*, 790−799.

(4) Broadbelt, L. J.; Stark, S. M.; Klein, M. T. Computer Generated Reaction Modelling: Decomposition and Encoding Algorithms for Determining Species Uniqueness. *Comput. Chem. Eng.* **1996**, *20*, 113−129.

(5) Weinberg, L. A Simple and Efficient Algorithm for Determining Isomorphism of Planar Triply Connected Graphs. *IEEE Trans. Circuit Theory* **1966**, *CT-13*, 142−148.

(6) Hopcroft, J. E.; Tarjan, R. E. Complexity of Computer Computations In *Isomorphism of Planar Graphs*; Miller, R. E., Thatcher, J. W., Eds.; Plenum Press: New York, 1972; pp 131−152.

(7) Tarjan, R. Depth-First Search and Linear Graph Algorithm. *SIAM J. Comput.* **1972**, *1*, 146−159.

(8) Luks, E. M. Isomorphism of Graphs of Bounded Valence Can Be Tested in Polynomial Time. *J. Comput. Sys. Sci.* **1982**, *25*, 42−65.

(9) Hoffmann, C. M. *Group-Theoretic Algorithms and Graph Isomorphism*; Springer-Verlag: New York, 1982.

(10) Babai, L.; Luks, E. M. Canonical Labeling of Graphs. *Proc. 15th ACM Symp. Theory Comput.* **1983**, 171−183.

(11) McKay, B. D. Practical Graph Isomorphism. *Congressus Numerantium* **1981**, *30*, 45−87.

(12) Read, R. C.; Corneil, D. G. The Graph Isomorphism Disease. *J. Graph Theory* **1977**, *1*, 339−363.

(13) Rücker, G.; Rücker, C. Computer Perception of Constitutional (Topological) Symmetry: TOPSYM, a Fast Algorithm for Partitioning Atoms and Pairwise Relations among Atoms into Equivalence Classes. *J. Chem. Inf. Comput. Sci.* **1990**, *30*, 187−191.

(14) Randić, M. On Unique Numbering of Atoms and Unique Codes for Molecular Graphs. *J. Chem. Inf. Comput. Sci.* **1975**, *15*, 105−108.

(15) 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.

(16) Faulon, J.-L. Isomorphism, Automorphism Partitioning, and Canonical Labeling Can Be Solved in Polynomial-Time for Molecular Graphs. *J. Chem. Inf. Comput. Sci.* **1998**, *38*, 432−444.

(17) Liu, X.; Balasubramanian, K.; Munk, M. E. Computational Techniques for Vertex Partitioning of Graphs. *J. Chem. Inf. Comput. Sci.* **1990**, *30*, 263−269.

(18) 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.

(19) Shelley, C. A.; Munk, M. E. Computer Perception of Topological Symmetry. *J. Chem. Inf. Comput. Sci.* **1977**, *17*, 110−113.

(20) Blair, J.; Gasteiger, J.; Gillespie, C.; Gillespie, P. D.; Ugi, I. Representation of the Constitutional and Stereochemical Features of Chemical Systems in the Computer Assisted Design of Syntheses. *Tetrahedron* **1974**, *30*, 1845−1859.

(21) Shelley, C. A.; Munk, M. E. An Approach to the Assignment of Canonical Connection Tables and Topological Symmetry Perception. *J. Chem. Inf. Comput. Sci.* **1979**, *19*, 247−250.

(22) Baase, S.; Gelder, A. V. *Computer Algorithms: Introduction to Design and Analysis*; Addison-Wesley Longman: New York, 2000.

(23) Prickett, S. E.; Mavrovouniotis, M. L. Construction of Complex Reaction Systems- II. Molecule Manipulation and Reaction Application Algorithms. *Comput. Chem. Eng.* **1997**, *21*, 1237−1254.

(24) Weininger, D. SMILES. 2. Algorithm for Generation of Unique SMILES Notation. *J. Chem. Inf. Comput. Sci.* **1989**, *29*, 97−101.

(25) Schubert, W.; Ugi, I. Constitutional Symmetry and Unique Descriptors of Molecules. *J. Am. Chem. Soc.* **1978**, *100*, 37−41.

(26) 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.

(27) Downs, G. M.; Gillet, V. J.; Holliday, J. D.; Lynch, M. F. Review of Ring Perception Algorithms for Chemical Graphs. *J. Chem. Inf. Comput. Sci.* **1989**, *29*, 172−187.

(28) Gasteiger, J.; Jochum, C. An Algorithm for the Perception of Synthetically Important Rings. *J. Chem. Inf. Comput. Sci.* **1979**, *19*, 43−48.

(29) Wipke, W. T.; Dyott, T. M. Use of Ring Assemblies in a Ring Perception Algorithm. *J. Chem. Inf. Comput. Sci.* **1975**, *15*, 140−147.

(30) Corey, E. J.; Wipke, W. T. Computer-Assisted Design of Complex Organic Syntheses. *Science* **1969**, *166*, 178−192.

CI020343B