MATHEMATICAL MODELING FOR RING SYSTEMS IN MOLECULAR NETWORKS

by

Dalal Alrowaili

A Dissertation Submitted in Partial Fulfillment

of the Requirements for the Degree of

Doctor of Philosophy in Computational Sciences

Middle Tennessee State University

December 2017

Dissertation Committee:

Prof. Dong Ye, Chair, Mathematics Department

Prof. Xiaoya Zha, Mathematics Department

Prof. Chris Stephens, Mathematics Department

Prof. Anatoliy Volkov, Chemistry Department

Prof. John Wallin, Physics and Astronomy Department

ACKNOWLEDGMENTS

I would like to thank Dr. John Wallin and Middle Tennessee State University for the opportunity to study as a part of the Computational Science Ph.D. program. I would like to give my dissertation committee of Dr. Xiaoya Zha, Dr. Chris Stephens, Dr. Anatoliy Volkov, and Dr. Dong Ye my sincere thanks. I would like to thank my many classmates and collaborators at Middle Tennessee State University for our productive discussions. Most importantly, I would like to express my deepest gratitude to my supervisor Dr. Ye. I have been incredibly lucky to have such a supervisor. He has been a most valued collaborator. Without his support and guidance, none of this would have been possible. I would like to thank my mom and my brothers, sisters and friends for their love and support throughout this long process.

ABSTRACT

In computational mathematics, Graph Theory serves as an abstract model for chemical compounds. Induced cycles, i.e. cycles with no chords, are referred to rings in molecules, and these rings have an important physical meaning in Chemistry. The mathematical analysis and development of algorithms for the ring perception problem is analogous to cycle detection in graph theory. In this work, we are interested in the representation of chemical structures using graphs and the detection of rings in these structures. In the first chapter, we develop a polynomial time algorithm for the detection of all small induced cycles in a given graph G. We achieve a complexity of $\mathcal{O}(m^3n+n^2)$ for a graph of m edges and n vertices. Then, we apply this approach to several chemical compounds such as fullerenes, cata-condensed benzenoids, protein structures and others.

Many chemical properties of fullerenes and benzenoid systems can be explained in Mathematics in terms of the number of perfect matchings, the Clar number, the Fries number, the HOMO-LUMO energy gap, etc. These are some of predictors of molecules stability. In the second chapter, we investigate the Fries number and Clar number for hexagonal systems and show that a cata-condensed hexagonal system has a maximum resonant set containing a maximum independent resonant set, which is conjectured for all hexagonal systems. Further, our computation results demonstrate that there exist many contra-pairs, and, for stability predictor of hexagonal systems, the Clar number is better than Fries number. Lastly, we compute the Clar number and Fries number of all isomers of fullerenes $C_{20}-C_{60}$ by using integer linear programming in addition to calculating the HOMO-LUMO energy gap of all fullerenes isomers.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

The ring perception problem (i.e. the identification of cyclic substructures) is used in a diverse range of problems including the analysis of electrical circuits, the analysis of communication networks, operations research, and the identification and classification of chemical structures. It is a fundamental part of chemical structure storage and retrieval systems [35]. The concept of rings in chemistry is defined as cycles of the linear graph in mathematics [96]. In chemical structures, the presence of small cycles can signify important physical properties; for example, the structural integrity of organic compounds [50].

Ring perception is fundamentally important not only for identifying rings but also for computer aided synthesis goals [108, 39]. It is also useful in identifying structural fragments associated with rings. For this reason, the synthesis of cyclic structures is usually desirable for researchers. The presence of rings also controls the possible conformations of a molecule by limiting the intramolecular reactions [108]. The synthesis of cyclic structures can be difficult due to these restrictions on conformational freedom [50, 108, 73].

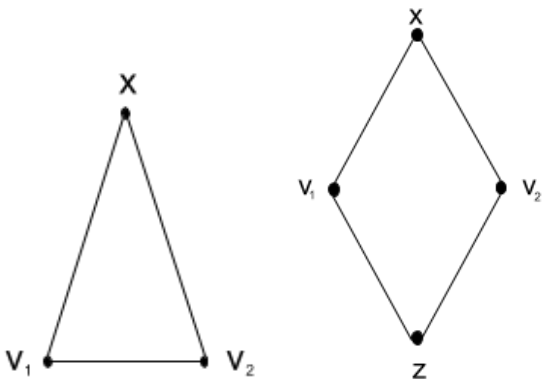
Another important property that is an indicator of the exceptional stability of a cycle is the classification of the aromaticity [108, 22]. A bond or atom is aromatic if it is a member of a cycle that contains $4n+2$ π-electrons where $n \geq 0$. The aromaticity of molecules implies that it only undergoes certain reactions and the cycles are of particular lengths [50, 108, 22].

Chemically interesting cycles are usually small. Depending on the application, the maximum length of these cycles is usually anywhere from six to eight. Rings that are bigger than six edges have limited synthetic reactions [35]. For this reason, the perception of rings of length six or smaller are of particular interest to chemists [35]. The reactions that use larger rings are only successful when the ring is known to be “real”, i.e. a ring from a maximum proper covering set [22]. Therefore, rings that are to be used as synthetic units

are all cycles of length six or smaller and all cycles larger than six that are known to be "real" [22].

The length of the smallest ring in a molecule usually determines the ease of which a bond will form [22]. For instance, consider the formation of a carbon-carbon bond using an aldol reaction to form a 6-membered ring fused to either a 3-membered ring or a 9-membered ring [22]. The reaction works much more easily when the 6-membered ring is fused to the 3-membered ring [22].

Finding large rings is a more difficult problem. Garey and Jonson [89] showed that finding the longest cycle or longest induced cycle is an $\mathcal{NP}$-complete problem. However, Lokshtanov [81] produced an algorithm to determine the longest isometric cycle of a given connected graph in polynomial time.



(A) C_3 (B) C_4

Figure 1: Induced Cycles of Length 3 and 4

Definitions

The mathematical analysis and development of algorithms for the ring perception problem are analogous to cycle detection in graph theory. In this project, we are interested in the representation of chemical structures using graphs and the detection of rings in these structures. Any molecule may be represented by a graph G, where every atom is a vertex

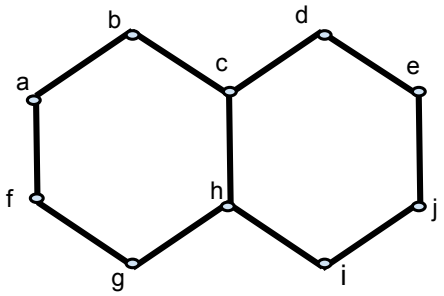


Figure 2: Isometric Cycles

v and every bond is an edge e. The Frerejacque number or the cyclomatic number of the molecule is represented as the graphs *nullity* [35]. The degree of valency of the molecule is represented as the graph's *connectivity*. In this section, we will provide some definitions from graph theory and chemistry, as they will be helpful in the discussion of cycle detection.

Let graph $G = \{V, E\}$ where V and E are sets of all vertices and edges of G. Alternatively, the vertex set and edge set of G can be written as $V(G)$ and $E(G)$, respectively. G is *connected* if for any two vertices u and $v \in V(G)$, there is a path joining them. If G is *disconnected* graph then it has at least two components (subgraphs of G). The *distance* between two vertices u and v is the number of edges that make the shortest path joining u and v , denoted by $dist_G(u,v)$.

A sequence of vertices where all vertices are distinct is a *path*. A closed path that begins and ends at the same vertex is a *cycle*. A cycle of a graph G is *induced* if it has no chords, see Figure 1A and 1B. An *isometric* cycle C is a cycle such that for any two vertices $u, v \in V(C)$, $\text{dist}_G(u,v) = \text{dist}_C(u,v)$. For example in Figure 2, the isometric cycles of this example are $\{abchgfa,\ cdejihc\}$, but the cycle $\{abcdejihgfa\}$ is not isometric because $\text{dist}_G(c,h) \neq \text{dist}_C(c,h)$. For any two vertices $u, v \in V(P)$ said P is a shortest path if $\text{dist}_G(u,v) = \text{dist}_P(u,v)$. A cycle of a graph G is *shortest* if for any two vertices $u, v \in V(C)$, the distance between u and v is the length of the shortest path joining them in G.

A cycle is called the longest isometric cycle of a graph if it has the largest cycle length such that for any two vertices u and v on the cycle, $\text{dist}_G(u,v) = \text{dist}_C(u,v)$ [81]. A cycle of a graph G is *longest cycle* if any two vertices $u,v \in V(C)$, the distance between u and v is the longest distance $\text{dist}_G(u,v) = \max(\text{dist}_C(u,v))$. The diameter of a graph G is the greatest distance between u and v where $u,v \in V(G)$ [29]. An alternating sequence $v_1e_1v_2e_2v_{i-1}e_{i-1}v_i$ of vertices and edges of G, such that starting and ending at vertices and e_{i-1} is incident with v_{i-1} and v_i, is a *walk* . A closed walk that starts and ends with the same vertex is a *circuit*. A *cut vertex* is a vertex such that removal of this vertex disconnects and divides a graph into two or more components. Similarly, a *cut edge* is an edge such that the removal of this edge disconnects and divides into two or more components. A *block* is a component that does not have cut vertices. A *bridge* is a series of one or more cut edges connecting two blocks of a component. This is not to be confused with the term bridge from chemistry, which indicates a series of two or more edges crossing a cycle [35].

A *spanning tree* is a tree that contains all vertices of a graph G. The *chords* are the minimum number of removal edges that make a cyclic graph to an acyclic graph. The *nullity* of a graph is the number of chords in a graph. A *neighbor* of a vertex u is a vertex v such that $uv \in E(G)$. The set of the neighborhood of u is a set of all neighbors of u denoted $N(u)$. The ith neighborhood of u is the set of vertices such that for every vertex $v \in N_i(u)$ that $dist_G(u,v) = i$. $E(G[N_i]) = \{\overline{uv} \mid \overline{uv} \in E \quad \text{and} \quad u,v \in N_i(x)\}$ and $E(G^c[N_i])\{\overline{uv} \mid \overline{uv} \notin E \quad \text{and} \quad u,v \in N_i(x)\}$.

Breadth First Search algorithm (BFS) is a search algorithm that starts with a given vertex x and finds all adjacent vertices. Then, for every vertex y in the neighbor of x, it will find all unvisited vertices in the neighbor of y and so on.

1.1 Cycles in Chemical Graphs

Many algorithms have been implemented to attain a variety of ring sets for connected graphs [35]. Not all algorithms find a general set of all rings [35]. Some are developed for

specific sets of rings with particular sets of desirable properties.

One of these sets is known as the Smallest Set of the Smallest Rings (SSSR). This set was first described by Plotkin [96] and is a fundamental type of ring set. Plotkin defines the SSSR to be the maximal linearly independent set of rings S in the set of all rings C. Therefore, the SSSR is also a minimum spanning set or a minimum covering set. This is a particularly important feature of the SSSR as the conformation of a ring system can be determined by the conformational possibilities of the smallest rings [73]. For instance, Hydrindane has three rings each with a length of five, six, and nine see Figure 3. However, the SSSR only contains the rings of lengths five and six. This set contains the information for the entire ring system because the ring of length nine is a linear combination of rings in the SSSR [73].

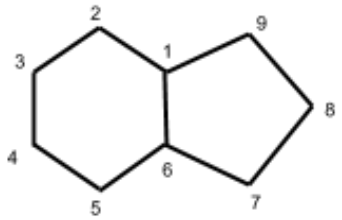


Figure 3: Hydrindane

Finding the SSSR can be used for synthesis planning by using the fundamental ring systems features. Plotkin [96] and Gasteiger & Jochum [50] developed methods for finding the SSSR. Plotkins algorithm includes all rings up to length seven and eight. This can be beneficial compared with other algorithms if larger rings are present in the structure. For example, in the Cubane molecule in Figure 4, many ring algorithms will only find 16 rings of length six and 6 rings of length four, while Plotkins algorithm will add 6 rings of length eight [35]. Although these algorithms are successful for analyzing structure, they are not always valid in complex structures [35].

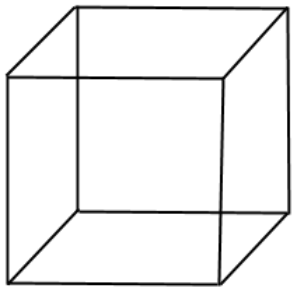


Figure 4: Cubane

1.2 Conjugated Cycles

Many cyclic molecules in organic chemistry are conjugated, but some of these compounds are aromatic and others antiaromatic. Let H be a hexagonal system. A *perfect matching* M of H is a set of disjoint edges such that every vertex of H is incident with exactly one edge in M. These edges of M are the edges that previously presented as double bonds of benzenoid systems in the Kekule structure, see Figure 5. A perfect matching of a hexagonal system is also called a *Kekulé structure* in chemistry literature [88].

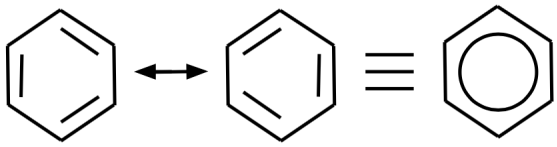


Figure 5: Kekulé Structures for Benzene or Resonance Structures

Definition 1. A cycle C of H is *M-alternating* (or resonant) if the edges of C alternate between M and $E(H)\backslash M$.

A conjugated cyclic molecule is very stable and aromatic if it obeys H*ü*ckel's rule, that is, the number of π-electrons of this molecule is $4n+2$ where $n \geq 0$ [88, 98].

Aromatic molecules are planar and that can be estimated from the H*ü*ckel's rule [88]. Aromatic molecules are very stable due to the delocalized π-electrons (or have large resonance energy values) [88]. The resonance stabilization energy (or delocalization energy) of a molecule can be estimated by the following formula

$$RE = \frac{1}{K}\sum_{n\geq 1}(R_n(\#^{4n+2}) + Q_n(\#^{4n})$$

Where K is the number of Kekulé structures (perfect matchings) of the molecule, and Q_n and R_n are oppositely signed parameters that decrease roughly geometrically with respect to the length of the cycle [99].

Some conjugated cycle molecules are planar and don't follow the the H*ü*ckel's rule and others are non-planar. In both of these cases, if the molecule contains $4n$ π-electrons, then they are antiaromatic [88]. These antiaromatic compounds have a low positive resonance energy values and small HOMO-LUMO energy gap value and thus, are unstable [88]. Figure 6 shows some examples for $(4n+2)$ π-electrons (or aromatic) and $(4n)$ π-electrons (or antiaromatic) molecules. Benzene C_6H_6 is the most stable molecule and a common example of the aromatic molecules because it has 6 π-electrons that form three alternated double bonds. However, Cyclooctatetraene C_8H_8 has 8 π-electrons and is not aromatic because it has $(4n)$ π-electrons.

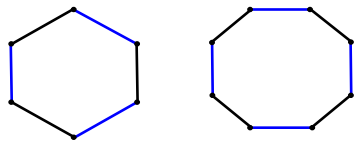


Figure 6: Cycles of Size 6 and 8

Definition 2. The hexagonal rings that have precisely three bounding edges in a given Kekulé structure are called *benzene faces*, see Figure 7.

Aromaticity is one of the most important concepts that has been attracting scientists'

Figure 7: A Benzene Face

attention since 1865 when Kekulé [65, 64] presented the cyclic structure of benzene. Later, several studies explained the characterization of aromaticity in terms of conjugated cycles [101, 104, 102, 103, 100].

Conjugated cycles have been used as a tool to solve many difficult problems of aromaticity [97]. In 1976, conjugated π-systems were noticed as a result of observing the difference between the three Kekulé valence structures of the "naphthalene" molecule. The naphthalene has one unique Kekulé structure where both hexagons have alternating single and double bonds see Figure 8 (b). This unique structure is the most stable resonance structure, see Figure 8. Notice that the other structures in Figure 8 (a and c) have only one hexagon with three double bonds.

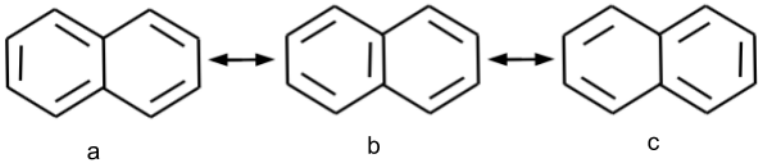


Figure 8: Different Kekulé Structures for Naphthalene

CHAPTER 2

SMALL CYCLES

In the following chapter, we give a brief review of cycle algorithms, a mathematical description of the problem setting, including definitions. We then outline our algorithms and give examples of this work. Next, we provide theorems for the analysis.

2.1 Cycle Algorithms

Computer aided cycle detection is a powerful tool. The programmatic representation of graphs is usually achieved by either a connection table or an adjacency matrix [35]. A connection table can easily be transformed into an adjacency matrix of a graph or incidence matrix [35]. Algorithms can be dependent on which of these representations are used.

There are two primary methods for ring perception, walking methods and matrix manipulation methods [35]. Walking methods are very clear and were the first to be developed. These methods 'walk' through the connection table starting with a vertex and tracing paths and branch points [35]. If we get back to the starting vertex, then a cycle is found and stored. Otherwise, we find the final branch vertex and then that path is contracted to that final vertex. This method is terminated when all paths are searched from each branch vertex. Applying this method on undirected graphs produces duplicate cycles because paths are found from both sides around the cycle, which adds unnecessary complexity. Therefore, some rules should be used to search the right path in a short time [35]. This method can work directly from the connection table and generate an ordered list of vertices and edges that form the desired rings [35].

The second fundamental method uses manipulations of trees, sets, or matrices to identify the desired set of rings [35]. Most algorithms that use this method depend on matrix manipulations to construct a spanning tree to generate a fundamental basis of cycles. The

fundamental basis is the set of the minimum number of linearly independent cycles that cover all vertices and edges of each ring [35]. However, the sets of fundamental bases of cycles are not unique.

Vector space algorithms are also used to find cycles [35]. By using different starting vertices and expanding spanning trees from the ends of the chords, we may produce more trees, and then take all the combinations of the resulting rings for additional fundamental bases. This method will derive rings which have an un-ordered sequence of vertices. Another step is necessary for this method to produce an ordered sequence of vertices or edges.

Many algorithms have been implemented to attain a variety of cycle sets of connected graphs [35]. Not all algorithms find a general set of all cycles for a given graph. Some are developed for a specific set of cycles with a particular set of desirable properties [35]. Most algorithms that we review are classified in four categories: 1) algorithms that find all possible cycles and then determine a specific set of cycles, 2) algorithms that find all possible simple cycles and then determine a specific set of cycles, 3) algorithms that find a fundamental basis of cycles, and 4) algorithms that find the smallest fundamental basis cycles set which is called the smallest set of smallest rings (SSSR). A short review is given in the next section for algorithms that find all possible cycles of a graph.

2.1.1 Algorithms to Find All Cycles

In 1970, Tiernan [118] developed an algorithm for cycle detection in directed graphs. In general, directed graphs are much easier to search. This algorithm is less efficient than others if it is used for undirected graphs. Tiernan uses a walking method to search a $|V| \times |V|$ adjacency matrix. The first vertex in the adjacency matrix is chosen to be the start vertex to search a path. The next start vertex chosen should not be in a found path and should not be adjacent to the last vertex of the found path. Also, the label of this vertex must be greater than that of the last start vertex. This algorithm finds an initial ring set containing all cycles and then finds the final ring set, which is the set of all simple cycles. A similar approach is

the backtracking algorithm presented by Beiss, Janicke, and Meissler [58]. In this approach the complexity analysis shows that it is more efficient than Tiernans algorithm.

Numerous algorithms are developed to obtain all rings in directed graphs. Using adjacency matrices manipulations approach to finding all cycles of directed graphs is easy, but additional steps are required in the process of selecting the desired cycles from the set of all cycles [35]. Many of these algorithms have not been implemented to chemical graphs.

Corey et. al. [22] established the first algorithm used in the LHASA organic synthesis program for ring perception. By producing a simple spanning tree using depth first search from the connection table, cycles are found and each one is represented as a cycle vector. From the set of all cycles, rings can be divided into real rings and pseudo rings. A set of real rings contains all cycles that are found from a maximum proper covering set and all other cycles are pseudo rings.

Shelley [114] used ring perception as a step to generate the coordinates for presenting chemical structures by graphs. The process of ring identification is used for assigning relative vertices coordinates for each ring system. This algorithm uses breadth the first search algorithm to produce a spanning tree to find all chords. Next, it finds all paths between every two vertices of each chord by using the depth first search algorithm and then adds that chord to the tree. An initial set of all cycles is found, all smaller cycles that contain nachbarpunkte are removed from the set to obtain a set of all simple cycles, and these rings are included in the set of $\mathcal{K}$-rings.

2.1.2 Algorithms to Find All Simple Cycles

An algorithm presented by Nickelsen [92] is used to find a set of simple cycles. This algorithm is used to find β-rings, which are required for mathematically based ring perception for the implementation in the GREMAS system. Nickelsen presents four criteria for the selection of β-rings. These rings are defined to be a set of simple cycles where each

has just three or four vertices that cannot be constructed from three or more smaller simple cycles. The concept of β-rings has some problems as it is difficult to apply manually, it requires producing all simple cycles, and it needs a large number of ring sum operations for a complex ring system. Furthermore, it does not produce the set of rings that conforms to the chemists concept of "real rings".

2.1.3 Algorithms for the Fundamental Basis Set of Cycles

A fundamental basis set may be preferred to finding the set of all rings. A spanning tree may be constructed by either manipulating the adjacency matrix or by a path searching algorithm with a connection table [35]. The spanning tree is then used to determine a fundamental basis set by observing the chords. Finding one ring for each chord can create a basis. The fundamental basis deduction is important in graph theory and network theory. Taking the ring sum of combinations of the basis cycle vectors after creating a fundamental basis can generate all cycles. A vector space and a backtracking algorithm have been developed to generate all possible cycles and find a fundamental basis set [35]. Perhaps the most important type of these sets is the smallest fundamental basis cycle set, which is called the smallest set of the smallest rings (SSSR) [35].

Welch [122] presented an algorithm to find the fundamental basis then conclude a set of all cycles. A $|V| \times |E|$ incidence matrix is used where columns are rearranged and partitioned in a form to apply matrix manipulations easily and then attain such set. The manipulation of the matrix uses a process for isolating the cycle vectors. Cycle vectors are stored in the basis cycle matrix as rows so that standard ring sum operations can be completed in stages. Then, for each component of the graph structure, a fundamental basis is attained and all combinations of fundamental cycle vectors can be generated. Welch used two additional stages to minimize the number of ring sum operations to find all cycles.

Paton [93] applied depth first search algorithm to produce a spanning tree by using adjacency matrix manipulations and a stack (or pushdown list) of unvisited vertices. For

adjacency matrix A of $|E|$ edges and $|V|$ vertices, let B be the spanning tree matrix, T be a set of vertices that are in the spanning tree, and X be a set of unvisited vertices which is equal to the set of vertices V. The algorithm starts by choosing a vertex from X to be the root of the tree. The starting vertex is stored on the stack and in the matrix B. While the stack is not empty, the algorithm loops over the following steps: 1) Pick a vertex from the stack and use A to find the associated edge. Let the set of these edges be Y. 2) Choose an edge from Y. If this edge is in T, then it is a chord and we backtrace the fundamental cycle found in B. Otherwise, this edge is added to B and the associated vertex is stored in T. In either case, delete this edge from Y and repeat this step until Y is empty. 3) Delete the chosen vertex from A and X. When the algorithm terminates, the spanning tree is complete and the fundamental cycles have been found. Paton shows that using depth first search algorithm in the procedure of selecting the next vertex is more efficient than using breadth first search algorithm. Jovanovich [63] modified Paton's algorithm to make it more efficient. Kizawas algorithm [62] has a similar matrix manipulation process and proves that producing a spanning tree using a list is more efficient by using breadth first search algorithm.

Corey and Petersson [21] developed another algorithm that is used in the LHASA synthesis system. This algorithm finds a fundamental basis and then derives the SSSR. A spanning tree is constructed by using Paton's algorithm. A set of edges and vertices is used to save every vertex in the tree, and a ring sum of the right pair of edge sets is used to derive a reduced fundamental basis cycle. This step costs extra storage, but eliminates backtracking the fundamental cycle like in Paton's algorithm [93]. Then the algorithm reduces the set of fundamental basis rings to obtain a reduced basis by taking the ring sums of each pair of fundamental basis cycles vectors to find the SSSR. The SSSR and all cycles of length 7 and less can be obtained by selecting the smallest rings from the set of reduced basis.

The Casteiger and Jochums [50] algorithm determines the SSSR by using the spanning tree method and produces a set of fundamental cycles that contains all possible smallest

cycles. In their paper, they do not focus on finding a set of all cycles but argue for the use of the SSSR. This algorithm does not use the method of reduced basis that is implemented in some other algorithms such as Corey and Petersson [21] or Wipke and Dyott [127], but can obtain the SSSR by directly constructing the smallest fundamental cycles. If an SSSR is not found, then other fundamental cycle sets containing the smallest cycles are produced.

This algorithm has two main stages. In the first stage, it constructs a spanning tree by searching for chords from a starting vertex using breadth-first search. Each chord is saved and ordered according to its distance from the starting vertex. Then backtracking is used back to the root for one end of each chord to find the first shared vertex between the two paths. This guarantees that the closest path is discovered and a set of fundamental basis cycles is produced. This fundamental cycle set is usually the SSSR. A second stage is implemented in cases where the SSSR is not found. This stage requires the following definition for the complexity of a ring system C:

$$C = \frac{|V(C)|}{|V(R)|}.$$

The second stage repeats itself until $C \leq 1.5$. It begins by selecting a vertex to be the new starting root vertex. This root must be chosen such that it is contained in a smallest cycle that does not include any pervious roots or otherwise chosen arbitrarily if this is not possible. It also must have the greatest ring connectivity for the cycle from which it was chosen. This is repeated for the new root vertex for a maximum of three iterations. Finally, an SSSR is found from the accumulated ring set.

2.1.4 Algorithms That Find a Smallest Set of Smallest Rings Directly

The SSSR has an important effect on the consistent characterization of a molecule structure since it contains the smallest fundamental basis cycles. A small basis is useful for fast processing of a graph; especially if the set is extended to comprise other cycles [35].

Although, an SSSR might not be unique in some complex structures that contains many equivalent rings, the SSSR is considered sufficient for most applications used to make a deeper ring analysis[35].

Plotkin [96] gave one of the first descriptions of the independent class of rings called the Smallest Set of the Smallest Rings (SSSR) as we mentioned previously. This algorithm is implemented in the Chemical Information and Data System (CIDS). The rings found by this algorithm are all the cycles of the length eight or smaller, or classified as K-rings. The class K is defined as a set of all SSSR rings that one ring cannot be indicated by the sum of the other smallest rings.

The first stage of this algorithm is finding an SSSR for the given graph. The second stage uses this SSSR to derive all rings of class K.

Stage one of the algorithm is to eliminate all edges that are not contained in any rings. Then Plotkin's theorem is applied for stratifying ring perception:

Theorem 2.1.1. *[96] If P is an unforked path in a structure G, and there is a shortest ring R through P such that $|R| \leq 2|P|$, there is an SSSR of G that contains R and no other ring in which P occurs.*

Therefore, the next step is to search for a suitable unforked path P and ring R by testing each edge of the structure. Then a longest unforked path P is determined containing an edge uv, and paths are constructed by searching from one end of the edge uv and smallest rings are derived. If the end vertex v is reached where path distance $L \leq |P|$ is the distance from vertex u, then all paths of length L, which join u and v with P are smallest rings R with length $|R| = L + |P| \leq 2|P|$. Thus, one or more suitable rings are found and contained in a path P, so we store one in the set of the smallest rings and delete the path P from the graph. Then the same process is repeated for $d(C)$ iterations where $d(C)$ is the number of rings in the set of all rings C. Therefore, no rings are left and an SSSR is found.

Edge uv is chosen to lie within the largest ring found. If the graph does not contain any suitable path P containing uv, then we simply delete this edge. Otherwise, an edge is picked at random if there no cycles are found. Then we repeat the procedure to find the SSSR with each edge removed.

In the second stage of the algorithm, the following theorem is applied to find all rings of the class K:

Theorem 2.1.2. *[96] Given an SSSR, if R_m, is the longest ring of the SSSR and if P is an unforked path that is part of R_m, but contained in no other ring in the SSSR, then the rings of class K that pass through P are the rings of length $|R_m|$ that pass through P. These are the shortest rings through P.*

Therefore, a longest unforked path P is selected through the largest ring in the SSSR previously found. All rings of length $|R_m|$ through P, are found from the previous step are stored in the set of class K. Then we delete the path P from the graph and delete R_m from the SSSR. Repeat the same process for $d(C)$ iterations until no rings are left. The algorithm terminates with the set of all K-ring. Plotkins algorithm is invalid for certain structures, because the rules of choosing an edge sometimes delete an edge required to find an SSSR.

Bersohn [11] establishes an algorithm to produce a set of synthetically important rings as they are described by Corey and Petersson [21]. These rings are all small cycles of size six or less or are not envelope cycles. If the value of nullity $|R|$ is greater than zero this indicates a ring's existence. Any acyclic chains is deleted (i.e. chains that do not belong to any rings) from the connection table to make a new structure. Bersohn initialized $k = 3$ and apply the following steps:

1. If $|R| = 1$, the new structure is an isolated cycle found and then the procedure stops. Else, choose a start vertex.

2. Use this starting vertex to search for all paths of length k. If the path reaches the start vertex, the cycle is stored (unless it had been previously found). Then if the cycle discovered has a path of length 7 or more, the algorithm deletes all paths starting with the new path sequence. Otherwise, if the path has a length less than 6, the algorithm deletes these paths once the lengths of these paths are more than 6. The application of this step of the algorithm guarantees that once a smallest ring is found, only symmetrically equivalent rings or rings of size six or less are searched from the chosen start vertex and larger envelope rings are removed. The algorithm continues searching paths by the depth-first search algorithm from each start vertex until all possible paths are removed by the reasons listed above. If there is no cycle discovered, then let $k := k + 1$ and the second stage is repeated.

3. A new start vertex is chosen and the second stage is repeated. The iteration stops if the nullity number of rings is found or there are no more vertices left.

Zamora [130] developed an algorithm that uses basic rules to attain three types of ring systems with respect to the SSSR. This algorithm starts with choosing a starting vertex of high ring connectivity. Depth-first search is used to generate a path. If this path returns to the start vertex, a smallest ring is found. Zamora mentions that a breadth-first search algorithm can be used to find the smallest cycle, or cycles first. This algorithm uses principles that provide for efficient path searching and for cycle analysis.

Zamora's algorithm is used to find three classes of cycle systems [130]:

1. A class where all of the vertices of the ring system are not contained by any subset of the smallest cycles.

2. A class where the subset of the smallest cycles includes all vertices but not all edges.

3. A class where the subset of the smallest cycles includes all vertices and edges.

Roos-Kozel and Jorgensen [108] developed an algorithm that can produce an SSSR and all symmetrically equivalent rings (all rings of class *K* mentioned by Plotkin [96]). They used the fundamental basis method that is used in many algorithms. It starts by deleting edges that do not belong to any ring structure. A spanning tree is constructed and paths are searched by using the breadth-first search. A path searching is started from a chosen vertex of highest connectivity until a cycle is found or the path is forked then it stops to avoid finding any envelope ring. The selection of a starting vertex requires that this vertex does not already exist in two or more cycles and other unvisited vertices are available. This procedure stops if all vertices are used or iterations reach the number of nullity and the set of rings contains all vertices. Each found cycle is checked to avoid having double cycles.

This algorithm can fail to find embedded rings if paths tracing is stopped by a branch vertex [108]. However, this problem is fixed if for each starting vertex, the number of cycles is less than the number of paths divided by 2 plus 1. There are separate steps applied to check for some ring classes such that asteranes, cyclophanes, or porphyrins, if all starting vertices are utilized, but the nullity is not reached. When 2 cycles are found, the existence of 4 starting vertices implies a cyclophane, more than 4 starting vertices implies an asterane, and more than 4 start vertices and more than 2 cycles implies a porphyrin. Hence, finding a branch point stops the path tracing which cause failures, and extra steps are required to solve these problems. Expanding the capabilities of the original path tracing may be a solution instead of involving more procedures.

Lee et. al. [74] present an RP-path method for finding the SSSR by using a Path Included Distance Matrix (PID). This method claims to be quicker than other methods for forming the SSSR. The complexity cost of this method for the worst case is $\mathcal{O}(n^3)$. However, it needs more storage space than other algorithms. Nevertheless, the storage problem is graph dependent, and many graphs do not present any memory problems.

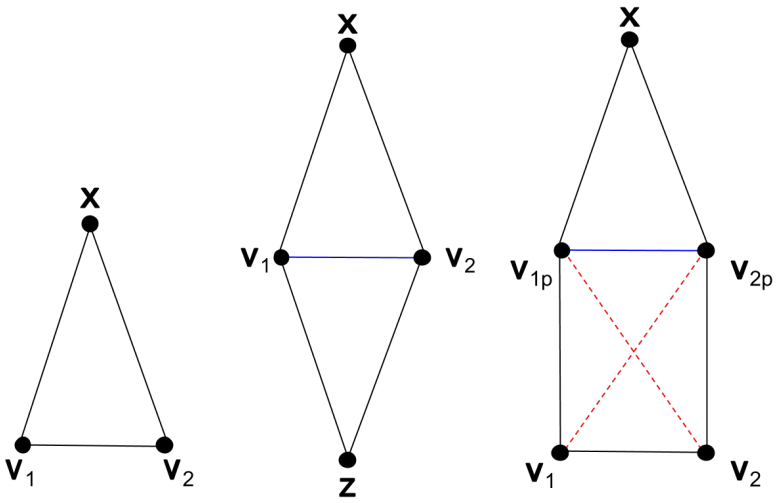
2.2 Small Cycles Algorithm

We develop an algorithm to detect sets of all small cycles of lengths 3,4,5,6,7, and 8. Many algorithms reviewed before are set to reveal cycles of particular requirements. However, our algorithm guarantees the detection of all cycles for any given complex graph. This method not only finds the small cycles but also provides the number of these cycles in each set. A vertex x is chosen to begin the process. Once the algorithm is complete, all small cycles that contain x will be found. All cycles can be found by deleting x from the graph and then choosing a new vertex until all vertices have been checked. The set C_3, all cycles of length 3, is found by searching for the edges that connect every pair of vertices in $N_1(x)$, that is we search $E(G[N_1(x)])$ Figure 9A. Set C_4 is generated by searching through $E(G^c[N_1(x)])$, the graph complement of first neighborhood of x. For every edge $\overline{v_1v_2}$ in $E(G^c[N_1(x)])$, we find the first neighborhood of each end and intersect both neighborhoods with the second neighborhood of x Figure 9B. Similarly, for set C_5 we use $E(G^c[N_1(x)])$ to check for edges connecting vertices in the first neighborhood of x Figure 9C. For each edge $\overline{v_{1p}v_{2p}}$ in $E(G^c[N_1(x)])$, we find the first neighborhood of each end. Then, we use $E(G[N_2(x)])$ to find the edge $\overline{v_1v_2}$, where v_1 is adjacent to v_{1p} and not connected to v_{2p} and v_2 is adjacent to v_{2p} and not connected to v_{1p}. The set C_6 has two cases. The first is that we have a path of length 2 in $E(G[N_2(x)])$, and the second is the isometric cycle case where we have one vertex in $N_3(x)$ Figure 10A and Figure 10B. Then we apply similar checking steps. We do the same for C_7 and C_8 Figure 11A and Figure 22). Also, we can continue in this fashion to find a set of cycles of any length.

Algorithm 2.2.1. FIND ALL SMALL CYCLES OF LENGTHS 3,4,5,6,7, AND 8

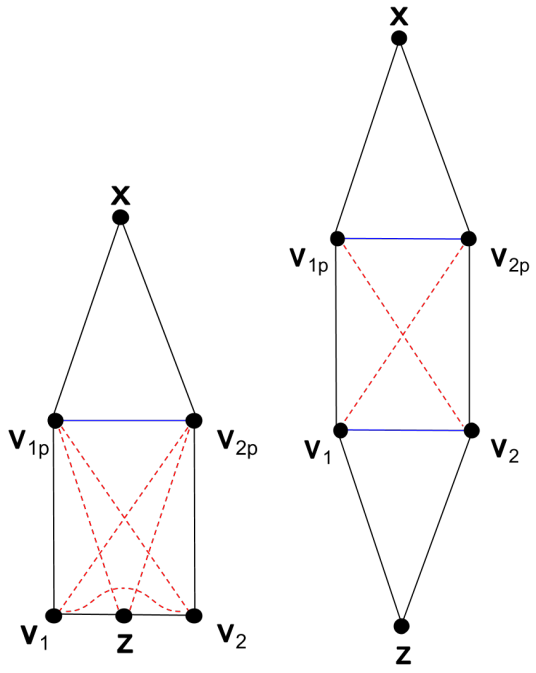
Input: A connected graph G and a given vertex x.

Output: The sets of all cycles of lengths 3,4,5,6,7, and 8 containing the given vertex x.



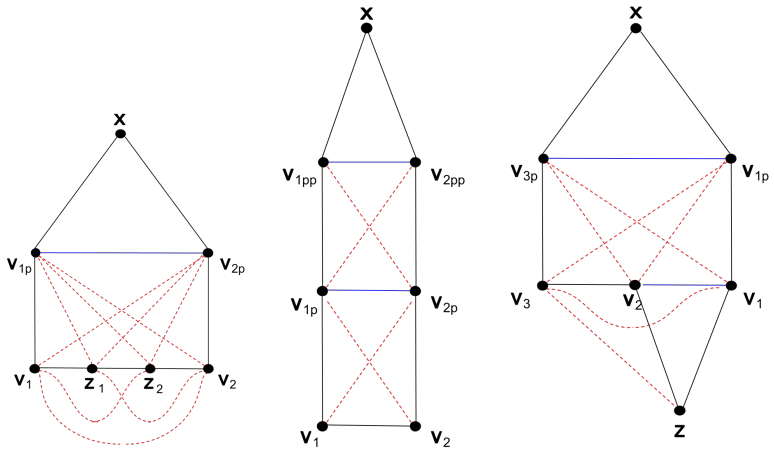
(A) C_3 (B) C_4 (C) C_5

Figure 9: Small Cycles of Length 3,4 and 5.



(A) Case 1 (B) Case 2

Figure 10: Small Cycles of Length 6 Cases.



(A) Case 1 (B) Case 2 (C) Case 3

Figure 11: Small Cycles of Length 7 Cases.

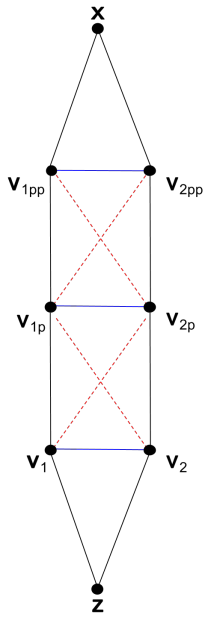


Figure 12: Small Cycles of Length 8 (Isometric Cycles Case)

Step 0. Set $C_i \leftarrow \emptyset$, for $i = 3,4,5,6,7$, and 8.

Step 1. Find $N_1(x)$.

Step 2. Find $E(G[N_1])$ and $E(G^c[N_1])$.

Step 3. For every edge $\overline{v_1v_2} \in E(G[N_1(x)])$

1. $C_3 = C_3 \cup \{xv_1v_2x\}$

Return C_3.

Step 4. Find $N_2(x) = \{u | u \in N_1(v_i) \setminus N_1(x) \setminus N_0(x)$, for every $v_i \in N_1(x)\}$

Step 5. For every edge $\overline{v_1v_2} \in E(G^c[N_1(x)])$

1. Find $z \in (N_1(v_1) \cap N_1(v_2) \cap N_2(x))$,
2. Then $C_4 \leftarrow C_4 \cup \{xv_1zv_2x\}$.

Return C_4.

Step 6. Find $E(G[N_2(x)])$ and $E(G^c[N_2])$.

Step 7. For every edge $\overline{v_1v_2} \in E(G[N_2(x)])$

1. Let $N'(v_1) = N_1(v_1) \cap N_1(x)$ and $N'(v_2) = N_1(v_2) \cap N_1(x)$
2. For $\overline{v_{1p}v_{2p}} \in E(G^c[N_1])$
   (a) If $v_{1p} \in N'(v_1) \setminus N'(v_2)$ and $v_{2p} \in N'(v_2) \setminus N'(v_1)$, then $C_5 \leftarrow C_5 \cup \{xv_{1p}v_1v_2v_{2p}x\}$.

Return C_5.

Step 8. Find $N_3(x) = \{u | u \in N_1(v_i) \setminus N_2(x) \setminus N_1(x)$, for every $v_i \in N_2(x)\}$

Two cases to find all cycles of length 6.

Case I: 3 vertices in $N_2(x)$.

Step 9. For every edge $\overline{v_1v_2} \in E(G^c[N_2(x)])$

1. Let $N'(v_1) = (N_1(v_1) \cap N_1(x)) \setminus N(v_2) \setminus N_2(x)$, $N'(v_2) = (N_1(v_2) \cap N_1(x)) \setminus N(v_1) \setminus N_2(x)$, and $Z = N_2(x) \cap N_1(v_1) \cap N_1(v_2)$.

2. For $\overline{v_{1p}v_{2p}} \in E(G^c[N_1])$

   - For $z \in Z$

     – If $v_{1p} \in N'(v_1) \setminus N(z)$ and $v_{2p} \in N'(v_2) \setminus N(z)$ then, $C_6 \leftarrow C_6 \cup \{xv_{1p}v_1zv_2v_{2p}x\}$.

Case II: One vertex in $N_3(x)$-(isometric cycles).

Step 10. For every edge $\overline{v_1v_2} \in E(G^c[N_2(x)])$

1. Let $N'(v_1) = (N_1(v_1) \cap N_1(x)) \setminus N(v_2) \setminus N_2(x)$, $N'(v_2) = (N_1(v_2) \cap N_1(x)) \setminus N(v_1) \setminus N_2(x)$, and $Z = N_3(x) \cap N_1(v_1) \cap N_1(v_2)$.

2. For $\overline{v_{1p}v_{2p}} \in E(G^c[N_1])$

   - For $z \in Z$

     – If $v_{1p} \in N'(v_1)$ and $v_{2p} \in N'(v_2)$ then, $C_6 \leftarrow C_6 \cup \{xv_{1p}v_1zv_2v_{2p}x\}$.

Return C_6.

Three cases to find all cycles of length 7.

Case I: 4 vertices in $N_2(x)$.

Step 11. For every edge $\overline{v_1v_2} \in E(G^c[N_2(x)])$

1. Let $N'(v_1) = (N_1(v_1) \cap N_1(x)) \setminus N(v_2) \setminus N_2(x)$, $N'(v_2) = (N_1(v_2) \cap N_1(x)) \setminus N(v_1) \setminus N_2(x)$, $N(z_1) = N_2(x) \cap N_1(v_1) \setminus N_1(v_2)$, and $N(z_2) = N_2(x) \cap N_1(v_2) \setminus N_1(v_1)$.

2. For $\overline{v_{1p}v_{2p}} \in E(G^c[N_1])$

   - For $\overline{z_1z_2} \in E(G[N_2])$

     – If $v_{1p} \in N'(v_1) \setminus N(z_1) \setminus N(z_2)$ and $v_{2p} \in N'(v_2) \setminus N(z_1) \setminus N(z_2)$. then, $C_7 \leftarrow C_7 \cup \{xv_{1p}v_1z_1z_2v_2v_{2p}x\}$.

Step 12. Find $E(G[N_3(x)])$ and $E(G^c[N_3])$

Case II: An edge (2 vertices) in $N_3(x)$-(isometric cycle).

Step 13. For every edge $\overline{v_1v_2} \in E(G[N_3(x)])$

1. Let $N'(v_1) = (N_1(v_1) \cap N_2(x)) \setminus N(v_2)$, and $N'(v_2) = (N_1(v_2) \cap N_2(x)) \setminus N(v_1)$.

2. For $\overline{v_{1p}v_{2p}} \in E(G^c[N_2])$

   - If $v_{1p} \in N'(v_1)$ and $v_{2p} \in N'(v_2)$, Let $N'(v_{1p}) = (N_1(v_{1p}) \cap N_1(x)) \setminus N(v_{2p})$, and $N'(v_{2p}) = (N_1(v_{2p}) \cap N_1(x)) \setminus N(v_{1p})$.

     – For $\overline{v_{1pp}v_{2pp}} \in E(G^c[N_1])$.

       * If $v_{1pp} \in N'(v_{1p})$ and $v_{2pp} \in N'(v_{2p})$, then $C_7 \leftarrow C_7 \cup \{xv_{1pp}v_{1p}v_1v_2v_{2p}v_{2pp}x\}$.

Case III: An edge (2 vertices) in $N_2(x)$ and A vertex in $N_3(x)$.

Step 14. For every set of vertices v_1,v_2, and v_3 such that $\overline{v_1v_2}, \overline{v_1v_3} \in E(G^c[N_2(x)])$,and $\overline{v_2v_3} \in E(G[N_2(x)])$

1. Let $N'(v_1) = (N_1(v_1) \cap N_1(x)) \setminus N(v_2) \setminus N(v_3) \setminus N_3(x)$, $N'(v_3) = (N_1(v_3) \cap N_1(x)) \setminus N(v_1) \setminus N(v_2) \setminus N_3(x)$, and $Z = (N_3(x) \cap N_1(v_1) \cap N_1(v_2)) \setminus N(v_3)$.

2. For $\overline{v_{1p}v_{3p}} \in E(G^c[N_1])$

   - For $z \in Z$

     – If $v_{1p} \in N'(v_1)$ and $v_{3p} \in N'(v_3)$ then, $C_7 \leftarrow C_7 \cup \{xv_{1p}v_1zv_2v_3v_{3p}x\}$.

Return C_7.

Step 15. Find $N_4(x) = \{u | u \in N_1(v_i) \setminus N_3(x) \setminus N_2(x) \setminus N_1(x)$, for every $v_i \in N_3(x)\}$

The isometric case for cycles of length 8.

Case I: One vertex in $N_4(x)$.

Step 16. For every edge $\overline{v_1v_2} \in E(G^c[N_3(x)])$

1. Let $N'(v_1) = (N_1(v_1) \cap N_2(x)) \setminus N(v_2)$, $N'(v_2) = (N_1(v_2) \cap N_2(x)) \setminus N(v_1)$, and $Z = N_4(x) \cap N_1(v_1) \cap N_1(v_2)$.

2. For $\overline{v_{1p}v_{2p}} \in E(G^c[N_2])$

   (a) Let $N'(v_{1p}) = (N_1(v_1p) \cap N_1(x)) \setminus N(v_2p) \setminus N_2(x), N'(v_{2p}) = (N_1(v_2p) \cap N_1(x)) \setminus N(v_1p) \setminus N_2(x)$.

   (b) For $\overline{v_{1pp}v_{2pp}} \in E(G^c[N_1])$

   - For $z \in Z$
     - If $v_{1pp} \in N'(v_1p)$ and $v_{2pp} \in N'(v_2p)$ then, $C_8 \leftarrow C_8 \cup \{xv_{1pp}v_{1p}v_1zv_2v_{2p}v_{2pp}x\}$.

   Return C_8.

2.2.1 The Computational Complexity

Theorem 2.2.1. *Let G be a connected graph. The computational complexity of the small cycles algorithms of lengths 3-8 is* $\mathcal{O}(m^3n + n^2)$ *where* $m = |E(G)|$ *and* $n = |V(G)|$.

Proof: The time complexity of step 0 is $\mathcal{O}(1)$, finding $N_1(x)$ is $\mathcal{O}(n)$, and the complexity for finding $E(G[N_i(x)])$, $E(G^c[N_i(x)])$, and $N_i(x)$ for $i = 2,3,4,...$ is $\mathcal{O}(n^2)$. One or more of these sets is constructed to search for each cycle set, but this construction can take place before the search. Therefore, there is an additional complexity of $\mathcal{O}(n^2)$ for each cycle set.

To find all cycles of length 3 by searching the edge set $E(G[N_1(x)])$ is $\mathcal{O}(m)$. With the additive complexity explained previously we get $\mathcal{O}(m + n^2)$.

To find all cycles of length 4, we search for appropriate combinations of vertices in $N_2(x)$ and edges in $G^c[N_1(x)]$, giving a complexity of $\mathcal{O}(mn+n^2)$ total.

To find all cycles of length 5, we search for appropriate combinations of edges in $G^c[N_1(x)]$ and edges in $G[N_2(x)]$, giving $\mathcal{O}(m^2+n^2)$ total.

Cycles of length 6 have 2 cases. In the first case we search for appropriate combinations of edges in $G^c[N_1(x)]$, edges in $G^c[N_2(x)]$, and vertices in $N_2(x)$, giving $\mathcal{O}(m^2n+n^2)$ total. In the second case we search for appropriate combinations of edges in $G^c[N_1(x)]$, edges in $G^c[N_2(x)]$, and vertices in $N_3(x)$, giving $\mathcal{O}(m^2n+n^2)$ total.

Cycles of length 7 have 3 cases. In the first case we search for appropriate combinations of edges in $G^c[N_1(x)]$, edges in $G^c[N_2(x)]$, and edges in $G[N_2(x)]$, giving $\mathcal{O}(m^3+n^2)$ total. In the second case we search for appropriate combinations of edges in $G^c[N_1(x)]$, edges in $G^c[N_2(x)]$, and edges in $G[N_3(x)]$, giving $\mathcal{O}(m^3+n^2)$ total. In the third case we search for appropriate combinations of edges in $G^c[N_1(x)]$, 2 edges in $G^c[N_2(x)]$, edges in $G[N_2(x)]$, and vertices in $N_3(x)$, giving $\mathcal{O}(m^3n+n^2)$ total.

Cycles of length 8 have only one case because we are only interested in isometric cycles of this length. For these cycles we search for appropriate combinations of edges in $G^c[N_1(x)]$, edges in $G^c[N_2(x)]$, edges in $G^c[N_3(x)]$, and vertices in $N_4(x)$, giving $\mathcal{O}(m^3n+n^2)$ total.

2.2.2 Implementations and Results

The algorithm is written in C++ (see Figure 14 shows the procedures). Our programs contain about a thousand of lines of codes. All codes are performed and executed on Intel Core i7-6820HQ CPU 2.7 GHz, 16 GB RAM, Windows 7 Enterprise 64-bit Operating System. We run our program for a set of actual molecular structures listed in Table 2.

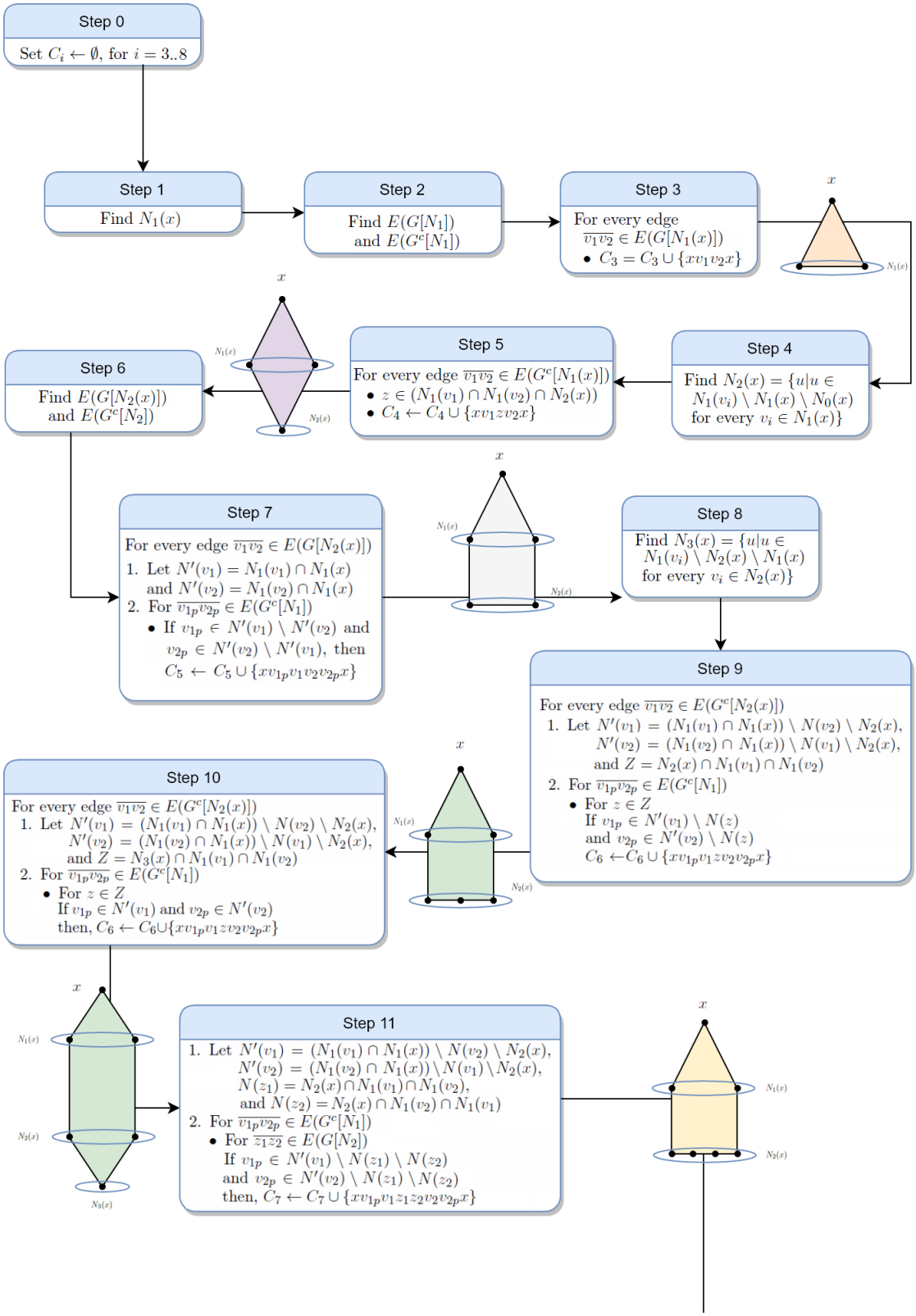


Figure 13: Flow Chart for Small Cycles Algorithm part 1.

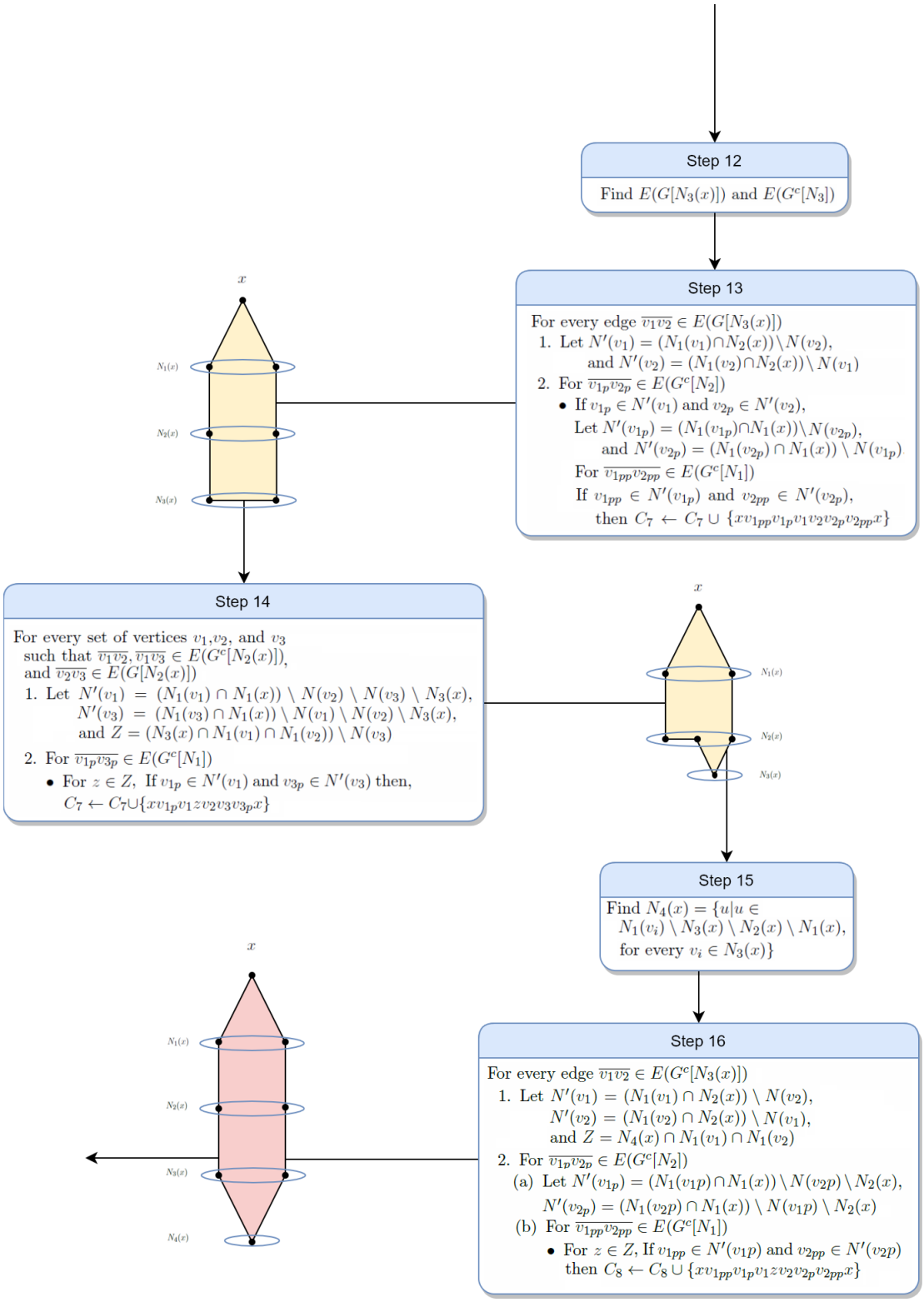


Figure 14: Flow Chart for Small Cycles Algorithm part 2.

These data are for chemical compounds. We used CCDC identifier codes from the Cambridge Structural Database (CSD) and Protein Data Bank (PDB) to represent each molecule, see Table 1.

Table 1: Molecular Structure Examples From CSD and PDB

CSD code	Chemical Name	Reference
1HPH	a protein structure (a structure of human parathyroid hormone)	[87]
ABECAL	9-(4-biphenyl)cyclopenta[a]phenalene	[113]
BOXGAW01	anthra(2,1,9,8-hijkl)benzo(de)naphtho(2,1,8,7-stuv)pentacene	[86]
CORONE	coronene	[41]
AMCOCB10	methyltriethylammonium 8,8'-oxido-3,3'commo-bis(3-cobalta-1,2-dicarba-closo-dodecaborate)	[95]
CIMCAE	octadecachloro-C_{48}graphene	[117]
GEBGEZ	bis(tetramethylammonium) octadecadiborate(20)	[43]
GUCFOA	tetracosachloro-C_{76}fullerene	[61]
GUTCUT	(1212)pentamantane	[25]
JALXEZ	1,10,11,20-tetrakis(Benzyloxy)nonacyclo(12.6.0.02,6.04,11.05,9.07,20.010,17.012,16.015,19)eicosane	[83]
LOPYOG	11,14-Di-t-butyl-7,8,9,16,17,18-hexamethoxy-1,3,4,6-tetraazahexabenzo[bc,ef,hi,kl,no,qr]coronene	[123]
NIHZAF	(1a,8a,9a,13a,14a,15a,16a,17a,18a,19a,20a,20a,24a,25a,26a)-endo-2,endo-7-Dibromo-11,22-dioxadecacyclo-(13.9.2.03,19.04,25.05,17.06,14.08,16.09,13.018,26.020,24)-hexaeicosane	[77]
OLADOU	tridecacyclo(12.10.0.01,18.02,6.02,10.03,13.04,11.05,9.010,17.012,16.014,21.015,19.020,24)tetracosa-7,22-diene	[16]
SELMON	t-Butyl 3-fluoro-4-hydroxy-1-((4-methylphenyl)sulfonyl)-2-phenyl-1,2,3,4-tetrahydroquinoline-3-carboxylate	[75]
UJISAH	dodecacyclo(12.12.0.02,11.03,16.05,18.06,11.07,20.09,22.010,2 5.012,17.012,21.013,24)hexacosane	[25]
ULOXID	3,6-bis(7b-Fluoradenyl)-N-hexylcarbazole	[107]
ULOXUP	4,4'-bis((7b-Fluoradenyl)phenyl)aniline	[107]
WUCHIN	1,2,3,4,5,7,8,9,10,11,12,13,15,16-tetradecakis(phenylsulfanyl)thieno[2''',3''',4''',5''':4'',5'']phenanthro[1'',10'',9'',8'':5',6',7',8']piceno[1',14',13',12':4,5,6,7,8,9]perylo[1,12-bcd]thiophene	[116]
YOFCUR	benzo(1,2,3-bc:4,5,6-b',c')dicoronene	[51]

Also, small cycle algorithm has been implemented on many fullerene graphs and cata-condensed benzenoid structures listed and explained in detail in the next sections.

The adjacency matrix G for each graph is constructed as where matrix entry g_{ij} is defined as

$$g_{i,j} = \begin{cases} 1 & \text{if } \exists \text{ an edge between vertex } i \text{ and vertex } j, \text{ that is edge } (i,j) \\ 0 & \text{otherwise.} \end{cases}$$

Note that an adjacency matrix is $n \times n$ where n is the number of vertices.

On completion our C++ program provides the results which are summarized in Table 2 output as sets of all smallest small cycles of lengths 3-8 and the total number of rings for each set. Table.1 also shows the CPU time for the algorithm for each data set. If we look at the smallest example we have in Figure 15, it has $|n| = 36$ and $|m| = 42$. The outputs of this structure is shown below:

```
7 small cycle(s) of length 6:
 1  2  4  6 17 16  1
 1 14 12 11 18 16  1
 6  7  9 29 36 17  6
11 19 20 21 35 18 11
16 18 35 34 36 17 16
21 22 23 24 34 35 21
24 32 30 29 36 34 24
```

It takes 0.001 second to find all smallest small cycles.

Table 2: Implementations of Small Cycles Algorithm and CPU Times (C++)

Example	n	m	C_3	C_4	C_5	C_6	C_7	C_8	CPU(s)
1HPH	626	631	0	0	4	2	0	0	0.037
ABECAL	46	51	0	0	1	5	0	0	0.001
AMCOCB10	65	102	40	1	24	80	0	0	0.005
BOXGAW01	56	66	0	0	0	11	0	0	0.001
CIMCAE	66	81	0	0	0	16	0	0	0.001
CORONE	36	42	0	0	0	7	0	0	0.001
GEBGEZ	51	71	22	0	6	4	2	0	0.002
GUCFOA	100	138	0	0	12	28	0	0	0.004
GUTCUT	58	68	0	0	0	16	0	15	0.002
JALXEZ	100	112	0	0	8	4	0	2	0.004
LOPYOG	104	116	0	0	0	13	0	0	0.003
NIHZAF	56	65	0	0	2	15	0	6	0.002
OLADOU	44	56	0	0	16	8	4	1	0.002
SELMON	63	66	0	0	0	4	0	0	0.002
UJISAH	56	67	0	0	0	18	0	9	0.003
ULOXID	98	110	0	0	5	8	0	0	0.002
ULOXUP	92	104	0	0	4	9	0	0	0.002
WUCHIN	212	240	0	0	2	27	0	0	0.006
YOFCUR	68	82	0	0	0	15	0	0	0.002

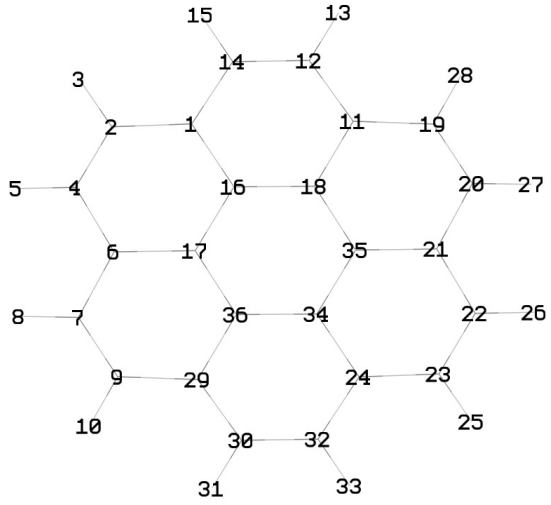


Figure 15: Coronene Molecule (CORONE) [41]

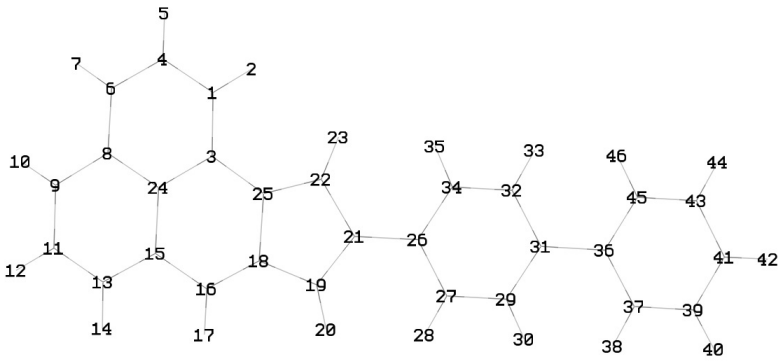


Figure 16: 9-(4-Biphenyl)Cyclopenta[a]Phenalene (ABECAL) [113]

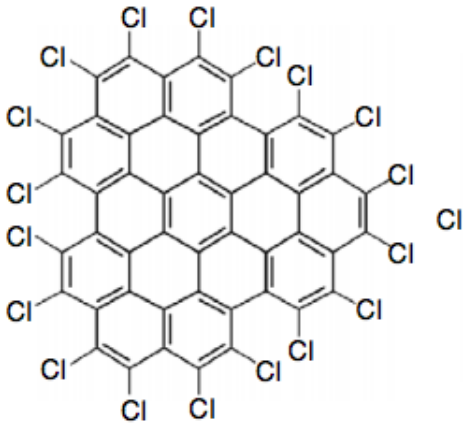


Figure 17: Chlorinated Nanographenes Molecule (CIMCAE) [117]

2.2.3 Conclusions

We present a polynomial time algorithm that finds all small cycles of length 3 to 8 for a given vertex and the number of cycles in each set. Our algorithm covers all cycles in any complex molecule graph. The breadth first search algorithm is used to walk through an adjacency matrix and trace paths. Compared with other algorithms, our method is efficient, simple and easy to implement for complex undirected graphs and molecule structures. Many algorithms don't discover all small cycles in some graphs for instance in the cube graph

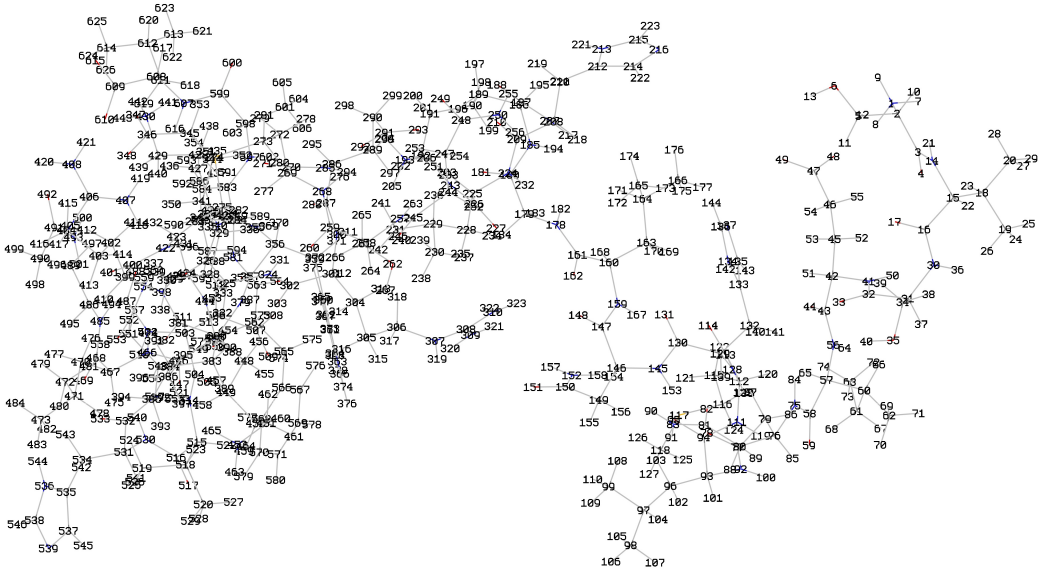


Figure 18: A Protein Structure of A Structure of Human Parathyroid Hormone 1HPH [87]

(Figure.21), the cycle 12781 won't be found. However, this algorithm guarantees to detect all these small cycles. This algorithm has several sections which can be applied at once or separately on a given graph to attain a set of all cycles of specific length. This algorithm may be useful in different fields of science.

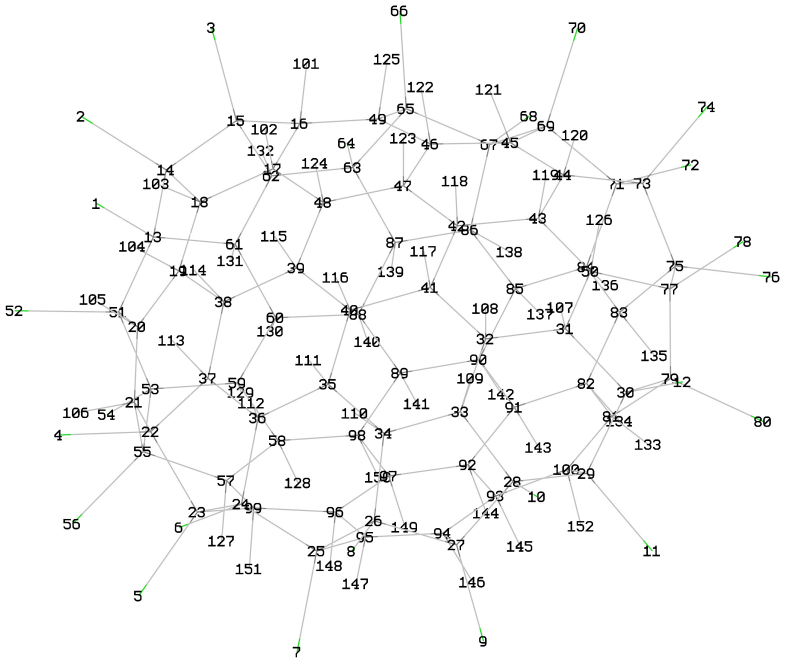


Figure 19: Tetracosachloro-C_{76}Fullerene (GUCFOA) [61]

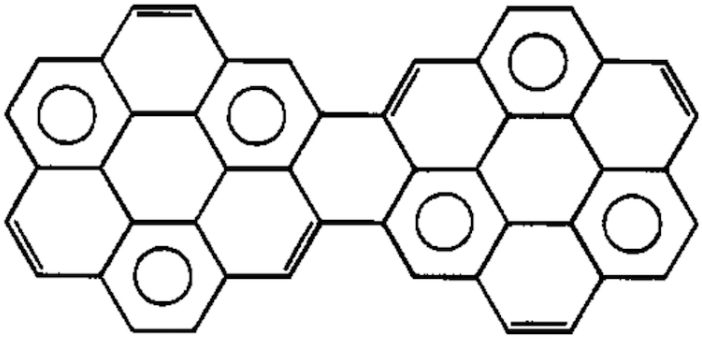


Figure 20: Benzo[1,2,3-bc:4,5,6-b'c'] Dicoronene Molecule (YOFCUR) [51]

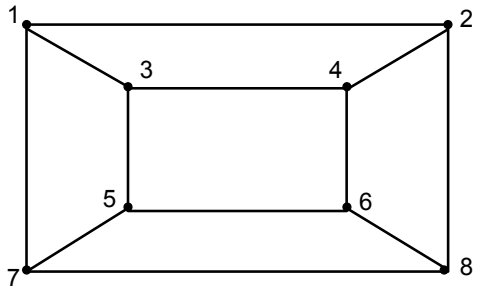


Figure 21: A Cube Graph

2.3 Shortest Paths Algorithm

We also developed another algorithm to find all shortest paths joining given vertices u and v and any given graph G. For any two vertices $u, v \in V(P)$ and $uv \in E(G)$ said P is a shortest path if $\text{dist}_G(u,v) = \text{dist}_P(u,v)$. To determine the shortest cycles, we first omit an edge between the two adjacent vertices u and v. The proposed fixed end algorithm then finds all shortest paths between the two given vertices. Evaluating these paths can determine a set of cycles, which can then be shown on a graph. In order to show this representation, we propose that the primary function of this research is two-fold; to use two algorithms in order to find all possible shortest paths in a graph and to compute all shortest cycles of a graph that is containing the given edge.

For each $uv \in E(G)$, we delete the edge that is joining them. Then, the vertices u and v are chosen to begin the process where u is the start vertex and v is the destination vertex. BFS algorithm is used to trace paths if the destination vertex is found. If v is not found then, we pick another vertex and continue tracing its neighborhood. Otherwise, all found paths are saved and we delete all paths that contain the same vertex, which is the vertex adjacent to (or before) the destination vertex v.

Once the algorithm is complete all shortest paths that are joining u and v will be found. All cycles can be found by deleting u and v from the graph and then choosing new vertices until all vertices have been checked. This algorithm guarantees that there are no two paths sharing a vertex or an edge. We have proved this in this research.

In the following sections we give a mathematical description of this problem setting, including definitions. We will then provide proofs and theorems for the analysis. Next we outline our algorithm and give examples of this work.

2.3.1 Properties

Let u and v be two vertices of G such that $uv \in E(G)$. A path P joining u and v is *semi-isometric* if for any two vertices $x, y \in V(P)$ such that $\text{dist}_P(u,x) < \text{dist}_P(u,y)$,

$$\text{dist}_G(x,y) = \min\{\text{dist}_P(x,y), \text{dist}_P(x,u) + \text{dist}_P(y,v) + 1\}.$$

Proposition 2.3.1. *Let G be a graph and C be a cycle. Assume that uv is an edge of C. Then C is a isometric cycle if and only if $C - uv$ is a semi-isometric path joining u and v.*

Proof. Let $P = C - uv$. Let $x, y \in V(C)$ such that $\text{dist}_P(u,x) < \text{dist}_P(u,y)$. Note that

$$\min\{\text{dist}_P(x,y), \text{dist}_P(x,u) + \text{dist}_P(y,v) + 1\} = \text{dist}_C(x,y).$$

The proposition follows directly from the definitions. □

Let $N_i(u)$ be the i-th neighborhood of u, i.e., $N_i(u) = \{x | \text{dist}_G(u,x) = i\}$. Let G_α be the subgraph induced by all edges contained by at least one semi-isometric path of length α.

Lemma 2.3.2. *Let G be a graph and u, v two vertices of G. Then $G_\alpha \cap G_\beta = \{u, v\}$ and $E[V(G_\alpha), V(G_\beta)] = \emptyset$ for $\alpha \neq \beta$.*

Proof. Let u, v two vertices of G and G_α be a subgraph containing all semi-isometric paths from u to v with α length and G_β be a subgraph containing all semi-isometric paths from u to v with β length.

Assume $\alpha > \beta$. Suppose on the contrary that either $G_\alpha \cap G_\beta - \{u, v\} \neq \emptyset$ or $E[V(G_\alpha), V(G_\beta)] \neq \emptyset$.

First suppose $G_\alpha \cap G_\beta \neq \emptyset$. Let $w \in G_\alpha \cap G_\beta - \{u, v\}$.

Note that

$$\max\{\text{dist}_{P_\alpha}(u,w), \text{dist}_{P_\alpha}(w,v)\} > \max\{\text{dist}_{P_\beta}(u,w), \text{dist}_{P_\beta}(w,v)\}$$

because $\alpha > \beta$. So $\text{dist}_{P_\alpha}(u,w) + \text{dist}_{P_\alpha}(w,v) > \text{dist}_{P_\beta}(u,w) + \text{dist}_{P_\beta}(w,v)$
that means $\text{dist}_G(u,w) + \text{dist}_G(w,v) = \text{dist}_{P_\beta}(u,w) + \text{dist}_{P_\beta}(w,v)$ then $\text{dist}_{P_\beta}(u,v)$ is a semi-isometric path and $\text{dist}_{P_\alpha}(u,v)$ is not a semi-isometric path a contradiction and therefore $G_\alpha \cap G_\beta = \{u,v\}$

Now assume $xy \in E[V(G_\alpha), V(G_\beta)]$ and $x \in V(G_\alpha)$ and $y \in V(G_\beta)$. Since $\alpha \neq \beta$, $x \in N_i(z)$ and $y \in N_j(z)$ for some $z \in \{u,v\}$ and $i \neq j$.
Without loss of generality,

assume $z = u$ and $i > j$. Then $i = j+1$ because x is in the first neighbor of y.
Then $\text{dist}_{P_\alpha}(y,v) > \text{dist}_{P_\beta}(y,v)$ because $\alpha > \beta$.
From the semi-isometric path definition $\text{dist}_G(y,v) = \text{dist}_{P_\beta}(y,v)$ and $\text{dist}_G(y,v) = \text{dist}_{P_\alpha}(y,v)$
That means $\text{dist}_G(y,v) > \text{dist}_G(y,v)$ is a contradiction. Therefore, y is not on a semi-isometric path of length α.

□

Algorithm 2.3.1. FIND ALL SHORTEST PATHS WITH FIXED ENDS ALGORITHM

Input: A connected graph G and two vertices u and v.

Output: The set of all shortest paths joining u and v.

Step 0. Set $\mathcal{P}_0 = \{u\}$, $N_0(u) = \{u\}$, $Q = N_0(u)$, $\mathcal{P} = \emptyset$ and $i = 0$.

Step 1. If $i < n$, then set $N_{i+1}(u) = \emptyset$, and $\mathcal{P}_{i+1} = \emptyset$, go to Step 2. Otherwise, return $\mathcal{P}$.

Step 2. If $N_i(u) \neq \emptyset$, choose a vertex $x \in N_i(u)$ and go to Step 3. Otherwise, go to Step 4.

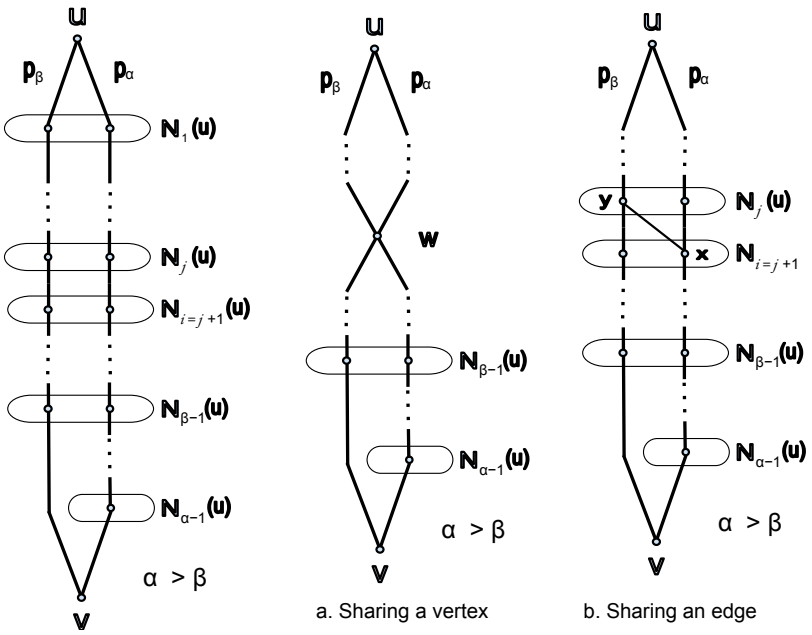


Figure 22: Lemma Example

Step 3. Set $Y = \{y | y \in N_1(x) \backslash Q\}$ and $N_{i+1}(u) \leftarrow N_{i+1}(u) \cup Y$, $\mathcal{P}_{i+1} = \mathcal{P}_{i+1} \cup \{P + xy | P \in \mathcal{P}_i$ with end u and $x\}$. Set $N_i(u) = N_i(u) \backslash \{x\}$.

1. If $v \notin Y$, then return to Step 2.

2. Otherwise, add $Q \leftarrow Q \cup (Y \backslash v)$, $Y = \{v\}$. set $\mathcal{P} \leftarrow \mathcal{P} \cup \mathcal{P}_{i+1}$ and delete all paths from $\mathcal{P}_i$ with x as an end vertex and return to Step 2.

Step 4. Update $Q \leftarrow Q \cup N_{i+1}(u)$ and set $i \leftarrow i+1$ and return to Step 1.

Theorem 2.3.3. *Let G be a graph. The computational complexity of the shortest paths algorithm is $\mathcal{O}(n+m)$ where $m = |E(G)|$ and $n = |V(G)|$.*

Proof. The time complexity of steps 1, 2, and 4 are n since we are going to explore all vertices in each level. In line 3, we find all adjacent vertices of a vertex x, that means we

go through each edge. Then, this step needs to be done in m time. The complexity time is $\mathcal{O}(3n+m)$. Therefore, the final complexity time of this algorithm is $\mathcal{O}(n+m)$ where $n = |V(G)|$ and $m = |E(G)|$ because we explore all vertices and edges.

□

Theorem 2.3.4. *Algorithm finds all shortest paths from u to v.*

Proof. We want to prove that P is the set of all shortest paths of G. That means we prove that

$\forall p \in P_{uv}$, p is the shortest path joining u and v.

Let q be a shortest path from u to v, then $q \in P$.

Frist, $\forall p \in P_{uv}$, p is the shortest path joining u and v.

let $p = uw_1w_2...w_i...w_j...w_kv$.Take any two vertices w_i and w_j where $i < j$ and $w_i \in N_i(u)$ & $w_j \in N_j(u)$

$\text{dist}_G(w_i, w_j) \leq \text{dist}_p(w_i, w_j)$

Assume that $\text{dist}_G(w_i, w_j) < \text{dist}_p(w_i, w_j)$ $(*)$

$\text{dist}_G(u, w_i) = i$,

$\text{dist}_G(u, w_j) = j$,

$\text{dist}_p(w_i, w_j) = j - i$

$\text{dist}_G(w_i, w_j) = \alpha$

from $(*)$ $\alpha < j - i$

$$\text{dist}_G(u, w_j) \leq \text{dist}_G(u, w_i) + \text{dist}_G(w_i, w_j)$$

$$\leq i + \alpha$$

$$< i + (j - i)$$

$$j < j$$

Contradicts our assumption

Now, we need to prove that if q is a shortest path from u to v, then $q \in P$.

let $q = uw_1w_2...w_kv$, and $w_i \in N_i(u)$ for $i = 1,2,...n$

w_i is not adjacent to v for $1 \leq i \leq k-1$. Suppose $w_i \in N_\alpha(u)$ Consider $i < \alpha$ and $i > \alpha$ by the definition $N_\alpha(u)$, $\text{dist}_G(u,x) = \alpha$ for any $x \in N_\alpha(u)$.

That means $q \in P$.

□

2.3.2 Implementations and Results

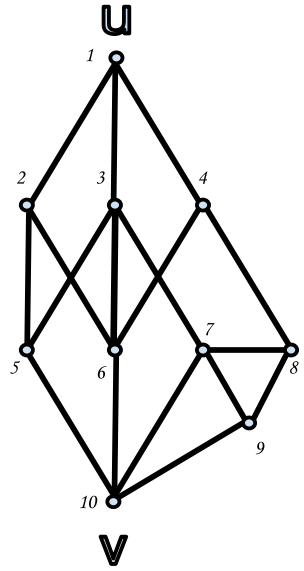


Figure 23: Shortest Paths Example

The graph in Figure 23 is used as an example for the shortest paths algorithm.

On completion our FORTRAN 90 program provides the following output as the set of all shortest paths of G. All codes are performed and executed on a MacBook Pro running a 2.5 GHz Intel Core i5 processor, with 16 GB of 1600 MHz DDR3 memory, and with the macOS Sierra 10.12.3 operating system.

```
 P =
      1      2      5     10
      1      3      5     10
      1      2      6     10
      1      3      6     10
      1      4      6     10
      1      3      7     10
```

Also, we run our algorithm for several actual molecular structures. The meanings of these chemical compounds are explained in detail in the previous section. The results, which are summarized in Table 3, output the total number of shortest cycles that join two vertices u and v. Also, shortest paths can be printed. The CPU time (per second) is also shown for the algorithm for each data.

Table 3: Implementations of Shortest Paths Algorithm and CPU Times (FORTRAN 90)

Example	n	m	number of cycles	CPU(s)
1HPH	626	631	6	26.9
ABECAL	46	51	6	0.1
BOXGAW01	56	66	11	0.2
CORONE	36	42	7	0.07
AMCOCB10	65	102	41	0.5
CIMCAE	66	81	18	0.3
GEBGEZ	51	71	29	0.2
GUCFOA	100	138	39	0.9
GUTCUT	58	68	16	0.2
JALXEZ	100	112	13	0.6
LOPYOG	104	116	13	0.7
NIHZAF	56	65	17	0.2
OLADOU	44	56	16	0.2
SELMON	63	66	4	0.3
UJISAH	56	67	19	0.3
ULOXID	98	110	13	0.6
ULOXUP	92	104	13	0.6
WUCHIN	212	240	30	3.0
YOFCUR	68	82	15	0.4

2.3.3 Conclusions

The Shortest Paths algorithm is easy to implement and can be used for any chemical graph. The concept of this algorithm is similar to the BFS algorithm. By this algorithm, a

set of shortest paths is generated from given two vertices $u, v \in V(G)$.

The distance between any given pair of vertices can be found easily by applying the shortest paths algorithm. We can compute a set of all shortest cycles in a graph containing edge uv by removing this edge and applying the shortest paths algorithm. This strategy is effective only if the graph is sparse. A graph is sparse if it has few edges. However, we have applied this algorithm to different actual molecular structures to show that it works for all graphs.

CHAPTER 3

CLAR STRUCTURES VS FRIES STRUCTURES IN HEXAGONAL SYSTEMS

A *hexagonal system* is a finite 2-connected plane bipartite graph in which every interior face is bounded by a regular hexagon. Hexagonal systems can be used to model molecular skeletons of graphens or benzenoid hydrocarbons such that a vertex represents a carbon atom and an edge models a bond between two adjacent carbon atoms. Many chemical properties of graphenes or benzenoid hydrocarbons can be estimated through the computing some topological indices of hexagonal systems, particularly perfect matchings or Kekulé structures [26, 55]. One of challenge problems in material science and chemistry is to identify the stable molecules among all possible structures through massive computation.

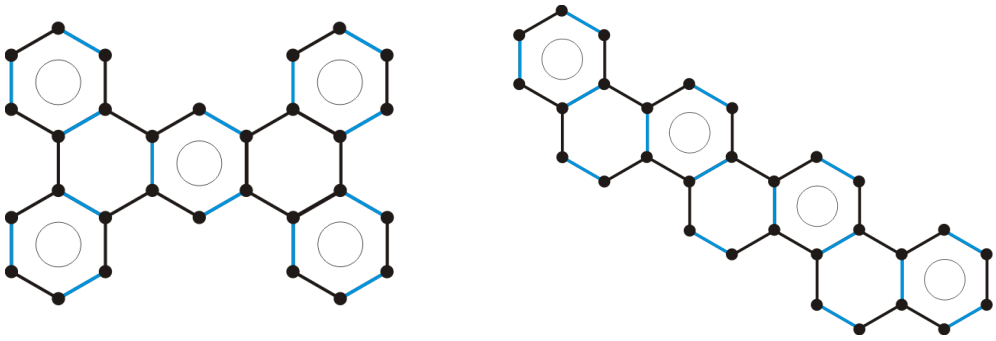
Let H be a hexagonal system. A *perfect matching* M of H is a set of disjoint edges such that every vertex of H is incident with exactly one edge in M. A perfect matching of a hexagonal system is also called a *Kekulé structure* in chemistry literatures. A graph with a perfect matching is called *matchable* and a graph is *matching-covered* if every edge is contained in a perfect matching. A hexagonal system is *catacondensed* if all vertices appear on its boundary. A catacondensed hexagonal system H is matching-covered. The computation of the number of distinct perfect matchings for a given bipartite graph is #P-complete [120]. For planar graphs, the number of perfect matchings can be computed in polynomial time by the Pfaffian orientation method (cf. Chapter 8 in [82]) which provides a polynomial time algorithm to compute the permanent of the adjacency matrix. Li et. al. [76] characterize hexagonal chains maximizing the coefficients sum of permanent polynomial. For a given perfect matching M, a cycle C of H is *M-alternating* (or resonant) if the edges of C alternate between M and $E(H)\backslash M$.

A *resonant set* $\mathcal{S}$ of H is a set of hexagons such that all hexagons are M-alternating for some perfect matching M. A maximum independent resonant set is also called *a Clar formula* of H. The *Fries number* of H is the size of a maximum resonant set, denoted by $\mathtt{fr}(H)$, and the *Clar number* of H is the size of a maximum independent resonant set (or Clar formula), denoted by $\mathtt{cl}(H)$. The Clar number and Fries number of hexagonal systems can be computed by the integer programming [59].

Abeledo and Atkinson [1] show that the integer programming for Clar problem can be relaxed to a linear integer programming. The integer programming can also be used to compute the Clar problem and Fries problem of fullerenes [17, 110]. There is a natural connection between the Clar number and the forcing number of hexagonal systems [134], and this connection could be considered in general for resonant sets and forcing sets (for more details on enumeration of forcing sets, see [133]).

For a given number of vertices, there are many non-isomorphic hexagonal systems, so-called isomers in chemistry. Brinkmann et al. [15] developed an algorithm to construct all non-isomorphic hexagonal systems. By the Clar theory [20] and Randić conjugated circuit model [99], usually, an isomer of a hexagonal system with larger Clar number is more stable, and the same holds for the Fries number. However, the Clar numbers and the Fries numbers of hexagonal systems are not always consistent, even for cata-condensed hexagonal systems. For example, the following two hexagonal systems H_1 in Figure 24A and H_2 in Figure 24B satisfy $\mathtt{cl}(H_1) = 5 > 4 = \mathtt{cl}(H_2) = 4$ but $\mathtt{fr}(H_1) = 6 < 7 = \mathtt{fr}(H_2)$.

A pair of hexagonal systems H_1 and H_2 with the same number of vertices is called a *contra pair* if $\mathtt{cl}(H_1) > \mathtt{cl}(H_2)$ and $\mathtt{fr}(H_1) < \mathtt{fr}(H_2)$. In this work, we characterize the structures of cata-condensed hexagonal systems maximizing Clar number and Fries number among all their isomers, and the contra pairs of cata-condensed hexagonal systems. Our computation results demonstrate correlations between the Fries number and Clar number of cata-condensed systems hexagonal systems. In a contra pair H_1 and H_2, the system with



(A) H_1 (B) H_2

Figure 24: A Contra Pair H_1 and H_2: Blue Edges form a Perfect Matching and Hexagons with Circles form a Clar Formula.

larger Clar number has larger HOMO-LUMO gap, which shows that Clar number is a better stability predictor than Fries number.

The following conjecture shows the connection between Clar structures and Fries structures of hexagonal systems, which is circulated among the mathematical chemistry community [91] and first appears in [54].

Conjecture 3.0.5 ([54])**.** *Let H be a hexagonal system. Then H has a maximum resonant set containing a maximum independent resonant set.*

Graver, Hartung and Souid [54] verified the conjecture for a family of hexagonal systems admitting a special face-coloring. In this project, we confirm the above conjecture for all cata-condensed hexagonal systems.

3.1 Maximizing Fries Number and Clar Number

Let H be a hexagonal system. The *inner dual* H^* of a hexagonal system H is a graph such that the vertex set consists of center points of all hexagons of H and two vertices of H^* are joined by a straight line segment if the corresponding two hexagons sharing an edge. A hexagonal system is *catacondensed* if its inner dual is a tree. A *linear chain* is a hexagonal

system whose inner dual is a straight line segment. Clearly, a linear chain is a catacondensed hexagonal system. A linear hexagon chain of length k consists of k hexagons $h_1, h_2, \ldots, h_k$ such that $h_i \cap h_{i+1}$ is an edge in the opposite position of $h_{i-1} \cap h_i$ (the subscribes are taken modulo k). A linear chain of H is maximal if it is not contained by another linear chain of H. A hexagon of H is called a *kink-hexagon* if it belongs to two maximal linear chains. A hexagon of H is called a *leaf-hexagon* if the corresponding vertex in its inner dual is a leaf.

Observation 3.1.1. *Let H be a linear chain with at least two hexagons. Then* $\mathtt{cl}(H) = 1$ *and* $\mathtt{fr}(H) = 2$.

Let H be a catacondensed hexagonal system and h be a hexagon which is contained by maximal linear chains $L_1, \ldots, L_k$ ($k \leq 3$). Choose one hexagon h_i from L_i for each i with $1 \leq i \leq k$. Note that h_i may be the hexagon h. Then $H - \cup_{i=1}^{k} V(h_i)$ has a perfect matching, which implies the following observation.

Observation 3.1.2. *Let H be a catacondensed hexagonal system and let $\mathbb{S}$ be a set of disjoint hexagons. If $\mathbb{S}$ contains at most one hexagon from each maximal linear chain, then $H - V(\mathbb{S})$ has a perfect matching. In other words, $\mathbb{S}$ is a independent resonant set of H.*

The above observation show that a Clar formula is a maximum set of disjoint hexagons which intersects every maximal linear chain of H. The following is a very important property for maximum independent resonant sets of hexagonal systems.

Theorem 3.1.3 (Zheng and Chen, [85])**.** *Let H be a hexagonal system and $\mathbb{S}$ be a maximum independent resonant set of H. Then $H - V(\mathbb{S})$ has unique perfect matching.*

The following is a very useful technical lemma due to Kotzig [69]. A short proof can also be found in [128].

Lemma 3.1.4 (Kotzig, [69])**.** *Let G be a graph with a unique perfect matching M. Then G has a cut-edge e which belongs to M.*

Theorem 3.1.5. *Let H be a hexagonal system with a perfect matching. Let $\phi(H)$ be the number of hexagons of H and $\tau(H)$ be the total number of vertices not on the boundary of H. Then*

$$1 \leq \mathtt{fr}(H) \leq \phi(H) - \frac{\tau(H)}{6}.$$

Furthermore, for cata-condensed hexagonal systems H, the upper bound holds if and only if every maximal linear chain of H has length 2.

Proof. Let H be a hexagonal system with a perfect matching M. By Lemma 3.1.4, H has another perfect matching $M' \neq M$ since H is 2-connected. Let $\mathcal{S}$ be a maximum independent resonant set. By Theorem 3.1.3, $H - V(\mathcal{S})$ has a unique perfect matching. It follows immediately that $\mathcal{S}$ is not empty. So $\mathtt{fr}(H) \geq \mathtt{cl}(H) \geq |\mathcal{S}| \geq 1$.

Let v be a vertex not on the boundary of H. For a perfect matching M of H, let $uv \in M$. Then the hexagon containing v but not uv is not M-alternating. Note that a hexagon has 6 vertices. Therefore, there are at least $\tau(H)/6$ hexagons which are not M-alternating. So $\mathtt{fr}(H) \leq \phi(H) - \tau(H)/6$ and the upper bound follows. Note that the upper bound is sharp for infinitely many hexagonal systems (see Figure **??**).

Note that, for catacondensed hexagonal system, $\tau(H) = 0$. In the following, we are going to show $\mathtt{fr}(H) = \phi(H)$ if and only if H is a catacondensed hexagonal system in which every maximal linear chain has length at most 2.

First assume that $\mathtt{fr}(H) = \phi(H)$. Then H has a perfect matching M such that all hexagons are M-alternating. If H has a vertex v which is not on the boundary of H, let $uv \in M$ and H has a hexagon h which contains v but not uv. Then h is not M-alternating, a contradiction. Hence all vertices of H are on the boundary of H. So H is catacondensed. If H has a linear chain L with length at least 3, by Observation 2.1, $\mathtt{fr}(L) = 2$ and hence L has a hexagon which is not M-alternating, a contradiction again. So every maximal linear chain of H has length at most 2.

Now, assume that H is catacondensed and every linear chain of H has length at most

2. We prove that $\mathtt{fr}(H) = \phi(H)$ by induction on the number of hexagons. If H has only one hexagon, the result trivially holds. So assume that H has more than one hexagons and let h be a leaf-hexagon of H. Let H' be a subgraph containing all hexagons of H except h. Then H' has no linear chain of length bigger than 2. By inductive hypothesis, H' has a perfect matching M such that all hexagons of H' are M-alternating. Then M contains all edges shared by two hexagons of H. It implies that the edge in $h \cap H'$ belongs to M too. So the edge of $h \cap H' \in M$. Let M_h be the perfect matching of h such that $h \cap H' \in M_h$. Then $M_h \cup M$ is a perfect matching of H such that all hexagons of H are $(M_h \cup M)$-alternating. Therefore, $\mathtt{fr}(H) = \phi(H)$. □

In the following, we consider the Clar number of hexagonal systems.

Theorem 3.1.6. *Let H be a hexagonal system with n vertices and admitting a perfect matching. Then*

$$1 \leq \mathtt{cl}(H) \leq \lfloor \frac{n}{6} \rfloor,$$

and the both bounds are sharp.

Proof. Let H be a hexagonal system with n vertices and a perfect matching. The lower bound follows from $\mathtt{fr}(H) \geq 1$ by Theorem 2.2. The bound is sharp for all linear hexagon chains.

Let $\mathcal{S}$ be a maximum independent resonant set. Since all hexagons in $\mathcal{S}$ are disjoint, $6|\mathcal{S}| \leq |V(H)| = n$. So $\mathtt{cl}(H) = |\mathcal{S}| \leq \lfloor n/6 \rfloor$. There are infinitely many cata-condensed hexagonal systems with $\mathtt{cl}(H) = \lfloor n/6 \rfloor$ (see the following theorem). □

Let L be the infinite hexagon lattice, i.e., a hexagon tiling of the plane. A hexagonal system H is a finite subgraph of the lattice consisting of a cycle together with its interior. Let $\mathcal{F}(L)$ be the set of all hexagonal faces of L. Then L admits a proper 3-face-coloring $c : \mathcal{F}(L) \to \mathbb{Z}_3$, which can be constructed from one hexagon h with color 0: color the

hexagons adjacent to h by 1 and 2 and extend the coloring to all faces. Note that this 3-face-coloring is unique up to permutations of colors. The restriction of the coloring c of faces of L on H is a proper 3-face-coloring of all inner faces of H, called a *canonical 3-coloring* of H. From the 3-face-coloring of L, define a proper 3-edge-coloring $\alpha : E(L) \to \mathbb{Z}_3$ of L such that $\alpha(e) = c(h_1) + c(h_2) \pmod 3$ where e is a common edge of hexagons h_1 and h_2. The restriction of α on H is a proper 3-edge-coloring of H. Note that, if $\alpha^{-1}(i)$ is a perfect matching M of H, then the hexagons of H with color $i+1$ or $i+2$ (mod 3) are M-alternating.

A hexagonal system H is *fully* if $\mathtt{cl}(H) = |V(H)|/6$. The number of vertices of a fully hexagonal system is divisible by 6. The fully hexagonal systems is a very important family which usually have good chemical properties [24]. The following proposition shows a direct connection between the Clar number of a fully hexagonal system and its canonical 3-coloring.

Proposition 3.1.7. *Let H be a fully hexagonal system and let $c : \mathcal{F}(H) \to \mathbb{Z}_3$ be a canonical 3-coloring of H, where $\mathcal{F}(H)$ is the set of all hexagons of H. Then*

$$\max\{|c^{-1}(i)| \,\big|\, i \in \mathbb{Z}_3\} = \mathtt{cl}(H).$$

Proof. Let $\mathcal{S}$ be a maximum independent resonant set of H. Since H is fully, we have $|\mathcal{S}| = n/6$ and $H - V(\mathcal{S}) = \emptyset$. Let $c : \mathcal{F}(H) \to \mathbb{Z}_3$ be the canonical 3-coloring of H.

Claim. *All hexagons of $\mathcal{S}$ are colored by same color.*

Proof of Claim. If not, assume hexagons of $\mathcal{S}$ are colored by at least two colors. Let $\mathcal{S}_1 \subset \mathcal{S}$ be the set of hexagons colored by the same color, say 1, which is not empty. Then $\mathcal{S}\backslash\mathcal{S}_1$ is not empty neither. Let $h_1 \in \mathcal{S}_1$ and $h_2 \in \mathcal{S}\backslash\mathcal{S}_1$ such that the distance of the two vertices corresponding to h_1 and h_2 in the inner dual H^* of H is as small as possible. Since both h_1 and h_2 belong to $\mathcal{S}$, $h_1 \cap h_2 = \emptyset$. Hence the distance between h_1 and h_2 is bigger than 1. Let $h_1 h_{i_1} h_{i_2} \ldots h_{i_k} h_2$ be a shortest path in H^* connecting h_1 and h_2. Then $k \geq 1$, and all

$h_{i_1}, h_{i_2}, \ldots, h_{i_k}$ do not belong to $\mathcal{S}$ by the choice of h_1 and h_2. Since all vertices of h_{i_1} is covered by $\mathcal{S}$, the other vertices not in h_1 are covered by hexagons from $\mathcal{S}$. Since c is the canonical 3-coloring, it follows that all vertices of h_{i_1} are covered by hexagons colored by 1. Therefore, all three disjoint hexagons adjacent to h_{i_1} belong to $\mathcal{S}$ and they are colored by 1. Note that, h_{i_2} (may be h_2) is adjacent to h_{i_1}, and hence is adjacent to at least a hexagon $h \in \mathcal{S}$ colored by 1 which is adjacent to h_{i_1}. So h has a shorter distance to h_2 as $hh_{i_2} \ldots h_{i_k} h_2$ is shorter than $h_1 h_{i_1} h_{i_2} \ldots h_{i_k} h_2$, a contradiction to the choice of h_1 and h_2. This completes the proof of Claim.

By the Claim, $\mathcal{S} \subseteq c^{-1}(i)$ for some $i \in \mathbb{Z}_0$. Note that $c^{-1}(i)$ is an independent hexagon set and hence $|c^{-1}(i)| \leq n/6$. It follows that $\mathcal{S} = c^{-1}(i)$ and this completes the proof. □

The following is a direct corollary of Proposition 3.1.7 since the canonical 3-coloring of a hexagonal system is unique up to the permutation of colors.

Corollary 3.1.8 (Gutman and Salem [57])**.** *Let H be a fully hexagonal system. Then H has a unique maximum independent resonant set.*

The following is a result characterizing the catacondensed hexagonal systems maximizing the Clar number among all its isomers. For a vertex set S, let S^c to denote its complement.

Theorem 3.1.9. *Let H be a catacondensed hexagonal system with n vertices and let H^* be the inner dual of H. Let $V_i = \{v | d_{H^*}(v) = i\}$ for $i = 1, 2, 3$. Then H^* has an independent set S such that its complement $S^c = V(H^*) \backslash S$ satisfies*

(1) $\mathtt{cl}(H) = n/6$ *if and only if $S^c \subseteq V_3$ is an independent set;*

(2) $\mathtt{cl}(H) = (n-2)/6$ *if and only if one of the follows holds:*

(2.1) *$S^c \subseteq V_3$ induces exactly one edge; or*

(2.2) *S^c is independent and $|S^c \cap V_1| = 0$ and $|S^c \cap V_2| = 1$;*

(3) $\mathtt{cl}(H) = (n-4)/6$ *if and only if one of the following properties holds:*

(3.1) $S^c \subseteq V_3$ *induces two edges; or*

(3.2) S^c *induces one edge and* $|S^c \cap V_2| = 1$ *and* $|S^c \cap V_1| = 0$*;*

(3.3) S^c *is independent and* $|S^c \cap V_2| + 2|S^c \cap V_1| = 2$.

Proof. Let H be a catacondensed hexagonal system and H^* be the inner dual of H. Then H^* is a tree.

(1) If $\mathtt{cl}(H) = n/6$, then H has a maximum independent resonant set $\mathcal{S}$ such that $H - V(\mathcal{S}) = \emptyset$. Then every hexagon not in $\mathcal{S}$ is adjacent to three hexagons in $\mathcal{S}$. Let S be the set of vertices of H^* corresponding to these hexagons in $\mathcal{S}$. Then the vertex of H^* in $S^c = V(H) \backslash S$ is adjacent to three vertices in S. So it follows that $S^c \subseteq V_3$ is an independent set due to the maximum degree of H^* is 3.

Now assume that H^* has an independent set S such that $S^c \subseteq V_3$ is also an independent set. Let $\mathcal{S}$ be the set of all hexagons corresponding to vertices in S. Then $\mathcal{S}$ is a set of disjoint hexagons as S is independent. For any vertex $x \in S^c$, all three neighbors of x are in S. It follows that $H - V(\mathcal{S})$ is empty. So $\mathcal{S}$ is a independent resonant set, and hence $\mathtt{cl}(H) = |\mathcal{S}| = n/6$.

(2) If $\mathtt{cl}(H) = (n-2)/6$, then H has a independent resonant set $\mathcal{S}$ such that $H - V(\mathcal{S})$ is an single edge e. If the edge e is not on the boundary, then it is a common edge of two hexagons which are adjacent to three disjoint hexagons since $H - V(\mathcal{S}) = e$. So the two vertices corresponding to the two hexagons containing e have degree-3 and are joined by an edge in H^*, and hence (2.1) holds. If the edge e is on the boundary of H, then e is contained by a kink hexagon and hence (2.2) holds.

Now, suppose that H^* has an independent set S such that either (2.1) or (2.2) holds. Let $\mathcal{S}$ be the set of disjoint hexagons corresponding to vertices in S. Then $H - V(\mathcal{S})$ is an single edge e which has a perfect matching. Then $\mathcal{S}$ is a independent resonant set. Since the hexagons in $\mathcal{S}$ cover all vertices of H except the two end-vertices of e, it follows that $|\mathcal{S}| = (n-2)/6$. By Theorem 3.1.6, it follows that $\mathtt{cl}(H) = |\mathcal{S}| = (n-2)/6$.

(3) If $\mathtt{cl}(H) = (n-4)/6$, then H has a independent resonant set $\mathcal{S}$ such that $H - V(\mathcal{S})$ has a matching of size 2. Let S be the set of vertices H^* corresponding to hexagons in $\mathcal{S}$ and $S^c = V(H^*) \backslash S$. If $H - V(\mathcal{S})$ has one connected component which is a path P of length 3, then the path is contained in a hexagon h of H because all other vertices are covered by $\mathcal{S}$. If h is a leaf-hexagon, then h is corresponding to a degree-1 vertex in S^c and all other vertices of S^c have degree-3. So (3.3) follows. So, in the following, assume that h is not a leaf-hexagon. If h is adjacent to two hexagons of H, then one of its adjacent hexagons h_1 belongs to $\mathcal{S}$ but another adjacent hexagon h_2 does not belong to S. Then h_2 has two vertices are not covered by $\mathcal{S}$ and four vertices are covered by $\mathcal{S}$. Hence h_2 is corresponding to a vertex of degree-3 in S^c, and all other hexagons not in $\mathcal{S}$ and different from h and h_2 are corresponding to vertices of degree-3 in S^c. Hence (3.2) follows. If h is adjacent to three hexagons h_1, h_2 and h_3 of H, then one of them, say h_1, belongs to $\mathcal{S}$ but other two hexagons do not belong to $\mathcal{S}$. Then, in H^*, there is an edge joining the two vertices corresponding to h and h_2 and an edge joining the two vertices corresponding to h and h_3. Therefore, (3.1) holds. So we may assume that $H - V(\mathcal{S})$ is not connected. In other words, $H - V(\mathcal{S})$ is a union of two isolated edges e_1 and e_2. Then e_1 and e_2 belong to different hexagons h_1 and h_2 of H such that $h_1 \cap h_2 = \emptyset$ (otherwise, $H - V(\mathcal{S})$ is connected). Each hexagon h_i for $i \in \{1,2\}$ are adjacent to two hexagons in $\mathcal{S}$. If both h_1 and h_2 are corresponding to degree-2 vertices in S^c, then (3.3) holds. If h_1 and h_2 are corresponding to one degree-2 vertex and one degree-3 vertex, then (3.2) holds. If both h_1 and h_2 are corresponding to degree-3 vertices in S^c, then h_i is adjacent to one hexagon $h_i' \notin \mathcal{S}$ (i.e., $h_i \cap h_i' = e_i$) for $i \in \{1,2\}$. It follows that the vertex of H^* corresponding to h_i for $i \in \{1,2\}$ is adjacent to the vertex corresponding to h_i'. Note that $h_1' \neq h_2'$. Otherwise, both e_1 and e_2 are contained in $h_1' = h_2'$, and hence $H - V(\mathcal{S})$ is connected, a contradiction. Therefore, S^c induces two edges and (3.1) follows because all other hexagons (different from h_1 and h_2) not in $\mathcal{S}$ are adjacent to three hexagons in $\mathcal{S}$.

Now, suppose that H^* has an independent set S satisfying either (3.1), or (3.2) or (3.3).

Let $\mathcal{S}$ be the set of hexagons corresponding to vertices in S. Since S is independent, all hexagons in $\mathcal{S}$ are disjoint from each other. If H satisfies (3.1), then $H - V(\mathcal{S})$ have two edges e_1 and e_2 corresponding to the two edges induced by vertices of S^c. Note that each e_i is contained by two hexagons corresponding to degree-3 vertices in S^c because $S^c \subseteq V_3$. So e_1 and e_2 are disjoint and hence form a perfect matching of $H - V(\mathcal{S})$. Therefore, $\mathcal{S}$ is a independent resonant set. If H satisfies (3.2), let h be the hexagon of H corresponding to the vertex in $S^c \cap V_2$ and the edge induced by S^c is corresponding to an edge e_1 in $H - V(\mathcal{S})$. Then e_1 is contained by two hexagons of H corresponding to two vertices in S^c. Note that h is adjacent to two hexagons of H. The hexagon h contains an edge e_2 whose end-vertices do not belong to any other hexagons of H. So $H - V(\mathcal{S})$ is a union of two disjoint edges e_1 and e_2. So $\mathcal{S}$ is a independent resonant set. If H satisfies (3.3), there are two possibilities that either $|S^c \cap V_2| = 2$ and $|S^c \cap V_1| = 0$, or $|S^c \cap V_2| = 0$ and $|S^c \cap V_1| = 1$. If $|S^c \cap V_2| = 2$, let h_1 and h_2 be two hexagons corresponding to the two vertices in $S^c \cap V_2$. Since S^c is also independent, both h_1 and h_2 have an edge, say e_1 and e_2 respectively, whose end-vertices do not belong to any other hexagons. Hence $H - V(\mathcal{S})$ is the union of e_1 and e_2. Therefore, $\mathcal{S}$ is a independent resonant set. If $S^c \cap V_2 = \emptyset$, then $S^c \cap V_1$ has exactly one vertex corresponding to a leaf-hexagon h of H. Then $H - V(\mathcal{S})$ is a path of length 3 which is contained by h. Therefore, $\mathcal{S}$ is a independent resonant set. Hence $\mathtt{cl}(H) \geq |\mathcal{S}| = (n-4)/6$. By Theorem 3.1.6, $\mathtt{cl}(H) = (n-4)/6$. This completes the proof. □

For any given number $n \equiv 2 \pmod 4$, we can construct a tree satisfying the properties in Theorem 3.1.9, and hence a catacondensed hexagonal system with $\mathtt{cl}(H) = \lfloor n/6 \rfloor$. So the following corollary follows.

Corollary 3.1.10. *For each n with $n \equiv 2 \pmod 4$, there is a catacondensed hexagonal system H with $\mathtt{fr}(H) = (n-2)/4$ and $\mathtt{cl}(H) = \lfloor n/6 \rfloor$.*

3.2 Contra Pairs

In general, for two hexagonal systems H_1 and H_2, one with a larger Fries number also has larger Clar number, and vice versa. However, it is not always true in the mathematical sense. For examples, Figure 25 demonstrates the distribution of catacondensed hexagonal systems with 10 hexagons according to Clar number and Fries number. There are 213 non-isomorphic catacondensed hexagonal systems with 10 hexagons having $\texttt{cl}(H) = 5$ and $\texttt{fr}(H) = 10$, while there are 458 non-isomorphic catacondensed hexagonal systems with 10 hexagons having $\texttt{cl}(H) = 6$ and $\texttt{fr}(H) = 9$.

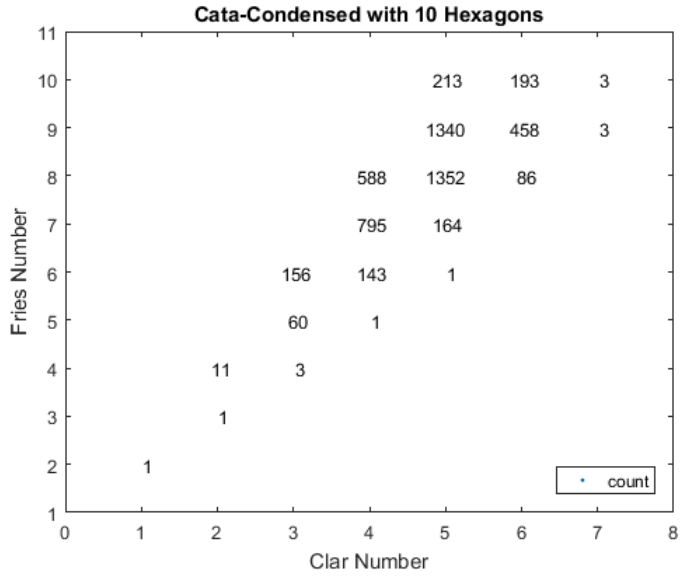


Figure 25: Distribution of Catacondensed Hexagonal Systems with 10 Hexagons with respect to Clar Number and Fries Number.

A pair of hexagonal systems (H_1, H_2) is called a *contra pair* if $\texttt{cl}(H_1) > \texttt{cl}(H_2)$ but $\texttt{fr}(H_1) < \texttt{fr}(H_2)$. From the distribution of cata-condensed hexagonal system with 10 hexagons, contra pairs are not rare. It is interesting to seek a structure characterization for contra pairs of hexagonal systems. However, it seems that such characterization may not be simple. The following result gives the range of the Clar number for hexagonal systems maximizing their Fries numbers.

Theorem 3.2.1. *Let H be a hexagonal system with n vertices and $\phi(H)$ hexagons. If $\mathtt{fr}(H) = \phi(H)$, then*

$$\frac{n-2}{8} \leq \mathtt{cl}(H) \leq \frac{n}{6}$$

and both bounds are sharp.

Proof. Let H be a hexagonal system with $\mathtt{fr}(H) = \phi(H)$. Then H has a perfect matching M such that every hexagon of H is M-alternating. Let H^* be the inner dual of H. Then the Clar number of H is equal to the independent number of H^*.

By Theorem 3.1.5, H is catacondensed and $\mathtt{fr}(H) = \phi(H) = (n-2)/4$. Hence its inner dual H^* is a tree has independent number at least $|V(H^*)|/2 = \phi(H)/2 = (n-2)/8$. It follows that

$$\mathtt{cl}(H) \geq \frac{n-2}{8}$$

which is attained by all hexagonal chains with $\mathtt{fr}(H) = (n-2)/4$. The upper bounds follows directly from Theorem 3.1.6. This completes the proof. □

A direct corollary of above theorem shows the gap between the Clar number and Fries number.

Corollary 3.2.2. *Let H be a hexagonal system with n vertices and $\phi(H)$ hexagons. If $\mathtt{fr}(H) = \phi(H)$, then*

$$\frac{n-6}{12} \leq \mathtt{fr}(H) - \mathtt{cl}(H) \leq \frac{n-2}{8}$$

and both bounds are sharp for infinitely many hexagonal systems.

In the above corollary, the upper bound holds for all hexagonal chains H with $\mathtt{fr}(H) = (n-2)/4$ and the lower bound holds for all catacondensed systems with $\mathtt{fr}(H) = (n-2)/4$ and $\mathtt{cl}(H) = \lfloor n/6 \rfloor$ whose existence follows from Corollary 3.1.10. The corollary implies that there is a infinite sequence of hexagonal systems $\{H_n\}$ such that $\lim_{n\to\infty}(\mathtt{fr}(H_n) - \mathtt{cl}(H_n)) \to \infty$, a result obtained by Klavžar et. al. [66].

For hexagonal systems maximizing Clar numbers, especially fully hexagonal systems, the following theorem shows the range of their Fries numbers.

Theorem 3.2.3. *Let H be a hexagonal system with $\phi(H)$ hexagons and $\tau(H)$ vertices not on its boundary. If H is a fully hexagonal system, then*

$$\frac{\phi(H)+\mathtt{cl}(H)}{2} \leq \mathtt{fr}(H) \leq \phi(H) - \frac{\tau(H)}{6}.$$

If H is catacondensed (not necessarily fully), then

$$\frac{4\mathtt{cl}(H)}{3} \leq \mathtt{fr}(H) \leq \phi(H).$$

Proof. The upper bounds follows directly from Theorem 3.1.5. So it suffices to show the lower bounds.

Let H be a fully hexagonal system. Then $\mathtt{cl}(H) = n/6$. By Proposition 3.1.7, the maximum independent resonant set $\mathcal{S}$ of H is the same as the maximum set of monotonic colored hexagons in the canonical 3-coloring $c : \mathcal{F}(H) \to \mathbb{Z}_3$ of H where $\mathcal{F}(H)$ is the set of all hexagons of H. Without loss of generality, assume that $\mathcal{S} = c^{-1}(1)$, i.e., all hexagons in $\mathcal{S}$ are colored by 1.

Let S be the set of all vertices in the inner dual H^* of H corresponding to all hexagons in $\mathcal{S}$. Then it follows that $H^* - S$ is a bipartite graph because all hexagons of $\mathcal{F}(H)\backslash\mathcal{S}$ can be colored by two colors 0 and 2. We may assume that $|c^{-1}(0)| \geq |c^{-1}(2)|$. Note that H is a subgraph of the infinite lattice L bounded by the boundary of H. Define an edge coloring of H such that $\alpha : E(H) \to \mathbb{Z}_3$ such that $\alpha(e) = c(h_1) + c(h_2) \pmod 3$ where h_1 and h_2 are two hexagons of L both containing e. Let $M = \alpha^{-1}(1) \subset E(H)$. Then every hexagon h colored by 1 contains 3 disjoint edges from M since there are three hexagons colored by 0 surrounding h. Therefore, all hexagons in $\mathcal{S} = c^{-1}(1)$ are M-alternating. Since $|\mathcal{S}| = n/6$, it follows that M is a perfect matching of H. On the other hand, all hexagons of H colored

by 0 are also M-alternating since every hexagon colored by 0 is adjacent to three disjoint hexagons colored by 1 in L. Hence $c^{-1}(1) \cup c^{-1}(0)$ is a resonant set of H. Therefore,

$$\mathtt{fr}(H) \geq |c^{-1}(1) \cup c^{-1}(0)| = |c^{-1}(1)| + |c^{-1}(0)| \geq \mathtt{cl}(H) + (\phi(H) - \mathtt{cl}(H))/2 = \frac{\phi(H) + \mathtt{cl}(H)}{2}.$$

If H is catacondensed, let $\mathcal{S}$ be the maximum independent resonant set. Choose a perfect matching M of H such that: (1) all hexagons in $\mathcal{S}$ are M-alternating; (2) subject to (1), the total number of M-alternating hexagons is maximum. Let $\mathcal{S}'$ be the set of all M-alternating hexagons. Every hexagon $h \in \mathcal{S}$ is contained by at least a maximal linear chain of H which contains at most two M-alternating hexagons which are adjacent to each other by Observation 3.1.1. However, an M-alternating hexagon which is not in $\mathcal{S}$ could be adjacent to at most three hexagons in $\mathcal{S}$. Therefore,

$$\mathtt{fr}(H) \geq |\mathcal{S}'| = |\mathcal{S}| + |\mathcal{S}' \backslash \mathcal{S}| \geq \mathtt{cl}(H) + \frac{1}{3}\mathtt{cl}(H) = \frac{4\mathtt{cl}(H)}{3}.$$

This completes the proof. □

In the following, we focus on catacondensed hexagonal systems. Let $\mathcal{H}(n)$ be the family of all catacondensed hexagonal systems with n vertices. Define $\mathtt{fr}(\mathcal{H}(n)) = \{\mathtt{fr}(H) | H \in \mathcal{H}(n)\}$ and $\mathtt{cl}(\mathcal{H}(n)) = \{\mathtt{cl}(H) | H \in \mathcal{H}(n)\}$. Note that, for any given $n \equiv 2 \pmod 4$, we can easily construct a hexagonal system of n vertices having any Fries number $2 \leq \mathtt{fr}(H) \leq \phi(H)$ and also a system of n vertices having any Clar number $1 \leq \mathtt{cl}(H) \leq \lfloor n/6 \rfloor$ as follows.

Let $n \equiv 2 \pmod 4$ and $2 \leq k \leq (n-2)/4$. In order to construct a catacondensed hexagonal system H with n vertices and $\mathtt{fr}(H) = k$, we construct a catacondensed hexagonal system H' with k hexagons in which every maximal linear chain has length 2. Then by Theorem 3.1.5, we have $\mathtt{fr}(H') = k$. Next, extend one maximal linear chain of H' to a linear chain with $\phi(H) - k + 2$ hexagons. Let H be the resulting catacondensed hexagonal system which has n vertices and $\mathtt{fr}(H) = k$. Note that, there is no catacondensed hexagonal system

with $n \geq 10$ vertices and $\mathtt{fr}(H) = 1$.

Let $n \equiv 2 \pmod 4$ and $1 \leq k \leq n/6$. For constructing a catacondensed hexagonal system H with n vertices and $\mathtt{cl}(H) = k$, first we construct a fully catacondensed hexagonal system H' with $\mathtt{cl}(H') = k$ by Proposition 3.1.7. Then H' has $6k$ vertices. Then choose a maximal linear chain of H' containing a leaf-hexagon and extend it to a linear chain with $(n-k)/4+2$ hexagons. Then the resulting catacondensed hexagonal system has n vertices and $\mathtt{cl}(H) = k$. Then we have the following proposition.

Proposition 3.2.4. *Let $\mathcal{H}(n)$ be the family of catacondensed hexagonal systems with $n \geq 10$ vertices. Then $\mathtt{fr}(\mathcal{H}(n)) = \{2,3,...,(n-2)/4\}$ and $\mathtt{cl}(\mathcal{H}(n)) = \{1,2,...,\lfloor \frac{n}{6} \rfloor\}$.*

3.3 Clar Structure and Fries Structure

In this section, we show that Conjecture 3.0.5 holds for all catacondensed hexagonal systems. Let H be a hexagonal system. If H has a perfect matching M realizing both Clar number and Fries number, then the maximum set $\mathcal{S}'$ of M-alternating hexagons has size $\mathtt{fr}(H)$ and the maximum set $\mathcal{S}$ of independent M-alternating hexagons has size $\mathtt{cl}(H)$. Therefore, Conjecture 3.0.5 is equivalent to the following conjecture. Klavžar et. al. [66] show that a maximal resonant set may not contain a maximum independent resonant set.

Theorem 3.3.1. *Let H be a catacondensed hexagonal system. Then H has a maximum resonant set containing a maximum independent resonant set.*

Proof. Use induction on the number of maximal linear chains of H. If H is a linear chain, then the result holds trivially. So assume that H has at least two maximal linear chains.

Let L be a maximal linear chain of H containing a leaf-hexagon. Without loss of generality, assume that $L = h_1h_2\ldots h_k$ with $k \geq 2$ and h_1 is the leaf-hexagon. Let H' be the hexagonal system obtained from H by deleting all vertices in $(\cup_{i=1}^{k-1} V(h_i)) \backslash V(h_k)$. Label

the vertices of h_k by $v_1, v_2, \ldots, v_6$ such that $v_1v_2 \in h_{k-1} \cap h_k$. By inductive hypothesis, H' has a maximum resonant set containing a maximum independent resonant set. Among all such maximum resonant sets and maximum independent resonant sets, choose a maximum resonant set $\mathcal{S}'$ and a maximum independent resonant set $\mathcal{S} \subseteq \mathcal{S}'$ such that

(i) $h_k \notin \mathcal{S}'$ if possible;

(ii) if $h_k \in \mathcal{S}'$, choose $\mathcal{S}$ such that $h_k \notin \mathcal{S}$ if possible.

Since $\mathcal{S}$ is a maximal independent resonant set, $H' - V(\mathcal{S})$ has a unique perfect matching by Theorem 3.1.3, denoted by M. By Observation 3.1.2, $\mathcal{S}$ intersects every maximal linear chain of H, which implies that $v_1v_2 \in M$ if $h_k \notin \mathcal{S}$.

If $h_k \notin \mathcal{S}'$, then $\mathcal{S} \cup h_1$ is a maximum independent resonant set by Observation 3.1.2 since $\mathcal{S}$ is a maximum independent resonant set of H'. If $k = 2$, it is clear that $\mathcal{S}' \cup h_1$ is a maximum resonant set of H which contains $S \cup h_1$. So assume that $k \geq 3$. Choose a perfect matching M' of H' such that $M \subseteq M'$ and all hexagons in $\mathcal{S}'$ are M'-alternating which can be done by adding a perfect matching carefully chosen from every hexagon h in $\mathcal{S}$ to M so that $h \cap h' \in M'$ where the hexagon $h' \in \mathcal{S}' \backslash S$ and is adjacent to h. Let M'' be a perfect matching of $L - V(h_k \backslash h_{k-1})$ such that both h_1 and h_2 are M''-alternating. Then $M' \backslash \{v_1v_2\} \cup M''$ is a perfect matching of H such that all hexagons in $\mathcal{S}' \cup \{h_1, h_2\}$ are alternating with respect to $M' \backslash \{v_1v_2\} \cup M''$. Note that $\mathcal{S}'$ is a maximum resonant set of H' and $\{h_1, h_2\}$ is a maximum resonant set of L. It follows that $\mathcal{S}' \cup \{h_1, h_2\}$ is a maximum resonant set of H which contains $\mathcal{S} \cup h_1$. The theorem follows.

In the following, assume that $h_k \in \mathcal{S}'$. Suppose first that $h_k \notin \mathcal{S}$. Then $\mathcal{S} \cup h_{k-1}$ is an independent resonant set of H by Observation 3.1.2. Since $\mathcal{S}$ is a maximum independent resonant set of H', it follows that $\mathcal{S} \cup h_{k-1}$ is a maximum independent resonant set of H because L contains a most one hexagon from a maximum independent resonant set.

Let M' be the perfect matching of H' containing such that all hexagons in $\mathcal{S}'$ are M'-alternating and let M'' be the unique perfect matching of $L - V(h_k)$. Note that h_{k-1} is

$(M'' \cup M')$-alternating because $v_1v_2 \in M \subset (M' \cup M'')$. Therefore, all hexagons of $\mathcal{S}' \cup h_{k-1}$ are $(M' \cup M'')$-alternating. So $\mathcal{S}' \cup h_{k-1}$ is a resonant set of H. Since $\mathcal{S}'$ is maximum in H' and $\{h_{k-1}, h_k\}$ is maximum in L, it follows that $\mathcal{S}' \cup h_{k-1}$ is a maximum resonant set of $H = H' \cup L$. Note that $\mathcal{S} \cup h_{k-1} \subseteq \mathcal{S}' \cup h_{k-1}$, and hence the theorem holds. So assume that $h_k \in \mathcal{S}$. Then any independent resonant set without h_k has size smaller than $\mathcal{S}$ by (ii). Therefore, $\mathcal{S}$ is a maximum independent resonant set of H. Similarly, we have that $\mathcal{S}' \cup h_{k-1}$ is a maximum resonant set of H which contains $\mathcal{S}$. This completes the proof. □

3.4 Linear Programming

Linear programming has been a very important and useful tool for solving many combinatorial optimization problems [82].

3.4.1 The Clar Number and Integer Linear Programming

In a Clar formula of a cata-condensed hexagonal system H, every vertex i has to belong to a hexagon containing a circle or an end vertex of a double bound. Therefore, we formulate the problem as the following: Let m be the number of hexagons of a given catacondensed benzenoid H, h_k be a hexagon of H where $k = 1,2,...m$ and y_k be a variable such that:

$$y_k = \begin{cases} 1 & \text{if } h_k \text{ contains a circle} \\ 0 & \text{if } h_k \text{ otherwise} \end{cases}$$

Let (i, j) be the edge joining vertices i and j of H and $x_{i,j}$ be a variable such that:

$$x_{i,j} = \begin{cases} 1 & \text{if the edge } (i,j) \text{ is a double bound.} \\ 0 & \text{if otherwise} \end{cases}$$

Where y_k and $x_{i,j}$ are constraint variables of the integer linear program problem. The neighborhood set of a vertex i is the set of all adjacent vertices of i and is denoted $N(i)$. Let $V(H)$ be the set of vertices, $E(H)$ be the set of edges of H, and $H(i)$ is the set of hexagons containing the vertex i in H.

Let the function that we want to optimize be the objective function and the constraints functions be an integer linear program as follows: The objective function:

$$\text{maximize} \quad z = \sum_{h_k \in H} y_k \quad \text{where} \quad k = 1,2,...m \tag{3.1}$$

Subject to the constraints:

$$\sum_{j \in N(i)} x_{ij} + \sum_{h_k \in H} y_k = 1 \tag{3.2}$$

$$x_{ij}, y_k \in \{0,1\} \quad \text{for} \quad (i,j) \in E(H), \quad h \in H \quad \text{and} \quad k = 1,2,...,m \tag{3.3}$$

The feasible solution for the maximization problem is determined by the constraint (3.3) that confirms that each vertex belongs to a double bound or a circle (perfect matching). We formulate the Clar numbers problem as integer programming (IP) problems. Then, we relax the (IP) to the linear programming (LP). Then, solving the linear programming relaxation of the integer programming by using branch and bound method. Then, we obtain optimum integer solutions.

We substitute the integer optimal solution of this (IP) in the objective function to obtain the optimal objective value, which is the Clar number of H.
The Clar formula of this problem can be attained from the optimum integer solutions as the following: A hexagon h_k contains a circle if $y_h = 1$ and the edge (i, j) is a double bound if $x_{i,j} = 1$.

3.4.2 The Fries Number and Integer Linear Programming

The following model is used for building the Fries optimization system for each data structure.

$$\text{maximize} \quad \sum_{k=1}^{m} y_k \tag{3.4}$$

Subject to the constraints:

$$\sum_{j\in N(i),j<i} x_{ij} + \sum_{j\in N(i),j>i} x_{ji} = 1 \qquad \text{for} \quad i = 1,2,...,n \tag{3.5}$$

$$\left(\sum_{i\in h_k} \sum_{j\in N(i)\cap h_k, j>i} x_{ij}\right) - 3y_k \geq 0 \qquad \text{for} \quad k = 1,2,...,m \tag{3.6}$$

$$x_{ij} \in \{0,1\} \quad i = 1,2,...,n; j \in N(i), j > i \tag{3.7}$$

$$y_k \in \{0,1\} \quad k = 1,2,...,m \tag{3.8}$$

The feasible solution for the maximization problem is determined by the constraints (3.5-3.8). The constraint (3.5) confirms that each vertex is incident with one double bond (i.e. no two double bonds share a vertex) and all edges satisfying this constraint build a perfect matching of the given molecule graph. The constraint (3.6) emphasizes that the *k*th hexagon has three double bounds in the perfect matching formed in (3.5).

3.5 Clar Number vs Fries Number: Implementations

In this section, we include some computational results for cata-condensed hexagonal systems with 10 hexagons. We use the algorithm of Brinkmann, Caporossi and Hansen [13] to construct all 5572 non-isomorphic cata-condensed hexagonal systems of 10 hexagons. The Clar number and Fries number can be computed by binary integer programming [59] and can be relaxed into linear programming [1]. For computing Clar number and Fries number of fullerenes, see [17, 110].

We compute the Clar numbers and the Fries numbers for catacondensed benzenoids using integer linear programming. The catacondensed benzenoids are saved as a binary code called "planar code" format for saving planar graphs [14]. For each catacondensed benzenoids isomer, we build two different systems, one for computing the Clar number and

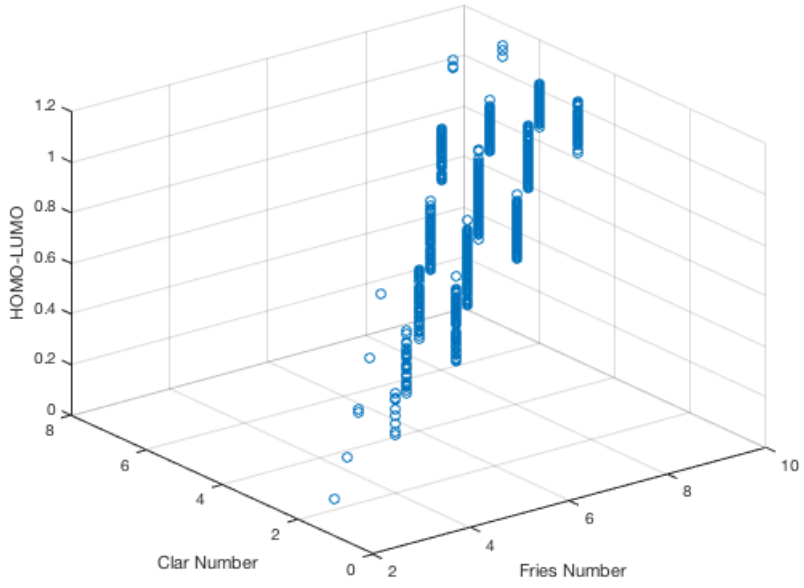


Figure 26: Distribution of HOMO-LUMO Gaps of Catacondensed Hexagonal Systems with 10 Hexagons with respect to Clar Number and Fries Number.

the other for computing the Fries number using the following steps:

- Convert the planar code to readable adjacency lists using Brinkmann et al. [14] program "planarread-mac" and the C program "planarcode"; then, use the executable file (*./planarread*) [14].
- Get the adjacency matrix from the adjacency list for each catacondensed benzenoid isomer.
- Use the adjacency matrix $|V| \times |V|$ for each isomer to construct two different incident matrices as the following:
  - To construct the Clar number model, the first part of the constraint is $\sum_{j \in N(i)} x_{i,j}$ as mentioned previously in section 3.4.1. The construction of this section of the Clar number model will form $|V| \times |E|$ incident matrix for each isomer. The

other part of constraint is $\sum_{h_k \in H} y_k$, and we also use the the adjacency matrix $|V| \times |V|$ and apply our small cycle algorithm to find all the hexagons (cycles of length 6) and their vertices. The algorithm will detect all cycles of length 6 and print the number of these found cycles. Then the hexagon incident matrix $|V| \times |H|$ is generated and constructed for each vertex $i \in V(h_k)$, for all k. The combination of the two incident matrices $[|V| \times |E|, |V| \times |H|]$ forms the whole constraints system.

- To construct the Fries optimization system, the first constraint is $\sum_{j \in N(i), j<i} x_{ij} + \sum_{j \in N(i), j>i} x_{ji}$ as mentioned in section 3.4.2. We use the adjacency matrix $|V| \times |V|$ to build this part of the system. This constraint will form $|V| \times |E|$ incident matrix. For the second constraint, the procedures are similar to what we do in the previous step for the Clar model because it has two parts. That means from this constraint, we build two incident matrices $|V| \times |E|$ and $|V| \times |H|$. The two incident matrices together will fulfill the second constraint. The combination of two constraints will construct the whole problem.

- Relax the constraints variables y_k and $x_{i,j}$ of the integer linear program problem to $0 \leq x_{i,j}, y_k \leq 1$ to obtain the linear program (LP).

- Solve for the integer optimum solution with linear programming method (simplex method).

- If the obtained optimal solution is an integer (i.e. all variables are 0 or 1 given the constraint $0 \leq x_{i,j}, y_k \leq 1$), then we are done. If the optimal solution has a non-integer part, then add two new constraints where the non-integer variable equals 0 and 1 one at a time defining two new LP (branch and bound method), return to the previous step to solve it and do the same steps until the optimal solution is an integer.

3.5.1 Computational Results

Our computational results show that hexagonal systems with large Clar number usually also have large Fries number and vice versa, which is believed to be true in general for hexagonal systems. However, this conclusion does not hold exactly from the mathematics point view because many contra pairs exits.

The computation the Clar numbers and Fries numbers for all catacondensed benzenoid isomers for $|h = 10|$ 5572 different isomers was done by using MATLAB(R2016a) and C++ (see Figure 40 shows the procedures). Our programs contain thousands of lines of codes. All codes are performed and executed on a MacBook Pro running a 2.5 GHz Intel Core i5 processor, with 16 GB of 1600 MHz DDR3 memory, and with the macOS Sierra 10.12.3 operating system.

From Clar's Theory [20] and Randić Conjugated Model [99], an isomer has larger resonance energy if it has a larger Clar number or Fries number. However, Clar number and Fries number do not always match exactly because of the existence of contra pairs. The HOMO-LUMO gap of a molecule is the energy difference between the highest occupied molecular orbital (HOMO) and the lowest unoccupied molecular orbital (LUMO), which is often used as a index to describe the stability of molecules (see [47, 68]). We compute the HOMO-LUMO gaps of catacondensed hexagonal systems with 10 hexagons and their HOMO-LUMO distribution with respect to Clar number and Fries number is shown in Figure 26. The distribution shows that the HOMO-LUMO gap increases when both Clar number and Fries number increase.

Figures 27 and 28 demonstrate the distributions of HOMO-LUMO gaps with respect to Clar number and Fries number respectively, which shows that the mean value of HOMO-LUMO gaps increase faster with respect to Clar number than Fries number. The mean value of HOMO-LUMO gap with respect to the maximum value Clar number 7 is much larger

than the mean value of HOMO-LUMO gaps with respect to the maximum Fries number 10. On the other hand, the standard deviation of HOMO-LUMO gap with respect to the maximum Clar number is much smaller than the standard deviation of HOMO-LUMO gap with respect to the maximum Fries number. In conclusion, Clar number is better than Fries number as a stability predictor for hexagonal systems.

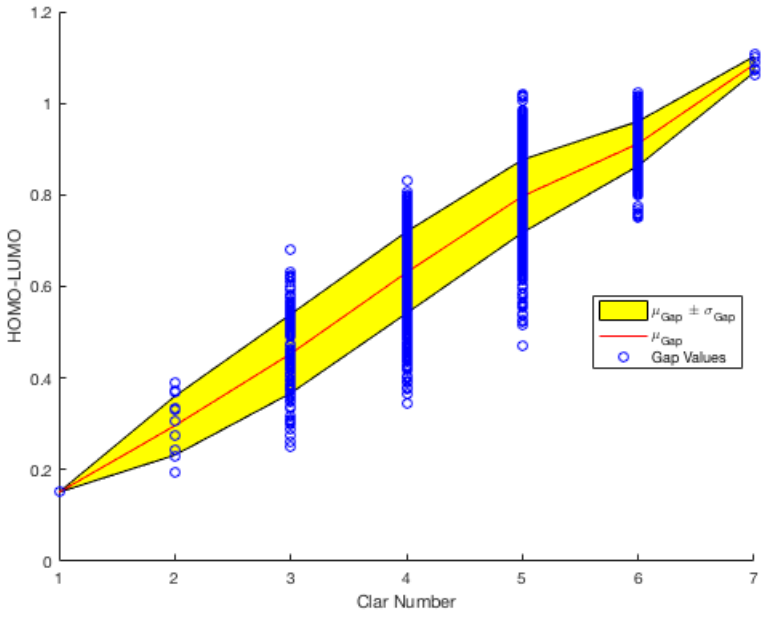


Figure 27: Distribution of HOMO-LUMO Gaps of Catacondensed Hexagonal Systems with 10 Hexagons with respect to Clar Number.

3.5.2 Conclusions

The Clar number and Fries number are used as indices for the stability of a hexagonal system. We investigates the Fries number and Clar number for hexagonal systems, and show that a cata-condensed hexagonal system has a maximum resonant set containing a maximum independent resonant set, which is conjectured for all hexagonal systems. Usually, an isomer of a hexagonal system with a larger Clar number is more stable, and the same holds for the Fries number. However, our computation results show that the Clar numbers and the

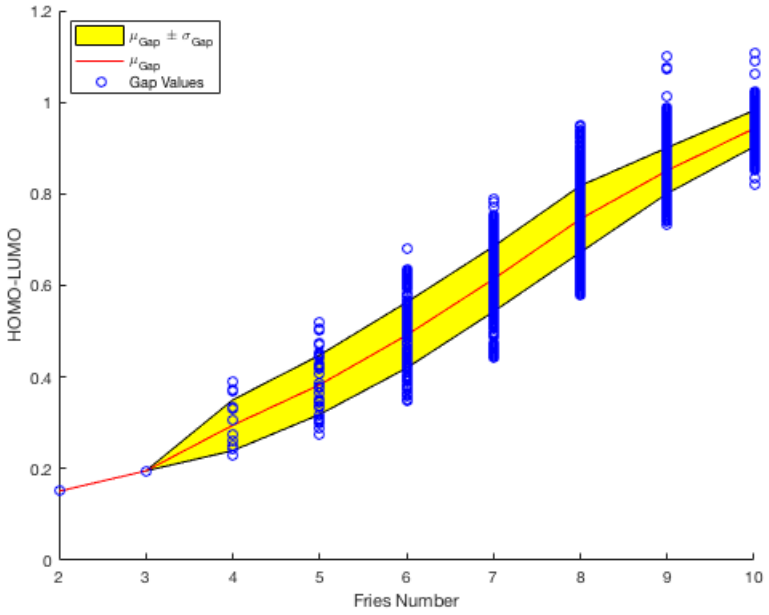


Figure 28: Distribution of HOMO-LUMO Gaps of Catacondensed Hexagonal Systems with 10 Hexagons with respect to Fries Number.

Fries numbers of hexagonal systems are not always consistent, even for cata-condensed benzenoid systems and demonstrate that there exist many contra-pairs. In conclusion, the Clar number is better than Fries number for stability predictor of hexagonal systems. The HOMO-LUMO energy gaps are also calculated for these contra pair isomers. In conclusion, Clar number is better than Fries number as a stability predictor for hexagonal systems.

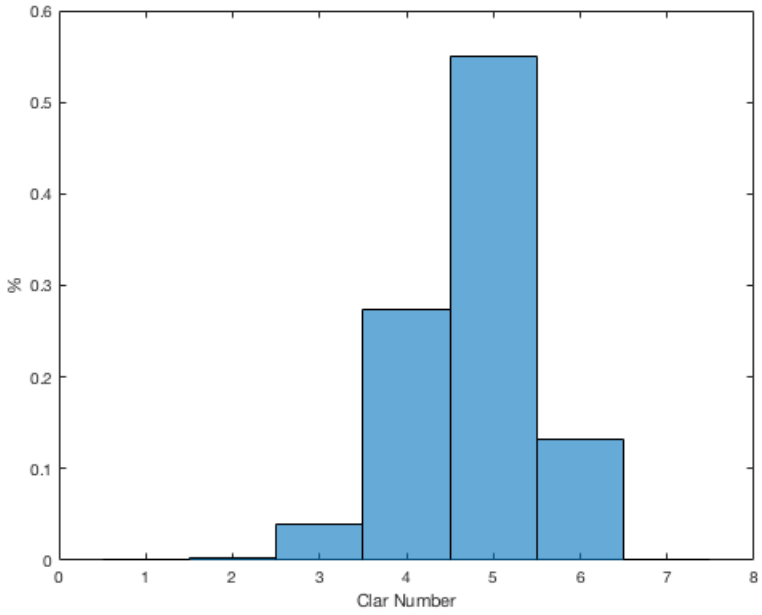


Figure 29: The Normalization Scheme of the Clar Numbers of Catacondensed Hexagonal Systems with 10 Hexagons.

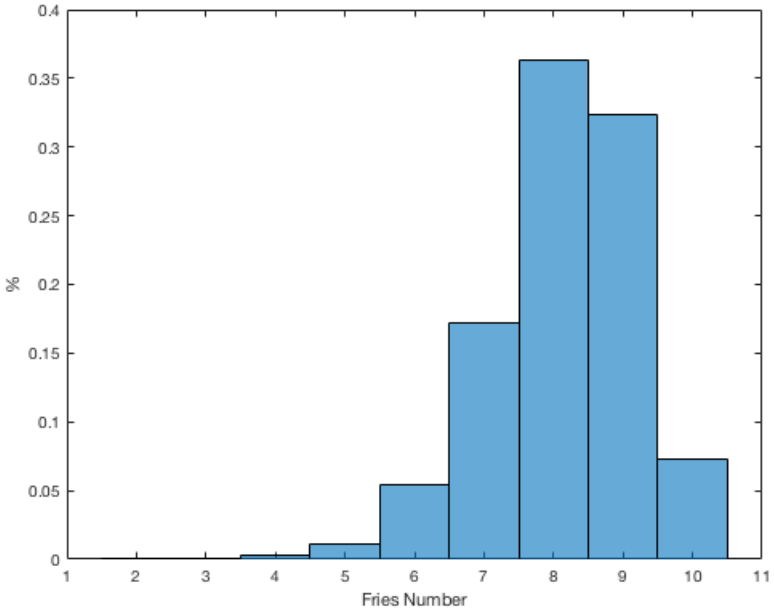


Figure 30: The Normalization Scheme of the Fries Numbers of Catacondensed Hexagonal Systems with 10 Hexagons.

CHAPTER 4

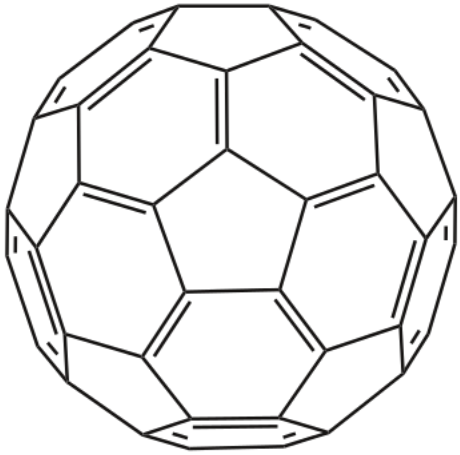
CLAR NUMBERS OF FULLERENES

4.1 Fullerenes

Fullerenes are polyhedral molecules built entirely of carbon atoms with either pentagons or hexagons as faces [6, 5]. In 1985 the first fullerene molecule was found by Kroto et. al. [71]. They discovered the famous Buckminsterfullerene (or buckyball), C_{60} (Figure 31). Later, similar polyhedral structures with 70, 76, 78, 82, 84, 90, 94, and 96 carbon atoms were found in nature [6, 5]. In 1991, the Buckminsterfullerene (or buckyball) was pronounced "Molecule of the year" and five years later Kroto et. al. were awarded the Nobel Prize in Chemistry for their discovery of fullerenes [6, 5]. Fullerenes have been examined and studied from different perspectives and have many applications in fields such as Mathematics, Chemistry, Biology, and Physics due to the thier unique strctuctures. [12, 31, 106, 37, 121, 131, 40].

Banhart et. al. [9] noticed that molecules that had sharp angles between the faces of their structure were less stable then those without. Therefore, fullerenes, which are approximately spherical, are fairly stable molecules. Due to their use in synthetic structures and that they are entirely built of carbon, fullerenes continue to be an important area of research [12, 106, 37, 121, 131, 40].

Graph theory techniques can be applied to represent a fullerene's molecular structure. Graph theoretical terms are used to describe predictors of fullerene molecular stability. The number of perfect matchings, the HOMO-LUMO gap, and other properties are used as predictors of the stability of different fullerene isomers [32, 38].



[126]

Figure 31: Buckminsterfullerene C_{60}

4.1.1 Fullerene Graphs

A graph is used to model the structure of these molecules with vertices representing carbon atoms and edges representing the bonds between them. With fullerenes, every vertex has degree 3. A *fullerene graph* with n vertices is a cubic, planar, and 3-connected graph which has exactly 12 pentagonal faces and $h = \frac{1}{2}n - 10$ hexagonal faces.

Fowler [46] showed that fullernes with n vertices exist for all even vertices $n \geq 20$ except $n = 22$. Chemists use C_n to indicate a fullerene with n vertices. All fullerenes have the same number of pentagons, and these pentagonal faces are important in forming the structure of the corresponding fullerene molecule.

Fowler and Manolopoulos [46, 19] mentioned that the number of fullerene isomers with n vertices is $\mathcal{O}(n^9)$, except for fullerenes with $n = 20$, 24, or 26 vertices because of their unique design. Many algorithms and theoretical studies of the fullerenes were developed to generate a list of fullerene structures, such as to generate all such IPR buckyballs [84, 79, 109].

Barnette [10] developed a mathematical method for generating cyclically n-connected

plane graphs. Brinkmann and Dress [15], developed a fast and complete method for the enumeration and generation of fullerene isomers. For example, with $n = 60$ vertices there exist 1812 different fullerene graphs, including C_{60}. However, their algorithm works only for 3-regular graphs that contain no more than 6-member rings, i.e. the boundary of every face in the graph cannot have more than 6 edges.

Fullerene Types

There are different classes of fullerene graphs regarding where the pentagons are positioned [6]. The *IPR fullerenes (Isolated Pentagon Rule)* are fullerenes where no two pentagons share vertices or edges, i.e. all pentagons are isolated and separated by hexagons. This type of fullerene is more stable [72, 19] because all pentagonal faces are separated which minimizes the steric strain energy and enhance the resonance stabilization of the IPR isomer [46]. If two pentagon faces are adjacent, the two fused pentagons form a cycle of size 8 that means $4n$ anti-aromatic conjugated cycle (i.e. does not satisfy H*ü*ckel rule) [46]. Only one isomer of C_{60} and C_{70} are IPR fullerenes among the 1812 and 8149 possible isomers respectively. There are no IPR fullerene isomers for C_{62},C_{64},C_{66} and C_{68} [72, 67]. For each fullerene C_{70} and larger, there exist at least one IPR isomer [32]. Adjacent pentagons will cause destabilization that results from (i) antiaromaticity (pentalene eight-π-electron system around the ring) and (ii) higher strain energy as a result of angle strain [60]. Higher fullerenes are all fullerene C_n where $n \geq 76$ and giant fullerenes are C_n where $n \geq 240$ (Figure 32).

A leapfrog fullerene is a very important type of fullerene because it has some special chemical properties [45, 78]. It has been proven that this type of fullerne contains at least one Fries Kekulè structure (i.e. a perfect matching which includes the maximum number of benzenoid system $\frac{n}{3}$) [45]. Therefore, a leapfrog fullerene must be the most stable fullerene.

This type of fullerene can be generated by applying some geometrical steps. The first leapfrog fullerene graph is C_{60} and can be constructed from the dodecahedron fullerene graph C_{20}.

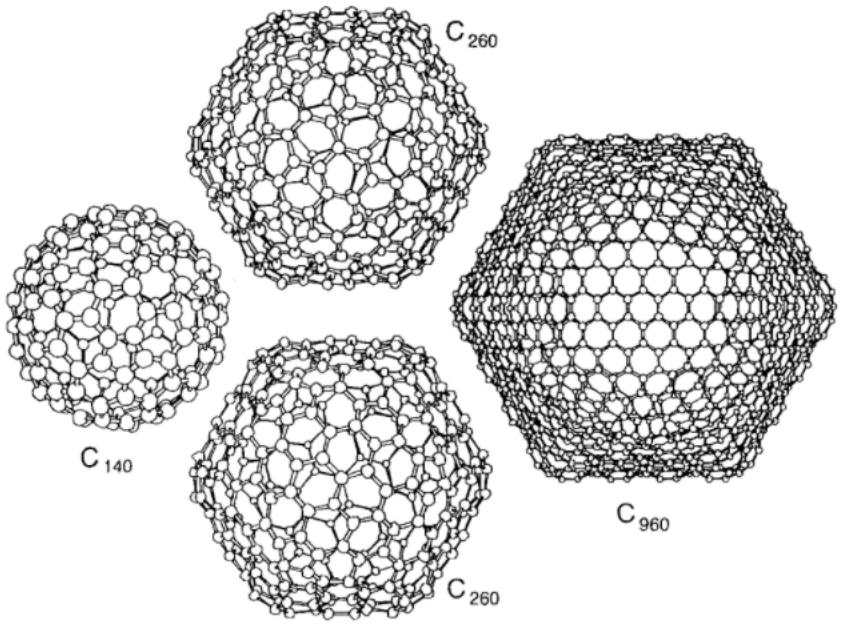


Figure 32: Giant Fullerene Graphs [36]

A fullerene graph is icosahedral if pentagons are equally distributed and the center of all pentagonal faces, are vertices of trianglar faces. All icosahedral fullerene graphs have common geometrical features. The smallest icosahedral fullerene graph is the dodecahedron C_{20} (Figure 34). C_{20} is the only icosahedral fullerene that does not fulfill the IPR and has the greatest possible degree of symmetry of non-IPR fullerenes. C_{60} is the most icosahedral symmetric fullerene graph [112] (Figure 33).

Coxeter [23] and Caspar and Klug [18] presented the tessellation method that generates all possible icosahedral structures from the triangulated duals of the spheres that have 12 pentagonal faces and some hexagonal faces. The icosahedral fullerene graphs were first explained by Goldberg [52, 53]. He applied his method on hexagonal grid to find the vertices of triangular faces and construct the triangulated dual. He also presented the Goldberg

equation that is applied to find the number of vertices in an icosahedral fullerene

$$n = 20(i^2 + ij + j^2)$$

where $0 \leq i \leq j$. For ($i = j$ or $i = 0$), the fullerene graph is of the full icosahedral symmetry group I_h and has a mirror symmetry. For ($i \neq j$ or $j \neq 0$), the fullerene is of the regular icosahedral group I. The Goldberg vector is defined as $\vec{G} = (i, j)$. This vector can be used to find the location of the (i, j) triangles in a hexagonal grid which is helpful for surrounding the 12 pentagonal rings with more hexagonal rings. In addition to icosahedral, other symmetry groups are used to define fullerene graph structure types such as tetrahedral, prismatic, antiprismatic, dihedral, etc.

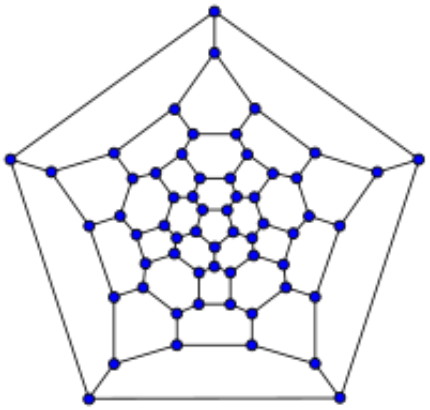


Figure 33: Icosahedral C_{60} Graph [124].

Nanotubical fullerenes are a type of fullerene that have a cylindrical shape. These tubular shapes have two ends that are covered by subgraphs [6]. Each subgraph includes 6 pentagonal rings and some hexagonal rings, but it can be open ended as well (Figure 36). The cylindrical part is basically a rolled hexagonal grid. This hexagonal grid can be used with the Goldberg vector to determine the class of nanotube graph. A nanotube graph structure is of the class (i, j) where the sum of i and j is called the perimeter. The (i, j)-nanotubes where $i = 0$ are called zig-zag, and if $i = j$, they are called armchair. The zig-zags and

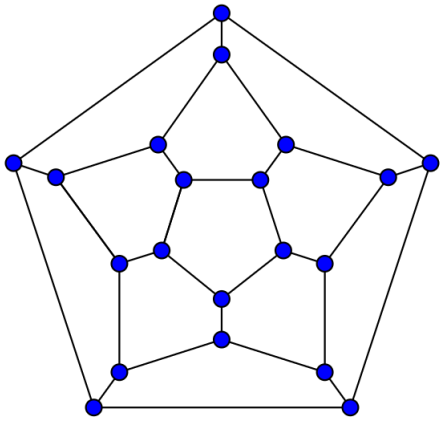


Figure 34: Dodecahedron C_{20} Graph [125].

armchair nanotube structure classes are the only classes that have a mirror symmetry in the hexagonal grid part. The Buckminsterfullerene can be considered the smallest nanotube of class $(5,5)$ whose caps are fulfilling the IPR rule (Figure 35).

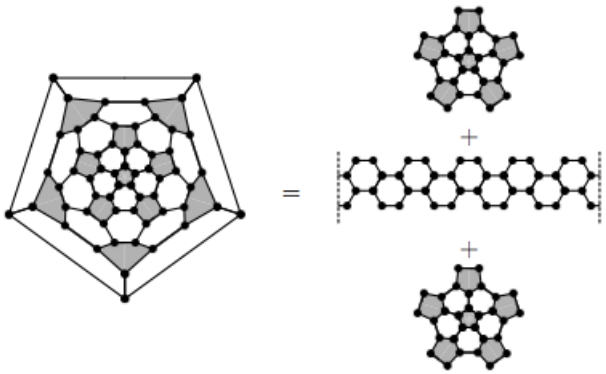


Figure 35: The Smallest Nanotube of Type (5,5)- Buckyball Fullerene C_{60} [6].

4.1.2 Perfect Matching of Fullerenes

The *Kekulé structures* of an organic molecule are also known as its *perfect matchings* M [6]. This is represented in its molecular graph G, by a set of edges such that each vertex in G is incident with exactly one edge in M. The edges of M were previously presented

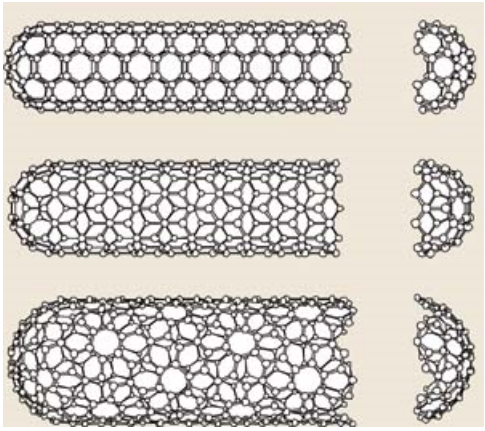


Figure 36: Nanotubes [115].

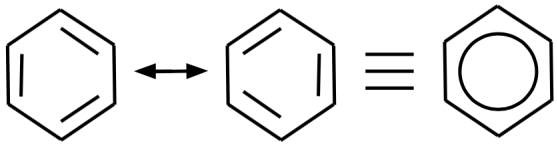


Figure 37: Perfect Matchings for Benzene or Resonance Structures.

as double bonds of fullerene and benzenoid hydrocarbon molecules (Figure 37). The size of a matching M is defined to be the number of edges in the matching, $|E(M)|$. Other fundamental studies of structural properties of fullerene graphs were made and present the character of all distinct Kekulé valence structures of the Buckminsterfullerene [105, 7, 42].

The computation and estimation of the number of Kekulé structures for a given benzenoid system or fullerene molecule is a problem of particular interest because of its important relationship with the molecule's stability and energy level. Babić developed a method to calculate the numbers of perfect matchings of C_{60} and C_{70} to be 12500 and 52168 respectively [8].

Theorem 4.1.1. *[30] If G is a cubic graph with at most 2 cut edges then G has a perfect*

matching.

As a result of this theorem, fullerene graphs all contain a perfect matching [94]. Some linear lower bounds for the number of perfect matchings in fullerene graphs exist with respect to the number of vertices in its graph [132, 30, 31]. Došlić [31] presented some structural properties of fullerene graphs and showed the lower bounds for the number of perfect matchings for fullerene graphs.

Theorem 4.1.2. *[31] A fullerene graph with n vertices has at least* $\frac{n}{4}+2$ *perfect matchings.*

Theorem 4.1.3. *[31] Every fullerene graph with n vertices contains at least* $\frac{n}{2}+4$ *perfect matchings.*

Došlić [30] determined a better lower bound by applying the Cathedral construction algorithm on fullerene graphs. The resulting bound for any fullerene graph with n vertices must include at least $\frac{n}{2}+1$ different perfect matchings.

Theorem 4.1.4. *[30] Every fullerene graph contains at least three different perfect matchings.*

Zhang et. al. [132] established that all fullerene graphs are 2-extendable, then proved an even lower bound for fullerene graphs as in the following theorem:

Theorem 4.1.5. *[132] Any fullerene graph with n vertices can include at least* $\lceil\frac{3(n+2)}{4}\rceil$ *different perfect matchings.*

Došlić [33] proved that the numeration of the perfect matchings is exponentially many for some types of fullerene graphs with n vertices. The same author in [34] shows that the number of perfect matchings of a leapfrog fullerene graph C_n with n vertices is at least $2^{\frac{n}{8}}$.

4.2 Clar Numbers and Fries Numbers of Fullerenes and Integer Linear Programming

In the previous chapter, we applied our small cycles algorithm that described previously in Chapter 2 to find all hexagons of a hexagonal system. This step is very important for the formulation of the problem. Then, we computed the Clar numbers of cata-condensed benzenoid systems. Similarly, we calculate the Clar number index for all fullerene graphs and solve the problem similar to the implementations that are explained in Chapter 3.

Let H be a set of all hexagonal faces of a fullerene C_n with n vertices and m hexagons. Let y_{ij} be an variable such that $x_{ij} = 1$ if $ij \in E(H)$, and 0 otherwise; and let z_{ik} be the vertex-hexagon incident variable such that $y_{ik} = 1$ if the vertex i belongs to the hexagon h_k, and 0 otherwise. Define $x(h_k)$ to be indicator of the hexagon h_k such that, $x(h_k) = 1$ if h_k belongs to a resonant set, and 0 otherwise. Then the Clar number problem can be formulated as follows:

$$\begin{aligned} \text{maximize} \quad & \sum_{h_k \in \mathcal{F}(H)} x(h_k) \\ \text{subject to} \quad & \sum_{0 \le j \le n} y_{ij} + \sum_{0 \le k \le m} z_{ik} x(h_k) = 1, \ \text{ for } i = 1, 2, \ldots, n \\ & y_{ij} \in \{0,1\}, z_{ik} \in \{0,1\} \text{ and } x(h_k) \in \{0,1\}. \end{aligned}$$

And the Fries number problem can be formulated as the following programming:

$$\begin{aligned}
\text{maximize} \quad & \sum_{h_k \in \mathcal{F}(H)} x(h_k) \\
\text{subject to} \quad & \sum_{1 \le j \le n} y_{ij} = 1, \ \text{ for } i = 1, 2, \ldots, n \\
& \sum_{\substack{1 \le i,j \le n \\ i,j \in h_k}} y_{ij} - 3x(h_k) \ge 0, \ \text{ for } k = 1, 2, \ldots, m \\
& y_{ij} \in \{0,1\}, z_{ik} \in \{0,1\} \text{ and } x(h_k) \in \{0,1\}.
\end{aligned}$$

Table 4: The Clar Numbers of Fullerene Isomers C_{20} - C_{60}.

Fullerene	Number of isomers	min	max
C_{20}	1	0	0
C_{24}	1	2	2
C_{26}	1	1	1
C_{28}	2	1	2
C_{30}	3	1	2
C_{32}	6	2	2
C_{34}	6	2	3
C_{36}	15	2	4
C_{38}	17	2	3
C_{40}	40	2	4
C_{42}	45	3	5
C_{44}	89	2	4
C_{46}	116	3	5
C_{48}	199	3	6
C_{50}	271	3	5
C_{52}	437	4	6
C_{54}	580	4	7
C_{56}	924	4	6
C_{58}	1205	4	7
C_{60}	1812	4	8

Table 5: The Fries Numbers of Fullerene Isomers C_{20} - C_{60}.

Fullerene	Number of isomers	min	max
C_{20}	1	0	0
C_{24}	1	2	2
C_{26}	1	1	1
C_{28}	2	1	4
C_{30}	3	2	4
C_{32}	6	4	4
C_{34}	6	4	5
C_{36}	15	4	8
C_{38}	17	4	6
C_{40}	40	4	10
C_{42}	45	5	10
C_{44}	89	4	10
C_{46}	116	6	10
C_{48}	199	6	12
C_{50}	271	6	12
C_{52}	437	7	12
C_{54}	580	8	12
C_{56}	924	8	14
C_{58}	1205	8	16
C_{60}	1812	8	20

4.3 HOMO-LUMO Energy Gap of Fullerenes

Electronic transitions can be explained in terms of graph eigenvalues in conjugated π-system molecular graphs. From the H*ü*ckel method, the eigenvectors x_i and eigenvalues λ_i are the molecular orbitals of the π-system and the orbital energies respectively [47].

Fukui [49] noticed that the reactivity of conjugated hydrocarbons is related to the frontier orbitals HOMO-LUMO. There is an interaction between the "frontier" electrons, the highest occupied energy (HOMO) of a given molecule and the lowest unoccupied orbitals (LUMO) of another molecule [49]. A molecule has a large HOMO-LUMO energy gap implies a high stability [49].

The HOMO is the highest energy orbital, and LUMO is the lowest energy orbital, HOMO and LUMO together are frontier molecular orbitals [90]. The energy differences between HOMO and LUMO is called the HOMO-LUMO gap. While many methods can be used to predict the HOMO-LUMO energy gap, the H*ü*ckel theory or H*ü*ckel molecular orbit energies method (HMO) is the most familiar [56, 47]. There is a linear relationship between these energy orbitals or HMO and the eigenvalues of a molecule graph. The HOMO and LUMO eigenvalues are used to explain the π-electron systems. A given molecule graph with n vertices has n electrons; there are $\lambda_1, \lambda_2, ..., \lambda_n$ eigenvalues arranged in decreasing order.

The orbital energy of a molecule graph can be represented by the formula $E_i = \alpha + \lambda_i \beta$ where $i = 1, 2, .., n$ and the integrals α and β are negative energies [56, 47]. This formula explains the relation between the HMO approximations and the eigenvalues and shows that the H*ü*ckel molecular orbital energies can be represented as linear functions of eigenvalues of a molecule graph [56]. The β is used as the unit of the energy scale and β = -137.0 kJ/mol = -1.4199 eV [111].

Then, if the number of vertices n is even, we can define the HOMO eigenvalue to

be $\lambda_{HOMO} = \lambda_{\frac{n}{2}}$ and the LUMO eigenvalue to be $\lambda_{LUMO} = \lambda_{\frac{n}{2}+1}$. Otherwise, $\lambda_{HOMO} = \lambda_{LUMO} = \lambda_{\frac{n+1}{2}}$.

Within the H*ü*ckel molecular orbital approximations, the total π-energy of a molecule $E_\pi = \sum_{i=1}^{n} n_i \lambda_i$ where $n_i = 2, 1$ or 0 is the number of $\pi-$electron [56]. The energy of a graph is $E(G) = \sum_{i=1}^{n} |\lambda_i|$ [56]. If a molecule graph is a bipartite graph and some others, then the orbital energy E_π equals the energy of a molecule graph $E(G)$ [56, 47].

In this work, we calculate the energy gap by computing the absolute value of the difference between the HOMO and LUMO eigenvalues.

The HOMO and the LUMO have essential functions in various chemical reactions. Any chemical reaction has kinetic and thermodynamic aspects. The kinetic stability means the stability of a molecule against any chemical reaction [84].

The estimation of the kinetic stability is not easy compared with the estimation of thermodynamic stability because it is related to chemical reactions, while the thermodynamic stability can be examined with different methods. The computation of the HOMO-LUMO energy gap is used to estimate the kinetic stability of fullerenes[84].

The fullerenes C_{60} and C_{70} isomer structures have a large HOMO-LUMO energy gap [28, 80]. The C_{74} and the metallofullerenes C_{80} isomer with icosahedral symmetry, have small HOMO-LUMO energy gaps and thus they are only kinetically unstable [28].

As the size of a fullerene graph increases, the kinetic stability will be raised and enhanced because the number of hexagonal faces increases to the pentagonal faces [44] and the pentagonal faces are in charge of reducing fullerene's aromaticity [119, 4]; however, the value of HOMO-LUMO energy gap becomes smaller due to the limitation of graphite [2].

Fowler [44] mentioned that the value of HOMO-LUMO energy gap sometimes cannot be adequate for predicting the kinetic stability of fullerenes and sometimes it is. In general, a molecule graph that has a large HOMO-LUMO energy gap is chemically unreactive because it has a high kinetic stability which means adding electrons to a high-lying LUMO

and removing electrons from a low-lying HOMO is undesirable [3, 84, 2]. Therefore, the HOMO-LUMO energy gap cannot be used as an index of kinetic stability for higher fullerenes with large π-electron systems.

4.4 Computational Results

This problem is solved by applying our small cycles algorithm that was described previously in Chapter 2 to formulate the Clar number and Fries number problems and then solving the problems by using integer linear programming. This is a similar method to what was described in Chapter 3. Then, we calculate the Clar numbers for all fullerene isomers from C_{20} to C_{60}. On completion our program provides the results which are summarized in Tables 4 and 5, showing the maximum and minimum Clar number values and Fries number values over all fullerene isomers.

The computation the Clar numbers and Fries numbers for all fullerene isomers is written in MATLAB(R2016a) and C++ (see Figure 40 shows the procedures). Our programs contain thousands of lines of codes. All codes are performed and executed a MacBook Pro running a 2.5 GHz Intel Core i5 processor, with 16 GB of 1600 MHz DDR3 memory, and with the macOS Sierra 10.12.3 operating system.

The interaction between the ”frontier” electrons, the highest occupied energy (HOMO) of a given molecule and the lowest unoccupied orbitals (LUMO) of another molecule has an effective role in many chemical reactions [49]. The HOMO-LUMO energy gaps are also calculated for all fullerene isomers from C_{24} to C_{100} in addition to C_{130} and C_{160}(IPR). Table 6 shows the maximum value of HOMO-LUMO energy gaps and corresponding isomer number of fullerene graphs. Figure 38 shows the largest maximum gaps over all fullerene isomers. Those fullerenes are C_{60} at isomer 1812, C_{72} at isomer 11190, C_{78} at isomer 24108, C_{84} at isomer 51588, C_{90} at isomer 99888, and C_{96} at isomer 191788 (Figure 41A to 41F. The same results are presented by fowler (see [48]). HOMO-LUMO energy gaps of

fullerene graphs C_{130} and C_{160}(IPR) are 0.40 $|\beta|$ and 0.39 $|\beta|$ respectively. The table shows that C_{60} has the largest HOMO-LUMO energy gap. And as the size of a fullerene increases, the HOMO-LUMO energy gap decreases and converges slowly to zero (Figure 39).

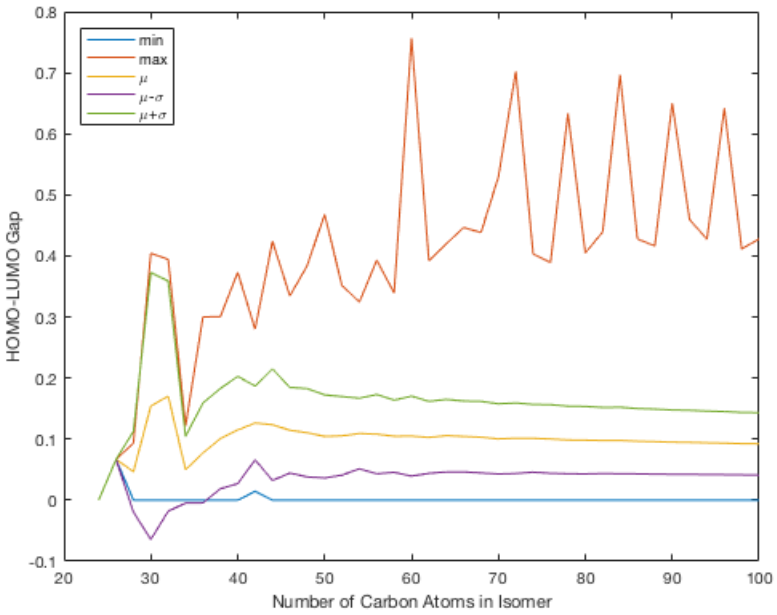


Figure 38: HOMO-LUMO Gap Statistics of Fullerene Graphs from C_{24} to C_{100}.

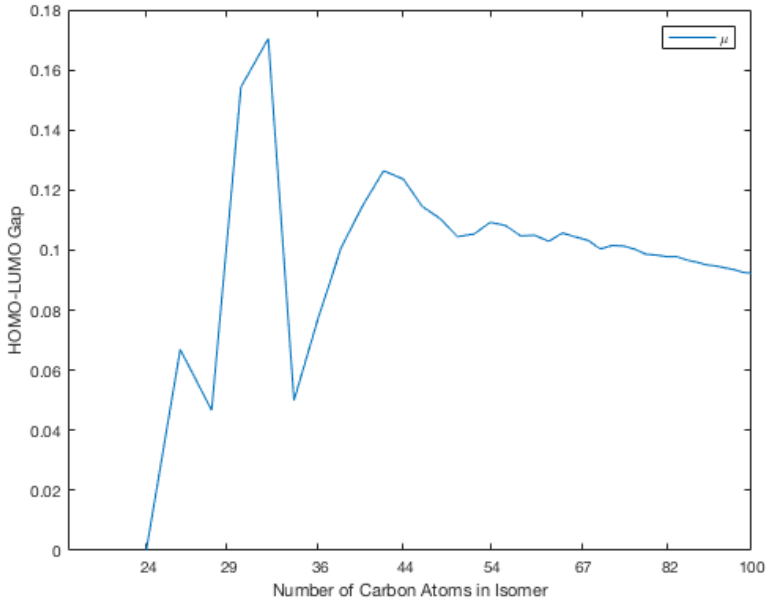


Figure 39: The Mean of HOMO-LUMO Gaps of Fullerene Graphs from C_{24} to C_{100}.

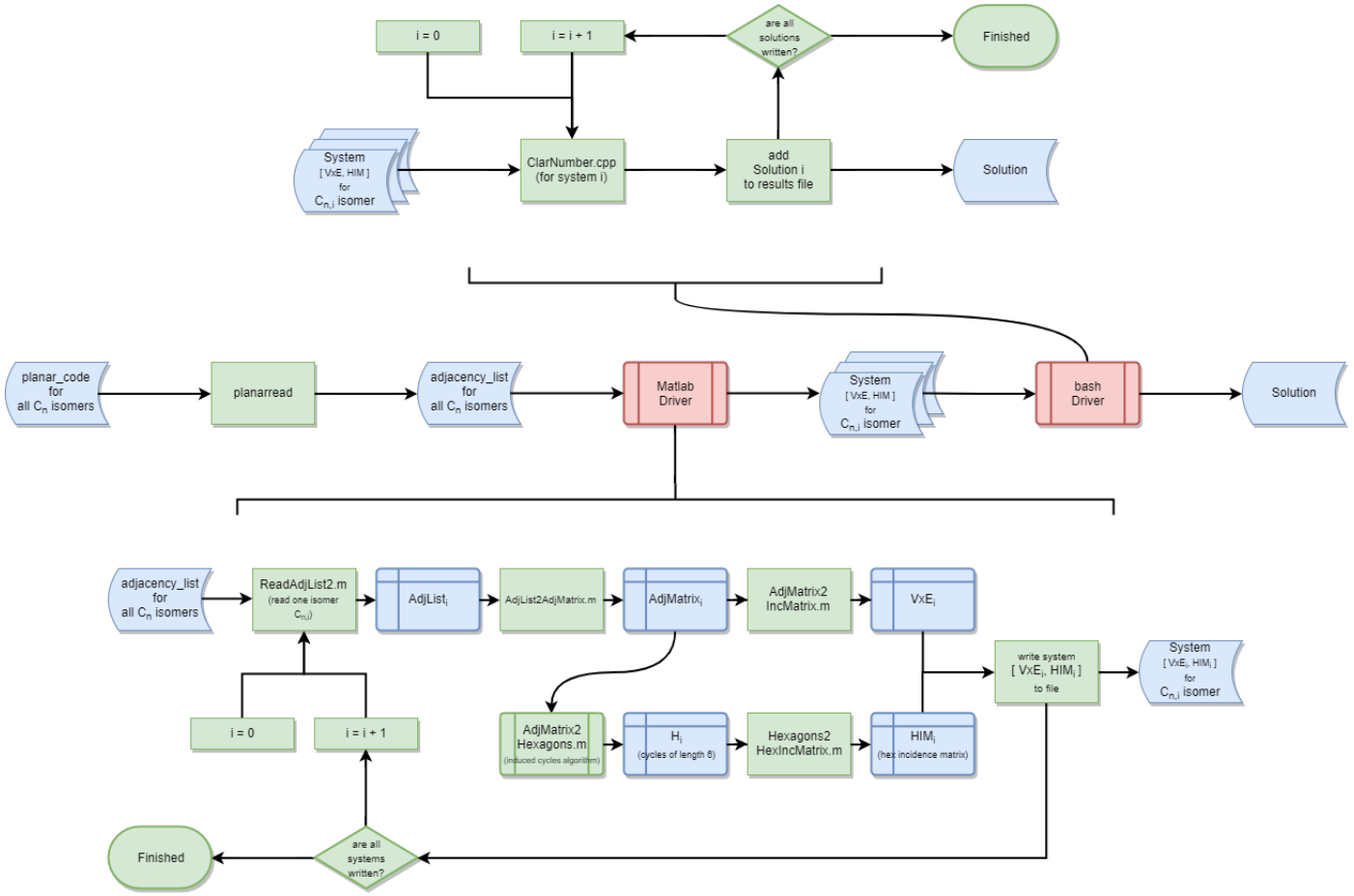


Figure 40: Flow Chart for the Computation of Clar and Fries Numbers of Fullerenes.

Table 6: The Maximum Value of HOMO-LUMO Energy Gaps

Fullerene	**Average Gap**	**Maximum Gap**	**Isomer Number** (of max value)
C_{24}	0	0	1
C_{26}	.0669	0.0669	1
C_{28}	0.0466	0.0933	1
C_{30}	0.1542	0.4044	2
C_{32}	0.1704	0.3942	5
C_{34}	0.0499	0.1217	3
C_{36}	0.0773	0.3002	14
C_{38}	0.1008	0.3004	17
C_{40}	0.1149	0.3731	39
C_{42}	0.1264	0.2800	45
C_{44}	0.1236	0.4241	72
C_{46}	0.1145	0.3345	107
C_{48}	0.1105	0.3842	163
C_{50}	0.1045	0.4679	270
C_{52}	0.1053	0.3516	333
C_{54}	0.1093	0.3244	256
C_{56}	0.1082	0.3929	865
C_{58}	0.1048	0.3393	1049
C_{60}	0.1049	0.7566	1812
C_{62}	0.1029	0.3919	1997
C_{64}	0.1057	0.4198	3452
C_{66}	0.1044	0.4464	4152

Continued on next page

Table 6 – continued from previous page

Fullerene	Average Gap	Maximum Gap	Isomer Number (of max value)
C_{68}	0.1033	0.4383	6290
C_{70}	0.1004	0.5293	8149
C_{72}	0.1015	0.7023	11190
C_{74}	0.1013	0.4031	14095
C_{76}	0.1003	0.3893	18720
C_{78}	0.0986	0.6333	24108
C_{80}	0.0983	0.4049	30469
C_{82}	0.0978	0.4393	39269
C_{84}	0.0978	0.6962	51588
C_{86}	0.0966	0.4277	57799
C_{88}	0.0959	0.4162	77956
C_{90}	0.0951	0.6499	99888
C_{92}	0.0947	0.4594	126351
C_{94}	0.0941	0.4273	140420
C_{96}	0.0935	0.6418	191788
C_{98}	0.0926	0.4115	183270
C_{100}	0.0923	0.4279	245666

4.5 Experimental Results

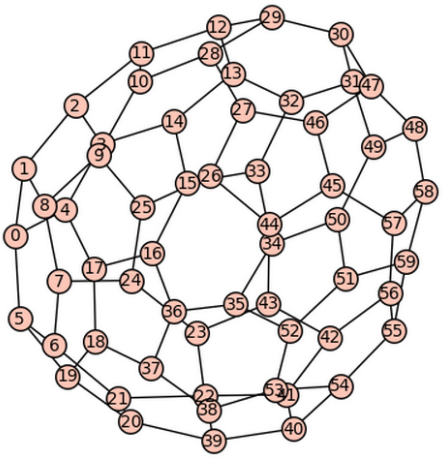
In 1985, C_{60} Buckminsterfullerene (or 60:1812) was the first synthesized fullerene by Kroto et al. [71]. Krätschmer et al. [70] reported that all carbon molecules can be produced in large quantities by the method of the evaporation of graphite. The C_{70} (or 70:8149) was

produced by the same processed method. There are many method that are used to produce fullerene molecules such as laser ablation, arc discharge, electron beam evaporation, heat resistive and etc [129].

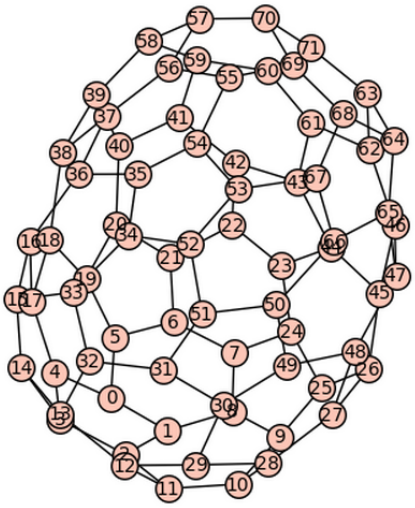
The experimental observation for higher fullerenes in general can be found as a mixture of isomers. In 1991, Diederich et. al [27] used the method that presented by Krätschmer [70] to produce IPR isomers of higher fullerenes. These are C_{78}, C_{84}, C_{90}, and C_{96}. However; C_{72} is not experimentally found [80]. These fullerene isomers plus C_{72} not only have unusual large HOMO-LUMO gaps but also have large resonance energy [80]. Thus, they are more stable than other IPR fullerene isomers [80]. Table.7 shows a list of fullerene isomers that have the largest HOMO-LUMO gap values and the synthesized ones.

Table 7: Fullerenes with the Largest Maximum Value of HOMO-LUMO Energy Gaps.

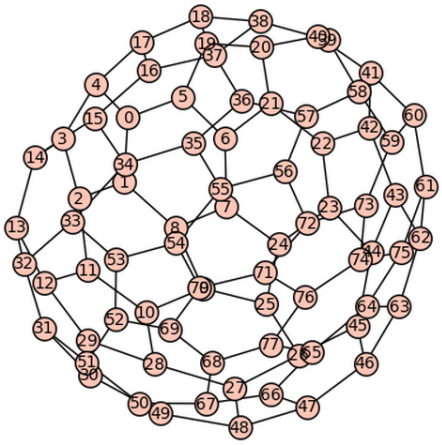
Fullerene	Isomer Number	Maximum Gap	IPR	Synthesized
C_{60}	1812	0.7566	Yes	Yes
C_{70}	8149	0.5293	Yes	Yes
C_{72}	11190	0.7023	Yes	No
C_{78}	24108	0.6333	Yes	Yes
C_{84}	51588	0.6962	Yes	Yes
C_{90}	99888	0.6499	Yes	Yes
C_{96}	191788	0.6418	Yes	Yes



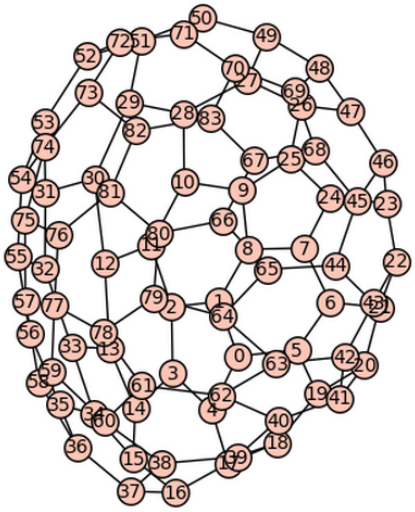
(A) C_{60}, Isomer Number 1812



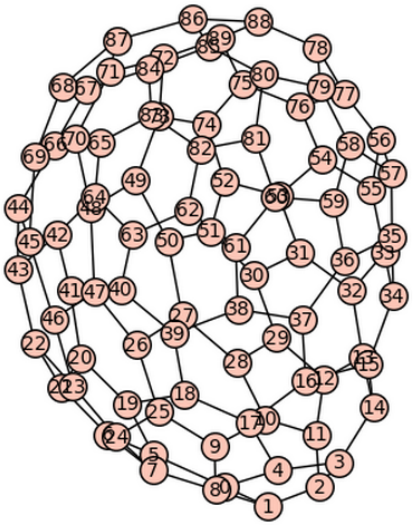
(B) C_{72}, Isomer Number 11190



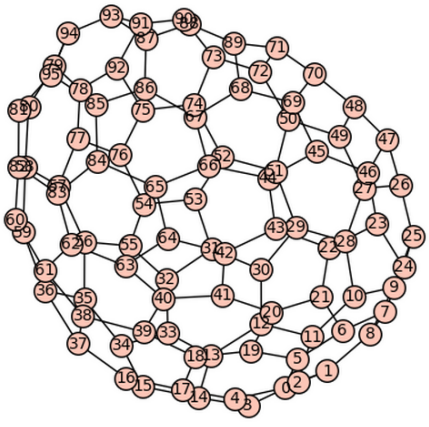
(C) C_{78}, Isomer Number 24108



(D) C_{84}, Isomer Number 51588



(E) C_{90}, Isomer Number 99888



(F) C_{96}, Isomer Number 191788

Figure 41: Fullerene Isomers with the Largest HOMO-LUMO Gaps.

CHAPTER 5

CONCLUSIONS AND FUTURE WORK

5.1 Conclusions

In this work we develop an effective algorithm to reveal all small cycles of size 3 to 8 for any complex chemical graphs. This algorithm has several sections where it can be applied at once or separately on a given graph to attain sets of all cycles of specific length. This algorithm is applied to various chemical structures such as fullerene graphs, benzenoid systems and others. There are some predictors of molecular stability like the Clar number, Fries number, perfect matching, the HOMO-LUMO energy gap value, etc. The Small Cycles algorithm is applied to the calculations of the Clar number and Fries number. After applying the small cycles algorithm on cata-condensed benzenoid systems and fullerene graphs, we compute the Clar numbers and Fries numbers for these chemical structures by using integer linear programming. we investigates the Fries number and Clar number for hexagonal systems, and show that a catacondensed hexagonal system has a maximum resonant set containing a maximum independent resonant set, which is conjectured for all hexagonal systems. Further, our computation results demonstrate that there exist many contra-pairs, and, for stability predictor of hexagonal systems. We show that the Clar number is better than Fries number.

In addition to calculating the HOMO-LUMO energy gap of all cata-condensed hexagonal systems and fullerene isomers. We demonstrated that as a fullerene girth increases the HOMO-LUMO energy gap decreases. However, Fullerenes C_{60} at isomer 1812, C_{72} at isomer 11190, C_{78} at isomer 24108, C_{84} at isomer 51588, C_{90} at isomer 99888, and C_{96} at isomer 191788 have the largest maximum HOMO-LUMO gaps; are the most stable isomers.

5.2 Future Work

We are going to compute the perfect matching of catacondensed benzenoid isomers and fullerene isomers by using integer linear programming. Then, we will try to prove if the following conjuncture is true: Let H_1 and H_2 be isomers of a catacondensed benzenoid with $|V(H_1)| = |V(H_2)|$, then $\phi(H_1) \geq \phi(H_2) \iff cl(H_1) \geq cl(H_2)$ where $\phi(H)$ is the number of perfect matchings of H. If that conjuncture is not true, we will study the isomer cases where it fails.

Also, for any given number $n \equiv 2 \pmod 4$, we can develop an algorithm to construct a tree satisfying the properties in Theorem 3.1.9, and hence a catacondensed hexagonal system with $cl(H) = \lfloor n/6 \rfloor$.

REFERENCES

[1] Abeledo, H. and Atkinson, G. W. Unimodularity of the clar number problem. *Linear algebra and its applications*, 420(2-3):441–448, 2007.

[2] Aihara, J.-i. Weighted homo-lumo energy separation as an index of kinetic stability for fullerenes. *Theoretical Chemistry Accounts: Theory, Computation, and Modeling (Theoretica Chimica Acta)*, 102(1):134–138, 1999.

[3] Aihara, J.-i., Yamabe, T., and Hosoya, H. Aromatic character of graphite and carbon nanotubes. *Synthetic metals*, 64(2):309–313, 1994.

[4] Ajayan, P., Ebbesen, T., Ichihashi, T., Iijima, S., Tanigaki, K., and Hiura, H. Opening carbon nanotubes with oxygen and implications for filling. *Nature*, 362(6420):522–525, 1993.

[5] Andova, V., Kardoš, F., and Škrekovski, R. Fullerene graphs and some relevant graph invariants. *Topics in Chemical Graph Theory*, pages 39–54, 2014.

[6] Andova, V., Kardoš, F., and Škrekovski, R. Mathematical aspects of fullerenes. *Ars Mathematica Contemporanea*, 11(2), 2016.

[7] Austin, S., Fowler, P., Hansen, P., Monolopoulos, D., and Zheng, M. Fullerene isomers of c 60. kekulé counts versus stability. *Chemical Physics Letters*, 228(4):478–484, 1994.

[8] Babić, D. and Ori, O. Matching polynomial and topological resonance energy of c 70. *Chemical Physics Letters*, 234(1):240–244, 1995.

[9] Banhart, F., Füller, T., Redlich, P., and Ajayan, P. The formation, annealing and self-compression of carbon onions under electron irradiation. *Chemical Physics Letters*, 269(3):349–355, 1997.

[10] Barnette, D. On generating planar graphs. *Discrete Mathematics*, 7(3-4):199–208, 1974.

[11] Bersohn, M. An algorithm for finding the synthetically important rings of a molecule. *Journal of the Chemical Society, Perkin Transactions 1*, pages 1239–1241, 1973.

[12] Bosi, S., Da Ros, T., Spalluto, G., and Prato, M. Fullerene derivatives: an attractive tool for biological applications. *European Journal of Medicinal Chemistry*, 38(11):913–923, 2003.

[13] Brinkmann, G., Caporossi, G., and Hansen, P. A constructive enumeration of fusenes and benzenoids. *Journal of Algorithms*, 45(2):155–166, 2002.

[14] Brinkmann, G., Coolsaet, K., Goedgebeur, J., and Mélot, H. House of graphs: A database of interesting graphs. *Discrete Applied Mathematics*, 161(1):311–314, 2013.

[15] Brinkmann, G. and Dress, A. W. A constructive enumeration of fullerenes. *Journal of Algorithms*, 23(2):345–358, 1997.

[16] Camps, P., Fernandez, J. A., Vazquez, S., Font-Bardia, M., and Solans, X. Generation, trapping, and dimerization of pentacyclo [6.4. 0.02, 10.03, 7.04, 9] dodeca-5, 8, 11-triene: An uncatalyzed thermal [2+ 2+ 2+ 2] cycloaddition. *Angewandte Chemie International Edition*, 42(34):4049–4051, 2003.

[17] Carr, J. A., Wang, X., and Ye, D. Packing resonant hexagons in fullerenes. *Discrete Optimization*, 13:49–54, 2014.

[18] Caspar, D. and Klug, A. Viruses, nucleic acids and cancer. In *17th Anderson Symposium, Williams & Wilkins, Baltimore*, 1963.

[19] Cioslowski, J. Note on the asymptotic isomer count of large fullerenes. *Journal of Mathematical Chemistry*, 52(1):1–5, 2014.

[20] Clar, E. Aromatic sextet. 1972.

[21] Corey, E. and Perersson, G. A. Algorithm for machine perception of synthetically significant rings in complex cyclic organic structures. *Journal of the American Chemical Society*, 94(2):460–465, 1972.

[22] Corey, E., Wipke, W. T., Cramer III, R. D., and Howe, W. J. Techniques for perception by a computer of synthetically significant structural features in complex molecules. *Journal of the American Chemical Society*, 94(2):431–439, 1972.

[23] Coxeter, H. Virus macromolecules and geodesic domes. *A Spectrum of Mathematics*, pages 98–107, 1971.

[24] Cyvin, S. J. and Gutman, I. *Kekulé structures in benzenoid hydrocarbons*, volume 46. Springer Science & Business Media, 2013.

[25] Dahl, J., Liu, S., and Carlson, R. Isolation and structure of higher diamondoids, nanometer-sized diamond molecules. *Science*, 299(5603):96–99, 2003.

[26] Dias, J. Handbook of polycyclic hydrocarbons. part a: Benzenoid hydrocarbons. 1987.

[27] Diederich, F., Ettl, R., Rubin, Y., Whetten, R. L., Beck, R., Alvarez, M., Anz, S., Sensharma, D., Wudl, F., Khemani, K. C., and others,. The higher fullerenes: Isolation and characterization of c _76, c _84, c _90, c _94, and c _70 o, and oxide of d _5h-c _70. *Science*, pages 548–551, 1991.

[28] Diener, M. D. and Alford, J. M. Isolation and properties of small-bandgap fullerenes. *Nature*, 393(6686):668–671, 1998.

[29] Diestel, R. *Graph theory {graduate texts in mathematics; 173}*. Springer-Verlag Berlin and Heidelberg GmbH & amp, 2000.

[30] Došlić, T. On lower bounds of number of perfect matchings in fullerene graphs. *Journal of Mathematical Chemistry*, 24(4):359–364, 1998.

[31] Došlić, T. On some structural properties of fullerene graphs. *Journal of Mathematical Chemistry*, 31(2):187–195, 2002.

[32] Došlić, T. Bipartivity of fullerene graphs and fullerene stability. *Chemical Physics Letters*, 412(4):336–340, 2005.

[33] Došlić, T. Fullerene graphs with exponentially many perfect matchings. *Journal of Mathematical Chemistry*, 41(2):183–192, 2007.

[34] Došlić, T. Leapfrog fullerenes have many perfect matchings. *Journal of Mathematical Chemistry*, 44(1):1–4, 2008.

[35] Downs, G. M., Gillet, V. J., Holliday, J. D., and Lynch, M. F. Review of ring perception algorithms for chemical graphs. *Journal of Chemical Information and Computer Sciences*, 29(3):172–187, 1989.

[36] Dresselhaus, M. S., Dresselhaus, G., and Eklund, P. C. *Science of fullerenes and carbon nanotubes: their properties and applications*. Academic press, 1996.

[37] Eckert, J.-F., Nicoud, J.-F., Nierengarten, J.-F., Liu, S.-G., Echegoyen, L., Barigelletti, F., Armaroli, N., Ouali, L., Krasnikov, V., and Hadziioannou, G. Fullerene-oligophenylenevinylene hybrids: synthesis, electronic properties, and incorporation in photovoltaic devices. *Journal of the American Chemical Society*, 122(31):7467–7479, 2000.

[38] Fajtlowicz, S. and Larson, C. Graph-theoretic independence as a predictor of fullerene stability. *Chemical Physics Letters*, 377(5):485–490, 2003.

[39] Fan, B. T., Panaye, A., Doucet, J. P., and Barbu, A. Ring perception. a new algorithm for directly finding the smallest set of smallest rings from a connection table. *Journal of Chemical Information and Computer Sciences*, 33(5):657–662, 1993.

[40] Faraji, A. H. and Wipf, P. Nanoparticles in cellular drug delivery. *Bioorganic & Medicinal Chemistry*, 17(8):2950–2962, 2009.

[41] Fawcett, J. and Trotter, J. The crystal and molecular structure of coronene. In *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, volume 289, pages 366–376. The Royal Society, 1966.

[42] Flocke, N., Schmalz, T., and Klein, D. Variational resonance valence bond study on the ground state of c60 using the heisenberg model. *The Journal of Chemical Physics*, 109(3):873–880, 1998.

[43] Fontaine, X. L., Greenwood, N. N., Kennedy, J. D., and MacKinnon, P. Boron-11 and proton nuclear magnetic resonance study of anti-b 18 h 22 and its anions, anti-[b 18 h 21]–and anti-[b 18 h 20] 2–. the crystal and molecular structure of [nme 4] 2 [anti-b 18 h 20]. *Journal of the Chemical Society, Dalton Transactions*, (7):1785–1793, 1988.

[44] Fowler, P. Aromaticity revisited. *Nature*, 350(6313):20–21, 1991.

[45] Fowler, P. and Pisanski, T. Leapfrog transformations and polyhedra of clar type. *Journal of the Chemical Society, Faraday Transactions*, 90(19):2865–2871, 1994.

[46] Fowler, P. W. and Manolopoulos, D. *An atlas of fullerenes*. Courier Corporation, 2006.

[47] Fowler, P. W. and Pisanski, T. Homo-lumo maps for chemical graphs. *MATCH Commun. Math. Comput. Chem*, 64(2):373–390, 2010.

[48] Fowler, P. W., Rogers, K. M., Fajtlowicz, S., Hansen, P., and Caporossi, G. Facts and conjectures about fullerene graphs: leapfrog, cylinder and ramanujan fullerenes. In *Algebraic Combinatorics and Applications*, pages 134–146. Springer, 2001.

[49] Fukui, K., Yonezawa, T., and Shingu, H. A molecular orbital theory of reactivity in aromatic hydrocarbons. *The Journal of Chemical Physics*, 20(4):722–725, 1952.

[50] Gasteiger, J. and Jochum, C. An algorithm for the perception of synthetically important rings. *Journal of Chemical Information and Computer Sciences*, 19(1):43–48, 1979.

[51] Goddard, R., Haenel, M. W., Herndon, W. C., Kruger, C., Zander, M., and others,. Crystallization of large planar polycyclic aromatic-hydrocarbons-the molecular and crystal-structures of hexabenzo (bc, ef, hi, kl, no, qr) coronene and benzo (l, 2, 3-bc/4, 5, 6-bc) dicoronene. *Journal of the American Chemical Society*, 117(1):30–41, 1995.

[52] Goldberg, M. The isoperimetric problem for polyhedra. *Tohoku Mathematical Journal, First Series*, 40:226–236, 1935.

[53] Goldberg, M. A class of multi-symmetric polyhedra. *Tohoku Mathematical Journal, First Series*, 43:104–108, 1937.

[54] Graver, J. E., Hartung, E. J., and Souid, A. Y. Clar and fries numbers for benzenoids. *Journal of Mathematical Chemistry*, 51(8):1981–1989, 2013.

[55] Gutman, I. and Cyvin, S. J. *Introduction to the theory of benzenoid hydrocarbons*. Springer Science & Business Media, 2012.

[56] Gutman, I. and Polansky, O. E. *Mathematical concepts in organic chemistry*. Springer Science & Business Media, 2012.

[57] Gutman, I. and Salem, K. A fully benzenoid system has a unique maximum cardinality resonant set. *Acta Applicandae Mathematicae*, 112(1):15–19, 2010.

[58] Hansel, K., Jänicke, W., and Wolin, J. M. Zur bestimmung aller elementarkreise in gerichteten graphen. *Optimization*, 9(3):427–440, 1978.

[59] Hansen, P. and Zheng, M. The clar number of a benzenoid hydrocarbon and linear programming. *Journal of Mathematical Chemistry*, 15(1):93–107, 1994.

[60] Hirsch, A. Principles of fullerene reactivity. In *fullerenes and related structures*, pages 1–65. Springer, 1999.

[61] Ioffe, I. N., Goryunkov, A. A., Tamm, N. B., Sidorov, L. N., Kemnitz, E., and Troyanov, S. I. Fusing pentagons in a fullerene cage by chlorination: Ipr d2-c76 rearranges into non-ipr c76cl24. *Angewandte Chemie International Edition*, 48(32):5904–5907, 2009.

[62] Ito, T. and Kizawa, M. The matrix rearrangement procedure for graph-theoretical algorithms and its application to the generation of fundamental cycles. *ACM Transactions on Mathematical Software (TOMS)*, 3(3):227–231, 1977.

[63] Jovanovich, A. D. Note on a modification of the fundamental cycles finding algorithm. *Information Processing Letters*, 3(1):33, 1974.

[64] Kekuié, A. Untersuchungen über aromatische verbindungen ueber die constitution der aromatischen verbindungen. i. ueber die constitution der aromatischen verbindungen. *European Journal of Organic Chemistry*, 137(2):129–196, 1866.

[65] Kekulé, A. Sur la constitution des substances aromatiques. *Bulletin mensuel de la Société Chimique de Paris*, 3:98, 1865.

[66] Klavžar, S., Salem, K., and Taranenko, A. Maximum cardinality resonant sets and maximal alternating sets of hexagonal systems. *Computers & Mathematics with Applications*, 59(1):506–513, 2010.

[67] Klein, D. J. and Liu, X. Theorems for carbon cages. *Journal of Mathematical Chemistry*, 11(1):199–205, 1992.

[68] Klein, D. J., Yang, Y., and Ye, D. Homo–lumo gaps for sub-graphenic and sub-buckytubic species. In *Proc. R. Soc. A*, volume 471, page 20150183. The Royal Society, 2015.

[69] Kotzig, A. On the theory of finite graphs with a linear factor. *Mat.-Fyz. Caspis. 1959, 9, 73*, 91:235, 1991.

[70] Krätschmer, W., Lamb, L. D., Fostiropoulos, K., and Huffman, D. R. Solid c60: A new form of carbon. *Nature*, 347(6291):354–358, 1990.

[71] Kroto, H. W., Heath, J. R., O'Brien, S. C., Curl, R. F., Smalley, R. E., and others,. C 60: buckminsterfullerene. *Nature*, 318(6042):162–163, 1985.

[72] Kroto, H. The stability of the fullerenes cn, with n= 24, 28, 32, 36, 50, 60 and 70. *Nature*, 329(6139):529–531, 1987.

[73] Leach, A. R., Dolata, D. P., and Prout, K. Automated conformational analysis and structure generation: algorithms for molecular perception. *Journal of Chemical Information and Computer Sciences*, 30(3):316–324, 1990.

[74] Lee, C. J., Kang, Y.-M., Cho, K.-H., and No, K. T. A robust method for searching the smallest set of smallest rings with a path-included distance matrix. *Proceedings of the National Academy of Sciences*, 106(41):17355–17358, 2009.

[75] Li, F., Nie, J., Wu, J.-W., Zheng, Y., and Ma, J.-A. Stereoselective synthesis of fluorinated 2, 3-dihydroquinolin-4 (1 h)-ones via a one-pot multistep transformation. *The Journal of Organic Chemistry*, 77(5):2398–2406, 2012.

[76] Li, W., Qin, Z., and Zhang, H. Extremal hexagonal chains with respect to the coefficients sum of the permanental polynomial. *Applied Mathematics and Computation*, 291:30–38, 2016.

[77] Lin, C.-T., Wang, N.-J., Tseng, H.-Z., and Chou, T.-C. Synthesis and transannular reactions of a polycyclic compound containing three parallel face-to-face double bonds. *The Journal of Organic Chemistry*, 62(14):4857–4861, 1997.

[78] Liu, X., Klein, D., and Schmalz, T. Preferable fullerenes and clar-sextet cages. *Fullerenes, Nanotubes, and Carbon Nanostructures*, 2(4):405–422, 1994.

[79] Liu, X., Klein, D., Schmalz, T., and Seitz, W. Generation of carbon-cage polyhedra. *Journal of Computational Chemistry*, 12(10):1252–1259, 1991.

[80] Liu, X., Schmalz, T., and Klein, D. Favorable structures for higher fullerenes. *Chemical Physics Letters*, 188(5):550–554, 1992.

[81] Lokshtanov, D. Finding the longest isometric cycle in a graph. *Discrete Applied Mathematics*, 157(12):2670–2674, 2009.

[82] Lovász, L. and Plummer, M. D. *Matching theory*, volume 367. American Mathematical Soc., 2009.

[83] Lutz, G., Hunkler, D., Prinzbach, H., and Rihs, G. The pagodane route to dodecahedranes: Chemistry and structure of bissecododecahedranes. *Angewandte Chemie International Edition*, 28(3):298–300, 1989.

[84] Manolopoulos, D. E., May, J. C., and Down, S. E. Theoretical studies of the fullerenes: C34 to c70. *Chemical Physics Letters*, 181(2-3):105–111, 1991.

[85] Mao-Lin, Z. and Rong-Si, C. A maximal cover of hexagonal systems. *Graphs and Combinatorics*, 1(1):295–298, 1985.

[86] Marsh, R. E. and Herbstein, F. H. More space-group changes. *Acta Crystallographica Section B: Structural Science*, 44(1):77–88, 1988.

[87] Marx, U. C., Austermann, S., Bayer, P., Adermann, K., Ejchart, A., Sticht, H., Walter, S., Schmid, F.-X., Jaenicke, R., Forssmann, W.-G., and others,. Structure of human parathyroid hormone 1–37 in solution. *Journal of Biological Chemistry*, 270(25):15194–15202, 1995.

[88] McNaught, A. D. and McNaught, A. D. *Compendium of chemical terminology*, volume 1669. Blackwell Science Oxford, 1997.

[89] Michael, R. G. and David, S. J. Computers and intractability: a guide to the theory of np-completeness. *WH Freeman & Co., San Francisco*, 1979.

[90] Muller, P. Glossary of terms used in physical organic chemistry (iupac recommendations 1994). *Pure and Applied Chemistry*, 66(5):1077–1184, 1994.

[91] Myrvold, W. Exploring connections between chemistry, computer science and graph theory. *Graph Theory Appl*, 115(45), 2015.

[92] Nickelsen, H. Ringbegriffe in der chemie-dokumentation. *Nachr. Dok*, 3:121–123, 1971.

[93] Paton, K. An algorithm for finding a fundamental set of cycles of a graph. *Communications of the ACM*, 12(9):514–518, 1969.

[94] Petersen, J. Die theorie der regulären graphs. *Acta Mathematica*, 15(1):193–220, 1891.

[95] Petřina, A., Petříček, V., Malỳ, K., Šubrtová, V., Linek, A., and Hummel, L. The crystal and molecular structure of methyltriethylammonium μ-8, 8-oxa-3, 3-commo-bis (undecahydro-1, 2-dicarba-3-cobalta-closo-ododecaborate)(1-),[n (c2h5) 3ch3]+[o (c2b9h10) 2co]-. *Zeitschrift für Kristallographie-Crystalline Materials*, 154(1-4):217–226, 1981.

[96] Plotkin, M. Mathematical basis of ring-finding algorithms in cids. *Journal of Chemical Documentation*, 11(1):60–63, 1971.

[97] Randić, M. Conjugated circuits and resonance energies of benzenoid hydrocarbons. *Chemical Physics Letters*, 38(1):68–70, 1976.

[98] Randic, M. Aromaticity and conjugation. *Journal of the American Chemical Society*, 99(2):444–450, 1977.

[99] Randić, M. Aromaticity of polycyclic conjugated hydrocarbons. *Chemical Reviews*, 103(9):3449–3606, 200 publisher=ACS Publications.

[100] Randić, M. Graph theoretical approach to π-electron currents in polycyclic conjugated hydrocarbons. *Chemical Physics Letters*, 500(1):123–127, 2010.

[101] Randić, M. π-electron currents in polycyclic conjugated hydrocarbons of decreasing aromatic character and a novel structural definition of aromaticity. *Open Org. Chem. J*, 5:11–26, 2011.

[102] Randić, M., Novič, M., Vračko, M., Vukičević, D., and Plavšić, D. π-electron currents in polycyclic conjugated hydrocarbons: Coronene and its isomers having five and seven member rings. *International Journal of Quantum Chemistry*, 112(4):972–985, 2012.

[103] Randić, M., Vukičević, D., Balaban, A. T., Vračko, M., and Plavšić, D. Conjugated circuits currents in hexabenzocoronene and its derivatives formed by joining proximal carbons. *Journal of Computational Chemistry*, 33(11):1111–1122, 2012.

[104] Randić, M., Vukičević, D., Novič, M., and Plavšić, D. π-electron currents in larger fully aromatic benzenoids. *International Journal of Quantum Chemistry*, 112(12):2456–2462, 2012.

[105] Ren, S. On some structural properties of generalized fullerene graphs with 13 pentagonal faces. *Discrete Mathematics*, 311(16):1666–1669, 2011.

[106] Rietmeijer, F. J. M. and others,. *Natural fullerenes and related structures of elemental carbon*, volume 285. Springer, 2006.

[107] Rong, L., Liu, Q., Shi, Y., and Tang, J. Fluoradenes via palladium-catalyzed intramolecular arylation. *Chemical Communications*, 47(7):2155–2157, 2011.

[108] Roos-Kozel, B. L. and Jorgensen, W. L. Computer-assisted mechanistic evaluation of organic reactions. 2. perception of rings, aromaticity, and tautomers. *Journal of Chemical Information and Computer Sciences*, 21(2):101–111, 1981.

[109] Sah, C.-H. Combinatorial construction of fullerene structures. *Croatica Chemica Acta*, 66(1):1–12, 1993.

[110] Salami, M. and Ahmadi, M. A mathematical programming model for computing the fries number of a fullerene. *Applied Mathematical Modelling*, 39(18):5473–5479, 2015.

[111] Schaad, L. and Hess Jr, B. A. Hueckel molecular orbital. pi.-resonance energies. question of the. sigma. structure. *Journal of the American Chemical Society*, 94(9):3068–3074, 1972.

[112] Schwerdtfeger, P., Wirz, L. N., and Avery, J. The topology of fullerenes. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 5(1):96–145, 2015.

[113] Shea, K. M., Lee, K. L., and Danheiser, R. L. Synthesis and properties of 9-alkyl-and 9-arylcyclopenta [a] phenalenes. *Organic Letters*, 2(15):2353–2356, 2000.

[114] Shelley, C. A. Heuristic approach for displaying chemical structures. *Journal of Chemical Information and Computer Sciences*, 23(2):61–65, 1983.

[115] Students, N. S. F. u. G. Buckyballs, 2005.

[116] Tan, Y.-Z., Osella, S., Liu, Y., Yang, B., Beljonne, D., Feng, X., and Müllen, K. Sulfur-annulated hexa-peri-hexabenzocoronene decorated with phenylthio groups at the periphery. *Angewandte Chemie International Edition*, 54(10):2927–2931, 2015.

[117] Tan, Y.-Z., Yang, B., Parvez, K., Narita, A., Osella, S., Beljonne, D., Feng, X., and Müllen, K. Atomically precise edge chlorination of nanographenes and its application in graphene nanoribbons. *Nature Communications*, 4, 2013.

[118] Tiernan, J. C. An efficient search algorithm to find the elementary circuits of a graph. *Communications of the ACM*, 13(12):722–726, 1970.

[119] Tsang, S., Harris, P., Green, M., and others,. Thinning and opening of carbon nanotubes by oxidation using carbon dioxide. *Nature-London-*, 362:520–520, 1993.

[120] Valiant, L. G. The complexity of enumeration and reliability problems. *SIAM Journal on Computing*, 8(3):410–421, 1979.

[121] Venturini, J., Koudoumas, E., Couris, S., Janot, J., Seta, P., Mathis, C., and Leach, S. Optical limiting and nonlinear optical absorption properties of c 60–polystyrene star polymer films: C 60 concentration dependence. *Journal of Materials Chemistry*, 12(7):2071–2076, 2002.

[122] Welch Jr, J. T. A mechanical analysis of the cyclic structure of undirected linear graphs. *Journal of the ACM (JACM)*, 13(2):205–210, 1966.

[123] Wijesinghe, L. P., Lankage, B. S., Máille, G. M. Ó., Perera, S. D., Nolan, D., Wang, L., and Draper, S. M. Methoxy functionalisation: exerting synthetic control of the supramolecular and electronic structure of nitrogen-doped nanographenes. *Chemical Communications*, 50(73):10637–10640, 2014.

[124] Wikipedia, the free encyclopedia,. Buckminsterfullerene, 2008.

[125] Wikipedia, the free encyclopedia,. Dodecahedron c_{20}, 2008.

[126] Wikipedia, the free encyclopedia,. Buckminsterfullerene, 2009.

[127] Wipke, W. T. and Dyott, T. M. Use of ring assemblies in ring perception algorithm. *Journal of Chemical Information and Computer Sciences*, 15(3):140–147, 1975.

[128] Wu, Y., Ye, D., and Zhang, C.-Q. Uniquely forced perfect matching and unique 3-edge-coloring. *Discrete Applied Mathematics*, 215:203–207, 2016.

[129] Xu, R. and Xu, Y. *Modern inorganic synthetic chemistry*. Elsevier, 2010.

[130] Zamora, A. An algorithm for finding the smallest set of smallest rings. *Journal of Chemical Information and Computer Sciences*, 16(1):40–43, 1976.

[131] Zhang, H., Ye, D., and Shiu, W. C. Forcing matching numbers of fullerene graphs. *Discrete Applied Mathematics*, 158(5):573–582, 2010.

[132] Zhang, H. and Zhang, F. New lower bound on the number of perfect matchings in fullerene graphs. *Journal of Mathematical Chemistry*, 30(3):343–347, 2001.

[133] Zhao, S. and Zhang, H. Forcing polynomials of benzenoid parallelogram and its related benzenoids. *Applied Mathematics and Computation*, 284:209–218, 2016.

[134] Zhou, X. and Zhang, H. Clar sets and maximum forcing numbers of hexagonal systems. *MATCH Commun. Math. Comput. Chem*, 74:161–174, 2015.