

Special Issue Reprint

Mathematical and Molecular Topology

Edited by
Lorentz Jäntschi and Mihaela Tomescu

www.mdpi.com/journal/mathematics

Mathematical and Molecular Topology

Mathematical and Molecular Topology

Editors

Lorentz Jäntschi

Mihaela Tomescu

MDPI • Basel • Beijing • Wuhan • Barcelona • Belgrade • Manchester • Tokyo • Cluj • Tianjin



Editors

Lorentz Jäntschi
Technical University of Cluj-Napoca
Cluj-Napoca
Romania

Mihaela Tomescu
University of Petroșani
Petroșani
Romania

Editorial Office

MDPI
St. Alban-Anlage 66
4052 Basel, Switzerland

This is a reprint of articles from the Special Issue published online in the open access journal *Mathematics* (ISSN 2227-7390) (available at: https://www.mdpi.com/journal/mathematics/special-issues/Mathematical_Molecular_Topology).

For citation purposes, cite each article independently as indicated on the article page online and as indicated below:

LastName, A.A.; LastName, B.B.; LastName, C.C. Article Title. <i>Journal Name</i> Year , <i>Volume Number</i> , Page Range.
--

ISBN 978-3-0365-8352-5 (Hbk)

ISBN 978-3-0365-8353-2 (PDF)

© 2023 by the authors. Articles in this book are Open Access and distributed under the Creative Commons Attribution (CC BY) license, which allows users to download, copy and build upon published articles, as long as the author and publisher are properly credited, which ensures maximum dissemination and a wider impact of our publications.

The book as a whole is distributed by MDPI under the terms and conditions of the Creative Commons license CC BY-NC-ND.

Contents

About the Editors	vii
Preface to "Mathematical and Molecular Topology"	ix
Ria Gupta and Ananga Kumar Das Some Variants of Normal Čech Closure Spaces via Canonically Closed Sets Reprinted from: <i>Mathematics</i> 2021 , <i>9</i> , 1225, doi:10.3390/math9111225	1
Deepak Kumar, Sunil Kumar, Janak Raj Sharma and Lorentz Jantschi Convergence Analysis and Dynamical Nature of an Efficient Iterative Method in Banach Spaces Reprinted from: <i>Mathematics</i> 2021 , <i>9</i> , 2510, doi:10.3390/math9192510	11
Donatella Bálint and Lorentz Jäntschi Comparison of Molecular Geometry Optimization Methods Based on Molecular Descriptors Reprinted from: <i>Mathematics</i> 2021 , <i>9</i> , 2855, doi:10.3390/math9222855	27
Kristjan Reba, Matej Guid, Kati Rozman, Dušanka Janežič and Janez Konc Exact Maximum Clique Algorithm for Different Graph Types Using Machine Learning Reprinted from: <i>Mathematics</i> 2022 , <i>10</i> , 97, doi:10.3390/math10010097	39
Muhammad Usman Ghani, Muhammad Kashif Maqbool, Reny George, Austine Efut Ofem and Murat Cancan Entropies via Various Molecular Descriptors of Layer Structure of H_3BO_3 Reprinted from: <i>Mathematics</i> 2022 , <i>10</i> , 4831, doi:10.3390/math10244831	53

About the Editors

Lorentz Jäntschi

Lorentz Jäntschi is Professor at Department of Physics and Chemistry from Technical University (UTCN) and Associate at Doctoral School of Chemistry from Babeş-Bolyai University (UBB), both from Cluj-Napoca. Graduated (BSc+MSc) in Informatics (1995) and Physics and Chemistry (1997) from UBB. PhD in Chemistry (UBB, 2000). MSc (Agriculture, 2002), PhD (Horticulture, 2010) and PostDoc (Horticulture, 2013) from University of Agricultural Sciences and Veterinary Medicine of Cluj-Napoca. Habilitated in Chemistry (2013). PhD advisor in Chemistry (at UBB, from 2013). With experience in programming and in mathematical and chemical modelling and analysis. Current interests include distribution analysis (statistics), conformation analysis (chemistry) and simulation (development of Monte-Carlo methods). Former full member of European Society of Mathematical Chemistry. Different editor roles for *Notulae Scientia Biologicae*, *Notulae Botanicae Horti Agrobotanici Cluj-Napoca* (Production editor, AcademicPres), *Open Agriculture* (Advisory editor, DeGruyter), *Foundations* (Board member) and *Symmetry* (Academic editor), *Mathematics* (Guest and Collection editor), *Computation, Chemistry* and *IJMS* (Topic editor), *AIMS*, and *Mathematics* (Reviewer).

Mihaela Tomescu

Mihaela Tomescu is Assistant professor at Department of Mathematics and Informatics from University of Petroşani (UP), Hunedoara County since 2008, after a stage as teaching assistant (1999–2001) and as assistant (2001–2008). Graduated (BSc+MSc) in Mathematics and Physics (1996) from Faculty of Mathematics of West University of Timișoara (WUT). MSc (Mathematics, 2003) from WUT after a specialization in Operational models applied in optimal control and differential geometry. PhD (Mathematics, 2012) from WUT with the Thesis “Stability and dichotomy for discrete variational systems in Banach spaces” supervised by Prof. Mihail Megan. Scientific secretary (2020– to date) of the Department of Mathematics and Informatics from UP. Director (2023–to date) of Petroşani College for Non-University Tertiary Education from UP. Guest editor for *Mathematics* (MDPI).

Preface to “Mathematical and Molecular Topology”

Topology is one of the fundamental tools in relating entities. Topology naturally finds application in all fields of engineering, physical sciences, life sciences, social sciences, medicine, business and even the arts. The motivating insight behind topology is that some geometric problems depend not on the exact shape of the objects involved, but rather on the way they are put together. Circa 1750, Euler stated the polyhedron formula, $V - E + F = 2$ (where V , E , and F respectively indicate the number of vertices, edges, and faces of the polyhedron), which may be regarded as the first theorem, signaling the birth of topology. Subjects included in topology are algebraic topology and graph theory. A related branch to graph theory is molecular topology (with concerns of either chemical or biological structure).

The Special Issue *Mathematical and Molecular Topology* received 10 submitted manuscripts from which 5 were accepted and published (50% success rate). Two manuscripts are related to mathematical topology, while the other three are related to molecular topology.

Mathematical Topology Related Works

A preclosure operator or Čech closure operator is a map between subsets of a set, similar to a topological closure operator, except that it is not required to be idempotent. π -normal, weakly π -normal and κ -normal generalizations of normality in Čech closure space are defined and characterized using canonically closed sets in [1]. An important result connects those spaces: the class of κ -normal spaces contains both the classes of weakly π -normal and almost normal Čech closure spaces.

A Banach space B is a complete normed vector space which, in terms of generality, lies in between a metric space (that has a metric, but no norm) and a Hilbert space (that has an inner-product, and hence a norm, that in turn induces a metric). In [2], the local convergence analysis of a fifth order method and its multi-step version in Banach spaces is studied. Starting from hypotheses based on the first Fréchet-derivative only, the proposed approach provides a computable radius of convergence, error bounds on the distances involved, and estimates on the uniqueness of the provided solution. Taylor expansions of higher-order derivatives may not exist or may be very expensive or impossible to compute, so approaches that use them do not produce such estimates. The authors provide numerical examples to validate the theoretical results. Basins of attraction are used to represent convergence domains of the methods and the boundaries of the basins reveal symmetric fractal-like shapes.

Molecular Topology Related Works

Graph algorithms, or algorithms operating on graphs is conceptually a branch of combinatorial algorithms having uses in many problems, from graph coloring, to finding a perfect matching and computing the lowest common ancestor, and to graph-based searching, routing and network theory. Complete subgraphs (or cliques) are subsets of vertices which are all adjacent one to the other, while maximal cliques are the largest such substructures in a graph. Maximal cliques of protein graphs serve to determine their similarity and function of the protein. In [3], improvements based on machine learning are added to a Maximum Clique Dynamic algorithm for finding the maximum clique in large graphs such as are protein graphs. The work is based on an algorithm published in 2007 [4] and has been widely used in bioinformatics since then, which uses an empirically determined parameter, T_{limit} , that determines the algorithm's flow. In [3], the authors extended the MCQD algorithm with an initial phase of a machine learning-based prediction of the T_{limit} parameter that is best suited for each input graph. The authors note that a such adaptability to graph types based on state-of-the-art

machine learning is a novel approach that has not been used in most graph-theoretic algorithms. It is shown empirically that the resulting new algorithm MCQD-ML improves search speed on certain types of graphs, in particular molecular docking graphs used in drug design where they determine energetically favorable conformations of small molecules in a protein binding site. In such cases, the speed-up is twofold.

Entropy is a fundamental concept associated with measuring the state of disorder, randomness, or uncertainty. Clausius, Boltzmann, or Gibbs (statistical) entropy and Shannon's (information) entropy are practically one and the same. The values of entropy are key parameters driving the direction of spontaneous change for many commonplace events. In [5] the authors use various computational and mathematical techniques to calculate atom–bond connectivity entropy, atom–bond sum connectivity entropy, the newly defined Albertson entropy using the Albertson index, and the IRM entropy using the IRM index. An example of the calculation is given on H_3BO_3 by using the subdivision and line graph of the layer structure.

Complementing the molecular topology of a molecule, molecular geometry provides surface and structural representation, and is the key element differentiating among various molecules sharing the same topology. Various methods (from molecular mechanics and semi-empirical to ab initio and density functional theory) are involved in geometry optimization (energy minimization) of the molecules. Having as template a series of 20 amino acids with near-optimal geometry were used to reach the optimum geometries by using 39 methods (HF, MP2, B3LYP included) in [6]. Next, a pool of molecular descriptors was used to characterize each optimized geometric conformation and cluster analysis and principal component analysis were performed to get the similarities between the different optimization methods. As authors noted, the results after the analysis are classified into three main groups and can provide alternate selection accordingly to solve different types of problems.

Several topology problems on topics such cohomology, compactness, connectedness, homeomorphisms, homology, homotopy, symmetry and similarity are still to be explored to provide further insight on theoretical aspects of mathematical and molecular topology and their applications.

References

1. Gupta, R.; Das, A.K. Some Variants of Normal Čech Closure Spaces via Canonically Closed Sets. *Mathematics* **2021**, *9*, 1225. <https://doi.org/10.3390/math9111225>.
2. Kumar, D.; Kumar, S.; Sharma, J.R.; Jantschi, L. Convergence Analysis and Dynamical Nature of an Efficient Iterative Method in Banach Spaces. *Mathematics* **2021**, *9*, 2510. <https://doi.org/10.3390/math9192510>.
3. Reba, K.; Guid, M.; Rozman, K.; Janežič, D.; Konc, J. Exact Maximum Clique Algorithm for Different Graph Types Using Machine Learning. *Mathematics* **2022**, *10*, 97. <https://doi.org/10.3390/math10010097>.
4. Konc, J.; Janezic, D. An improved branch and bound algorithm for the maximum clique problem. *Match Commun. Math. Comput. Chem.* **2007**, *58*, 569–590.
5. Ghani, M.U.; Kashif Maqbool, M.; George, R.; Ofem, A.E.; Cancan, M. Entropies Via Various Molecular Descriptors of Layer Structure of H_3BO_3 . *Mathematics* **2022**, *10*, 4831. <https://doi.org/10.3390/math10244831>.
6. Bálint, D.; Jantschi, L. Comparison of Molecular Geometry Optimization Methods Based on Molecular Descriptors. *Mathematics* **2021**, *9*, 2855. <https://doi.org/10.3390/math9222855>.

Lorentz Jantschi and Mihaela Tomescu

Editors

Article

Some Variants of Normal Čech Closure Spaces via Canonically Closed Sets

Ria Gupta and Ananga Kumar Das *

School of Mathematics, Shri Mata Vaishno Devi University, Katra, Jammu and Kashmir 182320, India; 14dmt001@smvdu.ac.in or riyag4289@gmail.com

* Correspondence: akdasdu@yahoo.co.in or ak.das@smvdu.ac.in

Abstract: New generalizations of normality in Čech closure space such as π -normal, weakly π -normal and κ -normal are introduced and studied using canonically closed sets. It is observed that the class of κ -normal spaces contains both the classes of weakly π -normal and almost normal Čech closure spaces.

Keywords: closure space; canonically closed; weakly normal; almost normal; π -normal; weakly π -normal; κ -normal

MSC: 04A05; 54D15

Citation: Gupta, R.; Das, A.K. Some Variants of Normal Čech Closure Spaces via Canonically Closed Sets. *Mathematics* **2021**, *9*, 1225. <https://doi.org/10.3390/math9111225>

Academic Editors: Lorentz Jäntschi and Mihaela Tomescu

Received: 12 April 2021
Accepted: 24 May 2021
Published: 27 May 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction and Preliminaries

It is evident from the literature that topological structures which are more general than the classical topology are more suitable for the study of digital topology, image processing, network theory, pattern recognition and related areas. Various generalized structures such as closure spaces, generalized closure spaces, Čech closure spaces, generalized topologies (GT), weak structures (WS), Generalized neighborhood systems (GNS) etc. were introduced and studied in the past (see [1–5]). However, recently Čech closure spaces attracted the attention of researchers due to its possibility of application in other applied fields discussed above. Usefulness of this Čech closure setting in variety of allied fields such as digital topology, computer graphics, image processing and pattern recognition are available in the literature [6–9]. Čech closure space was defined by Čech [1], are obtained from Kuratowski [10] closure operator by omitting the idempotent condition. In this setting Galton [11] studied the motion of an object in terms of a function giving its position at each time and systematically investigated what a continuous motion looks like. J. Šlapal [6] observed that this structure is more suitable than others for application in digital topology because Čech closure spaces are well-behaved with respect to connectedness. Allam et al. [12,13] introduced a new method for generating closure spaces via a binary relation which was subsequently used by G. Liu [14] to establish a one-to-one correspondence between quasi discrete closures and reflexive relation. Furthermore, J. Šlapal and John L. Pfaltz [15] studied network structures via associated closure operators. Higher separation axioms in Čech closure space was introduced by Barbel M. R. Stadler and F. Peter Stadler [16] in 2003 and discussed the concept of Urysohn functions, normal, regular, completely normal etc. in the form of neighborhood. In 2018 Gupta and Das [17] introduced higher separation axioms via relation. Since normality is an important topological property, many weak variants of normality introduced and studied in the past to properly study normality in general topology (See [18–22]). In the present paper, we introduced some variants of normality in Čech closure space as π -normal, weakly π -normal and κ -normal using canonically closed sets. It is observed that some characterizations of normality and almost normality which holds in topological spaces may not hold in

Čech closure spaces. Further relation between newly defined notions and already defined notions was also investigated.

A closure space is a pair (X, cl) , where X is any set and closure $cl : P(X) \rightarrow P(X)$ is a function associating with each subset $A \subseteq X$ to a subset $cl(A) \subseteq X$, called the closure of A , such that $cl(\emptyset) = \emptyset$, $A \subseteq cl(A)$, $cl(A \cup B) = cl(A) \cup cl(B)$. With any closure cl for a set X there is associated the interior operation int_{cl} , usually denoted by int , which is a single-valued relation on $P(X)$ ranging in $P(X)$ such that for each $A \subseteq X$, $int_{cl}(A) = X - cl(X - A)$. The set $int_{cl}(A)$ is called the interior of A in (X, cl) . In a closure space (X, cl) , a set A is closed if $cl(A) = A$ and open if its complement is closed i.e., if $cl(X - A) = (X - A)$. In other words, a set is open if and only if $int(A) = A$. Additionally, from closure axioms we have $cl(A \cap B) \subseteq cl(A) \cap cl(B)$ and $int(A) \cup int(B) \subseteq int(A \cup B)$. In a Čech closure space a canonically closed (regularly closed) set is a closed set A of X such that $cl(int(A)) = A$ and a canonically open (regularly open) set is an open set U of X such that $int(cl(U)) = U$.

Definition 1. [23] A Čech closure space (X, cl) is said to be

1. normal if for every two disjoint closed sets $A = cl(A)$ and $B = cl(B)$ there exist disjoint open sets U and V containing $cl(A)$ and $cl(B)$ respectively.
2. almost normal if for every two disjoint closed sets $cl(A) = A$ and $cl(B) = B$ out of which one is canonically closed there exist disjoint open sets U and V containing $cl(A)$ and $cl(B)$ respectively.
3. weakly normal if for every two disjoint closed sets $cl(A) = A$ and $cl(B) = B$ there exists an open set U such that $A \subseteq U$ and $int(cl(U)) \cap B = \emptyset$.

Remark 1. The notion of normality defined above in the Definition 1 is different from the notion of normality defined in [1]. A closure space is said to be normal [1] if every pair of sets with disjoint closures are separated by disjoint neighborhoods. The disjoint sets considered by Čech for separation in the definition of normality are not necessarily closed sets and neighborhoods need not be open. Throughout the present paper, we have taken the notion of normality only in the sense of Definition 1.

Lemma 1. [1] If U and V are subsets of a closure space (X, cl) such that $U \subseteq V$ then $cl(U) \subseteq cl(V)$.

Theorem 1. [23] Suppose (X, cl) is a Čech closure space such that $int(cl(U))$ is canonically open for every open set U . Then (X, cl) is weakly normal and almost normal implies (X, cl) is normal.

2. Variants of Normal Čech Closure Space

Definition 2. Let (X, cl) be a Čech closure space then A is said to be π -closed if it is equal to the intersection of two canonically closed set.

Example 1. Let $X = \{a, b, c, d\}$ be the set and define $cl : P(X) \rightarrow P(X)$ as $cl(\{a\}) = \{a\}$, $cl(\{b\}) = cl(\{a, b\}) = \{a, b\}$, $cl(\{c\}) = cl(\{a, c\}) = cl(\{c, d\}) = cl(\{a, c, d\}) = \{a, c, d\}$, $cl(\{d\}) = \{d\}$, $cl(\{a, d\}) = \{a, d\}$, $cl(\{b, c\}) = cl(\{a, b, c\}) = cl(\{b, c, d\}) = cl(X) = X$, $cl(\{b, d\}) = cl(\{a, b, d\}) = \{a, b, d\}$, $cl(\emptyset) = \emptyset$. Here, the set $A = \{a\}$ is π -closed as it is the intersection of two canonically closed set i.e., $\{a, c, d\}$ and $\{a, b\}$ but $\{a\}$ is not canonically closed. In this Čech closure space, $cl(A) = \{d\} = A$ is closed but not π -closed as it is not equal to the intersection of two canonically closed set.

The implications in Figure 1 are obvious from the definitions. However, none of these implications is reversible as shown in the above example.

canonically closed \longrightarrow π -closed \longrightarrow closed

Figure 1. Interrelation of types of closed sets.

Definition 3. A Čech closure space (X, cl) is π -normal if for every two disjoint closed sets one of which is π -closed there exist two disjoint open sets U and V containing the closed set and the π -closed set respectively.

It is obvious that in a Čech closure space (X, cl) , every normal space is π -normal. However, the converse need not be true as shown below.

Example 2. A Čech closure space which is π -normal but not normal.

Let $X = Y \cup \{p, q\}$ be an infinite set, then any set $A \in P(X)$ is one of the following four types of sets:

Type-I: A is finite in X .

Type-II: A is infinite in Y such that $p \notin A$ and $q \notin A$.

Type-III: $(Y - A)$ is finite and A contains either p or q .

Type-IV: $(Y - A)$ is finite and A contains both p and q .

Define $cl : P(X) \rightarrow P(X)$ by

$$cl(A) = \begin{cases} A, & \text{if } A \text{ is of type-I;} \\ A \cup \{p, q\}, & \text{if } A \text{ is of type-II;} \\ A \cup \{p, q\}, & \text{if } A \text{ is of type-III;} \\ A, & \text{if } A \text{ is of type-IV.} \end{cases}$$

In this Čech closure space, type-I and type-IV sets are closed sets. A set U is open if U is an infinite set containing p and/or q whose complement is finite. Additionally, a finite set U in Y whose complement is infinite is an open set in X . In this space only two types of sets are canonically closed. i.e., (1) Every finite set in Y is canonically closed (2) a set containing both p and q whose complement is finite in Y is canonically closed. This space is π -normal but not normal because for two disjoint closed sets $A = C \cup \{p\}$ and $B = D \cup \{q\}$, where C and D are finite in Y , there does not exist disjoint open sets satisfying the condition of normal Čech closure space.

Example 3. A space which is not π -normal.

Let X be the set of integers defined by

$$cl(\{x\}) = \begin{cases} x, & \text{if } x \text{ is even;} \\ \{x - 1, x, x + 1\}, & \text{if } x \text{ is odd.} \end{cases}$$

and $cl(A) = \bigcup_{x \in A} cl(x)$.

This Čech closure space is not π -normal because for the π -closed set $cl(A) = \{4\} = A$ and a closed set $cl(B) = \{0, 1, 2\} = B$ there does not exist disjoint open sets containing $cl(A)$ and $cl(B)$ respectively.

Following examples establish that the notion of weak normality defined earlier, and the notion of π -normality are independent notions.

Example 4. A space which is weakly normal but not π -normal.

Let X be the set of positive integers. Define $cl : P(X) \rightarrow P(X)$ as defined in Example 3. Here, the Čech closure space (X, cl) is weakly normal but not π -normal as shown in Example 3.

Example 5. A space which is π -normal but not weakly normal.

Let $X = \{a, b, c, d\}$ be the set and define $cl : P(X) \rightarrow P(X)$ as $cl(\{a\}) = \{a\}$, $cl(\{b\}) = \{b\}$,

$cl(\{c\}) = \{a, c, d\}$, $cl(\{d\}) = \{d\}$, $cl(\{a, b\}) = \{a, b\}$, $cl(\{a, c\}) = \{a, c, d\}$, $cl(\{a, d\}) = \{a, d\}$, $cl(\{b, c\}) = X$, $cl(\{b, d\}) = \{b, d\}$, $cl(\{c, d\}) = \{a, c, d\}$, $cl(\{a, b, c\}) = X$, $cl(\{a, b, d\}) = \{a, b, d\}$, $cl(\{a, c, d\}) = \{a, c, d\}$, $cl(\{b, c, d\}) = X$, $cl(X) = X$, $cl(\emptyset) = \emptyset$. Here, (X, cl) is a π -normal Čech closure space which fails to be weakly normal because for two disjoint closed sets $A = \{a\} = cl(A)$ and $B = \{d\} = cl(B)$ there does not exist an open set U such that $cl(A) \subseteq U$ and $int(cl(U)) \cap B = \emptyset$.

Theorem 2. *If (X, cl) is a π -normal Čech closure space then for every π -closed set $cl(A) = A$ and for every open set U containing $cl(A)$ there exists an open set V such that $cl(A) \subseteq V \subseteq cl(V) \subseteq U$.*

Proof. Let $cl(A) = A$ be a π -closed set and U be an open set containing $cl(A)$. Since, (X, cl) is π -normal, there exist disjoint open sets V and W such that $cl(A) \subseteq V$ and $(X - U) \subseteq W$ implies $V \subseteq (X - W)$. Thus, by Lemma 1, $cl(V) \subseteq cl(X - W)$ implies $W \subseteq X - cl(V)$. Therefore, $(X - U) \subseteq W \subseteq X - cl(V)$ and hence $cl(A) \subseteq V \subseteq cl(V) \subseteq U$. \square

Theorem 3. *If (X, cl) is a π -normal Čech closure space then for every closed set $cl(A) = A$ and for every π -open set U containing $cl(A)$ there exists an open set V such that $cl(A) \subseteq V \subseteq cl(V) \subseteq U$.*

Proof. Let $cl(A) = A$ be a closed set and U be a π -open set containing $cl(A)$ implies $(X - U)$ is a π -closed set which is disjoint from the closed set A . Since, (X, cl) is π -normal, there exist disjoint open sets V and W such that $cl(A) \subseteq V$ and $(X - U) \subseteq W$. Thus, $V \subseteq (X - W)$ implies $cl(V) \subseteq cl(X - W) = (X - W)$, and so, $W \subseteq (X - cl(V))$. Therefore, $(X - U) \subseteq W \subseteq (X - cl(V))$ and hence $cl(A) \subseteq V \subseteq cl(V) \subseteq U$. \square

Definition 4. [24] *A Čech closure space (X, cl) is said to be regular if for a closed set $cl(A) = A$ and a point $x \notin cl(A)$ there exist disjoint open sets U and V such that $x \in U$ and $cl(A) \subseteq V$.*

Definition 5. [1] *A Čech closure space is said to be*

1. T_1 if for two distinct points x and y , we have $x \notin cl(\{y\})$ and $y \notin cl(\{x\})$.
2. T_2 if any two distinct points x and y are separated.

Remark 2. *In a Čech closure space, every normal T_1 space is regular and T_2 . but if we replace normal by π -normal then the result need not be true. Consider the space defined in Example 2 which is π -normal and T_1 but neither T_2 nor regular. The space is not T_2 because disjoint points 'p' and 'q' cannot be separated and is not regular because for closed set $A = C \cup \{p\}$ where C is finite in Y and a point 'q' there does not exist disjoint open sets satisfying the required condition.*

Definition 6. [24] *A Čech closure space is said to be almost regular if for canonically closed set $cl(int(A)) = A$ and a point $x \notin cl(int(A))$ there exist disjoint open sets U and V such that $x \in U$ and $cl(int(A)) \subseteq V$.*

Theorem 4. *In a Čech closure space, every π -normal T_1 space is almost regular.*

Proof. let $cl(int(A)) = A$ be a canonically closed set and $x \notin cl(int(A))$ be a point. Since the space is a T_1 Čech closure space, the singleton set $\{x\}$ is closed. As every canonically closed set is π -closed, by π -normality there exist disjoint open sets U and V such that $cl(int(A)) \subseteq U$ and $\{x\} \subseteq V$. Hence (X, cl) is an almost regular Čech closure space. \square

Definition 7. *A Čech closure space is said to be weakly π -normal if for two disjoint π -closed sets there exist disjoint open sets separating them.*

Definition 8. *A Čech closure space is said to be κ -normal if for two disjoint canonically closed sets A and B there exist disjoint open sets U and V containing A and B respectively.*

From the definitions it is observed that every π -normal space is weakly π -normal, every weakly π -normal space as well as every almost normal space is κ -normal. Thus, the implications in Figure 2 are obvious but none of them is reversible which is exhibited below by Examples.

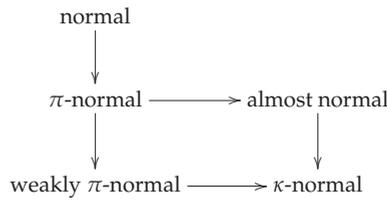


Figure 2. Interrelation of variants of normality.

Example 6. A Čech closure space which is weakly π -normal but not π -normal.

Let $X = \{a, b, c, d\}$ be a set and define $cl : P(X) \rightarrow P(X)$ as $cl(\{a\}) = cl(\{b\}) = cl(\{a, b\}) = \{a, b\}$, $cl(\{c\}) = \{c\}$, $cl(\{d\}) = cl(\{c, d\}) = \{b, c, d\}$, $cl(\{a, c\}) = cl(\{b, c\}) = cl(\{a, b, c\}) = \{a, b, c\}$, $cl(\{a, d\}) = cl(\{b, d\}) = cl(\{a, b, d\}) = cl(\{a, c, d\}) = cl(\{b, c, d\}) = cl(X) = X$, $cl(\emptyset) = \emptyset$. This space is vacuously weakly π -normal but not π -normal because for the π -closed set $\{a, b\}$ and a closed set $\{c\}$, there does not exist disjoint open sets containing $\{a, b\}$ and $\{c\}$.

Example 7. A Čech closure space which is weakly π -normal but not almost normal.

Let $X = \{a, b, c, d\}$ be the set and define $cl : P(X) \rightarrow P(X)$ as $cl(\{a\}) = \{a\}$, $cl(\{b\}) = cl(\{a, b\}) = \{a, b\}$, $cl(\{c\}) = cl(\{a, c\}) = cl(\{c, d\}) = cl(\{a, c, d\}) = \{a, c, d\}$, $cl(\{d\}) = \{d\}$, $cl(\{a, d\}) = \{a, d\}$, $cl(\{b, d\}) = cl(\{a, b, d\}) = \{a, b, d\}$, $cl(\{b, c\}) = cl(\{a, b, c\}) = cl(\{b, c, d\}) = cl(X) = X$, $cl(\emptyset) = \emptyset$. Clearly, (X, cl) is a Čech closure space which is vacuously weakly π -normal but not almost normal because for the canonically closed set $cl(int(A)) = \{a, b\} = A$ and the closed set $cl(B) = \{d\} = B$ there does not exist disjoint open sets containing A and B respectively.

Example 8. A Čech closure space which is κ -normal but not almost normal.

The Čech closure space defined in Example 7 is vacuously κ -normal but not almost normal as shown in Example 7.

Example 9. A Čech closure space which is κ -normal.

Let $X = Y \cup \{p, q\}$ be an infinite set. Define $cl : P(X) \rightarrow P(X)$ as in Example 2. Here, the closure space (X, cl) is κ -normal as for two disjoint canonically closed sets there exist disjoint open sets containing them.

Example 10. A Čech closure space which is not κ -normal.

Let X be the set of integers and define $cl : P(X) \rightarrow P(X)$ as shown in Example 3. This Čech closure space (X, cl) is not κ -normal because for two disjoint canonically closed sets $A = \{0, 1, 2\} = cl(int(A))$ and $B = \{4, 5, 6\} = cl(int(B))$ there does not exist disjoint open sets containing them.

Theorem 5. If (X, cl) is a weakly π -normal Čech closure space then for every π -closed set A and for every π -open set U containing A there exists an open set V such that $A \subseteq V \subseteq cl(V) \subseteq U$.

Proof. Let $cl(A) = A$ be a π -closed set and U be a π -open set containing $cl(A)$. Since, (X, cl) is weakly π normal, there exist disjoint open sets V and W such that $cl(A) \subseteq V$ and $(X - U) \subseteq W$. Thus, $V \subseteq X - W$ implies $cl(V) \subseteq cl(X - W) = (X - W)$. Therefore, $A \subseteq V \subseteq cl(V) \subseteq U$. \square

Theorem 6. *If (X, cl) is a κ -normal Čech closure space then for every canonically closed set $cl(int(A)) = A$ and for every canonically open set $int(cl(U)) = U$ containing $cl(int(A))$ there exists an open set V such that $cl(int(A)) \subseteq V \subseteq cl(V) \subseteq int(cl(U))$.*

Proof. Proof of this theorem is similar to the proof of Theorem 5. \square

Theorem 7. *Suppose (X, cl) is a Čech closure space such that $int(cl(U))$ is canonically open for every open set U . Then (X, cl) is weakly normal and κ -normal implies (X, cl) is almost normal.*

Proof. let $cl(int(A)) = A$ be a canonically closed set and $cl(B) = B$ be a closed set disjoint from canonically closed set $cl(int(A)) = A$. Since, (X, cl) is a weakly normal Čech closure space, there exists an open set U such that $A \subseteq U$ and $int(cl(U)) \cap B = \emptyset$. Since $int(cl(U))$ is canonically open, $X - int(cl(U))$ is canonically closed containing $cl(B)$. Thus, by κ -normality there exist disjoint open sets P and Q such that $cl(int(A)) \subseteq P$ and $cl(B) \subseteq X - (int(cl(U))) \subseteq Q$. Hence (X, cl) is an almost normal Čech closure space. \square

Theorem 8. *Suppose (X, cl) is a T_1 Čech closure space such that $int(cl(U))$ is canonically open for every open set U . Then (X, cl) is weakly π -normal and weakly normal implies (X, cl) is almost regular.*

Proof. let $cl(int(A)) = A$ be a canonically closed set and $x \notin cl(int(A))$ be a point. Since (X, cl) is T_1 , the singleton set $\{x\}$ is closed. By weak normality, there exists an open set U such that $A \subseteq U$ and $int(cl(U)) \cap \{x\} = \emptyset$. Since $int(cl(U))$ is canonically open, $X - int(cl(U))$ is canonically closed containing $\{x\}$. Thus, by weak π -normality, there exist disjoint open sets P and Q such that $cl(int(A)) \subseteq P$ and $\{x\} \subseteq X - (int(cl(U))) \subseteq Q$. Hence (X, cl) is an almost regular Čech closure space. \square

It is clear from Example 11 that the T_1 axiom cannot be relaxed from the Theorem 8 as the space is weakly π -normal and weakly normal but not almost regular.

Example 11. *Let $X = \{a, b, c\}$ be the set and define $cl : P(X) \rightarrow P(X)$ as $cl(\{a\}) = \{a\}$, $cl(\{b\}) = \{a, b\}$, $cl(\{c\}) = \{a, c\}$, $cl(\{a, b\}) = \{a, b\}$, $cl(\{a, c\}) = \{a, c\}$, $cl(\{b, c\}) = cl(X) = X$, $cl(\emptyset) = \emptyset$. Clearly, (X, cl) is a Čech closure space which is weakly π -normal and weakly normal but not almost regular.*

Definition 9. [24] *A Čech closure space (X, cl) is said to be β -normal if for two disjoint closed sets $cl(A) = A$ and $cl(B) = B$ there exist disjoint open sets U and V whose closures are disjoint such that $cl(A \cap U) = cl(A)$ and $cl(B \cap V) = cl(B)$.*

Definition 10. [24] *A Čech closure space is extremally disconnected (E. D) if for every open set U , $cl(U)$ is open.*

Example 12. *A Space which is extremally disconnected.*

Let $X = \{a, b, c, d\}$ be the set. Define $cl : P(X) \rightarrow P(X)$ as $cl(\{a\}) = cl(\{a, c\}) = \{a, c\}$, $cl(\{b\}) = \{b\}$, $cl(\{c\}) = \{c\}$, $cl(\{d\}) = cl(\{b, d\}) = \{b, d\}$, $cl(\{a, b\}) = cl(\{a, b, c\}) = \{a, b, c\}$, $cl(\{b, c\}) = \{b, c\}$, $cl(\{c, d\}) = cl(\{b, c, d\}) = \{b, c, d\}$, $cl(\{a, d\}) = cl(\{a, b, d\}) = cl(\{a, c, d\}) = cl(X) = X$, $cl(\emptyset) = \emptyset$. In this space, closure of every open set is open. Thus, the space is extremally disconnected.

Theorem 9. *In an extremally disconnected Čech closure space (X, cl) , every β -normal space is κ -normal.*

Proof. Let $cl(int(A)) = A$ and $cl(int(B)) = B$ be two disjoint canonically closed sets. Thus, $cl(int(A))$ and $cl(int(B))$ are two disjoint closed sets. We must show (X, cl) is κ -normal. Since (X, cl) is β -normal, there exist disjoint open sets U and V such that $cl(cl(A) \cap U) =$

$cl(A), cl(cl(B) \cap V) = cl(B)$ and $cl(U) \cap cl(V) = \emptyset$. Thus, $cl(A) = cl(cl(A) \cap U) \subseteq cl(U)$ and $cl(B) = cl(cl(B) \cap V) \subseteq cl(V)$. By extremally disconnectedness of (X, cl) , $cl(U)$ and $cl(V)$ are two disjoint open sets containing $cl(A)$ and $cl(B)$ respectively. Hence (X, cl) is κ -normal. \square

Example 13. A Čech closure space which is κ -normal but not β -normal.

Let $X = Y \cup \{p, q\}$ be an infinite set. Define $cl : P(X) \rightarrow P(X)$ as in Example 2. Here, the closure space (X, cl) is κ -normal but not β -normal because for two disjoint closed sets $cl(A) = C \cup \{p\}$ and $cl(B) = D \cup \{q\}$, where C and D are finite in Y , there does not exist disjoint open sets satisfying the condition of β -normal Čech closure space.

Example 14. Let X be an infinite set. Define $cl : P(X) \rightarrow P(X)$ as defined in [1] by

$$cl(A) = \begin{cases} A, & \text{if } A \text{ is finite;} \\ X, & \text{otherwise.} \end{cases}$$

Here, (X, cl) is a Čech closure space which is T_1 almost normal but not regular because for closed set $cl(A) = A$ and a point disjoint from the closed set A there does not exist disjoint open sets separating them.

The following theorem directly follows from the Theorem 1.

Theorem 10. Suppose (X, cl) is a weakly normal Čech closure space such that $int(cl(U))$ is canonically open for every open set U . Then following are equivalent:

1. (X, cl) is normal.
2. (X, cl) is π -normal.
3. (X, cl) is weakly π -normal.
4. (X, cl) is κ -normal.
5. (X, cl) is almost normal.

3. Discussion and Conclusions

Closure space was first appeared in 1966 in the book “Topological Spaces” is popularly known as Čech closure space in the name of the author of the book E. Čech. After many decades of its introduction, it is now slowly becoming objects of increasing interest and importance. The purpose of this discussion is to discuss some important developments in this area in the last two decades. In 2003, some higher separation axioms including completely regular and completely normal spaces are studied in closure setting by Stadler et al. [16]. In 2008, Dimitrije Andrijević and others [25] considered families of subset of a closure space equipped with different Vietoris-like topologies and studied properties such as connectedness and compactness of the space and its hyperspaces. Subsequently in 2010, they generalized the notions of the compact-open and graph topology to the set of functions between two Čech closure spaces [26]. Additionally, they investigated how the separation properties (T_0 , T_1 and regular) of the initial spaces are related to those of function spaces.

Recently, in 2021, Antonio Rieser [27] studied homotopy theory on the category of Čech closure spaces, whose objects are sets endowed with a Čech closure operator and whose morphisms are the continuous maps between them. They introduced some new classes of Čech closure structures on metric spaces, graphs, and simplicial complexes.

Another approach of generating closure spaces via a binary relation was also adopted by many researchers to address various issues in mathematics and other allied fields (see [12–15]). In [17], we have introduced and studied some new separation axioms on closure spaces generated through binary relations.

Apart from this, Junsheng Qiao [28] shown that the category of Čech closure spaces can be embedded in the category of stratified L-Čech closure spaces as a coreflective

full subcategory. Perfilieva et al. [29] investigated the relationship between L-Fuzzy Čech closure spaces and L-Fuzzy co-topological spaces from the categorical viewpoint. Relational variants of categories related to L-Fuzzy closure spaces was studied in [30].

In this paper, we have defined and investigated few variants of normality in Čech closure spaces using canonically closed sets. Normality is an important topological property, and its importance is due to its behaviour as it behaves differently from other separation axioms for subspaces and products. Additionally, the class of normal spaces are more general than the important class of compact Hausdorff spaces. Normality involves separation of closed sets by open sets. On the other hand, in digital image processing a picture needs to be segmented into subsets where relationship of these subset from other neighboring subsets and adjoining points plays a prominent role for the processing of images. Such types of relationships between sets/points are either geometrical or topological. Geometrical relation involves position of points whereas topological relation involves concepts such as adjacency, neighborhood, separation, connectedness and compactness. So, the possibility of application of the notions defined in this paper in digital topology and digital image processing cannot be ruled out.

Author Contributions: Investigation: R.G. and A.K.D.; Supervision: A.K.D. All authors have read and agreed to the published version of the manuscript.

Funding: The first author received fellowship from Department of Science and Technology (DST), Government of India (IF140967).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors are thankful to the referee for many valuable remarks that essentially improved the paper, and the first author is thankful to Department of Science and Technology (DST), Government of India for awarding INSPIRE fellowship (IF140967).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Čech, E. *Topological Spaces (Revised edition by Zdeněk Frolík and Miroslav Katětov)*; Publishing House of The Czechoslovak Academy of Sciences: Prague, Czech Republic, 1966.
2. Császár, Á. Generalized Topology, Generalized Continuity. *Acta Math. Hung.* **2002**, *96*, 351–357. [[CrossRef](#)]
3. Császár, Á. Weak structures. *Acta Math. Hung.* **2011**, *131*, 193–195. [[CrossRef](#)]
4. Das, A.K. A note on weak structures due to Császár. *Bul. Acad. Ştiinţe Mold. Mat.* **2015**, *78*, 114–116.
5. Min, W.K. Some Results on generalized topological spaces and generalized systems. *Acta Math. Hung.* **2005**, *108*, 171–181. [[CrossRef](#)]
6. Šlapal, J. Closure operations for digital topology. *Theor. Comput. Sci.* **2003**, *305*, 457–471. [[CrossRef](#)]
7. Khalimsky, E.D.; Kopperman, R.; Meyer, P.R. Computer graphics and connected topologies on finite ordered sets. *Topol. Appl.* **1990**, *36*, 1–17. [[CrossRef](#)]
8. Rosenfeld, A.; Kak, A.C. *Digital Picture Processing*; Academic Press: New York, NY, USA, 1976.
9. Rosenfeld, A. Digital topology. *Amer. Math. Mon.* **1979**, *86*, 621–630. [[CrossRef](#)]
10. Kuratowski, C. Sur l'operation de l'Analysis Situs. *Fundam. Math.* **1922**, *3*, 182–199. [[CrossRef](#)]
11. Galton, A. A generalized topological view of motion in discrete space. *Theoret. Comput. Sci.* **2003**, *305*, 111–134. [[CrossRef](#)]
12. Allam, A.A.; Bakeir, M.Y.; Abo-Tabl, E.A. New approach for closure spaces by relations. *Acta Math. Acad. Paedagog. Nyregyhziensis* **2006**, *22*, 285–304.
13. Allam, A.A.; Bakeir, M.Y.; Abo-Tabl, E.A. Some methods for generating topologies by relations. *Bull. Malays. Math. Sci. Soc.* **2008**, *31*, 1–11.
14. Liu, G. Closures and topological closures in quasi-discrete closure spaces. *Appl. Math. Lett.* **2010**, *23*, 772–776. [[CrossRef](#)]
15. Šlapal, J.; Pfaltz, J.L. Closure operators associated with networks. *Hacet. J. Math. Stat.* **2017**, *46*, 91–101.
16. Stadler, B.M.R.; Stadler, P.F. Higher separation axioms in generalized closure spaces. *Comment. Math. Prace Mat.* **2003**, *43*, 257–273.
17. Gupta, R.; Das, A.K. New separation axioms on closure spaces generated by relations. *Proc. Jangjeon Math. Soc.* **2018**, *21*, 23–31.
18. Das, A.K.; Bhat, P.; Tartir, J.K. On a simultaneous generalization of β -normality and almost β -normality. *Filomat* **2017**, *31*, 425–430. [[CrossRef](#)]

19. Das, A.K. Simultaneous generalizations of regularity and normality. *Eur. J. Pure Appl. Math.* **2011**, *4*, 34–41.
20. Das, A.K.; Raina, S.S. On Relative β -Normality. *Acta Math. Hung.* **2020**, *160*, 468–477. [[CrossRef](#)]
21. Kalantan, L.; Saeed, M.M. L-normality. *Topol. Proc.* **2017**, *50*, 141–149.
22. Singal, M.K.; Singal, A.R. Mildly normal spaces. *Kyungpook Math. J.* **1973**, *13*, 27–31.
23. Das, A.K.; Gupta, R. *On Some Variants of Normality in Čech Closure Spaces*; Unpublished work.
24. Gupta, R.; Das, A.K. *On β -Normal Čech Closure Spaces*; Unpublished work.
25. Andrijević, D.; Jelić, M.; Mršević, M. Some properties of hyperspaces of Čech closure spaces with Vietoris-like topologies. *Filomat* **2010**, *24*, 53–61. [[CrossRef](#)]
26. Andrijević, D.; Jelić, M.; Mršević, M. On function spaces topologies in the setting of Čech closure spaces. *Topology Appl.* **2011**, *158*, 1390–1395. [[CrossRef](#)]
27. Rieser, A. Čech closure spaces: A unified framework for discrete and continuous homotopy. *Topology Appl.* **2021**, *296*, 107613. [[CrossRef](#)]
28. Qiao, J. On the reflective and coreflective subcategory of stratified L-Čech closure spaces. *Fuzzy Sets Syst.* **2020**, *390*, 105–117. [[CrossRef](#)]
29. Perfilieva, I.; Ramadan, A.A.; Elkordy, E.H. Categories of L-Fuzzy Čech Closure Spaces and L-Fuzzy Co-Topological Spaces. *Mathematics* **2020**, *8*, 1274. [[CrossRef](#)]
30. Močkoř, J. Functors among Relational Variants of Categories Related to L-Fuzzy Partitions, L-Fuzzy Pretopological Spaces and L-Fuzzy Closure Spaces. *Axioms* **2020**, *9*, 63. [[CrossRef](#)]

Article

Convergence Analysis and Dynamical Nature of an Efficient Iterative Method in Banach Spaces

Deepak Kumar ^{1,*}, Sunil Kumar ², Janak Raj Sharma ¹ and Lorentz Jantschi ^{3,4,*}

¹ Department of Mathematics, Sant Longowal Institute of Engineering and Technology, Longowal 148106, India; jrshira@yahoo.co.in

² Department of Mathematics, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Channai 601103, India; k_sunil@ch.amrita.edu

³ Department of Physics and Chemistry, Technical University of Cluj-Napoca, 400114 Cluj-Napoca, Romania

⁴ Chemical Doctoral School, Babes-Bolyai University, 400028 Cluj-Napoca, Romania

* Correspondence: deepak.babbi@gmail.com (D.K.); lorentz.jantschi@chem.utcluj.ro (L.J.)

Abstract: We study the local convergence analysis of a fifth order method and its multi-step version in Banach spaces. The hypotheses used are based on the first Fréchet-derivative only. The new approach provides a computable radius of convergence, error bounds on the distances involved, and estimates on the uniqueness of the solution. Such estimates are not provided in the approaches using Taylor expansions of higher order derivatives, which may not exist or may be very expensive or impossible to compute. Numerical examples are provided to validate the theoretical results. Convergence domains of the methods are also checked through complex geometry shown by drawing basins of attraction. The boundaries of the basins show fractal-like shapes through which the basins are symmetric.

Keywords: local convergence; nonlinear equations; Banach space; Fréchet-derivative

Citation: Kumar, D.; Kumar, S.; Sharma, J.R.; Jantschi, L. Convergence Analysis and Dynamical Nature of an Efficient Iterative Method in Banach Spaces. *Mathematics* **2021**, *9*, 2510. <https://doi.org/10.3390/math9192510>

Academic Editors: Juan Benigno Seoane-Sepúlveda and Alicia Cordero Barbero

Received: 14 August 2021

Accepted: 26 September 2021

Published: 7 October 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Let X, Y be Banach spaces and $D \subseteq X$ be a closed and convex set. In this study, we locate a solution x^* of the nonlinear equation

$$G(x) = 0, \quad (1)$$

where $G : D \subseteq X \rightarrow Y$ is a Fréchet-differentiable operator. In computational sciences, many problems can be written in the form of (1). See, for example, [1–3]. The solutions of such equations are rarely attainable in closed form. This is why most methods for solving these equations are usually iterative. The most well-known method for approximating a simple solution x^* of Equation (1) is Newton's method, which is given by

$$x_{m+1} = x_m - G'(x_m)^{-1}G(x_m), \quad \text{for each } m = 0, 1, 2, \dots \quad (2)$$

and has a quadratic order of convergence. In order to attain the higher order of convergence, a number of modified Newton's or Newton-like methods have been proposed in the literature (see [2–20]) and references cited therein. In particular, Sharma and Kumar [18] recently proposed a fifth order method for approximating the solution of $G(x) = 0$ using the Newton–Chebyshev composition defined for each $n = 0, 1, 2, \dots$ by

$$\begin{aligned} y_m &= x_m - \Gamma_m G(x_m), \\ z_m &= y_m - \Gamma_m G(y_m), \\ x_{m+1} &= z_m - (2I - \Gamma_n [z_m, y_m; G]) \Gamma_m G(z_m), \end{aligned} \quad (3)$$

where $\Gamma_m = G'(x_m)^{-1}$, and $[z_m, y_m; G]$ is the first order divided difference of G . The method has been shown to be computationally more efficient than existing methods of a similar nature.

The important part in the development of an iterative method is to study its convergence analysis. This is usually divided into two categories, namely the semilocal and local convergence. The semilocal convergence is based on the information around an initial point and gives criteria that ensure the convergence of iteration procedures. The local convergence is based on the information of a convergence domain around a solution and provides estimates of the radii of the convergence balls. Local results are important since they provide the degree of difficulty in choosing initial points. There exist many studies which deal with the local and semilocal convergence analysis of iterative methods such as [3–5,7–11,13,16,19,21–23]. The semilocal convergence of the method (3) in Banach spaces has been established in [18]. In the present work, we study the local convergence of this method and its multi-step version, including the computable radius of convergence, error bounds on the distances involved, and estimates on the uniqueness of the solution.

We summarize the contents of the paper. In Section 2, the local convergence (including radius of convergence, error bounds, and uniqueness results of method (3)) is studied. The generalized multi-step version is presented in Section 3. Numerical examples are performed to verify the theoretical results in Section 4. In Section 5, the basins of attractors are studied to visually check the convergence domain of the methods. Finally, some conclusions are reported in Section 6.

2. Local Convergence

The local convergence analysis of method (3) is presented in this section. Let $L_0 > 0$, $L > 0$, $L_1 > 0$, and $M \geq 0$ be given parameters. It is convenient to generate some functions and parameters for the local convergence study that follows. Define function $g_1(t)$ on interval $[0, \frac{1}{L_0})$ by

$$g_1(t) = \frac{Lt}{2(1 - L_0t)}$$

and parameter

$$r_1 = \frac{2}{2L_0 + L} < \frac{1}{L_0}. \tag{4}$$

Then, we have that $g_1(r_1) = 1$ and $0 \leq g_1(t) \leq 1$ for each $t \in [0, r_1)$. Moreover, define the function $g_2(t)$ and $h_2(t)$ on interval $[0, \frac{1}{L_0})$ by

$$g_2(t) = \left(1 + \frac{M}{1 - L_0t}\right)g_1(t)$$

and

$$h_2(t) = g_2(t) - 1.$$

We have that $h_2(0) = -1 < 0$ and $h_2(r_1) = \frac{M}{1 - L_0r_1} > 0$. According to the intermediate value theorem, function $h_2(t)$ has zeros in the interval $(0, r_1)$. Denote such zeros by r_2 . Finally, define functions $K(t)$, $g_3(t)$, and $h_3(t)$ on the interval $[0, \frac{1}{L_0})$ by

$$K(t) = 1 + \frac{1}{1 - L_0t}(L_0 + L_1t(g_2(t) + g_1(t)))t,$$

$$g_3(t) = \left(1 + \frac{MK(t)}{1 - L_0t}\right)g_2(t)$$

and

$$h_3(t) = g_3(t) - 1.$$

We have that $h_3(0) = -1 < 0$ and $h_3(r_2) = \frac{MK(r_2)}{1-L_0r_2} > 0$. According to the intermediate value theorem, function $h_3(t)$ has zeros in $(0, r_2)$. Denote such zeros by r_3 of function $h_3(t)$ in interval $[0, r_2)$. Set

$$r = \min\{r_i\}, \quad i = 1, 2, 3. \tag{5}$$

Then, we obtain that

$$0 < r \leq r_1. \tag{6}$$

Then, for each $t \in [0, r)$

$$0 \leq g_1(t) \leq 1, \tag{7}$$

$$0 \leq g_2(t) \leq 1 \tag{8}$$

and

$$0 \leq g_3(t) \leq 1. \tag{9}$$

Let $U(v, \rho)$ and $\bar{U}(v, \rho)$ symbolise the open and closed balls in X , with a radius $\rho > 0$ and a centre $v \in X$.

Using the above notations, we then describe the local convergence analysis of method (3).

Theorem 1. *Suppose $G : D \subseteq X \rightarrow Y$ is a Fréchet-differentiable function. Let $[\cdot, \cdot; G] : X \times X \rightarrow L(Y)$ be the divided difference operator. Consider that there exist $x^* \in D$, $L_0 > 0$, $L > 0$, $L_1 > 0$, and $M \geq 1$, such that for each $x, y \in D$*

$$G(x^*) = 0, \quad G(x^*)^{-1} \in L(Y, X), \tag{10}$$

$$\|G'(x^*)^{-1}(G'(x) - G'(x^*))\| \leq L_0\|x - x^*\|, \tag{11}$$

$$\|G'(x^*)^{-1}(G'(x) - G'(y))\| \leq L\|x - y\|, \tag{12}$$

$$\|G'(x^*)^{-1}G'(x)\| \leq M, \tag{13}$$

$$\|G'(x^*)^{-1}([x, y; G] - G'(x^*))\| \leq L_1(\|x - x^*\| + \|y - x^*\|), \tag{14}$$

and

$$\bar{U}(x^*, r) \subset D, \tag{15}$$

where r is defined by (5). Then, for each $m = 0, 1, \dots$, the sequence $\{x_m\}$ generated by method (3) for $x_0 \in U(x^*, r) - \{x^*\}$ is well defined, stays in $U(x^*, r)$, and converges to x^* . Furthermore, the following estimates hold:

$$\|y_m - x^*\| \leq g_1(\|x_m - x^*\|)\|x_m - x^*\| < \|x_m - x^*\| < r, \tag{16}$$

$$\|z_m - x^*\| \leq g_2(\|x_m - x^*\|)\|x_m - x^*\| < \|x_m - x^*\| < r \tag{17}$$

and

$$\|x_{m+1} - x^*\| \leq g_3(\|x_m - x^*\|)\|x_m - x^*\|, \tag{18}$$

where the “ g ” functions are defined previously. Furthermore, if there exists $T \in [r, \frac{2}{L_0})$ such that $\bar{U}(x^*, T) \subset D$, then x^* is the only solution of $G(x) = 0$ in $\bar{U}(x^*, T)$.

Proof. We shall show the estimates (16)–(18) using mathematical induction. Using (4), (11), and the hypotheses $x_0 \in U(x^*, r) - \{x^*\}$, we obtain that

$$\|G'(x^*)^{-1}(G(x_0) - G(x^*))\| \leq L_0\|x_0 - x^*\| < L_0r < 1. \tag{19}$$

It follows from (19) and the Banach Lemma [3] that $G'(x_0)^{-1} \in L(Y, X)$ and

$$\|G'(x_0)^{-1}G'(x^*)\| \leq \frac{1}{1 - L_0\|x_0 - x^*\|} < \frac{1}{1 - L_0r}. \tag{20}$$

Hence, y_0 is well defined for $m = 0$. Then, by using (4), (7), (12), and (20), we have

$$\begin{aligned} \|y_0 - x^*\| &\leq \|x_0 - x^* - G'(x_0)^{-1}G(x_0)\| \\ &\leq \|G'(x_0)^{-1}G'(x^*)\| \left\| \int_0^1 G'(x^*)^{-1}[G'(x^* + \theta(x_0 - x^*)) - G'(x_0)]d\theta \right\| \\ &\quad \times \|x_0 - x^*\| \\ &\leq \frac{L\|x_0 - x^*\|^2}{2(1 - L_0\|x_0 - x^*\|)} \\ &= g_1(\|x_0 - x^*\|)\|x_0 - x^*\| < \|x_0 - x^*\| < r, \end{aligned} \tag{21}$$

which shows (16) for $m = 0$ and $y_0 \in U(x^*, r)$.

Notice that for each $\theta \in [0, 1]$ and $\|x^* + \theta(x_0 - x^*) - x^*\| = \theta\|x_0 - x^*\| < r$. That is, $x^* + \theta(x_0 - x^*) \in U(x^*, r)$. We can write

$$G(x_0) - G(x^*) = \int_0^1 G'(x^* + \theta(x_0 - x^*))(x_0 - x^*)d\theta. \tag{22}$$

Then, using (13) and (21), we have

$$\begin{aligned} \|G'(x^*)^{-1}G(x_0)\| &= \left\| \int_0^1 G'(x^*)^{-1}G'(x^* + \theta(x_0 - x^*))(x_0 - x^*)d\theta \right\| \\ &\leq M\|x_0 - x^*\|. \end{aligned} \tag{23}$$

Similarly, we obtain

$$\|G'(x^*)^{-1}G(y_0)\| \leq M\|y_0 - x^*\|, \tag{24}$$

$$\|G'(x^*)^{-1}G(z_0)\| \leq M\|z_0 - x^*\|. \tag{25}$$

Using the second substep of method (3), (8), (20), (21), (27), and (24), we obtain that

$$\begin{aligned} \|z_0 - x^*\| &\leq \|y_0 - x^*\| + \|G'(x_0)^{-1}G(y_0)\| \\ &= \|y_0 - x^*\| + \|G'(x_0)^{-1}G'(x^*)\| \|G'(x^*)^{-1}G(y_0)\| \\ &\leq \|y_0 - x^*\| + \frac{M\|y_0 - x^*\|}{1 - L_0\|x_0 - x^*\|} \\ &\leq \left(1 + \frac{M}{1 - L_0\|x_0 - x^*\|}\right) \|y_0 - x^*\| \\ &\leq \left(1 + \frac{M}{1 - L_0\|x_0 - x^*\|}\right) g_1(\|x_0 - x^*\|)\|x_0 - x^*\| \\ &\leq g_2(\|x_0 - x^*\|)\|x_0 - x^*\| < \|x_0 - x^*\| < r. \end{aligned} \tag{26}$$

Which shows (17) for $m = 0$ and $z_0 \in U(x^*, r)$.

Next, we have the linear operator $A_0 = 2I - G'(x_0)^{-1}[y_0, x_0; G]$; by using (11), (14), and (20), we obtain

$$\begin{aligned}
 \|A_0\| &= \|2I - G'(x_0)^{-1}[z_0, y_0; G]\| \\
 &\leq 1 + \|G'(x_0)^{-1}(G'(x_0) - [z_0, y_0; G])\| \\
 &\leq 1 + \|G'(x_0)^{-1}G'(x^*)\| \|G'(x^*)^{-1}(G'(x_0) - [z_0, y_0; G])\| \\
 &\leq 1 + \|G'(x_0)^{-1}G'(x^*)\| \|G'(x^*)^{-1}(G'(x_0) - G'(x^*) + G'(x^*) - [z_0, y_0; G])\| \\
 &\leq 1 + \|G'(x_0)^{-1}G'(x^*)\| \left(\|G'(x^*)^{-1}(G'(x_0) - G'(x^*))\| + \|G'(x^*)^{-1}(G'(x^*) - [z_0, y_0; G])\| \right) \\
 &\leq 1 + \frac{2}{1-L_0\|x_0-x^*\|} \left(L_0\|x_0-x^*\| + L_1(\|z_0-x^*\| + \|y_0-x^*\|) \right) \\
 &\leq 1 + \frac{2}{1-L_0\|x_0-x^*\|} \left(L_0\|x_0-x^*\| + L_1(g_2(\|x_0-x^*\|) + g_1(\|x_0-x^*\|)) \|x_0-x^*\| \right) \\
 &\leq 1 + \frac{2}{1-L_0\|x_0-x^*\|} \left(L_0 + L_1(g_2(\|x_0-x^*\|) + g_1(\|x_0-x^*\|)) \right) \|x_0-x^*\| \\
 &= K(\|x_0-x^*\|).
 \end{aligned} \tag{27}$$

Then, using Equations (4), (9), (25), and (26), we obtain that

$$\begin{aligned}
 \|x_1 - x^*\| &\leq \|z_0 - x^*\| + \|A_0\| \|G'(x_0)^{-1}G(z_0)\| \\
 &= \|z_0 - x^*\| + \|A_0\| \|G'(x_0)^{-1}G'(x^*)\| \|G'(x^*)^{-1}G(z_0)\| \\
 &\leq \|z_0 - x^*\| + \frac{MK(\|x_0-x^*\|)\|z_0-x^*\|}{1-L_0\|x_0-x^*\|} \\
 &\leq \left(1 + \frac{MK(\|x_0-x^*\|)}{1-L_0\|x_0-x^*\|} \right) \|z_0 - x^*\| \\
 &\leq \left(1 + \frac{MK(\|x_0-x^*\|)}{1-L_0\|x_0-x^*\|} \right) g_2(\|x_0-x^*\|) \|x_0-x^*\| \\
 &\leq g_3(\|x_0-x^*\|) \|x_0-x^*\| < \|x_0-x^*\| < r,
 \end{aligned} \tag{28}$$

which proves the (18) for $m = 0$ and $x_1 \in U(x^*, r)$. By simply replacing x_0, y_0, z_0 , and x_1 by x_m, y_m, z_m , and x_{m+1} in the preceding estimates, we arrive at (16)–(18). Then, from the estimates $\|x_{m+1} - x^*\| < \|x_m - x^*\| < r$, we deduce that $\lim_{m \rightarrow \infty} x_m = x^*$ and $x_{m+1} \in U(x^*, r)$.

Finally, we show the uniqueness part; let $Q = \int_0^1 G'(y^* + t(x^* - y^*))dt$ for some $y^* \in \bar{U}(x^*, r)$ with $G(y^*) = 0$. Using (15), we obtain that

$$\begin{aligned}
 \|G'(x^*)^{-1}(Q - G'(x^*))\| &\leq \int_0^1 L_0 \|y^* + t(x^* - y^*) - x^*\| dt \\
 &\leq \int_0^1 (1-t) \|x^* - y^*\| dt \\
 &\leq \frac{L_0}{2} T < 1.
 \end{aligned} \tag{29}$$

It follows from (29) that Q is invertible. Then, from the identity $0 = G(x^*) - G(y^*) = Q(x^* - y^*)$, we deduce that $x^* = y^*$. \square

Remark 1. By (11) and the estimate

$$\begin{aligned}
 \|G'(x^*)^{-1}G'(x)\| &= \|G'(x^*)^{-1}(G'(x) - G'(x^*)) + I\| \\
 &\leq 1 + \|G'(x^*)^{-1}(G'(x) - G'(x^*))\| \\
 &\leq 1 + L_0\|x - x^*\|
 \end{aligned}$$

condition (13) can be dropped and be replaced by

$$M(t) = 1 + L_0t$$

or

$$M(t) = M = 2, \text{ since } t \in [0, \frac{1}{L_0}).$$

3. Generalized Method

The multistep version of (3) consisting of $q + 1$, ($q \in \mathbb{N}$), steps is expressed as

$$\begin{aligned} z_m^{(0)} &= y_m - \Gamma_m G(y_m), \\ z_m^{(1)} &= z_m - \psi(x_m, y_m, z_m) G(z_m), \\ z_m^{(2)} &= z_m^{(1)} - \psi(x_m, y_m, z_m) G(z_m^{(1)}), \\ &\dots\dots\dots \\ z_m^{(q-1)} &= z_m^{(q-2)} - \psi(x_m, y_m, z_m) G(z_m^{(q-2)}), \\ z_m^{(q)} &= x_{m+1} = z_m^{(q-1)} - \psi(x_m, y_m, z_m) G(z_m^{(q-1)}), \end{aligned} \tag{30}$$

where $y_m = x_m - \Gamma_m G(x_m)$, $z_m^{(0)} = z_m$, $\psi(x_m, y_m, z_m) = (2I - \Gamma_m [z_m, y_m; G]) \Gamma_m$, and $\Gamma_m = G(x_m)^{-1}$.

Next, we show that the generalized scheme (30) possesses convergence order $2q + 3$.

3.1. Order of Convergence

The definition of divided difference is required to derive (30) convergence order. Recalling the result of Taylor’s expansion on vector functions (see [24]) for this:

Lemma 1. $G : D \subset \mathbb{R}^n \rightarrow \mathbb{R}^n$ be r -times Fréchet differentiable in a convex set $D \subset \mathbb{R}^n$ then for any $x, h \in \mathbb{R}^n$, the following expression holds:

$$G(x + h) = G(x) + G'(x)h + \frac{1}{2!}G''(x)h^2 + \frac{1}{3!}G'''(x)h^3 + \dots + \frac{1}{(r-1)!}G^{(r-1)}(x)h^{r-1} + R_r, \tag{31}$$

where

$$\|R_r\| \leq \frac{1}{r!} \sup_{0 \leq t \leq 1} \|G^{(r)}(x + th)\| \|h\|^r \text{ and } h^r = (h, h, \dots, h).$$

The divided difference operator $[\cdot, \cdot; G] : D \times D \subset \mathbb{R}^n \times \mathbb{R}^n \rightarrow L(\mathbb{R}^n)$ is defined by (see [24])

$$[x + h, x; G] = \int_0^1 G'(x + th) dt, \forall x, h \in \mathbb{R}^n. \tag{32}$$

When we expand $G'(x + th)$ in the Taylor series at point x and integrate, we obtain

$$[x + h, x; G] = \int_0^1 G'(x + th) dt = G'(x) + \frac{1}{2}G''(x)h + \frac{1}{6}G'''(x)h^2 + O(h^3). \tag{33}$$

where $h^i = (h, h, \dots, h)$, $h \in \mathbb{R}^n$.

Let $e_m = x_m - x^*$. Expanding $G(x_m)$ in a neighbourhood of x^* and assuming $\Gamma = G'(x^*)^{-1}$ exists, we obtain

$$G(x_m) = G'(x^*)(e_m + A_2(e_m)^2 + A_3(e_m)^3 + A_4(e_m)^4 + A_5(e_m)^5 + O((e_m)^5)), \tag{34}$$

where $A_i = \frac{1}{i!} \Gamma G^{(i)}(x^*) \in L_i(\mathbb{R}^n, \mathbb{R}^n)$ and $(e_m)^i = (e_m, e_m, \dots, e_m)$, $e_m \in \mathbb{R}^n$, $i = 2, 3, \dots$

Additionally,

$$G'(x_m) = G'(x^*)(I + 2A_2e_m + 3A_3(e_m)^2 + 4A_4(e_m)^3 + O((e_m)^4)), \tag{35}$$

$$G''(x_m) = G'(x^*)(2A_2 + 6A_3e_m + 12A_4(e_m)^2 + O((e_m)^3)), \tag{36}$$

$$G'''(x_m) = G'(x^*)(6A_3 + 24A_4e_m + O((e_m)^2)). \tag{37}$$

The inversion of $G'(x_m)$ yields

$$G'(x_m)^{-1} = (I - 2A_2e_m + (4A_2^2 - 3A_3)(e_m)^2 - (4A_4 - 6A_2A_3 - 6A_3A_2 + 8A_2^3)(e_m)^3 + O((e_m)^4))\Gamma. \tag{38}$$

We are in a position to investigate scheme (30)'s convergence behaviour. As a result, the following theorem is established:

Theorem 2. *Suppose that*

- (i) $G : D \subset \mathbb{R}^n \rightarrow \mathbb{R}^n$ is many times differentiable mapping.
- (ii) There exists a solution $x^* \in D$ of equation $G(x) = 0$ such that $G'(x^*)$ is nonsingular. Then, sequence $\{x_n\}$ generated by method (30) for $x_0 \in D$ converges to x^* with order $2q + 3$, $q \in \mathbb{N}$.

Proof. Employing (34) and (38) in the Newton iteration y_m , we obtain that

$$\begin{aligned} \tilde{e}_m = y_m - x^* &= A_2e_m^2 + (2A_2^2 - A_3)e_m^3 + (4A_2^3 - 4A_2A_3 - 3A_3A_2 + 3A_4)e_m^4 \\ &- (8A_2^4 + 6A_3^2 + 6A_2A_4 + 4A_4A_2 - 8A_2^2A_3 - 6A_2A_3A_2 - 6A_3A_2^2)e_m^5 + O(e_m^6). \end{aligned} \tag{39}$$

The Taylor series of $G(y_m)$ about x^* yields

$$G(y_m) = G'(x^*)(\tilde{e}_m + A_2\tilde{e}_m^2 + A_3\tilde{e}_m^3 + A_4\tilde{e}_m^4 + O(\tilde{e}_m^5)), \tag{40}$$

Substituting (38)–(40) in first step of (30), we obtain

$$\bar{e}_m = z_m - x^* = 2A_2^2e_m^3 + (4A_2A_3 - 9A_2^3 + 3A_3A_2)e_m^4 + O(e_m^5). \tag{41}$$

Using Equations (35)–(37) in (33) for $x + h = z_m$, $x = y_m$, and $h = \bar{e}_m - \tilde{e}_m$, it follows that

$$[z_m, y_m; G] = G'(x^*)(I + A_2(\bar{e}_m + \tilde{e}_m) + O((\bar{e}_m)^2, (\tilde{e}_m)^2))$$

and

$$\Gamma_m[z_m, y_m; G] = I - 2A_2e_m + (4A_2^2 - 3A_3)(e_m)^2 + A_2(\bar{e}_m + \tilde{e}_m) + O((e_m)^3).$$

As a result, we arrive at the conclusion

$$\psi(x_m, y_m, z_m) = (I - 5A_2^2(e_m)^2 + 2(10A_2^3 - 4A_2A_3 - 3A_3A_2)(e_m)^3 + O((e_m)^4))G'(x^*)^{-1}. \tag{42}$$

In addition, we have

$$G(z_m) = G'(x^*)(\bar{e}_m + O((\bar{e}_m)^2)). \tag{43}$$

Using (42) and (43) in the second step of method (30), it follows that

$$z_m^{(1)} - x^* = 10A_2^4(e_m)^5 + O((e_m)^6). \tag{44}$$

The expansion of $G(z_m^{(q-1)})$ about x^* yields

$$G(z_m^{(q-1)}) = G'(x^*)((z_m^{(q-1)} - x^*) + A_2(z_m^{(q-1)} - x^*)^2 + \dots). \tag{45}$$

Then, we have

$$\begin{aligned} \psi(x_m, y_m, z_m)G(z_m^{(q-1)}) &= (I - 5A_2^2(e_m)^2 + 2(10A_2^3 - 4A_2A_3 - 3A_3A_2)(e_m)^3 + O((e_m)^4))G'(x^*)^{-1} \\ &\quad \times G'(x^*)((z_m^{(q-1)} - x^*) + A_2(z_m^{(q-1)} - x^*)^2 + \dots) \\ &= (z_m^{(q-1)} - x^*) - 5A_2^2(z_m^{(q-1)} - x^*)(e_m)^2 + A_2(z_m^{(q-1)} - x^*)^2 + \dots \end{aligned} \tag{46}$$

Using (46) in (30), we obtain

$$z_m^{(q)} - x^* = 5A_2^2(z_m^{(q-1)} - x^*)(e_m)^2 - A_2(z_m^{(q-1)} - x^*)^2 + \dots \tag{47}$$

As we know from (44) that $z_m^{(1)} - x^* = 10A_2^4(e_m)^5 + O((e_m)^6)$, from (47) for $q = 2, 3$, we therefore have

$$\begin{aligned} z_m^{(2)} - x^* &= 5A_2^2(e_m)^2(z_m^{(1)} - x^*) + \dots \\ &= 50A_2^6(e_m)^7 + O((e_m)^8) \end{aligned}$$

and

$$\begin{aligned} z_m^{(3)} - x^* &= 5A_2^2(e_m)^2(z_m^{(2)} - x^*) + \dots \\ &= 250A_2^8(e_m)^9 + O((e_m)^{10}). \end{aligned}$$

Proceeding by induction, it follows that

$$e_{m+1} = z_m^{(q)} - x^* = 2 \cdot 5^q A_2^{2q+2} (e_m)^{2q+3} + O((e_m)^{2q+4}).$$

This completes the proof of Theorem 2. \square

Remark 2. Note that method (3) utilizes three functions, one derivative, and one inverse operator per full iteration and converges to the solution with the fifth order of convergence. The generalized scheme (30) based on (3) (for $q = 1$) generates the methods with increasing convergence orders 5, 7, 9, ... corresponding to $q = 1, 2, 3, \dots$ at an additional cost of one function evaluation per each iteration. This fulfils the main aim of developing higher order methods, keeping computational cost under control.

3.2. Local Convergence

Along the same lines as method (3), we offer the local convergence analysis of method (30). Define \bar{g}_2 , λ , μ , and h_μ on the interval $[0, r_2)$ by

$$\begin{aligned} \bar{g}_2(t) &= \frac{K(t)}{1 - w_0(t)}, \\ \lambda(t) &= 1 + \bar{g}_2(t)M, \\ \mu(t) &= \lambda^q(t)g_2(t)t^{\lambda-1} \end{aligned}$$

and

$$h_\mu(t) = \mu(t) - 1.$$

We have that $h_\mu(0) < 0$. Suppose that

$$\mu(t) \rightarrow +\infty \text{ or a positive number as } t \rightarrow r_2^-. \tag{48}$$

Denote by $r^{(q)}$ the smallest zero on the interval $(0, r_2)$ of function h_μ . Define r^* by

$$r^* = \min\{r_1, r^{(q)}\}. \tag{49}$$

Proposition 1. *Suppose that the conditions of Theorem 2 hold. Then, sequence $\{x_m\}$ generated for $x_0 \in U(x^*, r^*) - \{x^*\}$ by method (30) is well defined in $U(x^*, r^*)$, remains in $U(x^*, r^*)$, and converges to x^* . Moreover, the following estimates hold:*

$$\begin{aligned} \|y_m - x^*\| &\leq g_1(\|x_m - x^*\|)\|x_m - x^*\| \leq \|x_m - x^*\| < r^*, \\ \|z_m - x^*\| &\leq g_2(\|x_m - x^*\|)\|x_m - x^*\| \leq \|x_m - x^*\|, \\ \|z_m^{(i)} - x^*\| &\leq \lambda^i(\|x_m - x^*\|)\|z_m - x^*\| \\ &\leq \lambda^i(\|x_m - x^*\|)g_2(\|x_m - x^*\|)\|x_m - x^*\|^\lambda \\ &\leq \|x_m - x^*\|, \quad i = 1, 2, \dots, q - 1, \end{aligned} \tag{50}$$

and

$$\begin{aligned} \|x_{k+1} - x^*\| = \|z_m^{(q)} - x^*\| &\leq \lambda^q(\|x_m - x^*\|)\|z_m - x^*\| \\ &\leq \mu(\|x_m - x^*\|)\|x_m - x^*\|. \end{aligned} \tag{51}$$

Furthermore, x^* is the only solution of $G(x) = 0$ in $D_1 = D \cap U(x^*, r^*)$.

Proof. Only new estimations (50) and (51) will be shown. We show the first two estimations using the evidence of Theorem 1. Then, we will be able to obtain that

$$\begin{aligned} \|\psi(x_m, y_m, z_m)G'(x^*)\| &\leq \|(2I - G'(x_m)^{-1}[z_m, y_m; G])G'(x_m)^{-1}G'(x^*)\| \\ &\leq \|(2I - G'(x_m)^{-1}[z_m, y_m; G])\| \|G'(x_m)^{-1}G'(x^*)\| \\ &\leq \frac{K(\|x_m - x^*\|)}{1 - w_0(\|x_m - x^*\|)} \\ &\leq \bar{g}_2(\|x_m - x^*\|). \end{aligned} \tag{52}$$

Moreover, we have

$$\begin{aligned} \|z^{(1)} - x^*\| &= \|z_m - x^* - \psi(x_m, y_m)G(z_m)\| \\ &\leq \|z_m - x^*\| + \|\psi(x_m, y_m, z_m)G'(x^*)\| \|G'(x^*)^{-1}G(z_m)\| \\ &\leq \|z_m - x^*\| + \bar{g}_2(\|x_m - x^*\|)M\|z_m - x^*\| \\ &\leq \lambda(\|x_m - x^*\|)\|z_m - x^*\| \\ &\leq \mu(\|x_m - x^*\|)\|x_m - x^*\|. \end{aligned}$$

Similarly, we obtain

$$\begin{aligned} \|z_m^{(2)} - x^*\| &\leq \lambda(\|x_m - x^*\|)\|z_m^{(1)} - x^*\| \\ &\leq \lambda^2(\|x_m - x^*\|)\|z_m - x^*\| \\ &\dots\dots\dots \\ \|z_m^{(i)} - x^*\| &\leq \lambda^i(\|x_m - x^*\|)\|z_m - x^*\| \\ \|x_{m+1} - x^*\| &\leq \|z_m^{(q)} - x^*\| \leq \lambda^q(\|x_m - x^*\|)\|z_m - x^*\| \\ &\leq \mu(\|x_m - x^*\|)\|x_m - x^*\|. \end{aligned}$$

That is, we have $x_m, y_m, z_m, z_m^{(i)} \in U(x^*, r^*), i = 1, 2, \dots, q$, and

$$\|x_{m+1} - x^*\| \leq \bar{c}\|x_m - x^*\|, \tag{53}$$

where $\bar{c} = \mu(\|x_0 - x^*\|) \in [0, 1)$, so $\lim_{m \rightarrow \infty} x_m = x^*$ and $x_{m+1} \in U(x^*, r^*)$. The uniqueness result is standard, as shown in Theorem 1. \square

4. Numerical Examples

Here, we shall demonstrate the theoretical results of local convergence which we have proved in Sections 2 and 3. To do so, the methods of the family (30) of order five, seven, and nine are chosen. Let us denote these methods by M_5 , M_7 , and M_9 , respectively. The divided difference in the examples is computed by $[x, y; F] = \int_0^1 F'(y + \theta(x - y))d\theta$. We consider three numerical examples, which are presented as follows:

Example 1. Let us consider $B = \mathbb{R}^{m-1}$ for natural integer $m \geq 2$. B is equipped with the max-norm $\|x\| = \max_{1 \leq i \leq m-1} \|x_i\|$. The corresponding matrix norm is $\|A\| = \max_{1 \leq i \leq m-1} \sum_{j=1}^{j=m-1} |a_{ij}|$ for $A = (a_{ij})_{1 \leq i, j \leq m-1}$. Consider the two-point boundary value problem on interval $[0, 1]$:

$$\begin{cases} v'' + v^{3/2} = 0, \\ v(0) = v(1) = 0. \end{cases} \tag{54}$$

Let us denote $\Delta = 1/m$, $u_i = \Delta i$, and $v_i = V(u_i)$ for each $i = 0, 1, \dots, m$. We can write the discretization of v'' at points u_i in the following form:

$$v_i'' \simeq \frac{v_{i-1} - 2v_i + v_{i+1}}{\Delta^2} \text{ for each } i = 2, 3, \dots, m - 1.$$

Using the initial conditions in (54), we obtain that $v_0 = v_m = 0$, and (54) is equivalent to the system of the nonlinear equation $F(v) = 0$ with $v = (v_1, v_2, \dots, v_{m-1})$ in the following form:

$$\begin{cases} \Delta^2 v_1^{3/2} - 2v_1 + v_2 = 0, \\ v_{i-1} + \Delta^2 v_i^{3/2} - 2v_i + v_{i+1} = 0 \text{ for each } i = 2, 3, \dots, m - 1. \end{cases} \tag{55}$$

Using (55), the Fréchet-derivative of operator F is given by

$$F'(v) = \begin{pmatrix} \frac{3}{2}\Delta^2 v_1^{1/2} - 2 & 1 & 0 & \dots & 0 \\ 1 & \frac{3}{2}\Delta^2 v_1^{1/2} - 2 & 1 & \ddots & 0 \\ 0 & 1 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 1 \\ 0 & \dots & 0 & 1 & \frac{3}{2}\Delta^2 v_1^{1/2} - 2 \end{pmatrix}.$$

Choosing $m = 11$, the corresponding solution is $x^* = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0)^T$, and we have $L_0 = L = L_1 = 3.942631477$ and $M = 2$. The parameters using method (30) are given in Table 1.

Table 1. Numerical results for example 1.

M_5	M_7	M_9
$r_1 = 0.00791011$	$r_1 = 0.00791011$	$r_1 = 0.00791011$
$r^{(1)} = 0.00470691$	$r^{(2)} = 8.50886 \times 10^{-10}$	$r^{(3)} = 1.61122 \times 10^{-13}$
$r^* = 0.00470691$	$r^* = 8.50886 \times 10^{-10}$	$r^* = 1.61122 \times 10^{-13}$

Thus, it follows that the above-considered methods of scheme (30) converge to x^* and remain in $\bar{U}(x^*, r^*)$.

Example 2. Scholars have determined that the speed of blood in a course is an element of the distance of the blood from the conduit’s focal pivot (Figure 1). As per Poiseuille’s law, the speed (cm/s) of blood that is r cm from the focal hub of a supply route is given by the capacity

$$S(r) = C(R^2 - r^2), \tag{56}$$

where R is the range of the course, and C is a consistent that relies upon the thickness of the blood and the tension between the two closures of the vein. Assume that for a specific course,

$$C = 1.76 \times 10^5 \text{ cm/s}$$

and

$$R = 1.2 \times 10^{-2} \text{ cm.}$$

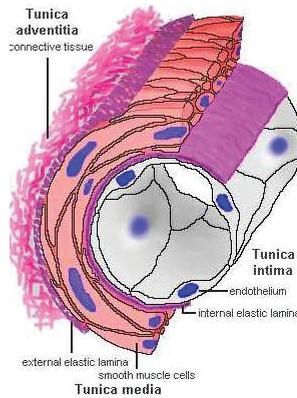


Figure 1. Cut-away view of an artery.

Using the numerical values, the problem reduces to

$$f_2(x) = 25.344 - 176,000x^2 = 0,$$

where $x = r$.

The graph of the function $f_2(x)$ is shown in Figure 2.

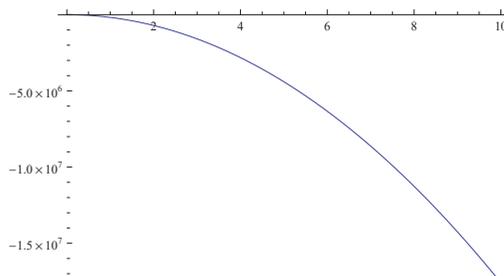


Figure 2. Graph of $f_2(x)$.

The zero of $f_2(x) = 0$ is $x^* = 0.012$; then, we have $L_0 = L = L_1 = 84.2803$ and $M = 5280$. The parameters using method (30) are given in Table 2.

It follows that the above-considered methods of scheme (30) will converge to x^* and remain in $\bar{U}(x^*, r^*)$ if r^* is chosen as shown in Table 2.

Table 2. Numerical results for example 2.

M_5	M_7	M_8
$r_1 = 0.169092$	$r_1 = 0.169092$	$r_1 = 0.169092$
$r^{(1)} = 0.0724823$	$r^{(2)} = 0.0331151$	$r^{(3)} = 0.0140628$
$r^* = 0.0724823$	$r^* = 0.0331151$	$r^* = 0.0140628$

Example 3. Consider the quasi-one-dimensional isentropic flow of a perfect gas through a variable-area channel, shown in Figure 3.

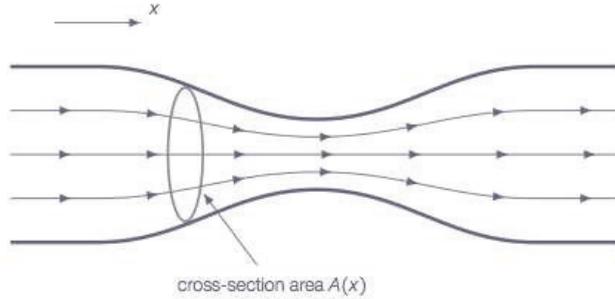


Figure 3. In quasi-one-dimensional flows, the stream tube cross section area is allowed to vary in one direction $A = A(x)$.

The relationship between the Mach number M and the flow area A , derived by Zucrow and Hoffman [25], is given by

$$\epsilon = \frac{A}{A^*} = \frac{1}{M} \left(\frac{2}{\gamma + 1} \left(1 + \frac{\gamma - 1}{2} M^2 \right) \right)^{(\gamma + 1)/2(\gamma - 1)}, \tag{57}$$

where A^* is the choking area (i.e., the area where $M = 1$), and γ is the specific heat ratio of the flowing gas shown in Figure 4.

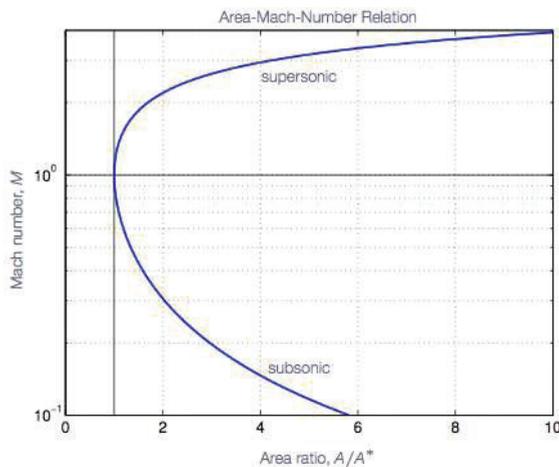


Figure 4. The area–Mach-number relation.

For each value of ϵ , two values of M exist, one less than unity (i.e., subsonic flow) and one greater than unity (i.e., supersonic flow). For the values of $\epsilon = 5.00$ and $\gamma = 1.4$, Equation (57) becomes

$$f_3(x) = 5 - \frac{0.578704(1 + 0.2x^2)^3}{x} \tag{58}$$

where $x = M$. The graph of the function $f_3(x)$ is shown in Figure 5, and the zero is $x^* = 0.116689$. Then, we have that

$$L = L_0 = L_1 = 8.137146, \text{ and } M = 0.610065.$$

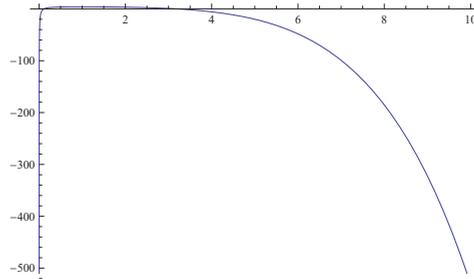


Figure 5. Graph of $f_3(x)$.

The parameters using method (30) are given in Table 3.

Table 3. Numerical results for example 3.

M_5	M_7	M_9
$r_1 = 0.0819303$	$r_1 = 0.0819303$	$r_1 = 0.0819303$
$r^{(1)} = 0.050974$	$r^{(2)} = 0.0355748$	$r^{(3)} = 0.0254287$
$r^* = 0.050974$	$r^* = 0.0355748$	$r^* = 0.0254287$

The computed values of r^* show that the considered methods of the scheme (30) will converge to x^* and remain in $\bar{U}(x^*, r^*)$.

5. Study of Complex Dynamics of the Method

To view the geometry of the methods of the family (30) of five, seven, and nine order methods, in the complex plane, we present the attraction of basins of the roots by performing the methods on some functions (see Table 4). The basins are displayed in Figures 6–8 concerning capacities. To draw basins, we use square shapes $R \in \mathbb{C}$ of size $[-2, 2] \times [-2, 2]$ and allot various shadings to the basins. The dark region is appointed to the focuses for which the strategy is disparate.

Table 4. Comparison of performance based on basins of attraction of methods.

S. No.	Test Problems	Roots	Color of Fractal	Best Performer	Poor Performer
1	$P_1(z) = z^2 - 4$	-2 2	red green	$M_5,$	M_7, M_9
2	$P_2(z) = z^3 - z$	-1 0 1	red green blue	M_5	M_7, M_9
3	$P_3(z) = z^6 + \frac{15}{7}z^5 + 5z^4 + \frac{7}{3}z^3 - z^2 + z + 1$	-0.8277... -0.7654 - 1.9514i -0.6562... -0.7654 + 1.9514i 0.4357 - 0.4786i 0.4357 + 0.4786i	cyan yellow purple blue green red	M_5, M_7	M_9

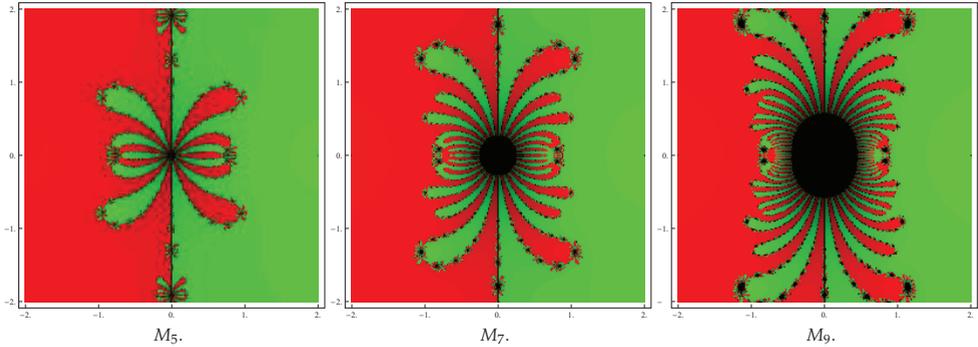


Figure 6. Basins of attraction of M_5 , M_7 , and M_9 for polynomial $P_1(z)$.

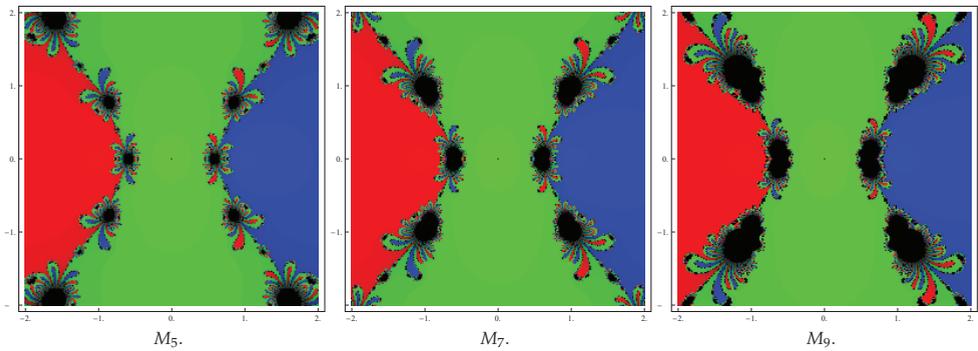


Figure 7. Basins of attraction of M_5 , M_7 , and M_9 for polynomial $P_2(z)$.

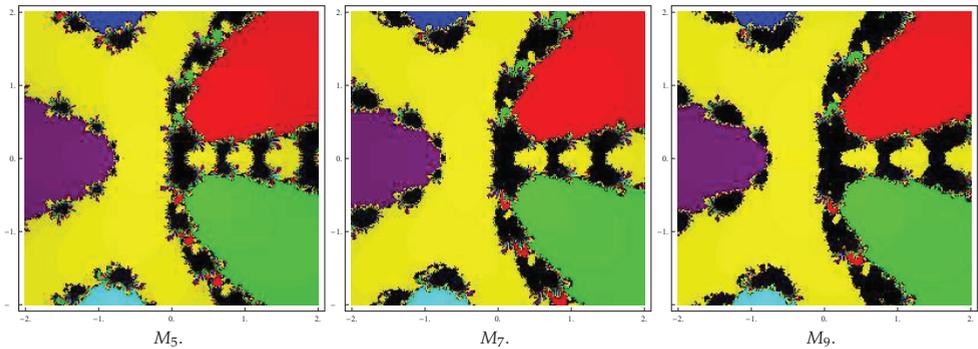


Figure 8. Basins of attraction of M_5 , M_7 , and M_9 for polynomial $P_3(z)$.

6. Conclusions

In this work, we have extended the utilization of technique (3) by introducing its assembly investigation and complex elements. Rather than using different procedures depending on the higher subordinate request just as a Taylor series, we have utilized only a subsidiary of request one, since this actually shows up in the technique. One more benefit of our methodology is the calculation of uniqueness balls where the repeats lie just as appraisals on $\|x_n - x^*\|$. These objectives are accomplished utilizing our Lipschitz-like conditions. The hypothetical outcomes so determined are confirmed on some useful issues.

Finally, we have checked the security of the technique through utilizing a complex element apparatus, specifically a bowl of fascination.

Author Contributions: Conceptualization, S.K. and J.R.S.; Formal analysis, S.K. and L.J.; Investigation, S.K. and L.J.; Methodology, D.K.; Writing—original draft, D.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Technical University of Cluj-Napoca open access publication grant.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: We would like to express our gratitude to the anonymous reviewers for their help with the publication of this paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Potra, F.-A.; Ptak, V. *Nondiscrete Induction and Iterative Process*, *Research Notes in Mathematics*; Pitman: Boston, MA, USA, 1984.
- Traub, J.F. *Iterative Methods for the Solution of Equations*; Prentice-Hall: Englewood Cliffs, NJ, USA, 1964.
- Argyros, I.K.; Hilout, S. *Computational Methods in Nonlinear Analysis*; World Scientific Publishing Company: Singapore, 2013.
- Amat, S.; Hernández, M.A.; Romero, N. Semilocal convergence of a sixth order iterative method for quadratic equation. *Appl. Num. Math.* **2012**, *62*, 833–841. [[CrossRef](#)]
- Argyros, I.K.; Ren, H. Improved local analysis for certain class of iterative methods with cubic convergence. *Numer. Algor.* **2012**, *59*, 505–521.
- Argyros, I.K.; Regmi, S. *Undergraduate Research at Cameron University on Iterative Procedures in Banach and Other Spaces*; Nova Science Publisher: New York, NY, USA, 2019.
- Argyros, I.K.; Sharma, J.R.; Kumar, D. Ball convergence of the Newton-Gauss method in Banach space. *SeMA* **2017**, *74*, 429–439. [[CrossRef](#)]
- Babajee, D.K.R.; Dauhoo, M.Z.; Darvishi, M.T.; Barati, A. A note on the local convergence of iterative methods based on Adomian decomposition method and 3-node quadrature rule. *Appl. Math. Comput.* **2008**, *200*, 452–458. [[CrossRef](#)]
- Candela, V.; Marquina, A. Recurrence relations for rational cubic methods I: The Halley method. *Computing* **1990**, *44*, 169–184. [[CrossRef](#)]
- Candela, V.; Marquina, A. Recurrence relations for rational cubic methods II: The Chebyshev method. *Computing* **1990**, *45*, 355–367. [[CrossRef](#)]
- Chun, C.; Stănică, P.; Neta, B. Third-order family of methods in Banach spaces. *Comput. Math. Appl.* **2011**, *61*, 1665–1675. [[CrossRef](#)]
- Cordero, A.; Torregrosa, J.R. Variants of Newton’s method using fifth-order quadrature formulas. *Appl. Math. Comput.* **2007**, *199*, 686–698. [[CrossRef](#)]
- Ezquerro, J.A.; Hernández, M.A. Recurrence relation for Chebyshev-type methods. *Appl. Math. Optim.* **2000**, *41*, 227–236. [[CrossRef](#)]
- Usurelu, G.I.; Bejenaru, A.; Postolache, M. Newton-like methods and polynomiographic visualization of modified Thakur processes. *Int. J. Comp. Math.* **2021**, *98*, 1049–1068. [[CrossRef](#)]
- Gdawiec, K.; Kotarski, W.; Lisowska, A. Polynomiography Based on the Nonstandard Newton-Like Root Finding Methods. *Abst. Appl. Anal.* **2015**, *2015*, 797594. [[CrossRef](#)]
- Hasanov, V.I.; Ivanov, I.G.; Nebzhibov, F. A new modification of Newton’s method. *Appl. Math. Eng.* **2002**, *27*, 278–286.
- Sharma, J.R.; Arora, H. On efficient weighted-Newton methods for solving systems of nonlinear equations. *Appl. Math. Comput.* **2013**, *222*, 497–506. [[CrossRef](#)]
- Sharma, J.R.; Kumar, D. A fast and efficient composite Newton Chebyshev method for systems of nonlinear equations. *J. Complex.* **2018**, *49*, 56–73. [[CrossRef](#)]
- Ren, H.; Wu, Q. Convergence ball and error analysis of a family of iterative methods with cubic convergence. *Appl. Math. Comput.* **2009**, *209*, 369–378. [[CrossRef](#)]
- Weerkoon, S.; Fernando, T.G.I. A variant of Newton’s method with accelerated third-order convergence. *Appl. Math. Lett.* **2000**, *13*, 87–93. [[CrossRef](#)]
- Hernández, M.A.; Salanova, M.A. Modification of Kantorovich assumptions for semi local convergence of Chebyshev method. *J. Comput. Appl. Math.* **2000**, *126*, 131–143. [[CrossRef](#)]
- Gutiérrez, J.M.; Magreñán, A.A.; Romero, N. On the semilocal convergence of Newton-Kantorovich method under center-Lipschitz conditions. *Appl. Math. Comput.* **2013**, *221*, 79–88.

23. Kou, J.S.; Li, Y.T.; Wang, X.H. A modification of Newton's method with third-order convergence. *Appl. Math. Comput.* **2006**, *181*, 1106–1111.
24. Ortega, J.M.; Rheinboldt, W.C. *Iterative Solutions of Nonlinear Equations in Several Variables*; Academic Press: New York, NY, USA, 1970.
25. Hoffman, J.D. *Numerical Methods for Engineers and Scientists*; McGraw-Hill Book Company: New York, NY, USA, 1992.

Article

Comparison of Molecular Geometry Optimization Methods Based on Molecular Descriptors

Donatella Bálint¹ and Lorentz Jäntschi^{1,2,*}

¹ Faculty of Chemistry and Chemical Engineering, Babeş-Bolyai University, 11 Arany Janos, 400082 Cluj-Napoca, Romania; donatella.balint@ubbcluj.ro

² Department of Physics and Chemistry, Technical University of Cluj-Napoca, 103-105 Muncii Blvd., 400641 Cluj-Napoca, Romania

* Correspondence: lorentz.jantschi@chem.utcluj.ro

Abstract: Various methods (Hartree–Fock methods, semi-empirical methods, Density Functional Theory, Molecular Mechanics) used to optimize a molecule structure feature the same basic approach but differ in the mathematical approximations used. The geometry optimization procedure calculates the energy at an initial geometry of a molecule and then proceeds to search a new geometry with a lower energy. Using the 3D structures collected from the PubChem database, 20 amino acid geometry optimization calculations were performed with several methods. The purpose of the study was to analyze these methods (39) to find the relationship between them and to determine which to use under different circumstances. Cluster analysis and principal component analysis were performed to evaluate the similarities between the different methods. The results after the analysis can be classified into three main groups and can be selected accordingly to solve different types of problems.

Keywords: Gaussian; optimization; geometry; molecular modeling; amino acids

Citation: Bálint, D.; Jäntschi, L. Comparison of Molecular Geometry Optimization Methods Based on Molecular Descriptors. *Mathematics* **2021**, *9*, 2855. <https://doi.org/10.3390/math9222855>

Academic Editor: Yang-Hui He

Received: 28 September 2021

Accepted: 7 November 2021

Published: 10 November 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

A basis set is essentially a finite number of atomic-like functions, over which the molecular orbital is formed via linear combination of atomic orbitals (LCAO). There are multiple choices for the basis set, such as Slater type orbitals [1] (STOs) or Gaussian-type orbitals [2] (GTOs). The wave functions are also called “stationary states” or “energy eigenstates”; in chemistry they are called “atomic orbitals” or “molecular orbitals”. Consequently, they are important in molecular modeling [3,4].

Stationary states be described by the time-independent Schrödinger equation:

$$H\psi = E\psi, \quad (1)$$

where ψ is the state vector of the quantum system, E is the energy, and H is the Hamiltonian operator.

In the time-independent Schrödinger equation, the operation may produce specific values for the energy called energy eigenvalues. In addition to its role in determining system energies, the Hamiltonian operator generates the time evolution of the wavefunction in the form:

$$H\psi = j\hbar \frac{\partial}{\partial t}, \quad (2)$$

where the j constant is the imaginary unit, \hbar is the reduced Planck constant, and t is time.

The Schrödinger equation provides a method for calculating the wave function of a system and its dynamic change over time. The equation is a wave equation in terms of the wave function which predicts analytically and precisely the probability of events or outcome. The spatial part needs to be solved for in time-independent problems, because the time-dependent phase factor is always the same.

The Schrödinger Equation (1) for molecular systems can only be solved approximately [5]. The energy operator can be replaced by the energy eigenvalue E , so the time-independent Schrödinger equation is an eigenvalue equation for the Hamiltonian operator. Approximation methods can be classified into ab initio or semi-empirical categories.

An unknown one-electron function, such as an orbital ψ_i can be expanded in a set of known functions χ_k ($k = 1, 2, \dots, M$), the basis set:

$$\psi_i = \sum_{k=1}^M c_{ki} \chi_k. \quad (3)$$

In Hartree–Fock (HF) and Kohn–Sham density function theory (DFT), the coefficients c_{ki} are determined by minimizing the total energy, which by traditional methods lead to a matrix eigenvalue problem that is solved iteratively to provide a self-consistent field (SCF) solution.

The foundations of the orbital theory were laid by Hartree, Fock, and Slater. If the $2n$ electrons in a molecule are assigned to a set of n molecular orbitals ψ_i ($i = 1, \dots, n$), the corresponding many electron wavefunction is:

$$\psi = (2n!)^{-1/2} \det[(\psi_{-1} \alpha)(\psi_{-1} \beta)(\psi_{-2} \alpha) \dots]. \quad (4)$$

The ψ_i are orthonormal and α and β are spin functions.

Slater-Type Orbitals (STOs) and Gaussian-Type Orbitals (GTOs) are used to describe AOs (atomic orbitals). STOs describe the shape of AOs more accurately than GTOs, but GTOs feature an advantage: they are much easier to compute. In fact, calculating multiple GTOs and combining them to describe an orbital is faster than calculating an STO. This is why combinations of GTOs are usually used to describe STOs, which, in turn, describe AOs.

The simplest and standard basis set in the Gaussian Program is Slater-Type-Orbitals simulated by three Gaussian functions each (STO-3G). Generally, if $n < 3$ the calculations produce poor results, in consequence, STO-3G is called the minimal basis set. We use minimal basis sets for qualitative results, very large molecules, or quantitative results for very small molecules (atoms) [6]. STOs represent the exact solutions for hydrogen-like atoms and provide a better representation than Gaussian functions for multielectron systems on a function-to-function comparison.

The most commonly used bases set for geometry optimization is 3-21G [7–9]. This method uses three Gaussians for the core orbitals and a two/one split for the valence functions. Usually, d orbitals for all heavy (non-hydrogen) atoms are added to improve a basis set. The polarization basis sets are those that include the d orbitals; they are indicated by the symbol “**”. A further development is the 6-31G** basis, in which a set of p orbitals is added to each hydrogen in the 6-31G* basis set [10].

A number of methods are used to optimize the geometry of molecules: empirical force field methods (molecular mechanics, a cheaper method in terms of computational speed, able to provide exceptional structural parameters), semi-empirical methods (to solve the Schrödinger equation, with certain approximations and description of the electron properties of atoms and molecules), and ab initio methods (e.g., Hartree–Fock, Post-Hartree–Fock, and Density Functional Theory) [6].

John A. Pople [11] pioneered the development of ab initio methods using Slater type bases sets or Gaussian orbitals to model the wave function. He defined models, selecting a combination of methods and bases sets, and compared the experimental results of the analysis. With his team, he established an extended basis of contracted Gaussian functions that considers the same properties but is still simple enough to be widely applied to organic molecules [10]. Gaussian-type atomic orbitals have been used broadly to calculate atomic and molecular wavefunctions. They were involved in the growth of one of the most common computational chemistry packages, the Gaussian programs.

For ab initio methods, the first step is a single-determinant SCF (self-consistent field) calculation. Ab initio quantum chemistry methods present the challenge of solving the

electronic Schrödinger equation based on the positions of the nuclei and the number of electrons to provide valuable data.

Their quality depends on the basis set used. The Hartree and Hartree–Fock methods can be regarded as reference methods for many calculations in complex systems. The first solutions to be obtained are used in the next iteration. Hartree–Fock equations must be solved by an iterative procedure and offer the second set of solutions. This approach, SCF, continues as long as the energies of all the electrons remain unaffected. Almost all ab initio calculations use GTO basis sets.

Pure density functional theory (DFT) [12,13] methods are characterized by pairing an exchange functional with a correlation functional. Most current DFT studies use BP86, B3LYP, or BPW91 functionals.

The combination of the method and the basis set determines the chemistry model as Gaussian, specifying a level of theory. HF methods are considered the default if no other keywords are mentioned. Most methods also require a basis set; if no basis set keyword is specified, then STO-3G is used automatically. Some examples of basis sets are: STO-3G, 3-21G, 6-21G, and 6-31G. Single first-polarization functions can also be requested by using the usual * or ** notation. 6-31G* (or 6-31G(d)) is 6-31G with additional d polarization functions on non-hydrogen atoms; 6-31G** (or 6-31G(d, p)) is 6-31G* plus p polarization functions for hydrogen [5]. The + and ++ diffuse functions are accessible with some basis sets. 6-31+G is 6-31G plus diffuse s and p functions for non-hydrogen atoms; 6-31++G also features diffuse functions for hydrogen. Thom Dunning introduced optimized basis sets with correlated wavefunctions: cc (correlations-consistent basis) or pV (polarized valence basis) [14]. The prefix aug (augmented) can be used to add diffuse functions. Which a basis set is used, it is related to the purpose of the calculation and the molecules to be studied. Even a large basis set is not always a guarantee of agreement with the experimental data [13,15].

Different approaches [16–18] to the comparison of basis sets agree that, even if they are similar, basis sets cannot be generalized. Some recommendations we found in the articles studied and by consulting Gaussian tutorials are:

A large basis set is not always the best (ex: cc-pVQZ is overkill for Hartree–Fock).

The minimal basis set (STO-3G) allows the analysis of the largest molecules while having the lowest resolution/quality for quantum level. In general, cc-pVDZ is equivalent to or worse than 6-31G (d, p).

cc-pVTZ is better than 6-311G(d,p) or similar.

The convergence of ab initio methods is time-consuming.

The following bases sets are approximately equivalent:

6-31G → cc-pVDZ
 6-311G → aug-cc-pVDZ
 6-31+G(d) → cc-pVTZ
 6-311+G(d) → aug-cc-pVTZ
 6-31++G(d,p) → cc-pVQZ
 6-311++G(d,p) → aug-cc-pVQZ

Due to the many basis sets and optimization methods, it is very difficult to find the optimal approach for scientific calculations. The choice of basis set for chemical calculations can have a major impact on the quality of the results, particularly for correlated ab initio methods [19]. The choice can be made based on the knowledge related to the design, development, and optimization of the latest developments in the field. For example, applications of basis sets are in the simulation and optimization of ultrasonic non-destructive tests, which are highly important in structural materials such as fiber composites, but also in columnar grained stainless steels [20]. Another approach could be functional cluster analysis (FCA) for multidimensional functional datasets, using orthonormalized Gaussian basis functions, which can be applied for example, to protein structures [21].

The purpose of this study was to analyze 39 optimization methods to find the relationship between them and determine which to use under different circumstances. Cluster

analysis, Statistical analysis (ANOVA), and principal component analysis (PCA) were performed to evaluate the similarities between the different methods.

2. Materials and Methods

The 20 amino acid structures (3D) shown in Table 1 were collected from the PubChem compound database [22].

Table 1. Amino acids used as input data to our algorithm.

Amino Acids (AA)			
Arginine	Lysine	Methionine	Leucine
Asparagine	Serine	Alanine	Phenylalanine
Aspartate	Threonine	Valine	Proline
Glutamate	Cysteine	Glycine	Tryptophan
Glutamine	Histidine	Isoleucine	Tyrosine

These 20 amino acids feature different forms, isomers, enantiomers, and conformers. In biological systems, amino acids feature the same chirality; most are levorotatory (*L*) and not dextrorotatory (*D*). Using the *L* conformer of these compounds, geometry optimizations were performed on the structures (Table 2). The most frequent procedure to establish the basis functions describing the occupied atomic orbitals by HF/DFT optimization followed by addressing the issue of polarization functions subsequently. Whatever optimization method is used, it defines a local minimum, and it is possible that optimization starting from different initial exponents will lead to different outcomes.

Table 2. Geometry optimization methods used in the calculations.

Gaussian Optimization Methods	
Semi-Empirical Methods (Default Spin)	1. Parameterized Model 6 PM6 (opt-pm6) 2. Austin Model 1 AM1 (opt-am1) 3. Parameterized Model 3 PM3 (opt-pm3) 4. Parameterized Model 3 (Molecular Mechanics correction) PM3MM (opt-pm3mm) 5. Pairwise Distance Directed Gaussian function PDDG (opt-pddg) 6. Complete Neglect of Differential Overlap CNDO (opt-cndo) 7. Intermediate Neglect of Differential Overlap INDO (opt-indo)
Density Functional Theory (Default Spin)	Becke(three-parameter)–Lee–Yang–Parr (functional) B3LYP 8. opt-b3lyp-sto-3g; 9. opt-b3lyp-3-21g; 10. opt-b3lyp-6-31g; 11. opt-b3lyp-6-311g; 12. opt-b3lyp-cc-pvdz; Local Spin Density Approximation LSDA 13. opt-lsda-3-21g; 14. opt-lsda-sto-3g; 15. opt-lsda-cc-pvdz; 16. opt-lsda-6-311g; 17. opt-lsda-6-31g; Perdew–Burke–Ernzerhof (functional) PBEPBE opt-pbepbe-sto-3g BVP86 19. opt-bvp86-sto-3g; 20. opt-bvp86-3-21g; 21. Opt-bvp86-6-31g; 22. opt-bvp86-6-311g; B3PW91 23. opt-b3pw91-sto-3g; 24. opt-b3pw91-6-31g; 25. opt-b3pw91-6-311g)
Møller–Plesset Perturbation Theory	MP2 26. opt-mp2-sto-3g; 27. opt-mp2-3-21g; 28. opt-mp2-6-31g; 29. opt-mp2-6-311g; 30. opt-mp2-cc-pvdz;)
Coupled-Cluster Theory	31. Coupled Cluster single-double CCSD (opt-ccsd-sto-3g)
Molecular Mechanics (Default Spin)	32. Universal Force Field UFF (opt-uff) 33. Dreiding (opt-dreiding)
Hartree–Fock (Default Spin)	34. STO-3G (opt-hf-sto-3g) 35. 3-21G (opt-hf-3-21g) 36. 3-21G* (opt-hf-3-21g*) 37. 6-31G (opt-hf-6-31g) 38. 6-311G (opt-hf-6-311g) 39. CC-pvdz (opt-hf-cc-pvdz)

The workflow is represented in the next figure (Figure 1). After the Gaussian program made the calculations based on the 39 methods selected, a family of molecular descriptors (FMPI- Fragmental Matrix Property Indices) [23] was also calculated to evaluate the degree of similarity between the methods. The results were submitted to cluster, PCA, and other statistical analyses.

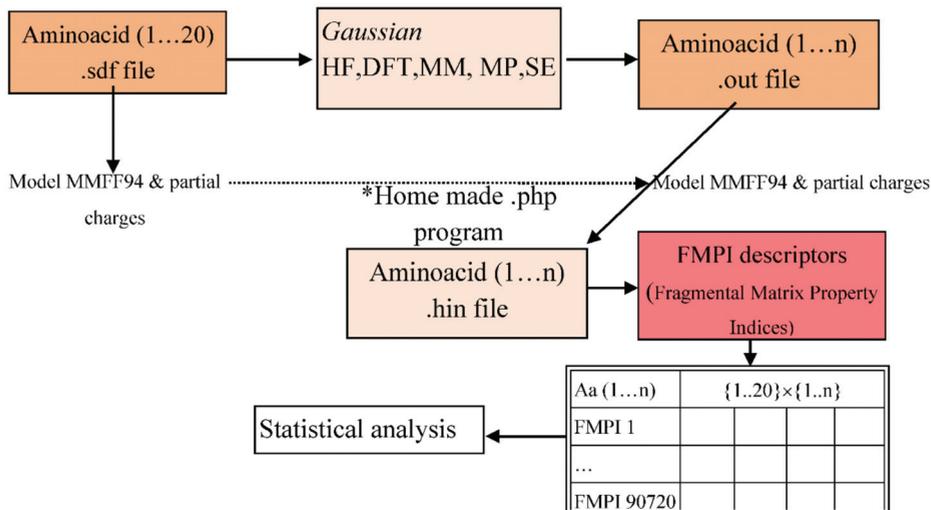


Figure 1. The working algorithm.

We collected the structures (3D) of the 20 essential amino acids (L conformers) from PubChem databases (.sdf files) and analysed them with the Gaussian program, taking into consideration the following steps, also shown in Figure 1:

- Enter the PubChem .sdf files to the Gaussian program.
- Save the file in .gjf file format (the input file format for the program).
- Analyse the amino acids using the following t command: Calculate → Gaussian Calculation Setup → Job type (Optimization).
- From the Calculation Setup menu select the Gaussian Geometry Optimization Methods one after another and run the calculations.
- Save for every calculation the .out file (the output file format for the program).

With a homemade *.php program we generated .hin files from the .sdf files and generated the molecular descriptors (FMPI). FMPI molecular descriptors are the improved version of SMPI (Szeged Matrix Property Indices) [24,25] descriptors. With SMPI, distance matrix are calculated, and then for each pair of (distinct) atoms the atoms closer to the first than to the second atom of the pair are collected into a matrix [15]. The improvement made to SMPI is the extension of the principle applied in Szeged fragments to the other two matrices collecting fragments from molecules for pairs of atoms.

Therefore, the gene sequence of FMPI was increased from SMPI with one gene and the number of descriptors was multiplied by three (arriving at 4536) [23]. After we obtained 4536 descriptors for every amino acid, we used the Statistica program to perform the clustering and PCA analysis.

3. Results and Discussion

After applying the algorithm described in the Methodology section, a principal component analysis (PCA) and a clustering analysis were performed. The next figure (Figure 2) shows the first and second component as the result of the PCA analysis.

Principal Components Analysis Summary								
Number of components is 10								
99.8851% of sum of squares has been explained by all the extracted components.								
Component	R ² X	R ² X(Cumul.)	Eigenvalues	Q ²	Limit	Q ² (Cumul.)	Significance	Iterations
1	0.712532	0.712532	27.78873	0.663751	0.025652	0.663751	S	4
2	0.149726	0.862258	5.83933	0.453947	0.026327	0.816391	S	7
3	0.065159	0.927417	2.54122	0.363794	0.027038	0.883187	S	9
4	0.038401	0.965818	1.49764	0.354425	0.027788	0.924588	S	8
5	0.018368	0.984187	0.71635	0.444229	0.028582	0.958088	S	8
6	0.009164	0.993350	0.35739	0.506863	0.029422	0.979332	S	5
7	0.002725	0.996075	0.10627	0.319829	0.030314	0.985942	S	9
8	0.001399	0.997475	0.05458	0.184278	0.031261	0.988533	S	7
9	0.000967	0.998441	0.03770	0.276891	0.032269	0.991708	S	7
10	0.000410	0.998851	0.01598	-0.064624	0.033344	0.991172	S	14

Figure 2. The explained variation (R²X) and the predictive variation (Q²X) of the PCA components.

Each principal component is a linear combination of the variables from the whole data set. A total of 90,720 descriptors for each component and each method were analysed, or a total of 3.538.080 descriptors.

The result of the PCA analysis indicated that the principal components (Figure 2) explained our large amount of data at 99.8851%, which reflected the variance of the data.

The first component accounted for a maximum amount of total variance (71.25%) in the data analysed. The second component accounted for the maximum variance that was not explained by the first component (14.9%). The third component also accounted for the maximum variance (6.51%) after the first two.

The R²X describes the predictive accuracy and takes values between 0 and 1. The more significant a principal component, the larger its R²X. The explained variance (R²X_{adj}) is simply the explained variation (R²X) adjusted for the degrees of freedom.

The quality assessment, goodness-of-prediction (Q²) statistic is typically reported as a result of cross-validation and provides a qualitative measure of consistency between the predicted and original data. As we add more variables to the PCA analysis, the value of Q² increases. Large values of Q² indicates a relevant and significant analysis.

In the next figure (Figure 3), a score loading plot, the distribution of component 1 versus component 2 is represented. The plot indicates that the similar methods are indeed roughly grouped together. Furthermore, the loadings define the orientation of the principal components in space. The loading vectors are p1 and p2. In our case, the first three components explained most of the data. In the next figure, a score loading plot, the distribution of component 3 versus component 2 is represented (Figure 4).

The classification of the methods into four categories (Semi-Empirical, Density Functional Theory, Molecular Mechanics Møller–Plesset Perturbation Theory, Coupled-Cluster Theory and Hartree–Fock) in the Section 2 is not entirely valid if we take into consideration the similarity between them. The degree of similarity between the methods grouped the data into the main categories presented above, but also into different and mixed groups. The PCA and cluster analyses produced comparable results.

The cluster analysis dendrogram (Figure 5) shows the Euclidean distances between the 39 methods compared. The single linkage or nearest neighbour technique is one of the simplest hierarchical clustering methods. The Euclidian distance between the methods due to the large data set (3.538.080 variables) was very high. To compare these methods, the standardization of the linkage distance was chosen on the X-axis. (Dlink/Dmax) *100 represents the linkage distances (Dlink) divided by maximum linkage distance (Dmax).

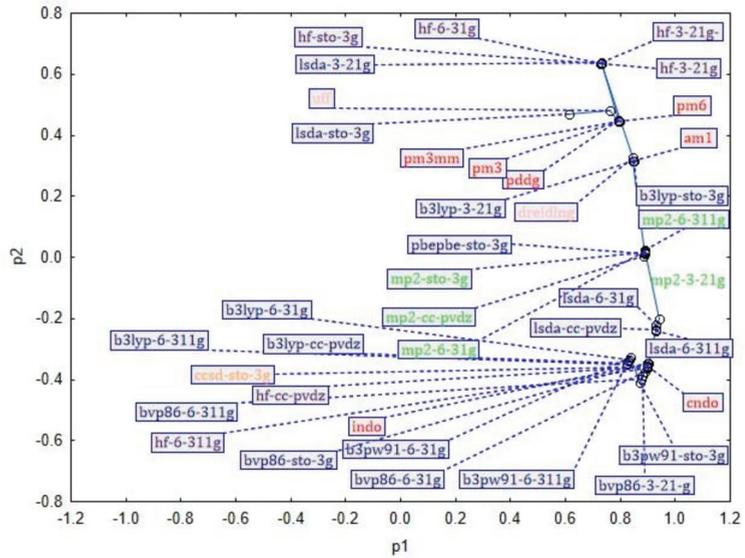


Figure 3. Score plot showing the distribution of the methods in the two principal components.

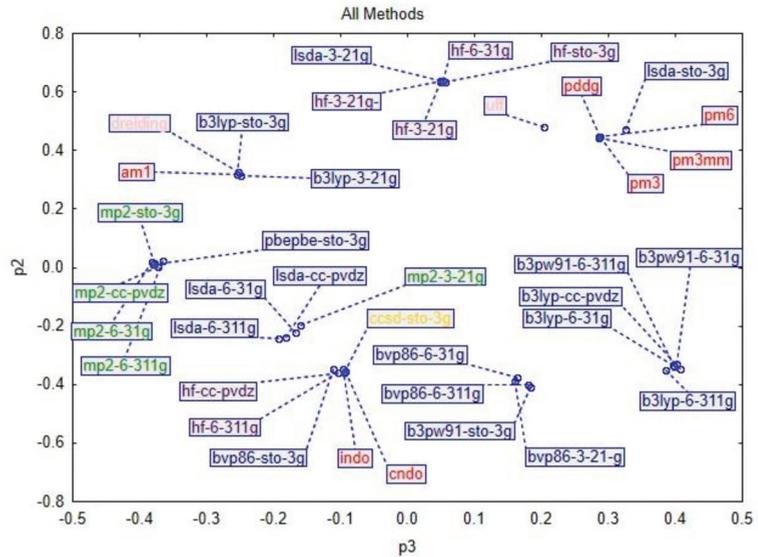


Figure 4. Score plot showing the distribution of the methods in the principal components p2 and p3.

The similarity between the optimization methods varies between the basis sets used. After obtaining the results of the PCA and clustering analyses, a classification can be made within the several different groups. The difference between the optimization methods was minimal; the tree clustering shows the relationship among them. For an extensive analysis, the data should be selected from different groups to obtain various results from multiple points of view.

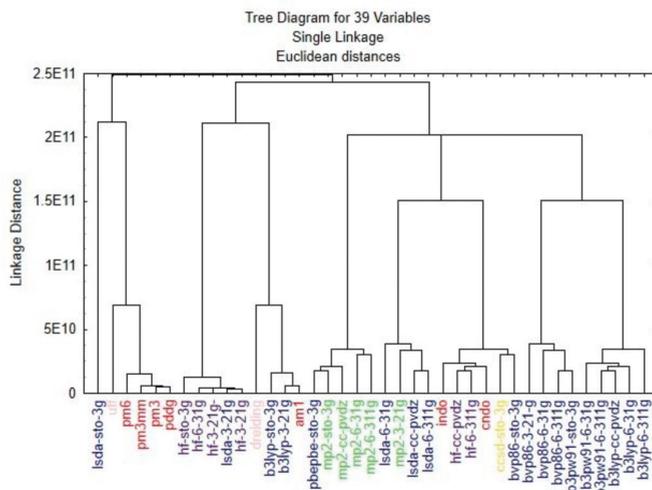


Figure 5. Clustering results.

Several studies use hybrid methods in their analysis [26–28] in order to obtain considerably better results. Davidson and Feller [28] in 1986 described a few criteria upon which a selection of the basis sets could be made, although since then many other methods have been introduced in computational chemistry. Because different theoretical methods and molecular properties have different basis set demands, different computer architectures and algorithms have different efficiency requirements, and the desired accuracy varies with the application, it is not possible to design one ‘optimum’ basis set.

Cramer [29] discussed the evolution of basis sets from the most widely used split-valence basis sets, such as 3-21G, 6-21G, 4-31G, 6-31G, and 6-311G [30], to modern examples of basis sets, such as cc-pCVDZ, cc-pCVTZ, etc. [31]. Comparing all the sets of comparisons, it is evident that the geometries for the molecules containing second-row elements are considerably more difficult to predict accurately than those for simpler organics. For example, it was found that AM1 is less successful when extended to these species than PM3 [32]. Furthermore, DFT methods feature limitations, such as different trends and high error accuracy [33].

The effort to determine the ‘best’ combinations of methods and basis sets that produce statistically good results for certain molecules and properties has become especially pronounced with the proliferation of the modern methods. Geometry optimization and energy minimization are fundamental tasks in molecular modelling and drug design. The failure to minimize energy and/or optimize geometry is directly converted to wrong molecular descriptors [34].

Because we used a very large data set, the results are more explicable if we divide them into different subgroups. After we performed the cluster and PCA analyses for every subgroup the following results were obtained.

For the semi-empirical methods, two principal components explain most of the data (Figure 6), and the cluster analysis showed the same tendency. The results can be divided into three main groups: am1; indo, cndo; and pm6, pm3mm, pm3, pddg. In conclusion, if we use one method from each group, this should be enough to describe our data.

For the Density Functional Theory methods, the statistical analysis looks a little different, because the dataset was larger this time. Most of the analyzed methods were part of this family.

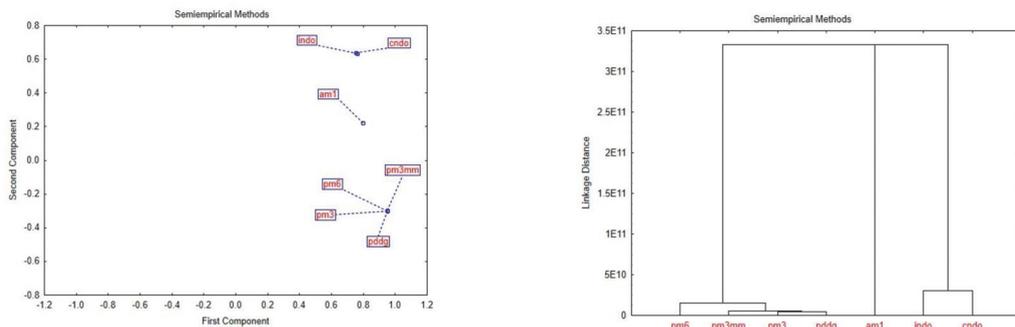


Figure 6. Score plot showing the distribution of the methods in the principal components p1 and p2 and cluster analysis.

Figure 7 demonstrates that the DFT methods are similar to each other, but also some ‘outlier’ methods can be observed. The methods can be divided into four major groups, and three methods, which are positioned separately.

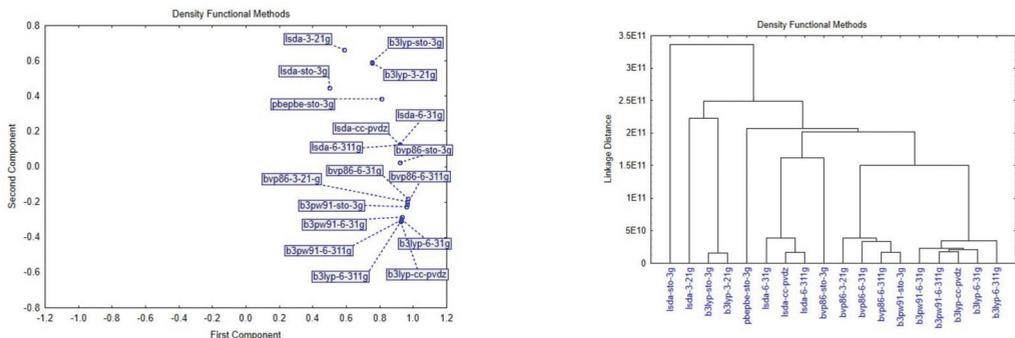


Figure 7. Score plot showing the distribution of the methods in the principal components p1 and p2 and cluster analysis.

In the Møller–Plesset Perturbation Theory methods, one principal component was identified (Figure 8). The methods are divided into two main groups, with one (mp2-3-21g) remaining a basis set.

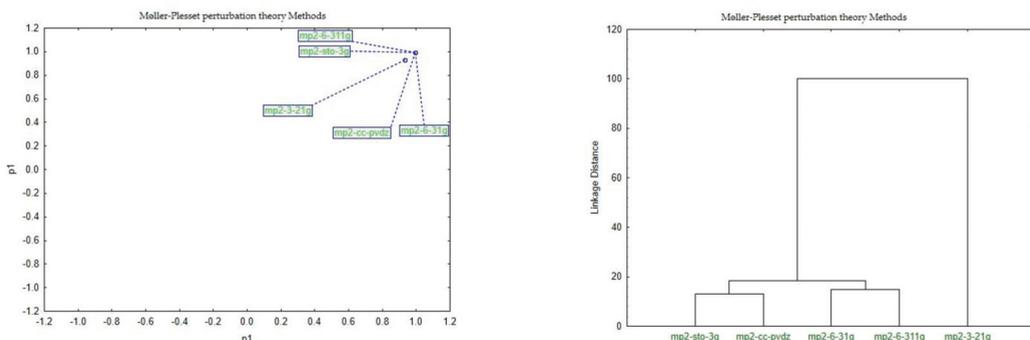


Figure 8. Score plot showing the distribution of the methods in the principal components p1 and p2 and cluster analysis.

The most widely used optimization calculation is the Hartree–Fock method. Based on our analysis, we identified two principal components (Figure 9) and two main groups.

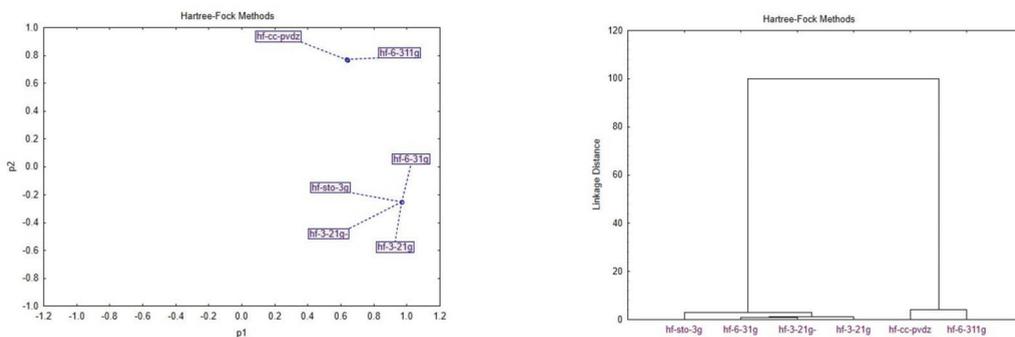


Figure 9. Score plot showing the distribution of the methods in the principal components p1 and p2 and cluster analysis.

The other methods, which are part of Coupled-Cluster Theory and Molecular Mechanics, could not be analysed separately because of the small dataset they represented. One method (CCSD) in Coupled-Cluster Theory and two methods (UFF, Dreiding) in Molecular Mechanics Theory did not reveal statistical significance if we analysed them alone. They are included in the first analysis, where all the methods are examined.

We performed another statistical analysis: the Single-Factor ANOVA test.

The reason for performing ANOVA was to see whether any difference existed between the groups for particular variables. The null hypothesis states that there was no significant difference between the methods analysed, based on the molecular descriptors calculated.

The *p*-value was 0.9995 > 0.05, so we accepted the null hypothesis, and concluded that there were no significant differences between the methods. In the Figure 10, the results of the ANOVA indicate that we cannot reject the null hypothesis.

ANOVA						
Source of Variation	SS	df	MS	F	P-value	F crit
Between Groups	2.92E+19	38	7.67E+17	0.405286	0.999584554	1.404833468
Within Groups	6.7E+24	3538041	1.89E+18			
Total	6.7E+24	3538079				

Figure 10. ANOVA test results.

4. Conclusions

In conclusion, we can state that the size of the basis set does not reflect its applicability in different circumstances. It is not possible to find the best basis set, only a couple of basis sets that fit our dataset. If we use different basis sets we obtain different results. Therefore, care must be taken to select the correct basis set. What makes the difference in results are the different selections and the correct use of optimization methods.

To find the best geometry optimization method to use in different situations, we must know are related. Two similar methods excluded each other in the analysis because they provided almost the same results. The results of our analysis show the correlation and the degree of relationship between the methods studied. The reclassification of the 39 examined methods facilitates the selection of the best basis sets for different study areas.

Author Contributions: Conceptualization, D.B.; Methodology, L.J.; Supervision, L.J.; Writing—original draft, D.B.; Writing—review & editing, L.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Technical University of Cluj-Napoca open access publication grant.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Slater, J.C. Atomic Shielding Constants. *Phys. Rev.* **1930**, *36*, 57. [CrossRef]
- Boys, S.F. Electronic Wave Functions. I. A General Method of Calculation for the Stationary States of any Molecular System. *Proc. R. Soc.* **1950**, *200*, 542–554.
- Joița, D.-M.; Tomescu, M.A.; Bălint, D.; Jäntschi, L. An Application of the Eigenproblem for Biochemical Similarity. *Symmetry* **2021**, *13*, 1849. [CrossRef]
- Jäntschi, L. The Eigenproblem Translated for Alignment of Molecules. *Symmetry* **2019**, *11*, 1027. [CrossRef]
- Schrödinger, E. An Undulatory Theory of the Mechanics of Atoms and Molecules. *Phys. Rev.* **1926**, *8*, 1049–1070. [CrossRef]
- Abegg, P.W.; Ha, T.-K. Ab initio calculation of spin-orbit-coupling constant from Gaussian lobe SCF molecular wavefunctions. *Mol. Phys.* **1974**, *27*, 763–767. [CrossRef]
- Binkley, J.S.; Pople, J.A.; Hehre, W.J. Self-consistent Molecular Orbital Methods. Small Split-valence Basis Sets for First-row Elements. *J. Am. Chem. Soc.* **1980**, *102*, 939–947. [CrossRef]
- Gordon, M.S.; Stephen, B.J.; Pople, J.A.; Pietro, W.J.; Hehre, W.J. Self-consistent Molecular-orbital Methods. Small Split-valence Basis Sets for Second-row Elements. *J. Am. Chem. Soc.* **1982**, *104*, 2797–2803. [CrossRef]
- Pietro, W.J.; Francl, M.M.; Hehre, W.J.; DeFrees, D.J.; Pople, J.A.; Binkley, J.S. Self-consistent Molecular Orbital Methods. Supplemented Small Split-valence Basis Sets for Second-row Elements. *J. Am. Chem. Soc.* **1982**, *104*, 5039–5048. [CrossRef]
- Pople, J.A. Quantum Chemical Models. *Angew. Chem.* **1999**, *38*, 13–14. [CrossRef]
- Ditchfield, R.; Hehre, W.J.; Pople, J.A. Self-Consistent Molecular-Orbital Methods. IX. An Extended Gaussian-Type Basis for Molecular-Orbital Studies of Organic Molecules. *J. Chem. Phys.* **1971**, *54*, 724–728. [CrossRef]
- Banerjee, T.; Ramalingam, A. Molecular Modeling and Optimization. In *Desulphurization and Denitrification of Diesel Oil Using Ionic Liquids: Experiments and Quantum Chemical Predictions*; Elsevier: Amsterdam, The Netherlands, 2015; pp. 11–38.
- Hohenberg, P.; Kohn, W. Inhomogeneous Electron Gas. *Phys. Rev.* **1964**, *136*, B864. [CrossRef]
- Diudea, M.V.; Gutman, L.; Jäntschi, L. *Molecular Topology*; Nova Science Publishers: New York, NY, USA, 2002.
- Petersson, G.A.; Malick, D.K.; Wilson, W.G.; Ochterski, J.W.; Montgomery, J.A., Jr.; Frisch, M.J. Calibration and comparison of the Gaussian-2, complete basis set, and density functional methods for computational thermochemistry. *J. Chem. Phys.* **1998**, *109*, 10570–10579. [CrossRef]
- Scuseria, G.E. Comparison of coupled-cluster results with a hybrid of Hartree-Fock and density functional theory. *J. Chem. Phys.* **1992**, *97*, 7528–7530. [CrossRef]
- Zheng, G.; Irle, S.; Morokuma, K. Performance of the DFTB method in comparison to DFT and semiempirical methods for geometries and energies of C₂₀–C₈₆ fullerene isomers. *Chem. Phys. Lett.* **2005**, *412*, 210–216. [CrossRef]
- Hill, J.G. Gaussian basis sets for molecular applications. *Int. J. Quantum Chem.* **2012**, *113*, 21–34. [CrossRef]
- Spies, M. Transducer field modeling in anisotropic media by superposition of Gaussian base functions. *J. Nondestruct. J. Acoust. Soc. Am.* **1999**, *105*, 633. [CrossRef]
- Kayano, M.; Dozono, K.; Konishi, S. Functional Cluster Analysis via Orthonormalized Gaussian Basis Expansions and Its Application. *J. Classif.* **2010**, *27*, 211–230. [CrossRef]
- Available online: <https://pubchem.ncbi.nlm.nih.gov/> (accessed on 5 September 2021).
- Jäntschi, L.; Bolboacă, S. Molecular modeling in compounds series with descriptors families. *An. Univ. Oradea Fasc. Chim.* **2016**, *23*, 5–14.
- Bolboacă, S.D.; Jäntschi, L. Nano-quantitative structure-property relationship modeling on C42 fullerene isomers. *J. Chem.* **2016**, *1*, 1. [CrossRef]
- Jäntschi, L. Predicția Proprietăților Fizico-Chimice Și Biologice Cu Ajutorul Descriptorilor Matematici. Ph.D. Thesis, Universitatea “Babeș-Bolyai” din Cluj-Napoca, Cluj-Napoca, Romania, 2000.
- Scott, A.P.; Radom, L. Harmonic Vibrational Frequencies: An Evaluation of Hartree-Fock, Møller-Plesset, Quadratic Configuration Interaction, Density Functional Theory, and Semiempirical Scale Factors. *J. Phys. Chem.* **1996**, *100*, 16502–16513. [CrossRef]
- Batra, P.; Bernd, G.; Gescheidt, M.S.G.; Houk, K.N. Calculations of Isotropic Hyperfine Coupling Constants of Organic Radicals. An Evaluation of Semiempirical, Hartree-Fock, and Density Functional Methods. *J. Phys. Chem.* **1996**, *100*, 18371–18379. [CrossRef]
- Russ, N.J.; Crawford, T.D.; Tschumper, G.S. Real versus artifactual symmetry-breaking effects in Hartree-Fock, density-functional, and coupled-cluster methods. *J. Chem. Phys.* **2004**, *120*, 7298. [CrossRef] [PubMed]
- Davidson, E.R.; Feller, D. Basis Set Selection for Molecular Calculations. *Chem. Rev.* **1988**, *86*, 661–696. [CrossRef]
- Cramer, C.J. *Essentials of Computational Chemistry: Theories and Models*; John Wiley and Sons, Ltd.: Hoboken, NJ, USA, 2002.
- Pople, J.A.; Beveridge, D.L.; Dobosh, P.A. Approximate Self-Consistent Molecular-Orbital Theory. Intermediate Neglect of Differential Overlap. *J. Chem. Phys.* **1967**, *47*, 2026. [CrossRef]
- Dunning, T.H. Gaussian basis sets for use in correlated molecular calculations. The atoms boron through neon and hydrogen. *J. Chem. Phys.* **1989**, *90*, 1007. [CrossRef]

32. Stewart, J.J.P. Optimization of parameters for semiempirical methods. Applications. *J. Comput. Chem.* **1989**, *10*, 221. [[CrossRef](#)]
33. Jensen, F. Atomic orbital bases sets. *WIREs Comput. Mol. Sci.* **2012**, *3*, 273–295. [[CrossRef](#)]
34. Jäntschi, L. Computer assisted geometry optimization for in silico modeling. *Appl. Med. Inform.* **2011**, *29*, 11–18.

Article

Exact Maximum Clique Algorithm for Different Graph Types Using Machine Learning

Kristjan Reba ^{1,2}, Matej Guid ², Kati Rozman ³, Dušanka Janežič ^{3,*} and Janez Konc ^{1,*}

¹ Theory Department, National Institute of Chemistry, Hajdrihova 19, 1000 Ljubljana, Slovenia; kristjan.reba96@gmail.com

² Faculty of Computer and Information Science, University of Ljubljana, Večna Pot 113, 1000 Ljubljana, Slovenia; matej.guid@fri.uni-lj.si

³ Faculty of Mathematics, Natural Sciences and Information Technologies, University of Primorska, Glagoljaška ulica 8, 6000 Koper, Slovenia; kati.rozman@gmail.com

* Correspondence: dusanka.janezic@upr.si (D.J.); konc@cmm.ki.si (J.K.)

Abstract: Finding a maximum clique is important in research areas such as computational chemistry, social network analysis, and bioinformatics. It is possible to compare the maximum clique size between protein graphs to determine their similarity and function. In this paper, improvements based on machine learning (ML) are added to a dynamic algorithm for finding the maximum clique in a protein graph, Maximum Clique Dynamic (MaxCliqueDyn; short: MCQD). This algorithm was published in 2007 and has been widely used in bioinformatics since then. It uses an empirically determined parameter, Tlimit, that determines the algorithm's flow. We have extended the MCQD algorithm with an initial phase of a machine learning-based prediction of the Tlimit parameter that is best suited for each input graph. Such adaptability to graph types based on state-of-the-art machine learning is a novel approach that has not been used in most graph-theoretic algorithms. We show empirically that the resulting new algorithm MCQD-ML improves search speed on certain types of graphs, in particular molecular docking graphs used in drug design where they determine energetically favorable conformations of small molecules in a protein binding site. In such cases, the speed-up is twofold.

Keywords: maximum clique; protein graphs; machine learning; ProBiS

Citation: Reba, K.; Guid, M.; Rozman, K.; Janežič, D.; Konc, J. Exact Maximum Clique Algorithm for Different Graph Types Using Machine Learning. *Mathematics* **2022**, *10*, 97. <https://doi.org/10.3390/math10010097>

Academic Editors: Lorentz Jäntschi, Mihaela Tomescu and Ioannis G. Tsoulos

Received: 11 November 2021

Accepted: 23 December 2021

Published: 28 December 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Finding the maximum clique in a graph is a well-studied NP-complete problem [1]. Recently developed algorithms significantly reduce the time required to search for a maximum clique, which is of great practical importance in many fields such as bioinformatics, social network analysis, and computational chemistry [2,3].

There have been many advances in the search for faster algorithms for maximum cliques, many of which focus on specific domains of graphs [4–7]. To make the algorithm work fast on general graphs, some good heuristics have been proposed to speed up the branch-and-bound search [1,4,8–15]. One such algorithm is MCQD, on which we have built [4]. It has been shown that the MCQD algorithm is faster than many other similar branch-and-bound algorithms in finding maximum cliques [1]. In the MCQD algorithm, there is a single parameter that can be set before the algorithm is executed. This parameter, called Tlimit, controls the fraction of a graph on which tighter upper bounds apply to the size of a maximal clique. These upper bounds require that $O(N^2)$ be computed. The fraction of a graph on which looser upper bounds are used ($O(N \log N)$) is empirically estimated to be 0.025 for random graphs. Even though MCQD seems to progress quickly with a default value of Tlimit in many graphs, there are some graphs where Tlimit performs poorly [4]. In particular, the Tlimit parameter is suboptimal in some dense and synthetic graphs of the DIMACS benchmark [16]. Here, we present an improvement to the original

MCQD algorithm that automatically determines the value of the Tlimit parameter for the MCQD algorithm. We predict that the Tlimit parameter uses machine learning for the input graph. The code used to perform the experiments is freely available at <http://insilab.org/mcqd-ml> (accessed on 9 November 2021).

1.1. Problem Description and Notation

Let $G = (V, E)$ be an undirected graph, where $V = 1, \dots, n$ is a set of vertices and $E \subset V \times V$ is a set of edges. A clique C in the graph G is a set of nodes defined such that there exists an edge between every two nodes in C . We say that C is a maximum clique if its cardinality $|C|$ is the largest among all cliques in the graph G . The maximum clique problem (MCP) is an optimization problem that seeks the maximum clique in a given graph. The clique number $w(G)$ of graph G is the number of nodes in the maximum clique of graph G . The maximum clique problem is strictly equivalent to a maximum independent set (MIS) as well as the minimum vertex cover problem (MVC). Finding the maximum clique is an NP-complete problem. We do not know if there is an algorithm for this group of problems that can find the solution in polynomial time. It is likely that no such algorithm exists.

1.2. Maximum Clique Dynamic (MCQD) Algorithm

The MCQD algorithm is based on a branch and bound principle [4]. It uses approximate graph coloring to estimate the upper bound of the maximum clique size and is shown in Algorithm 1.

Algorithm 1. Dynamic algorithm for maximum clique search.

```

1: procedure MaxCliqueDyn(R, C, level)
2:   S[level] ← S[level] + S[level - 1] - Sold[level]
3:   Sold[level] ← S[level - 1]
4:   while R ≠ ∅ do
5:     choose a vertex p with maximum C(p) (last vertex) from R
6:     R ← R \ {p}
7:     if |Q| + C[index_of_p_in_R] > |Qmax| then
8:       Q ← Q ∪ {p}
9:       if R ∩ Γ(p) ≠ ∅ then
10:        if S[level] / ALL_STEPS < Tlimit then
11:          calculate the degrees of vertices in G(R ∩ Γ(p))
12:          sort vertices in R ∩ Γ(p) in descending order with respect to
their degrees
13:          ColorSort(R ∩ Γ(p), C')
14:          S[level] ← S[level] + 1
15:          ALL_STEPS ← ALL_STEPS + 1
16:          MaxCliqueDyn(R ∩ Γ(p), C', level + 1)
17:        else if |Q| > |Qmax| then
18:          Qmax ← Q
19:          Q ← Q \ {p}

```

The algorithm stores the current clique in the variable Q and keeps track of the current maximum clique size in the variable Q_{max} . As an input, it accepts an ordered set of nodes based on their color, a set of colors, and the level variable which provides the current depth of the recursive function. The algorithm also uses two global variables, $S[level]$ and $Sold[level]$, which store the sum of steps up to the current level of algorithm progression and the previous level $Sold[level] = S[level - 1]$. With the Tlimit parameter, we can limit the use of the graph coloring of vertices R sorted by their degree. When the proportion of steps up to a certain level of recursion is less than Tlimit, we perform additional operations of recalculating the vertex degrees for the remainder of the graph and of resorting these vertices according to their descending degrees. This additional work increases the tendency of the ColorSort function to estimate a tighter upper bound for the size of a maximum

clique, generally reducing the number of steps and time necessary for the algorithm to find a maximum clique. The Tlimit value used in the original paper [4] was empirically determined on a sample of random graphs and was set to a value of 0.025.

1.3. Protein Product Graphs and Use of Molecular Docking Graphs in Drug Discovery

To move drugs from the research phase to the trial phase, the most promising molecules must be identified from a set of potential candidates. This requires a detailed knowledge of the functions of drug target proteins, which is often lacking. Protein functions can be determined by comparing the structure of unknown proteins to proteins with known functions [2]. To compare proteins with each other, we can represent them as protein graphs, such as we did with the ProBiS (Protein Binding Sites) algorithm [17]. Two protein graphs can be compared by constructing a protein product graph, which is a Cartesian product of the two protein graphs and captures all possible overlaps of one protein with the other. Finding a maximum clique in this protein product graph is directly equivalent to finding the alignment that overlaps most of the vertices of the protein graphs. The quality of the overlap is an indication of the similarity of the proteins.

Another application for maximum clique search is molecular docking, which is often performed as a high-throughput screening approach whose goal is to predict the binding position and binding affinity of potential ligands of a target protein [18]. In a particular class of molecular docking called fragment docking, which was explored in our ProBiS-Dock docking algorithm, a maximum clique algorithm is used to reconstruct a docking graph of the small molecule in a protein-bound conformation from fragments of the previously docked molecule. The calculated binding affinities of the docked fragments can be included in this graph as node weights, resulting in a weighted docking graph. A clique with maximum weight in such a graph represents the docked conformation of a small molecule with the highest binding affinity among all possible conformations of that small molecule. This allows the algorithm to discover potential new ligands of a protein that could become drugs in the future.

2. Overview of Graph Theory and Neural Networks Approaches

We describe the novel developed MCQD-ML (Maximum Clique Dynamic–Machine Learning) algorithm that was tested with different types of graphs and incorporates different machine learning models.

2.1. Graphs Used for Training and Testing

To train the machine learning algorithm, we first create a variety of graphs. In order to capture the largest possible variety of target graphs in our training set, we include 10,000 sparse and dense random graphs, as well as 15 complete protein graphs and 200 molecular docking graphs. The random graphs are generated such that each edge exists with probability d , where d is greater than 0.99 in dense graphs. The types of graphs are presented in the following sections.

2.2. Molecular Docking Graphs

To identify energetically preferred docking conformations of potential ligands, we performed a maximum clique search in molecular docking graphs. A molecular docking graph is a graph whose nodes are docked molecular fragments and in which two nodes are connected if the docked fragments can be connected with linker atoms to reconstruct the original docked molecule. Each node is assigned a weight representing the binding energy (or binding affinity) of a docked fragment. By performing a maximum weight clique search on docking graphs, we can find the combination of docked fragments that yields the conformation with the lowest energy of the docked small molecule with a given protein. We use the ProBiS-Dock algorithm to build molecular docking graphs. The algorithm is used to find the ligands with the highest potential when screening multiple ligands on a target protein [18,19].

2.3. Protein Product Graphs

In the ProBiS algorithm [17], proteins are represented as protein graphs. Each node in a protein graph represents the spatial coordinates of the surface amino acid functional groups. If the distance between nodes u and v is less than 15 Å, there is an edge between two nodes in a protein graph. We can formulate the comparison of two proteins as a maximum clique search by using the notion of a protein product graph. A maximum clique in a protein product graph is a superposition of protein graphs in which the majority of the nodes of two graphs are aligned. The protein product graph of two protein graphs G_1 and G_2 is defined by a set of nodes, $V(G_1, G_2) = V(G_1) \times V(G_2)$. Each node in a product graph consists of a node u from graph G_1 and a node v from graph G_2 , both of which represent a similar functional group in the original proteins. In general, a protein product graph can have $|V_1| \times |V_2|$ nodes, but this number is reduced by keeping only the nodes from the original protein graphs G_1 and G_2 that have similar neighbourhoods in a 6 Å sphere.

2.4. Small Protein Product Graphs

The problem with protein product graphs is the large size of the adjacency matrix, which can exceed the available memory depending on the size of the proteins being compared. It is possible to split a large protein product graph into smaller product graphs that are much denser and contain only a subset of the nodes of the original product graph. The advantage of smaller and denser graphs is the speed at which they can be processed. A disadvantage of smaller protein graphs is the loss of information. If we look for a maximum clique in a small product graph, there is no guarantee that the same clique will be the maximum clique in the entire protein product graph.

2.5. Protocol for Machine Learning on Graphs

To gather as much information as possible about the graph, it is necessary to perform machine learning directly on the graph. To this end, we tested several different graph neural network models and a support vector regression algorithm with the Weisfeiler–Lehman kernel function [20–30], which are listed in Table 1. We tested three different graph neural network models that can model data of different complexity with inductive biases. They are (i) Graph Convolutional Networks (GCN) [28], (ii) Graph Attention Networks (GAT) [29,30], and (iii) Graph Isomorphism Networks (GIN) [15,25]. We trained the models on a given training set and then used them to predict Tlimit values for graphs on the test set. The test set contained 15 dense random graphs, 10 small product graphs, 3 product graphs, and 10 docking graphs. We evaluated the performance of the algorithms and calculated the average speed of the standard MCQD algorithm for each set of test graphs. We also calculated the combined speed for the entire test set by summing the runtimes of the algorithms for many different types of graphs and dividing the sum by the runtime required for the MCQD algorithm.

Table 1. Different machine learning methods employed.

ML Method	Description	Works on Graphs	Representative Power	References
XGBoost	Ensemble of gradient boosted trees.	No. Best for tabular data.	Works well on tabular data and extracted features of a graph. Results depend on the quality of features extracted.	[21,22]
SVR-WL	Support vector machine with Weisfeiler–Lehman kernel	Yes.	Can distinguish non-iso-morphic graphs.	[24]
GNN	Graph Neural Networks (GCN, GAT, GIN)	Yes.	Can distinguish most graphs and learn good representations.	[15,25,28–30]

3. Materials and Methods

3.1. Preparation of a Labeled Training Set

Before attempting to use machine learning to improve the selection of the Tlimit parameter value for specific input graphs, we prepared a labeled training set in which different Tlimit values were identified for each graph with the time required to detect the maximum clique. So, we performed the maximum clique search with different Tlimit values on a set of graphs and recorded the time taken by the MCQD algorithm to find the maximum clique. For each generated graph, we ran the MCQD algorithm multiple times for different values of the Tlimit parameter to record the Tlimit values approximately uniformly on a logarithmic scale from 0 to 1. When running MCQD for many graphs and many Tlimit values for each graph, this step becomes computationally intensive. After collecting all Tlimit pairs and their corresponding computation time, we selected the Tlimit value with the lowest time as the best Tlimit value for a graph. This value was then used as the label value for training the machine learning models. The training set consists of graphs as input and the optimal Tlimit value for each graph as the target variable.

3.2. Maximum Clique Dynamic Algorithm with Machine Learning (MCQD-ML)

The idea behind the MCQD-ML algorithm is shown in Figure 1. The algorithm performs inference on the graph to determine a Tlimit parameter before the MCQD algorithm starts, and the MCQD algorithm then uses this parameter instead of the hard-coded parameter. In this way, we obtain the best Tlimit parameter for a given graph and use it to make the MCQD algorithm run faster.

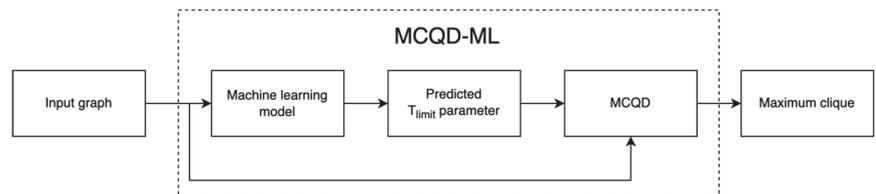


Figure 1. MCQD-ML algorithm architecture.

We used an implementation of the MCQD algorithm that can search for a maximum clique as well as a maximum weighted clique. This algorithm is available as source code at <https://gitlab.com/janezkonc/insidrug/-/blob/master/lib/glib/mcqd.cpp> (accessed on 9 November 2021). For experimental purposes, we created two training sets and two test sets for molecular docking graphs. One set contains the docking graphs with weights, and in the other set we omit the weights from the docking graphs and assume that all nodes have the same weight. All other graphs are unweighted.

3.3. Evaluation of Possible Acceleration of the MCQD Algorithm

To determine if any speed-ups are possible by tuning the parameter Tlimit, we plot the time needed for MCQD to find the maximum clique at different values of the Tlimit parameter. In Figure 2, it can be observed that on a random 150 node graph, the default value of parameter is well suited and the maximum clique can be found relatively quickly compared to other values of Tlimit.

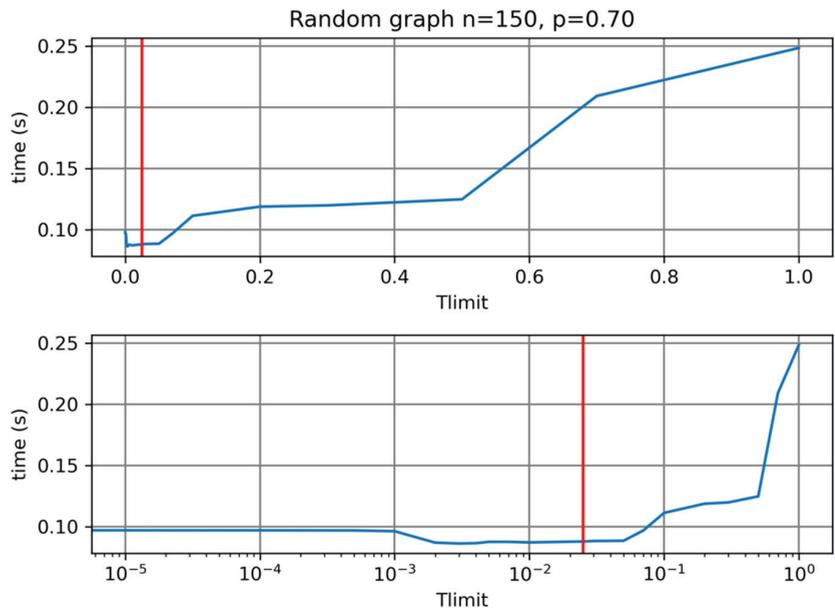


Figure 2. Time necessary for MCQD to find the maximum clique on a random graph with 150 nodes and density $p = 0.7$. The red line represents the default value (0.025) of the MCQD algorithm.

In Figure 3 we evaluate the impact of the initial sorting of vertices on the time required for MCQD to finish searching. We found that initial sorting of vertices has no significant impact on the time needed by MCQD to find the maximum clique.

3.4. Evaluation of the Effect of Machine Learning Models on Validation Sets

We perform an evaluation of the trained machine learning models we presented. The models are evaluated using the R^2 score on the validation set, which contains graphs from different domains. This value (also called coefficient of determination) is used in statistics to evaluate statistical models. Values of R^2 typically range from 0 to 1, with 1 being the best possible value. If the model predicts the mean of the data (constant value), the R^2 value is 0. The value can also be negative if the model does not perform as well as the mean of the data. The results of our evaluation are shown in Table 2.

We find that the model GAT achieves the highest R^2 value, with any machine learning model performing better than the standard MCQD parameter choice, which is nearly equal to 0. Thus, we expect the GAT model to perform the best, while the other models in the test set are not as fast. In the next section, we evaluate the models based on the time they take to find the maximum clique.

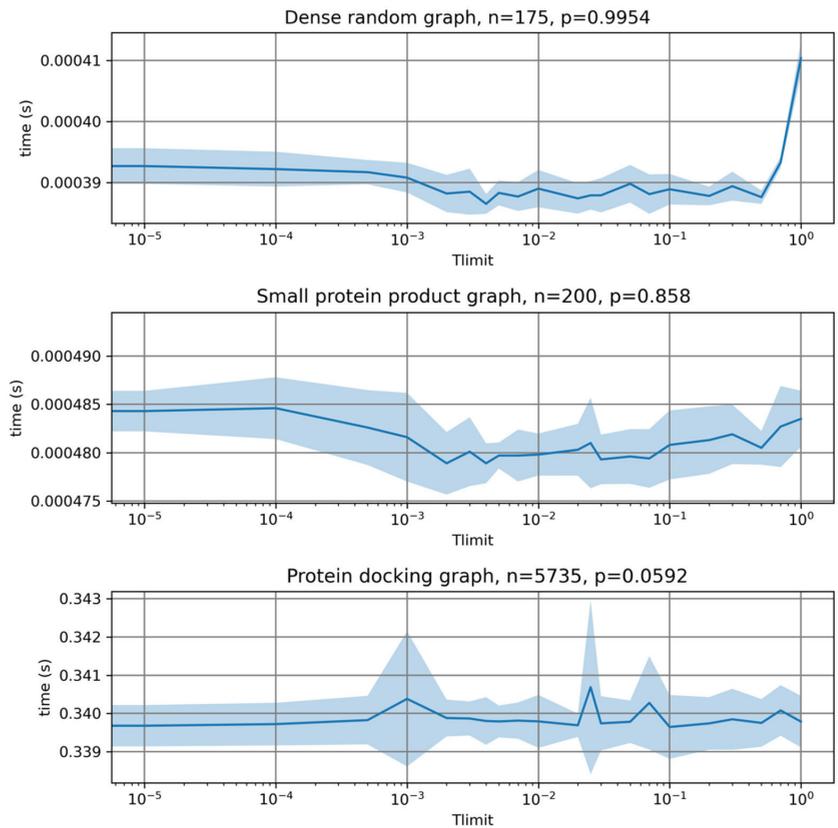


Figure 3. Time needed by the MCQD algorithm to find the maximum clique on different graphs independent of Tlimit. The blue line represents the mean time and the shaded area represents standard deviation over 20 runs of the MCQD algorithm.

Table 2. R² values from different machine learning models.

Model Name	R ² Score on Validation Set
MCQD	−0.02
XGB	0.15
SVR-WL	0.21
GCN	0.42
GAT	0.55
GIN	0.16

4. Results

Our Maximum Clique Dynamic–Machine Learning (MCQD-ML) algorithm was implemented in Python (ML part) and C++ (MCQD part) and uses only 1 CPU core. Here we evaluated the MCQD-ML algorithm on several previously described sets and compared the results with the standard MCQD algorithm. The MCQD algorithm was extensively compared and benchmarked [1,2,4]. The computational experiments were performed on an AMD Ryzen 9 3900X 12-core with a CPU frequency of 2 GHz. The MCQD-ML maximum clique algorithm was compared with the original MCQD algorithm on random graphs, protein product graphs, and molecular docking graphs. We limited the time available for the algorithms to 2000 s. To compare the performance of the algorithms, we use two metrics: (i) the speed-up on a test set, i.e., the time taken by the MCQD algorithm to find the

maximum clique for each graph in a test set divided by the time taken by the MCQD-ML algorithm to find the maximum clique on a given set of graphs and (ii) the average speed-up on a test set is calculated by taking the speed-up of the MCQD-ML algorithm for each graph and averaging it over all graphs.

We used various machine learning models to predict the value of the Tlimit parameter, and then used this value in the MCQD-ML algorithm to evaluate its performance on several test sets, including random graphs, protein product graphs, and molecular docking graphs. We compared it with the basic MCQD algorithm with default value Tlimit = 0.025. MCQD-ML is implemented with the following machine learning models: XGBoost (XGB), Graph Convolutional Neural Network (GCN), Graph Attention Neural Network (GAT), Graph Isomorphism Network (GIN), and Support Vector Regressor with the Weisfeiler–Lehman Kernel (SVR-WL). For each model, we record the time it takes MCQD to find the maximum clique with a predicted value of the parameter Tlimit.

4.1. Dense Random Graphs

In a series of tests with dense random graphs, we found that GAT outperforms other models, including the original MCQD algorithm. The faster speed of GAT compared to MCQD is not great, as GAT is about 18% faster on average and only 4% faster on the entire test set of dense random graphs.

From Table 3 and Figure 4 we can see that the default MCQD algorithm is nearly optimal for some graphs and almost two times slower compared to tests with a better choice of the value of the parameter. There exists no Tlimit for which MCQD will find the maximum clique substantially faster.

Table 3. Times needed by algorithms to find the maximum clique for each graph in a test set of dense random graphs. Best times are in bold.

n	p	MCQD	XGB	GCN	GAT	GIN	SVR-WL
63	0.9944	0.0008	0.0007	0.0007	0.0007	0.0011	0.0007
113	0.9987	0.0024	0.0022	0.0023	0.0022	0.0028	0.0023
121	0.9955	0.0044	0.0042	0.0042	0.0041	0.0065	0.0047
175	0.9954	0.0171	0.0159	0.0157	0.0151	0.0194	0.0157
304	0.9911	8.8271	6.4638	7.305	6.2747	8.6368	9.3515
414	0.9943	2.3677	1.8574	1.7514	1.2559	5.0611	1.9631
443	0.9938	57.898	55.2395	66.4033	58.8421	428.473	265.817
475	0.9979	0.2406	0.2327	0.2413	0.2305	0.2336	0.2287
476	0.9977	0.3262	0.2695	0.3024	0.2906	0.2703	0.2652
524	0.9992	0.5042	0.438	0.466	0.4482	0.4341	0.4278
622	0.9981	0.6802	0.6225	0.6212	0.6082	0.6253	0.612
690	0.9978	326.052	1124.65	511.101	428.92	115.922	−1.0000
828	0.9979	382.55	322.846	431.302	254.81	−1.0000	1217.84
931	0.9995	1.98	1.7438	1.7799	1.7807	1.7584	1.7017
941	0.9988	25.4684	12.2125	22.7202	20.2954	12.3739	12.044
Speedup			0.52	0.77	1.04	0.73	0.31
Average speedup			1.14	1.04	1.18	1.09	1.05

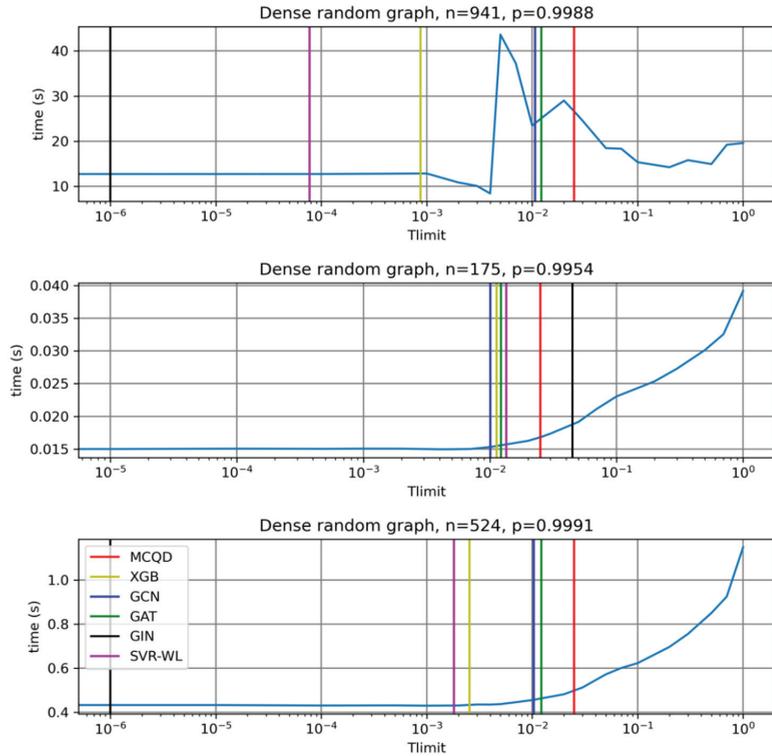


Figure 4. Time that MCQD algorithm and each variant of the MCQD-ML algorithm needs to find the maximum clique on three different graphs from a test set of dense random graphs dependent on the Tlimit parameter.

4.2. Small Protein Product Graphs

From Table 4 it can be observed that most ML models fail to reach the performance of the default MCQD algorithm.

Table 4. Times that algorithms need to find maximum clique for each graph from test set of small product graphs. Best times are in bold.

n	p	MCQD	XGB	GCN	GAT	GIN	SVR-WL
61	0.9792	0.0008	0.0008	0.0008	0.0007	0.0012	0.0008
138	0.9422	0.0079	0.0137	0.0078	0.0074	0.0102	0.0076
200	0.8581	0.0358	0.0398	0.0388	0.0381	0.0327	0.0393
271	0.9852	0.2062	0.2004	0.1972	0.1907	0.1831	0.1913
346	0.9091	2.3032	0.7774	2.8878	2.8278	14.3173	4.7920
451	0.9743	0.8956	0.8989	0.8955	0.8464	1.3257	1.3406
563	0.9800	1.7685	1.8496	1.7348	1.6936	1.7277	1.6994
655	0.9692	2.3652	2.3684	2.4533	2.6894	15.9674	15.8806
750	0.9625	4.7147	5.8504	4.2834	4.1741	8.0964	8.0182
905	0.9412	18.4683	16.2290	25.2455	18.5778	−1.0000	283.5820
Speedup			1.08	0.81	0.99	0.29	0.09
Average speedup			1.13	0.96	1.02	0.69	0.70

4.3. Protein Product Graphs

In Table 5 and Figure 5 we observe that any substantial speed-ups on product graphs are not achievable because the default value of parameter Tlimit is almost optimal for all product graphs in the test set.

Table 5. Times that algorithms need to find the maximum clique for each graph from a test set of full product graphs. Best times are in bold.

n	p	MCQD	XGB	GCN	GAT	GIN	SVR-WL
27,840	0.0069	9.8018	9.9147	10.2909	9.8759	10.9743	10.1547
36,841	0.0060	18.6482	19.7002	19.1695	19.0900	23.4188	19.9433
121,359	0.0024	198.5920	199.5000	199.4170	199.8520	378.1210	199.3480
Speedup			0.99	0.99	0.99	0.55	0.99
Average speedup			0.98	0.98	0.99	0.74	0.97

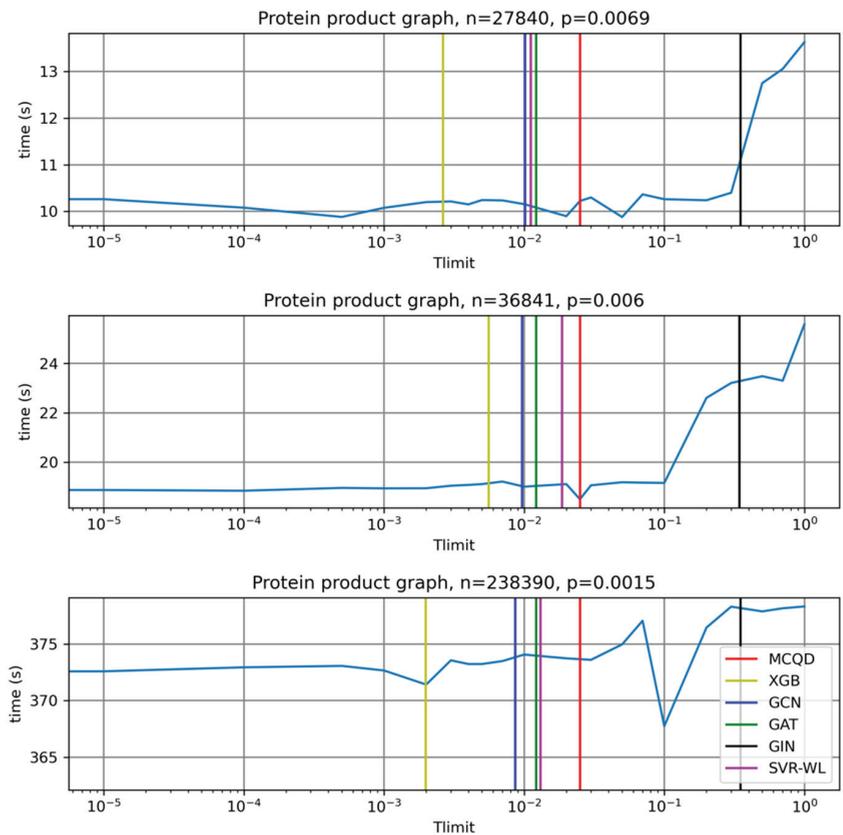


Figure 5. Time needed by MCQD algorithm and each variant of the MCQD-ML algorithm to find a maximum clique on three different graphs from test set of protein product graphs dependent of Tlimit parameter.

4.4. Molecular Docking Graphs

On the test set of molecular docking graphs, we observe in Table 6 that the GAT model and SVR-WL outperform every other model, including the MCQD algorithm. The performance of GAT and SVR-WL is almost two times faster with the whole test set, and 34% faster on average. On Figure 6 we observe that the molecular docking graphs vary

in the optimal parameter value. While on a graph with 1779 nodes the default value of the parameter is nearly optimal, it is not suitable for the graph with 5309 nodes where it is more than three times slower than with the optimal parameter value.

Table 6. Times that algorithms need to find maximum clique for each graph from test set of docking graphs. Best times are in bold.

n	p	MCQD	XGB	GCN	GAT	GIN	SVR-WL
345	0.1266	0.0025	0.0025	0.0026	0.0025	0.0026	0.0025
1779	0.1108	0.0940	0.0948	0.0943	0.0939	0.0952	0.0952
1851	0.1580	0.1606	0.1606	0.1829	0.1562	0.4394	0.1580
3233	0.1620	3.6941	3.4489	1.9791	1.9817	3.5981	1.9176
4211	0.0448	0.3889	0.3900	0.3990	0.3783	0.3967	0.3823
5293	0.1119	2.7147	6.0876	2.9925	2.6810	5.4606	2.7374
5309	0.1474	26.3695	19.1478	33.7628	7.7648	7.8752	7.5596
5735	0.0592	1.2673	1.2681	1.3803	1.2271	1.3196	1.2476
6294	0.1382	3.0941	15.8609	3.3343	3.0363	3.1517	3.0399
7211	0.1012	4.4230	11.2341	4.5631	4.2580	9.0498	4.3415
Speedup			0.73	0.86	1.96	1.41	1.96
Average speedup			0.84	1.01	1.34	1.10	1.34

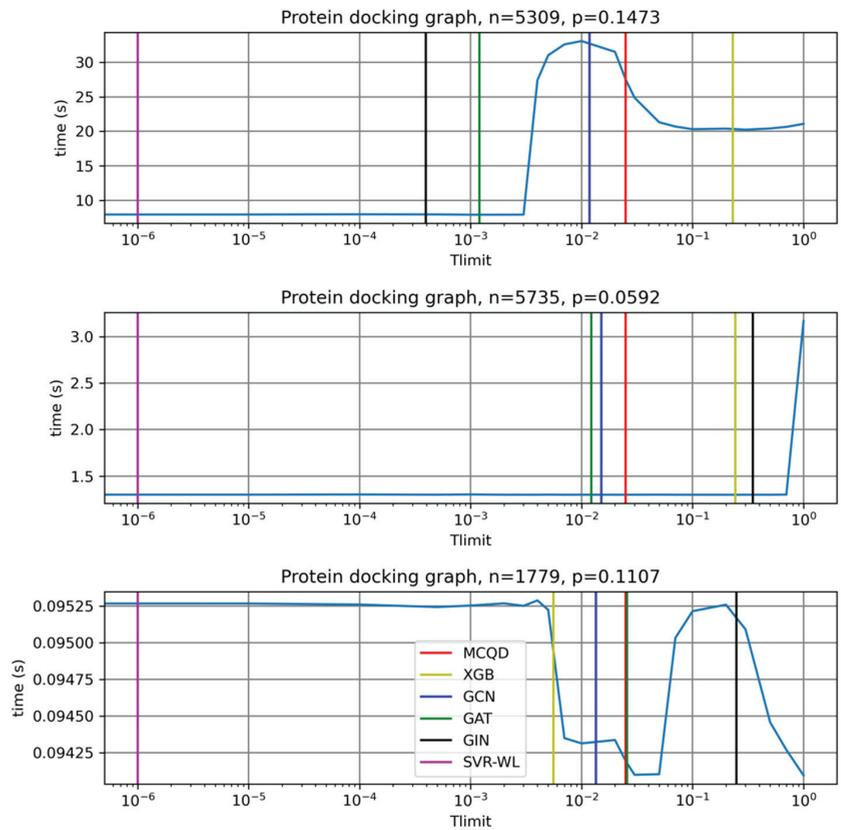


Figure 6. Time that MCQD algorithm and each variant of the MCQD-ML algorithm need to find the maximum clique on three different graphs from a test set of molecular docking graphs dependent of Tlimit parameter.

From these experiments we see that the prediction of Tlimit is not an easy task and differs between graphs from the same general domain. For the XGB model, we conclude that it does not have sufficient information about the graph to be able to predict a good Tlimit value. For models GCN and GIN, we hypothesize that due to their expressive power (GIN, for example, can distinguish between isomorphic graphs), they are harder to train with relatively small sets and thus perform more poorly than, for example, the GAT model.

4.5. Weighted Molecular Docking Graphs

On a test set of weighted molecular docking graphs, we observed that unlike with the set of unweighted docking graphs, there are only minor speed-ups with the GAT model (Table 7).

Table 7. Times that the MCQD algorithm and the MCQD-ML (variant with the GAT model) algorithm need to find the maximum clique for each graph from a test set of weighted molecular docking graphs.

n	p	MCQD	GAT
345	0.1266	0.0049	0.0048
1779	0.1108	0.1188	0.1122
1851	0.1580	1.0636	1.0563
3233	0.1620	107.3550	106.3990
4211	0.0448	0.4950	0.4960
5293	0.1119	1.1551	1.1550
5309	0.1474	16.5672	16.4841
5735	0.0592	1.3452	1.3416
6294	0.1382	11.4437	10.8346
7211	0.1012	6.8934	6.8882
	Speedup		1.01
	Average speedup		1.01

From these experiments, above we can see that we can speed up the maximum clique search with MCQD by augmenting it with the GAT model. The speed-ups were achieved on random graphs and docking graphs, while on other graph domains we saw very little improvement.

5. Conclusions

We have developed a new approach to find the maximum clique on a protein graph using both neural networks and artificial intelligence approaches. It is a new approach that has not been developed before, and its results show a remarkable speed-up in determining the correct maximum clique on the product graph. Therefore, we expect that this approach will be widely applicable in various scientific fields, such as computer science.

Having fast algorithms that solve maximum clique problem is of great importance in the discovery of new drugs and of protein behavior. We applied a couple of machine learning methods on a regression problem in order to speed up a dynamic algorithm for maximum clique search and obtained several variants of the new MCQD-ML algorithm, which we applied to graph topologies that are particularly important in bioinformatics.

We concluded that improvements using deep learning methods are possible. The most well-suited model that we tested is the graph attention network (GAT), which can speed up the maximum clique search on average by 18% on random graphs and by 34% on docking graphs. The computational cost introduced with the machine learning model is negligible compared to the maximum clique search.

From experiments on protein product graphs, we can assume that further improvements using the same MCQD algorithm are unlikely to be achievable. In further work, we could improve the quality of the set with more samples from different graph topologies such as social network graphs. It would be interesting to test possible speed-ups on other

algorithms that operate on a domain of graphs and use empirically determined parameters that determine the progress of the algorithm.

Author Contributions: Conceptualization, M.G., D.J. and J.K.; methodology, K.R. (Kristjan Reba), M.G., K.R. (Kati Rozman), D.J. and J.K.; software, K.R. (Kristjan Reba), M.G., K.R. (Kati Rozman), D.J. and J.K.; validation, K.R. (Kristjan Reba), M.G. and K.R. (Kati Rozman); formal analysis, M.G., K.R. (Kati Rozman), D.J. and J.K.; investigation, K.R. (Kristjan Reba), M.G., K.R. (Kati Rozman), D.J. and J.K.; resources, M.G., D.J. and J.K.; data curation, K.R. (Kristjan Reba), M.G., K.R. (Kati Rozman), D.J. and J.K.; writing—original draft preparation, K.R. (Kristjan Reba), M.G., D.J. and J.K.; writing—review and editing, K.R. (Kristjan Reba), M.G., K.R. (Kati Rozman), D.J. and J.K.; visualization, K.R. (Kristjan Reba); supervision, J.K., M.G. and D.J.; project administration, J.K., M.G. and D.J.; funding acquisition, J.K. and D.J. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Slovenian Research Agency Project, grants N1-0142, L7-8269, N1-0209, J1-1715, and J1-9186.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data supporting reported results can be found at <http://insilab.org/mcqd-ml> (accessed on 9 November 2021).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Wu, Q.; Hao, J.-K. A review on algorithms for maximum clique problems. *Eur. J. Oper. Res.* **2015**, *242*, 693–709. [CrossRef]
2. Depolli, M.; J Konc, K.R.; Trobec, R.; Janezic, D. Exact parallel maximum clique algorithm for general and protein graphs. *J. Chem. Inf. Model.* **2013**, *53*, 2217–2228. [CrossRef]
3. Butenko, S.; Wilhelm, W.E. Clique-detection models in computational biochemistry and genomics. *Eur. J. Oper. Res.* **2006**, *173*, 1–17. [CrossRef]
4. Konc, J.; Janezic, D. An improved branch and bound algorithm for the maximum clique problem. *Match Commun. Math. Comput. Chem.* **2007**, *58*, 569–590.
5. Prates, M.; Avelar, P.H.; Lemos, H.; Lamb, L.C.; Vardi, M.Y. Learning to solve np-complete problems: A graph neural network for decision tsp. *Proc. AAAI Conf. Artif. Intell.* **2019**, *33*, 4731–4738. [CrossRef]
6. Walteros, J.L.; Buchanan, A. Why is maximum clique often easy in practice? *Oper. Res.* **2020**, *68*, 1625–1931. [CrossRef]
7. Li, C.-M.; Jiang, H.; Manyà, F. On minimization of the number of branches in branch-and-bound algorithms for the maximum clique problem. *Comput. Oper. Res.* **2017**, *84*, 1–15. [CrossRef]
8. Tomita, E.; Seki, T. An efficient branch-and-bound algorithm for finding a maximum clique. In Proceedings of the International Conference on Discrete Mathematics and Theoretical Computer Science, Dijon, France, 7–12 July 2003; Springer: Berlin/Heidelberg, Germany, 2003; pp. 278–289.
9. Tomita, E.; Matsuzaki, S.; Nagao, A.; Ito, H.; Wakatsuki, M. A much faster algorithm for finding a maximum clique with computational experiments. *J. Inf. Process.* **2017**, *25*, 667–677. [CrossRef]
10. Carraghan, R.; Pardalos, P.M. An exact algorithm for the maximum clique problem. *Oper. Res. Lett.* **1990**, *9*, 375–382. [CrossRef]
11. Li, C.-M.; Quan, Z. An efficient branch-and-bound algorithm based on maxsat for the maximum clique problem. *Proc. AAAI Conf. Artif. Intell.* **2010**, *24*, 128–133.
12. Segundo, P.S.; Lopez, A.; Pardalos, P.M. A new exact maximum clique algorithm for large and massive sparse graphs. *Comput. Oper. Res.* **2016**, *66*, 81–94. [CrossRef]
13. Bengio, Y.; Lodi, A.; Prouvost, A. Machine learning for combinatorial optimization: A methodological tour d’horizon. *Eur. J. Oper. Res.* **2021**, *290*, 405–421. [CrossRef]
14. Abe, K.; Xu, Z.; Sato, I.; Sugiyama, M. Solving np-hard problems on graphs with extended AlphaGO Zero. *arXiv* **2019**, arXiv:1905.11623.
15. Zhou, J.; Cui, G.; Zhang, Z.; Yang, C.; Liu, Z.; Wang, L.; Li, C.; Sun, M. Graph neural networks: A review of methods and applications. *arXiv* **2018**, arXiv:1812.08434. [CrossRef]
16. Johnson, D.S.; Trick, M.A. *Cliques, Coloring, and Satisfiability: Second DIMACS Implementation Challenge, 11–13 October 1993*; American Mathematical Society: Boston, MA, USA, 1996.
17. Konc, J.; Janežič, D. ProBiS algorithm for detection of structurally similar protein binding sites by local structural alignment. *Bioinformatics* **2010**, *26*, 1160–1168. [CrossRef] [PubMed]
18. Fine, J.; Konc, J.; Samudrala, R.; Chopra, G. CANDOCK: Chemical atomic network-based hierarchical flexible docking algorithm using generalized statistical potentials. *J. Chem. Inf. Model.* **2020**, *60*, 1509–1527. [CrossRef]

19. Lešnik, S.; Konc, J. In silico laboratory: Tools for similarity-based drug discovery. In *Targeting Enzymes for Pharmaceutical Development*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 1–28.
20. Cortes, C.; Vapnik, V. Support-vector networks. *Mach. Learn.* **1995**, *20*, 273–297. [[CrossRef](#)]
21. Friedman, J.H. Greedy function approximation: A gradient boosting machine. *Ann. Stat.* **2001**, *29*, 1189–1232. [[CrossRef](#)]
22. Chen, T.; Guestrin, C. Xgboost: A scalable tree boosting system. In Proceedings of the 22nd ACM Sigkdd International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13 August 2016; pp. 785–794.
23. Shervashidze, N.; Vishwanathan, S.; Petri, T.; Mehlhorn, K.; Borgwardt, K. Efficient graphlet kernels for large graph comparison. *Artif. Intell. Stat. PMLR* **2009**, *5*, 488–495.
24. Shervashidze, N.; Schweitzer, P.; van Leeuwen, E.J.; Mehlhorn, K.; Borgwardt, K.M. Weisfeiler-lehman graph kernels. *J. Mach. Learn. Res.* **2011**, *12*, 2539–2561.
25. Leskovec, J. Stanford cs224w: Machine Learning with Graphs. Traditional Methods for Machine Learning in Graphs 2019. Available online: <http://web.stanford.edu/class/cs224w/> (accessed on 1 May 2021).
26. Xu, K.; Hu, W.; Leskovec, J.; Jegelka, S. How powerful are graph neural networks? *arXiv* **2018**, arXiv:1810.00826.
27. Khalil, E.; Dai, H.; Zhang, Y.; Dilkina, B.; Song, L. Learning combinatorial optimization algorithms over graphs. *Adv. Neural Inf. Process. Syst.* **2017**, *1704*, 6348–6358.
28. Kipf, T.N.; Welling, M. Semi-supervised classification with graph convolutional networks. *arXiv* **2016**, arXiv:1609.02907.
29. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is all you need. *arXiv* **2017**, arXiv:1706.03762.
30. Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; Bengio, Y. Graph attention networks. *arXiv* **2016**, arXiv:1609.02907.

Article

Entropies via Various Molecular Descriptors of Layer Structure of H_3BO_3

Muhammad Usman Ghani ^{1,†}, Muhammad Kashif Maqbool ^{2,†}, Reny George ^{3,*,†}, Austine Efut Ofem ^{4,†} and Murat Cancan ^{5,†}

¹ Institute of Mathematics, Khawaja Fareed University of Engineering & Information Technology, Abu Dhabi Road, Rahim Yar Khan 64200, Pakistan

² The Government Sadiq Egerton College Bahawalpur, Punjab 63100, Pakistan

³ Department of Mathematics, College of Science and Humanities in Al-Kharj, Prince Sattam Bin Abdulaziz University, Al-Kharj 11942, Saudi Arabia

⁴ School of Mathematics, Statistics and Computer Science, University of KwaZulu-Natal, Durban 4001, South Africa

⁵ Faculty of Education, Yuzuncu Yil University, Van 65080, Turkey

* Correspondence: renygeorge02@yahoo.com

† These authors contributed equally to this work.

Abstract: Entropy is essential. Entropy is a measure of a system's molecular disorder or unpredictability, since work is produced by organized molecular motion. Entropy theory offers a profound understanding of the direction of spontaneous change for many commonplace events. A formal definition of a random graph exists. It deals with relational data's probabilistic and structural properties. The lower-order distribution of an ensemble of attributed graphs may be used to describe the ensemble by considering it to be the results of a random graph. Shannon's entropy metric is applied to represent a random graph's variability. A structural or physicochemical characteristic of a molecule or component of a molecule is known as a molecular descriptor. A mathematical correlation between a chemical's quantitative molecular descriptors and its toxicological endpoint is known as a QSAR model for predictive toxicology. Numerous physicochemical, toxicological, and pharmacological characteristics of chemical substances help to foretell their type and mode of action. Topological indices were developed some 150 years ago as an alternative to the Herculean, and arduous testing is needed to examine these features. This article uses various computational and mathematical techniques to calculate atom–bond connectivity entropy, atom–bond sum connectivity entropy, the newly defined Albertson entropy using the Albertson index, and the IRM entropy using the IRM index. We use the subdivision and line graph of the H_3BO_3 layer structure, which contains one boron atom and three oxygen atoms to form the chemical boric acid.

Keywords: entropies via various molecular descriptors; H_3BO_3 layer structure; subdivision of H_3BO_3 ; line graph of H_3BO_3

MSC: 05C07; 05C09; 05C31; 05C76; 05C99

Citation: Ghani, M.U.; Kashif Maqbool, M.; George, R.; Ofem, A.E.; Cancan, M. Entropies via Various Molecular Descriptors of Layer Structure of H_3BO_3 . *Mathematics* **2022**, *10*, 4831. <https://doi.org/10.3390/math10244831>

Academic Editor: Vasily Novozhilov

Received: 9 November 2022

Accepted: 13 December 2022

Published: 19 December 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Theoretical chemistry and graph theory are combined in chemical graph theory (CGT). It makes a contribution to the modeling of actual and fictitious chemical substances, examines the mathematical structure and connectedness, and then unifies the mathematical and chemical notions [1]. A chemical compound is modeled by displaying its structural formula as a chemical graph, in which atoms are represented by vertices and chemical bonds by edges [2].

We determine a structure's distance-based entropy by using some well-known topological indices, which are the numbers that help characterize its topological features after

it has been reproduced. The many pharmacological, physicochemical (such as melting point, boiling temperature, volume, molecular weight, density, etc.), and toxicological properties of a chemical molecule have a link with these invariants [3–5]. Topological indices have the amazing feature of remaining constant over graph isomorphisms, making them typically graph-invariant [6–14]. Numerous topological indices based on chemical graphs that rely on the number of vertices have been discovered and studied [15–19]. The atom–bond connectivity index and its modified form, the atom–bond sum connectivity index, the Albertson index, and the IRM index, as well as their mathematical equations, are introduced and defined in this section. For more explanation, see [20–27].

The atom–bond connectivity index was established by Estrada et al. [28] and is a modified version of the connectivity index. It is described as

$$ABC(G, x) = \sum_{a_i \sim a_j} x \sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} \times V_{a_j})}} \quad \& \quad ABC = \sum_{a_i \sim a_j} \sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} \times V_{a_j})}} \quad (1)$$

Zhou and Trinajstić [29] proposed the sum-connectivity index, $\sum_{u,v \in \xi_g} \frac{1}{\sqrt{V_{a_i} + V_{a_j}}}$, an alternative to the connectivity index. The atom–bond sum-connectivity (ABS) index is a recently proposed modification of the atom–bond connectivity index that makes use of the fundamental concept of the sum-connectivity index [30]. A definition of the ABS index is

$$ABS(G, x) = \sum_{a_i \sim a_j} x \sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} + V_{a_j})}} \quad \& \quad ABS = \sum_{a_i \sim a_j} \sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} + V_{a_j})}} \quad (2)$$

To determine a graph’s irregularity, the authors in [31] established the Albertson index $A(G)$.

$$A(G, x) = \sum_{a_i \sim a_j} x^{|V_{a_i} - V_{a_j}|} \quad \& \quad A(G) = \sum_{a_i \sim a_j} |V_{a_i} - V_{a_j}| \quad (3)$$

The irregularities of the graph are gauged using the Albertson, Bell, and IRM indices [32]. The definition of IRM(G) is

$$IRM(G, x) = \sum_{a_i \sim a_j} x^{|V_{a_i} - V_{a_j}|^2} \quad \& \quad IRM(G) = \sum_{a_i \sim a_j} [V_{a_i} - V_{a_j}]^2 \quad (4)$$

In this paper, we work with Boric acid H_3BO_3 . It is an acid made up of four oxygen atoms, one phosphorus atom, and three hydrogen atoms. Boric acid is sometimes referred to as orthoboric acid, boracic acid, hydrogen borate, or acidum boricum. It possesses antiviral, antifungal, and antiseptic qualities and is a weak acid. Figure 1 depicts the boric acid complex, which consists of one boron atom, three oxygen atoms, and three hydrogen atoms. The floral pattern structure (base unit) depicted in Figure 1 is created by polymerizing the H_3BO_3 unit structure, which consists of six repeating units of H_3BO_3 .

The degree of unpredictability (or disorder) in a system is measured by entropy. It may also be considered a measurement of how evenly the molecules in the system distribute their energy. The number of alternative configurations of molecule position and the amount of kinetic energy at a specific thermodynamic state is known as a microstate.

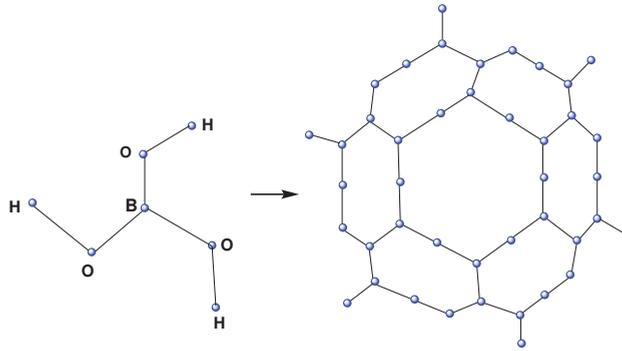


Figure 1. Boric acid H_3BO_3 .

Entropies via Various Molecular Descriptors

Ghani et al. in [33] and Manzoor et al. in [34] recently offered another strategy that is a little bit novel in the literature: applying the idea of Shannon’s entropy [35] in terms of topological indices. The following formula represents the graph entropy:

$$ENT_{\mu(G)} = - \sum_{a_i \sim a_j} \frac{\mu(V_{a_i} V_{a_j})}{\sum_{a_i \sim a_j} \mu(V_{a_i} V_{a_j})} \log \left\{ \frac{\mu(V_{a_i} V_{a_j})}{\sum_{a_i \sim a_j} \mu(V_{a_i} V_{a_j})} \right\}. \tag{5}$$

where a_1, a_2 represents atoms, ξ_G represents the edge set, and $\mu(V_{a_i} V_{a_j})$ represents the edge weight of edge $(V_{a_i} V_{a_j})$.

- **Entropy related to ABC index**

Let $\mu((a_i)(a_j)) = \left\{ \sqrt{\frac{V_{a_i} + V_{a_j} - 2}{V_{a_i} \times V_{a_j}}} \right\}$. Then ABC index (1) is given by

$$ABC_G = \sum_{a_i, a_j \in \xi_G} \left\{ \sqrt{\frac{V_{a_i} + V_{a_j} - 2}{V_{a_i} \times V_{a_j}}} \right\} = \sum_{a_i, a_j \in \xi_G} \mu((a_i)(a_j)).$$

Adding the parameters of ABC_G into Equation (5), then the atom–bond connectivity (ENT_{ABC}) entropy is

$$ENT_{ABC_G} = \log(ABC_G) - \frac{1}{ABC_G} \log \left\{ \prod_{a_i, a_j \in \xi_G} \left(\sqrt{\frac{V_{a_i} + V_{a_j} - 2}{V_{a_i} \times V_{a_j}}} \right) \left(\sqrt{\frac{V_{a_i} + V_{a_j} - 2}{V_{a_i} \times V_{a_j}}} \right) \right\}. \tag{6}$$

- **Entropy related to ABS index**

Let $\mu((a_i)(a_j)) = \left\{ \sqrt{\frac{V_{a_i} + V_{a_j} - 2}{V_{a_i} + V_{a_j}}} \right\}$. Then the ABS index (2) is given by

$$ABS_G = \sum_{a_i, a_j \in \xi_G} \left\{ \sqrt{\frac{V_{a_i} + V_{a_j} - 2}{V_{a_i} + V_{a_j}}} \right\} = \sum_{a_i, a_j \in \xi_G} \mu((a_i)(a_j)). \tag{7}$$

Adding the parameters of ABS_G into Equation (5), then the atom–bond sum connectivity ($ENT_{ABC(G)}$) entropy is

$$ENT_{ABS_G} = \log(ABS_G) - \frac{1}{ABS_G} \log \left\{ \prod_{a_i, a_j \in \xi_G} \left(\sqrt{\frac{V_{a_i} + V_{a_j} - 2}{V_{a_i} + V_{a_j}}} \right) \left(\sqrt{\frac{V_{a_i} + V_{a_j} - 2}{V_{a_i} + V_{a_j}}} \right) \right\}. \tag{8}$$

- **Entropy related to Albertson index**

Let $\mu((a_i)(a_j)) = \{|V_{a_i} - V_{a_j}|\}$. Then the Albertson entropy (3) is given by

$$A_{(G)} = \sum_{a_i, a_j \in \xi_G} \{|V_{a_i} - V_{a_j}|\} = \sum_{a_i, a_j \in \xi_G} \mu((a_i)(a_j)).$$

Adding the parameters of $A_{(G)}$ into Equation (5), then the Albertson (ENT_A) entropy is

$$ENT_{A_{(G)}} = \log(A_{(G)}) - \frac{1}{A_{(G)}} \log \left\{ \prod_{a_i, a_j \in \xi_G} (|V_{a_i} - V_{a_j}|)^{|V_{a_i} - V_{a_j}|} \right\}. \tag{9}$$

- **Entropy related to IRM index**

Let $\mu((a_i)(a_j)) = \{|V_{a_i} - V_{a_j}|^2\}$. Then the IRM entropy (4) is given by

$$IRM_{(G)} = \sum_{a_i, a_j \in \xi_G} \{|V_{a_i} - V_{a_j}|^2\} = \sum_{a_i, a_j \in \xi_G} \mu((a_i)(a_j)).$$

Adding the parameters of $IRM_{(G)}$ into Equation (5), then the IRM (ENT_{IRM}) entropy is

$$ENT_{IRM_{(G,x)}} = \log IRM_{(G)} - \frac{1}{IRM_{(G,x)}} \log \left\{ \prod_{a_i, a_j \in \xi_G} (|V_{a_i} - V_{a_j}|^2)^{|V_{a_i} - V_{a_j}|^2} \right\}. \tag{10}$$

2. Layer Structure of $H_3BO_3(s, t)$

In this section, we discuss the $H_3BO_3(s, t)$ layer structure, which serves as the foundation for its subdivision and line graph. The $H_3BO_3(s, t)$ unit structure polymerizes to generate the floral pattern structure (base unit) seen in Figure 2, which is made up of six repeating H_3BO_3 units. This layer structure may be stretched to whatever number of rows and columns is desired. The horizontal lines of floral pattern structures are characterized as rows “s”, while the vertical lines are designated as columns “t”. Figure 2 depicts $H_3BO_3(s, t)$ with one row and two columns, $s = 1$ and $t = 2$.

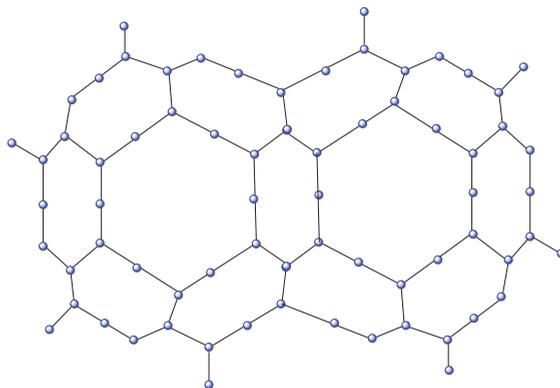


Figure 2. Layer structure of H_3BO_3 .

2.1. Subdivision of the Layer Structure $H_3BO_3(s, t)$

Figure 3 shows the subdivision of $H_3BO_3(s, t)$, the layer structure achieved by installing one atom between each atom–bond of Figure 2.

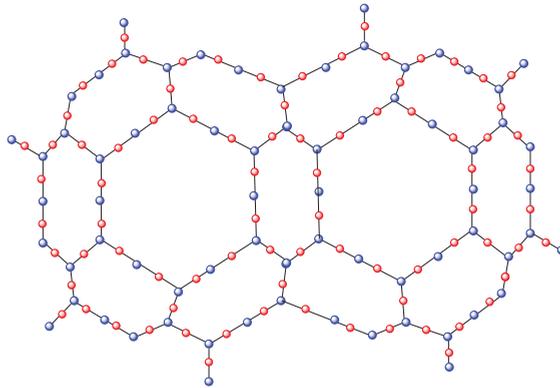


Figure 3. Subdivision of H_3BO_3 .

Result and Discussion

In subdivision of the layer structure $H_3BO_3(s, t)$, the atom–bond $E(G)$ is divided into three groups based on the degree of each edge’s end vertices. The set that is disjointed is shown by the symbols $\zeta_{(d(u_i), d(v_j))}$. The first set that is disjointed is $\zeta_{(1,2)}$, the second set that is disjointed is $\zeta_{(2,2)}$, and the third set that is disjointed is $\zeta_{(2,3)}$. The table below describes the different types of edges as well as the equations for calculating the number of edges in each type of the $SH_3BO_3(s, t)$ layer structure.

- **Entropy related to the ABC index of subdivision H_3BO_3**

Let $S(H_3BO_3)$ be a subdivision of $H_3BO_3(s, t)$. Then by using Equation (1) and Table 1, the atom–bond connectivity index is

$$\begin{aligned}
 ABC(S(H_3BO_3)) &= \sum_{\zeta_{(1\sim 2)}} x^{\sqrt{\frac{1+2-2}{1 \times 2}}} + \sum_{\zeta_{(2\sim 2)}} x^{\sqrt{\frac{2+2-2}{2 \times 2}}} + \sum_{\zeta_{(2\sim 3)}} x^{\sqrt{\frac{2+3-2}{2 \times 3}}} \\
 &= 2(s + t + 1)x^{\sqrt{\frac{1}{2}}} + 12(st + s + t)x^{\sqrt{\frac{1}{2}}} \\
 &\quad + 6(3s + 3t + 4st - 1)x^{\sqrt{\frac{1}{2}}}
 \end{aligned}
 \tag{11}$$

Differentiate (11) at $x = 1$; we get the atom–bond connectivity index

$$ABCS(H_3BO_3) = \sqrt{\frac{1}{2}}(32s + 32t + 36st - 4)
 \tag{12}$$

Here, we determine the atom–bond connectivity entropy by using Table 1 and Equation (12) in Equation (6) according to the following:

Table 1. Edge division based on vertices in the layer structure of subdivision $H_3BO_3(s, t)$.

Atomic bond type	$\zeta_{(1,2)}$	$\zeta_{2\sim 2}$	$\zeta_{2\sim 3}$
Number of atom bonds	$2(s + t + 1)$	$12(st + s + t)$	$6(3s + 3t + 4st - 1)$

$$\begin{aligned}
 ENT_{ABC}S(H_3BO_3) &= \log(ABC) - \frac{1}{ABC} \log \left\{ \prod_{\xi(1,2)} \left[\sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} \times V_{a_j})}} \right]^{\left\lfloor \sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} \times V_{a_j})}} \right\rfloor} \right. \\
 &\quad \times \left. \prod_{\xi(2,2)} \left[\sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} \times V_{a_j})}} \right]^{\left\lfloor \sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} \times V_{a_j})}} \right\rfloor} \times \prod_{\xi(2,3)} \left[\sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} \times V_{a_j})}} \right]^{\left\lfloor \sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} \times V_{a_j})}} \right\rfloor} \right\} \\
 &= \log \left(\sqrt{\frac{1}{2}} (32s + 32t + 36st - 4) \right) - \frac{1}{\sqrt{\frac{1}{2}} (32s + 32t + 36st - 4)} \\
 &\quad \times \log \left\{ 2(s + t + 1) \left(\sqrt{\frac{1}{2}} \right)^{\sqrt{\frac{1}{2}}} \times 12(st + s + t) \left(\sqrt{\frac{1}{2}} \right)^{\sqrt{\frac{1}{2}}} \right. \\
 &\quad \times \left. 6(3s + 3t + 4st - 1) \left(\sqrt{\frac{1}{2}} \right)^{\sqrt{\frac{1}{2}}} \right\}. \tag{13}
 \end{aligned}$$

• Entropy related to the ABS index of subdivision H_3BO_3

Let $S(H_3BO_3)$ be a subdivision of $H_3BO_3(s, t)$. Then by using Equation (2) and Table 1, the atom–bond sum connectivity is

$$\begin{aligned}
 ABSS(H_3BO_3) &= \sum_{\xi(1\sim 2)} x^{\sqrt{\frac{1+2-2}{1+2}}} + \sum_{\xi(2\sim 2)} x^{\sqrt{\frac{2+2-2}{2+2}}} + \sum_{\xi(2\sim 3)} x^{\sqrt{\frac{2+3-2}{2+3}}} \\
 &= 2(s + t + 1)x^{\sqrt{\frac{1}{3}}} + 12(st + s + t)x^{\sqrt{\frac{1}{2}}} \\
 &\quad + 6(3s + 3t + 4st - 1)x^{\sqrt{\frac{3}{5}}} \tag{14}
 \end{aligned}$$

Taking the first derivative of Equation (14) at $x = 1$, we get the atom–bond sum connectivity index

$$ABS(S(H_3BO_3)) = 2(s + t + 1)\sqrt{\frac{1}{3}} + 12(st + s + t)\sqrt{\frac{1}{2}} + 6(3s + 3t + 4st - 1)\sqrt{\frac{3}{5}}. \tag{15}$$

Here, we determine the atom–bond sum connectivity entropy by using Table 1 and Equation (15) in Equation (6) according to the following:

$$\begin{aligned}
 ENT_{ABS}S(S(H_3BO_3)) &= \log(ABS) - \frac{1}{ABS} \log \left\{ \prod_{\xi(1,2)} \left[\sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} + V_{a_j})}} \right]^{\left\lfloor \sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} + V_{a_j})}} \right\rfloor} \right. \\
 &\quad \times \prod_{\xi(2,2)} \left[\sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} + V_{a_j})}} \right]^{\left\lfloor \sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} + V_{a_j})}} \right\rfloor} \times \prod_{\xi(2,3)} \left[\sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} + V_{a_j})}} \right]^{\left\lfloor \sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} + V_{a_j})}} \right\rfloor} \left. \right\} \\
 &= \log(ABS) - \frac{1}{ABS} \log \left\{ 2(s + t + 1) \left(\sqrt{\frac{1}{3}} \right)^{\sqrt{\frac{1}{3}}} \times 12(st + s + t) \left(\sqrt{\frac{1}{2}} \right)^{\sqrt{\frac{1}{2}}} \right. \\
 &\quad \times \left. 6(3s + 3t + 4st - 1) \left(\sqrt{\frac{3}{5}} \right)^{\sqrt{\frac{3}{5}}} \right\}. \tag{16}
 \end{aligned}$$

• Entropy related to the Albertson index $S(H_3BO_3)$

Let $S(H_3BO_3)$ be a subdivision of $H_3BO_3(s, t)$. Then by using Equation (3) and Table 1, the atom–bond connectivity index is

$$\begin{aligned} A_{(G,x)}(S(H_3BO_3)) &= \sum_{\xi_{(1\sim 2)}} x^{|1-2|} + \sum_{\xi_{(2\sim 2)}} x^{|2-2|} + \sum_{\xi_{(2\sim 3)}} x^{|2-3|} \\ &= 2(s + t + 1)x + 12(st + s + t) + 6(3s + 3t + 4st - 1)x \end{aligned} \tag{17}$$

Differentiate (17) at $x = 1$; we get the atom–bond connectivity index

$$A_{(G,x)}S(H_3BO_3) = 32s + 32t + 36st - 4 \tag{18}$$

Here, we determine the atom–bond connectivity entropy by using Table 1 and Equation (18) in Equation (9) according to the following:

$$\begin{aligned} ENT_{A_{(G,x)}}S(H_3BO_3) &= \log(A_{(G,x)}) - \frac{1}{A_{(G,x)}} \log \left\{ \prod_{\xi_{(1,2)}} [|V_{a_i} - V_{a_j}|]^{|V_{a_i} - V_{a_j}|} \right. \\ &\quad \times \prod_{\xi_{(2,2)}} [|V_{a_i} - V_{a_j}|]^{|V_{a_i} - V_{a_j}|} \times \left. \prod_{\xi_{(2,3)}} [|V_{a_i} - V_{a_j}|]^{|V_{a_i} - V_{a_j}|} \right\} \\ &= \log(32s + 32t + 36st - 4) - \frac{1}{32s + 32t + 36st - 4} \log \left\{ 2(s + t + 1) \right. \\ &\quad \left. + 12(st + s + t) + 6(3s + 3t + 4st - 1) \right\}. \end{aligned} \tag{19}$$

• **Entropy related to the IRM index of subdivision H_3BO_3**

Let $S(H_3BO_3)$ be a subdivision of $H_3BO_3(s, t)$. Then by using Equation (4) and Table 1, the atom–bond connectivity index is

$$\begin{aligned} IRM_{(G,x)}(S(H_3BO_3)) &= \sum_{\xi_{(1\sim 2)}} x^{|1-2|^2} + \sum_{\xi_{(2\sim 2)}} x^{|2-2|^2} + \sum_{\xi_{(2\sim 3)}} x^{|2-3|^2} \\ &= 2(s + t + 1)x + 12(st + s + t) \\ &\quad + 6(3s + 3t + 4st - 1)x \end{aligned} \tag{20}$$

Differentiate (20) at $x = 1$; we get the atom–bond connectivity index

$$IRM_{(G,x)}S(H_3BO_3) = 32s + 32t + 36st - 4 \tag{21}$$

Here, we determine the atom–bond connectivity entropy by using Table 1 and Equation (21) in Equation (10) according to the following:

$$\begin{aligned} ENT_{IRM_{(G,x)}}S(H_3BO_3) &= \log(IRM_{(G,x)}) - \frac{1}{IRM_{(G,x)}} \log \left\{ \prod_{\xi_{(1,2)}} [|V_{a_i} - V_{a_j}|^2]^{|V_{a_i} - V_{a_j}|^2} \right. \\ &\quad \times \prod_{\xi_{(2,2)}} [|V_{a_i} - V_{a_j}|^2]^{|V_{a_i} - V_{a_j}|^2} \times \left. \prod_{\xi_{(2,3)}} [|V_{a_i} - V_{a_j}|^2]^{|V_{a_i} - V_{a_j}|^2} \right\} \\ &= \log(32s + 32t + 36st - 4) - \frac{1}{32s + 32t + 36st - 4} \log \left\{ 2(s + t + 1) \right. \\ &\quad \left. + 12(st + s + t) + 6(3s + 3t + 4st - 1) \right\}. \end{aligned} \tag{22}$$

2.2. Layer Structure of H_3BO_3 in the Form of a Line Graph

In the line graph of the layer structure $H_3BO_3(s, t)$, the atom–bond $E(G)$ is divided into five groups based on the degree of each edge’s end vertices. The set that is disjointed is shown by the symbols $\xi_{(d(u_i), d(v_j))}$. The first set that is disjointed is $\xi_{(2,3)}$, the second set that is disjoint is $\xi_{(2,4)}$, the third set that is disjointed is $\xi_{(3,3)}$, the fourth set that is disjointed is $\xi_{(3,4)}$, and the fifth set that is disjointed is $\xi_{(4,4)}$.

Figure 4 displays the $H_3BO_3(s, t)$ layer structure as a line graph.

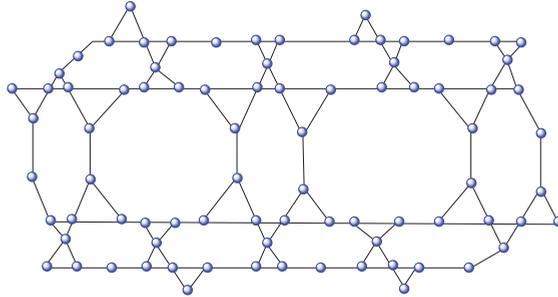


Figure 4. Line graph of H_3BO_3 .

• Entropy related to the ABC index of $L(H_3BO_3)$

Let $L(H_3BO_3)$ be a line graph of $H_3BO_3(s, t)$. Then by using Equation (1) and Table 2, the ABC polynomial is

$$\begin{aligned}
 ABCL(H_3BO_3) &= \sum_{\xi(2\sim 3)} x\sqrt{\frac{2+3-2}{2\times 3}} + \sum_{\xi(2\sim 4)} x\sqrt{\frac{2+4-2}{2\times 4}} + \sum_{\xi(3\sim 3)} x\sqrt{\frac{3+3-2}{3\times 3}} + \sum_{\xi(3\sim 4)} x\sqrt{\frac{3+4-2}{3\times 4}} + \sum_{\xi(4\sim 4)} x\sqrt{\frac{4+4-2}{4\times 4}} \\
 &= 6(1+t+s)x\sqrt{\frac{1}{2}} + 2(s+t+1)x^{\frac{1}{2}} + 4(s+t+3st-2)x^{\frac{2}{3}} \\
 &+ 2(5s+5t+6st-1)x\sqrt{\frac{5}{12}} + 2(s+t+3st-2)x\sqrt{\frac{3}{8}}.
 \end{aligned}
 \tag{23}$$

Taking the first derivative of Equation (23) at $x = 1$, we get the ABC index

$$\begin{aligned}
 ABCL(H_3BO_3) &= 6(1+t+s)\sqrt{\frac{1}{2}} + \frac{2}{3}(24st+11s+11t-5) + 2(5s+5t+6st-1)\sqrt{\frac{5}{12}} \\
 &+ 2(s+t+3st-2)\sqrt{\frac{3}{8}}
 \end{aligned}
 \tag{24}$$

Here, we determine the ABC entropy by using Table 2 and Equation (24) in Equation (6) according to the following:

$$\begin{aligned}
 ENT &= \log(ABC) - \frac{1}{ABC} \log \left\{ \prod_{\xi(2,3)} \left[\sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} \times V_{a_j})}} \right]^{\left[\sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} \times V_{a_j})}} \right]} \right. \\
 &\times \prod_{\xi(2,4)} \left[\sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} \times V_{a_j})}} \right]^{\left[\sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} \times V_{a_j})}} \right]} \times \prod_{\xi(3,3)} \left[\sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} \times V_{a_j})}} \right]^{\left[\sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} \times V_{a_j})}} \right]} \\
 &\times \prod_{\xi(3,4)} \left[\sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} \times V_{a_j})}} \right]^{\left[\sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} \times V_{a_j})}} \right]} \times \prod_{\xi(4,4)} \left[\sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} \times V_{a_j})}} \right]^{\left[\sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} \times V_{a_j})}} \right]} \left. \right\} \\
 &= \log(ABC) - \frac{1}{ABS} \log \left\{ 6(1+t+s)\left(\sqrt{\frac{1}{2}}\right)^{\frac{1}{\sqrt{2}}} + 2(s+t+1)\left(\frac{1}{2}\right)^{\frac{1}{2}} \right. \\
 &+ 4(s+t+3st-2)\left(\sqrt{\frac{2}{3}}\right)\sqrt{\frac{3}{8}} + 2(5s+5t+6st-1)\left(\sqrt{\frac{5}{12}}\right)\sqrt{\frac{5}{12}} \\
 &+ 2(s+t+3st-2)\left(\sqrt{\frac{3}{8}}\right)\sqrt{\frac{3}{8}} \left. \right\}.
 \end{aligned}
 \tag{25}$$

Table 2. Edge division based on vertices in the line graph $H_3BO_3(s, t)$ layer structure.

Atomic bonds	$\xi_{2\sim 3}$	$\xi_{2\sim 4}$	$\xi_{3\sim 3}$	$\xi_{3\sim 4}$	$\xi_{4\sim 4}$
Cardinality	$6(1 + t + s)$	$2(s + t + 1)$	$4(s + t + 3st - 2)$	$2(5s + 5t + 6st - 1)$	$2(s + t + 3st - 2)$

• **Entropy related to the ABS index of $L(H_3BO_3)$**

Let $L(H_3BO_3)$ be a line graph of $H_3BO_3(s, t)$. Then by using Equation (2) and Table 2, the ABS polynomial is

$$\begin{aligned}
 ABSL(H_3BO_3) &= \sum_{\xi_{(2\sim 3)}} x\sqrt{\frac{2+3-2}{2+3}} + \sum_{\xi_{(2\sim 4)}} x\sqrt{\frac{2+4-2}{2+4}} + \sum_{\xi_{(3\sim 3)}} x\sqrt{\frac{3+3-2}{3+3}} + \sum_{\xi_{(3\sim 4)}} x\sqrt{\frac{3+4-2}{3+4}} + \sum_{\xi_{(4\sim 4)}} x\sqrt{\frac{4+4-2}{4+4}} \\
 &= 6(1 + t + s)x\sqrt{\frac{3}{5}} + 2(s + t + 1)x\sqrt{\frac{2}{3}} + 4(s + t + 3st - 2)x\sqrt{\frac{2}{3}} \\
 &+ 2(5s + 5t + 6st - 1)x\sqrt{\frac{5}{7}} + 2(s + t + 3st - 2)x\sqrt{\frac{3}{4}}.
 \end{aligned} \tag{26}$$

Taking the first derivative of Equation (26) at $x = 1$, we get the ABS index

$$\begin{aligned}
 ABS(L(H_3BO_3)) &= 6(1 + t + s)\sqrt{\frac{3}{5}} + 2(6st + 3s + 3t - 3)\sqrt{\frac{2}{3}} + 2(5s + 5t + 6st - 1)\sqrt{\frac{5}{7}} \\
 &+ 2(s + t + 3st - 2)\sqrt{\frac{3}{4}}.
 \end{aligned} \tag{27}$$

Here, we determine the ABS entropy by using Table 2 and Equation (27) in Equation (6) according to the following:

$$\begin{aligned}
 ENT_{ABS}(L(H_3BO_3)) &= \log(ABS) - \frac{1}{ABS} \log \left\{ \prod_{\xi_{(2,3)}} \left[\sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} + V_{a_j})}} \right]^{\left[\sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} + V_{a_j})}} \right]} \right. \\
 &\times \prod_{\xi_{(2,4)}} \left[\sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} + V_{a_j})}} \right]^{\left[\sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} + V_{a_j})}} \right]} \times \prod_{\xi_{(3,3)}} \left[\sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} + V_{a_j})}} \right]^{\left[\sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} + V_{a_j})}} \right]} \\
 &\times \prod_{\xi_{(3,4)}} \left[\sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} + V_{a_j})}} \right]^{\left[\sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} + V_{a_j})}} \right]} \times \prod_{\xi_{(4,4)}} \left[\sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} + V_{a_j})}} \right]^{\left[\sqrt{\frac{(V_{a_i} + V_{a_j} - 2)}{(V_{a_i} + V_{a_j})}} \right]} \left. \right\} \\
 &= \log(ABS) - \frac{1}{ABS} \log \left\{ 6(1 + t + s)\left(\sqrt{\frac{3}{5}}\right)^{\sqrt{\frac{3}{5}}} + 2(s + t + 1)\left(\sqrt{\frac{2}{3}}\right)^{\sqrt{\frac{2}{3}}} \right. \\
 &+ 4(s + t + 3st - 2)\left(\sqrt{\frac{2}{3}}\right)^{\sqrt{\frac{2}{3}}} + 2(5s + 5t + 6st - 1)\left(\sqrt{\frac{5}{7}}\right)^{\sqrt{\frac{5}{7}}} \\
 &+ 2(s + t + 3st - 2)\left(\sqrt{\frac{3}{4}}\right)^{\sqrt{\frac{3}{4}}} \left. \right\}.
 \end{aligned} \tag{28}$$

• **Entropy related to the Albertson index of $L(H_3BO_3)$**

Let $L(H_3BO_3)$ be a line graph of $H_3BO_3(s, t)$. Then by using Equation (3) and Table 2, the Albertson index is

$$\begin{aligned}
 A_{(G,x)}L(H_3BO_3) &= \sum_{\xi_{(2\sim 3)}} x^{|2-3|} + \sum_{\xi_{(2\sim 4)}} x^{|2-4|} + \sum_{\xi_{(3\sim 3)}} x^{|3-3|} + \sum_{\xi_{(3\sim 4)}} x^{|3-4|} + \sum_{\xi_{(4\sim 4)}} x^{|4-4|} \\
 &= 6(1 + t + s)x + 2(s + t + 1)x^2 + 4(s + t + 3st - 2) \\
 &+ 2(5s + 5t + 6st - 1)x + 2(s + t + 3st - 2).
 \end{aligned} \tag{29}$$

Taking the first derivative of Equation (29) at $x = 1$, we get the Albertson index

$$A_{(G,x)}(L(H_3BO_3)) = 2(15st + 13s + 13t - 2) \tag{30}$$

Here, we determine the A entropy by using Table 2 and Equation (30) in Equation (9) according to the following:

$$\begin{aligned} ENT_{A_{(G,x)}}(L(H_3BO_3)) &= \log(A) - \frac{1}{A} \log \left\{ \prod_{\xi_{(2,3)}} [|V_{a_i} - V_{a_j}|]^{[|V_{a_i} - V_{a_j}|]} \right. \\ &\times \prod_{\xi_{(2,4)}} [|V_{a_i} - V_{a_j}|]^{[|V_{a_i} - V_{a_j}|]} \times \prod_{\xi_{(3,3)}} [|V_{a_i} - V_{a_j}|]^{[|V_{a_i} - V_{a_j}|]} \\ &\times \left. \prod_{\xi_{(3,4)}} [|V_{a_i} - V_{a_j}|]^{[|V_{a_i} - V_{a_j}|]} \times \prod_{\xi_{(4,4)}} [|V_{a_i} - V_{a_j}|]^{[|V_{a_i} - V_{a_j}|]} \right\} \\ &= \log 2(15st + 13s + 13t - 2) - \frac{1}{2(15st + 13s + 13t - 2)} \log \left\{ 6(1 + t + s) \right. \\ &+ 4(s + t + 1) + 4(s + t + 3st - 2) + 2(5s + 5t + 6st - 1) \\ &+ \left. 2(s + t + 3st - 2) \right\}. \end{aligned} \tag{31}$$

- **Entropy related to the IRM index of $L(H_3BO_3)$**

Let $L(H_3BO_3)$ be a line graph of $H_3BO_3(s, t)$. Then by using Equation (4) and Table 2, the IRM index is

$$\begin{aligned} IRM_{(G,x)}L(H_3BO_3) &= \sum_{\xi_{(2\sim3)}} x^{[2-3]^2} + \sum_{\xi_{(2\sim4)}} x^{[2-4]^2} + \sum_{\xi_{(3\sim3)}} x^{[3-3]^2} + \sum_{\xi_{(3\sim4)}} x^{[3-4]^2} + \sum_{\xi_{(4\sim4)}} x^{[4-4]^2} \\ &= 6(1 + t + s)x + 2(s + t + 1)x^4 + 4(s + t + 3st - 2) \\ &+ 2(5s + 5t + 6st - 1)x + 2(s + t + 3st - 2). \end{aligned} \tag{32}$$

Taking the first derivative of Equation (32) at $x = 1$, we get the IRM index

$$IRM_{(G,x)}(L(H_3BO_3)) = 30s + 30t + 30st. \tag{33}$$

Here, we determine the IRM entropy by using Table 2 and Equation (33) in Equation (10) according to the following:

$$\begin{aligned} ENT_{IRM}(L(H_3BO_3)) &= \log(IRM) - \frac{1}{IRM} \log \left\{ \prod_{\xi_{(2,3)}} [|V_{a_i} - V_{a_j}|^2]^{[|V_{a_i} - V_{a_j}|^2]} \right. \\ &\times \prod_{\xi_{(2,4)}} [|V_{a_i} - V_{a_j}|^2]^{[|V_{a_i} - V_{a_j}|^2]} \times \prod_{\xi_{(3,3)}} [|V_{a_i} - V_{a_j}|^2]^{[|V_{a_i} - V_{a_j}|^2]} \\ &\times \left. \prod_{\xi_{(3,4)}} [|V_{a_i} - V_{a_j}|^2]^{[|V_{a_i} - V_{a_j}|^2]} \times \prod_{\xi_{(4,4)}} [|V_{a_i} - V_{a_j}|^2]^{[|V_{a_i} - V_{a_j}|^2]} \right\} \\ &= \log(30s + 30t + 30st) - \frac{1}{30s + 30t + 30st} \log \left\{ 6(1 + t + s) + 8(s + t + 1) \right. \\ &+ \left. 4(s + t + 3st - 2) + 2(5s + 5t + 6st - 1) + 2(s + t + 3st - 2) \right\}. \end{aligned} \tag{34}$$

3. Comparison and Conclusions

Here, molecular descriptors for the subdivision and line graph of the layer structure of H_3BO_3 that are multiplicative and degree-based have been studied. Using these molecular descriptors, we compute the ABC entropy, ABS entropy, A entropy, and IRM entropy of the subdivision and line graph of the layer structure of H_3BO_3 . Our results (entropies) help to describe the randomness and disorder of a molecule of H_3BO_3 based on the number

of different arrangements available to it in a given system or reaction. For instance, the atom–bond connectivity (ABC) index offers excellent calculations of the strain energy of molecules via correlation. When the temperatures of the production of alkanes are described using the ABC-index, a good quantitative structure–property relationship (QSPR) model ($r = 0.9970$) is produced.

The values of four degree-based indices, namely, the ABC-index, ABS-index, A-index, and IRM-index, are presented in this work, numerically in Table 3 and graphically in Figure 5. As shown in Figure 5, the values of all indices are directly proportional to the values of (s, t) , with the values of (s, t) along the x-axes and the resultant of the indices along the y-axes. The disparities between each topological index for a certain structure are revealed by these charts. The results of the computations demonstrate that the degree-based indices and entropy estimates depend greatly on the values of s and t or the molecular structure.

Table 3. Numerical comparison of molecular descriptors.

Values of (s, t)	ABC-Index	ABS-Index	Albertson Index	IRM-Index
(1, 2)	115.948	120.98	164	164
(2, 3)	263.004	275.844	372	372
(3, 4)	460.964	484.636	652	652
(4, 5)	709.828	747.356	1004	1004
(5, 6)	1009.596	1064.004	1428	1428
(6, 7)	1360.268	1434.58	1924	1924
(7, 8)	1761.844	1859.084	2492	2492
(8, 9)	2214.324	2337.516	3132	3132
(9, 10)	2717.708	2869.876	3844	3844
(10, 11)	3271.996	3456.164	4628	4628
(11, 12)	3877.188	4096.38	5484	5484
(12, 13)	4533.284	4790.524	6412	6412

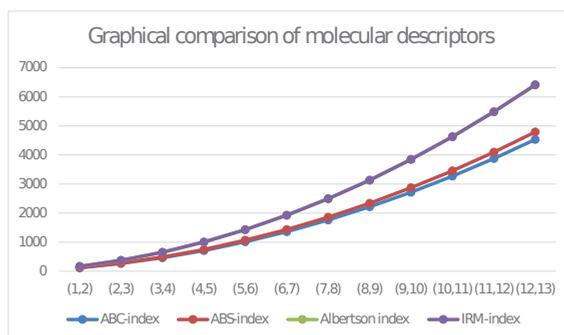


Figure 5. Graphical Comparison of ABC-index, ABS-index, Albertson index and IRM-index.

Author Contributions: Conceptualization, M.U.G.; Software, A.E.O.; Investigation, M.U.G.; Resources, M.C.; Writing—original draft, M.U.G.; Project administration, M.K.M.; Funding acquisition, R.G. These authors contributed equally to this work. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No data were used to support this study.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Zhang, Y.F.; Ghani, M.U.; Sultan, F.; Inc, M.; Cancan, M. Connecting SiO₄ in Silicate and Silicate Chain Networks to Compute Kulli Temperature Indices. *Molecules* **2022**, *27*, 7533. [[CrossRef](#)] [[PubMed](#)]
- Chu, Y.M.; Khan, A.R.; Ghani, M.U.; Ghaffar, A.; Inc, M. Computation of Zagreb Polynomials and Zagreb Indices for Benzenoid Triangular & Hourglass System. *Polycycl. Aromat. Compd.* **2022**, 1–10.
- De Bellis, J.; Bellucci, L.; Bottaro, G.; Labella, L.; Marchetti, F.; Samaritani, S.; Dell'Amico, D.B.; Armelao, L. Single-crystal-to-single-crystal post-synthetic modifications of three-dimensional LOFs (Ln = Gd, Eu): A way to modulate their luminescence and thermometric properties. *Dalton Trans.* **2020**, *49*, 6030–6042. [[CrossRef](#)]
- Alam, A.; Ghani, M.U.; Kamran, M.; Shazib Hameed, M.; Hussain Khan, R.; Baig, A.Q. Degree-Based Entropy for a Non-Kekulean Benzenoid Graph. *J. Math.* **2022**.
- Mikherdov, A.S.; Novikov, A.S.; Kinzhalov, M.A.; Zolotarev, A.A.; Boyarskiy, V.P. Intra-/intermolecular bifurcated chalcogen bonding in crystal structure of thiazole/thiadiazole derived binuclear (diaminocarbene) PdII complexes. *Crystals* **2018**, *8*, 112. [[CrossRef](#)]
- Ranjini, P.S.; Loksha, V.; Usha, A. Relation between phenylene and hexagonal squeeze using harmonic index. *Int. J. Graph Theory* **2013**, *1*, 116–121.
- Saeed, N.; Long, K.; Mufti, Z.S.; Sajid, H.; Rehman, A. Degree-based topological indices of boron b₆(3s + 3t + 4st – 1). *J. Chem.* **2021**. [[CrossRef](#)]
- Anjum, M.S.; Safdar, M.U. K Banhatti and K hyper-Banhatti indices of nanotubes. *Eng. Appl. Sci. Lett.* **2019**, *2*, 19–37. [[CrossRef](#)]
- Khan, A.R.; Ghani, M.U.; Ghaffar, A.; Asif, H.M. Characterization of temperature indices of silicates. *Res. Sq.* **2022**. [[CrossRef](#)]
- Kulli, V.R.; Chaluvaraju, B.; Boregowda, H.S. Connectivity Banhatti indices for certain families of benzenoid systems. *J. Ultra Chem.* **2017**, *13*, 81–87. [[CrossRef](#)]
- Kulli, V.R. On K Banhatti indices of graphs. *J. Comput. Math. Sci.* **2016**, *7*, 213–218.
- Ghani, M.U.; Sultan, F.; El Sayed, M.; Cancan, M.; Ali, S. SiO₄ characterization in a chain and C₆ H₆ embedded in a Non-kekulean structure for Kulli Temperature indices. *Res. Sq.* **2022**. [[CrossRef](#)]
- Kulli, V.R. On multiplicative K Banhatti and multiplicative K hyper-Banhatti indices of Vg Phenylene nanotubes and nanotorus. *Ann. Pure Appl. Math.* **2016**, *11*, 145–150.
- Munir, M.; Nazeer, W.; Rafique, S.; Kang, S.M. M-polynomial and degree-based topological indices of polyhex nanotubes. *Symmetry* **2016**, *8*, 149. [[CrossRef](#)]
- Mahmood, M.K.; Ali, S. A novel labeling algorithm on several classes of graphs. *Punjab Univ. J. Math.* **2017**, *49*, 23–35.
- Mahmood, M.K.; Ali, S. On super totient numbers, with applications and algorithms to graph labeling. *Ars Comb.* **2019**, *143*, 29–37.
- Ali, S.; Mahmood, M.K.; Shum, K.P. Novel classes of integers and their applications in graph labeling. *Hacetatepe J. Math. Stat* **2021**, *1*, 1–17. [[CrossRef](#)]
- Ali, S.; Mahmood, M.K.; Falcón Ganfornina, R.M. A paradigmatic approach to investigate restricted hyper totient graphs. *AIMS Math.* **2021**, *6*, 3761–3771. [[CrossRef](#)]
- Adonin, S.A.; Usoltsev, A.N.; Novikov, A.S.; Kolesov, B.A.; Fedin, V.P.; Sokolov, M.N. One-and two-dimensional iodine-rich iodobismuthate (III) complexes: Structure, optical properties, and features of halogen bonding in the solid state. *Inorg. Chem.* **2020**, *59*, 3290–3296. [[CrossRef](#)]
- Sakander, H. Computing distance-based topological descriptors of complex chemical networks: New theoretical techniques. *Chem. Phys. Lett.* **2017**, *688*, 51–58.
- Hu, M.; Ali, H.; Binyamin, M.A.; Ali, B.; Liu, J.B.; Fan, C. On distance-based topological descriptors of chemical interconnection networks. *J. Math.* **2021**.
- Munir, M.; Nazeer, W.; Nizami, A.R.; Rafique, S.; Kang, S.M. M-polynomials and topological indices of titania nanotubes. *Symmetry* **2016**, *8*, 117. [[CrossRef](#)]
- Bashir, Y.; Aslam, A.; Kamran, M.; Qureshi, M.I.; Jahangir, A.; Rafiq, M.; Bibi, N.; Muhammad, N. On forgotten topological indices of some dendrimers structure. *Molecules* **2017**, *22*, 867. [[CrossRef](#)] [[PubMed](#)]
- Husin, M.N.; Hasni, R.; Arif, N.E.; Imran, M. On topological indices of certain families of nanostar dendrimers. *Molecules* **2016**, *21*, 821. [[CrossRef](#)] [[PubMed](#)]
- Nandini, G.K.; Rajan, R.S.; Shantrinal, A.A.; Rajalaxmi, T.M.; Rajasingh, I.; Balasubramanian, K. Topological and thermodynamic entropy measures for COVID-19 pandemic through graph theory. *Symmetry* **2020**, *12*, 1992. [[CrossRef](#)]
- Deng, H.; Yang, J.; Xia, F. A general modeling of some vertex-degree based topological indices in benzenoid systems and phenylenes. *Comput. Math. Appl.* **2011**, *61*, 3017–3023. [[CrossRef](#)]
- Li, Y.; Zhou, Z.; Shen, P.; Chen, Z. Two-dimensional polyphenylene: Experimentally available porous graphene as a hydrogen purification membrane. *Chem. Commun.* **2010**, *46*, 3672–3674. [[CrossRef](#)]
- Estrada, E.; Torres, L.; Rodriguez, L.; Gutman, I. An atom-bond connectivity index: Modelling the enthalpy of formation of alkanes. *Indian J. Chem. A* **1998**, *37*, 849–855.
- Zhou, B.; Trinajstić, N. On a novel connectivity index. *J. Math. Chem.* **2009**, *46*, 1252–1270. [[CrossRef](#)]
- Ali, A.; Furtula, B.; Redžepović, I.; Gutman, I. Atom-bond sum-connectivity index. *J. Math. Chem.* **2022**, *60*, 2081–2093. [[CrossRef](#)]
- Albertson, I.; Michael, O. The irregularity of a graph. *Ars Comb.* **1997**, *46*, 219–225.

32. Hussain, Z.; Munir, M.; Rafique, S.; Min Kang, S. Topological characterizations and index-analysis of new degree-based descriptors of honeycomb networks. *Symmetry* **2018**, *10*, 478. [[CrossRef](#)]
33. Ghani, M.U.; Sultan, F.; Tag El Din, E.S.M.; Khan, A.R.; Liu, J.B.; Cancan, M. A Paradigmatic Approach to Find the Valency-Based K-Banhatti and Redefined Zagreb Entropy for Niobium Oxide and a Metal–Organic Framework. *Molecules* **2022**, *27*, 6975. [[CrossRef](#)]
34. Manzoor, S.; Siddiqui, M.K.; Ahmad, S. Computation of entropy measures for phthalocyanines and porphyrins dendrimers. *Int. J. Quantum Chem.* **2022**, *122*, e26854. [[CrossRef](#)]
35. Shannon, C.E. A mathematical theory of communication. *Bell Syst. Tech. J.* **1948**, *27*, 379–423. [[CrossRef](#)]

MDPI
St. Alban-Anlage 66
4052 Basel
Switzerland
Tel. +41 61 683 77 34
Fax +41 61 302 89 18
www.mdpi.com

Mathematics Editorial Office
E-mail: mathematics@mdpi.com
www.mdpi.com/journal/mathematics





Academic Open
Access Publishing

www.mdpi.com

ISBN 978-3-0365-8353-2