UNIVERZA NA PRIMORSKEM
FAKULTETA ZA MATEMATIKO, NARAVOSLOVJE IN INFORMACIJSKE TEHNOLOGIJE

DOKTORSKA DISERTACIJA
(DOCTORAL THESIS)

O JADRANSKIH INDEKSIH, SPEKTRALNIH LASTNOSTIH JADRANSKIH MATRIK IN PROGRAMSKA OPREMA ZA TOPOLOŠKE DESKRIPTORJE

(ON ADRIATIC INDICES, SPECTRAL PROPERTIES OF ADRIATIC MATRICES AND SOFTWARE FOR TOPOLOGICAL DESCRIPTORS)

ALEXANDER VASILYEV

KOPER, 2014

UNIVERZA NA PRIMORSKEM
FAKULTETA ZA MATEMATIKO, NARAVOSLOVJE IN INFORMACIJSKE TEHNOLOGIJE

DOKTORSKA DISERTACIJA
(DOCTORAL THESIS)

O JADRANSKIH INDEKSIH, SPEKTRALNIH LASTNOSTIH JADRANSKIH MATRIK IN PROGRAMSKA OPREMA ZA TOPOLOŠKE DESKRIPTORJE

(ON ADRIATIC INDICES, SPECTRAL PROPERTIES OF ADRIATIC MATRICES AND SOFTWARE FOR TOPOLOGICAL DESCRIPTORS)

ALEXANDER VASILYEV

KOPER, 2014 MENTOR: PROF. DR. DRAGAN STEVANOVIĆ

Acknowledgement

I would like to express my gratitude to my supervisor Dragan Stevanović for his help and support. I am also very grateful to Damir Vukičević for the great time in Split and his scientific advices. I thank Jelena Sedlar for her help and valuable discussions.

I would also like to thank all the personnel of the Faculty of Mathematics, Natural Sciences and Information Technologies and Andrej Marušič Institute who made the process of my study smooth.

Хочу поблагодарить своих родителей, а также своего брата, по примеру которого я когда-то выбрал направление «прикладная математика».

Также я бы хотел поблагодарить Виктора Ермолаева, как ключевого мотиватора попробовать податься в PhD, а также Александра Борисовича Батхина, моего научного руководителя в ВГИ ВолГУ.

Особую благодарность заслуживает моя горячо любимая жена Евгения, без которой эта работа вряд ли бы увидела свет. Спасибо моей доченьке Алисе, которая радовала меня все это время.

Alexander Vasilyev

Contents

Index of Figures **viii**

Index of Tables **x**

1 Introduction **1**

2 Background **5**

2.1 Graphs 5
2.2 Spectra of graphs 6
2.3 Molecular descriptors 7
2.3.1 Topological indices 7
2.3.2 Discrete Adriatic indices 10
2.4 Sage: mathematics software system 11

3 MathChem package **13**

3.1 Installation 13
3.1.1 Installing MathChem as a Python module 13
3.1.2 Installing MathChem as a Sage module 14
3.2 MathChem package structure 14
3.3 Input of molecular graphs 15
3.3.1 Constructing a molecular graph 15
3.3.2 Reading data from an external file 16
3.3.3 Processing large files 17
3.3.4 Downloading data from the NCI online database 18
3.4 MathChem properties and methods 18
3.4.1 Basic properties 19
3.4.2 Graph matrices and their spectral properties 19
3.4.3 Topological indices 22
3.5 More elaborate examples of MathChem use 24
3.5.1 Examples of integration with NetworkX and Sage 25
3.5.2 Correlation examples 27
3.5.3 Creating an interactive widget 29
3.6 Conclusion 32

4 Inverse Sum Indeg Index 33
4.1 Preliminaries . . . 34
4.2 Minimum values of the inverse sum indeg index . . . 36
4.3 Maximum values of the inverse sum indeg index . . . 42
4.4 Open problems . . . 46

5 Symmetric Division Deg Index 47
5.1 Minimum values of the symmetric division deg index . . . 48
5.2 Maximum values of the symmetric division deg index . . . 49

6 Trees of given order and independence number with minimal first Zagreb index 53
6.1 Proof of the main result . . . 54
6.2 Unicyclic graphs . . . 57

7 Applications of MatchChem 61
7.1 Correlation analysis of spectral properties of Adriatic matrices . . . 61
7.1.1 Octane isomers . . . 64
7.1.2 Polyaromatic hydrocarbons . . . 64
7.1.3 Polychlorobiphenyls . . . 65
7.1.4 Phenetyl-amines with two substituent sites . . . 66
7.1.5 Conclusion . . . 67
7.2 Network of topological indices . . . 69
7.2.1 Connected graphs . . . 73
7.2.2 Trees and unicyclic graphs . . . 74
7.2.3 Chemical graphs . . . 74
7.2.4 Conclusion . . . 76

8 Conclusion 77

Bibliography 79

Index 86

List of Figures

3.1 Molecular graph can be visualized with show() method from Sage. . 26

3.2 Bar chart of Randić index for connected NCI compounds with NSC numbers from 1 to 5000. 28

3.3 Scatter plot of Randić index versus the Harary index for connected NCI compounds with NSC numbers from 1 to 5000. 29

3.4 Histogram of orders of molecular graphs for connected NCI compounds with NSC numbers from 1 to 5000. 30

3.5 Interactive widget. 31

4.1 Two trees with the same value of the inverse sum indeg index, exactly one of which is a broom. 41

4.2 Trees on 18, 19 and 20 vertices having maximum inverse sum indeg index. 46

4.3 Chemical trees on 18, 19 and 20 vertices having maximum inverse sum indeg index. 46

5.1 Graph S_n^+ whose value of the SDD index is maximal in the class of unicyclic graphs. 51

6.1 Three non-isomorphic trees with minimum value of $M_1 = 36$. $n = 10$, $\alpha = 6$. 54

6.2 Two non-isomorphic trees with minimum value of $M_1 = 48$. $n = 11$, $\alpha = 8$. 54

6.3 All non-isomorphic graphs $U_{10,7}$. 58

7.1 Octanes. Correlation between MON and Randic type sdi energy . . . 66

7.2 Octanes. Correlation between TSA and Randic type lordeg spectral radius 66

7.3 Octanes. Correlation between S and sum sdeg energy 66

7.4 Octanes. Correlation between BP and inverse sum indeg energy . . . 67

7.5 Octanes. Correlation between MP and Randic type hadi spectral radius 67

7.6 Octanes. Correlation between HVAP and min-max rodeg energy . . 67

7.7 Octanes. Correlation between CP and Randic type lordeg spectral radius 68

7.8 Polychlorobiphenyls. Correlation between MP and min-max sdi energy 69

7.9 classes of networks of topological indices. 70

7.10 Network of topological indices. Connected graphs. $R^2 \geq 0.9$ 71
7.11 Network of topological indices. Trees. $R^2 \geq 0.9$ 72
7.12 Network of topological indices. Unicyclic graphs. $R^2 \geq 0.9$ 73
7.13 Correlation of MTI and incidence energy. 74
7.14 Correlation of first Zagreb index and the largest eigenvalue of Seidel matrix. 75
7.15 Correlation of geometric arithmetic index and fourth spectral moment of Seidel matrix. 75
7.16 Correlation of multiplicative Π_2 Zagreb index and normalized Laplacian energy. 76

List of Tables

3.1 Input file formats. 16
3.2 Functions for retrieving structures from the NCI online database. . . 18
3.3 Basic properties of *Mol* objects 19
3.4 Molecular graph matrices 19
3.5 Topological indices 22
3.6 The useful discrete Adriatic indices. 23

7.1 Molecular descriptors symbols 64
7.2 Correlation coefficients R^2 for Octane isomers (C8) 65
7.3 Correlation coefficients R^2 for Polyaromatic hydrocarbons (PAH) . . 68
7.4 Correlation coefficients R^2 for Polychlorobiphenyls (PCB) 68
7.5 Correlation coefficients R^2 for Phenetyl-amines (Phenet). 69

Abstract

ON ADRIATIC INDICES, SPECTRAL PROPERTIES OF ADRIATIC MATRICES AND SOFTWARE FOR TOPOLOGICAL DESCRIPTORS

This work primarily deals with study of topological indices used in the field of mathematical chemistry.

First we present a novel software tool MathChem aimed to calculate existing and define new topological indices based on the structure of molecular graphs. We describe in details all the aspects of MathChem usage and provide comprehensive examples.

Using MathChem we obtained a number of conjectures for different open problems, mostly related to discrete Adriatic indices. In general, the results can be represented as an answer to the following questions.

- What are the lower and upper bounds for a certain topological index among the graphs with given property?
- What is the structure of the corresponding extremal graphs?

In particular, the following problems are solved.

- We determine extremal values and extremal graphs of the *inverse sum indeg index* in several classes of graphs (with given number of vertices): the class of all connected graphs, the class of all trees, the class of all chemical graphs, the class of all chemical trees, the class of all graphs with given maximum degree, the class of all graphs with given minimum degree, the class of all graphs with given number of pendant vertices, and the class of all trees with given number of pendant vertices. We give some complete and some partial solutions for these problems.
- We determine extremal values and extremal graphs of the *symmetric division deg index* in several classes of graphs (with given number of vertices): the class of all connected graphs, the class of all trees, the class of all unicyclic graphs, the class of all chemical graphs, the class of all graphs with given minimum degree. We also determine maximum values of the index in the class of all graphs with given maximum degree.
- We give the lower bound of *Zagreb M_1 index* of trees and unicyclic graphs with fixed independence number. We also describe the structure of the corresponding extremal graphs.

Math. Subj. Class (2010): 05C05, 05C10, 05C40, 05C35, 68R10, 92E10

Key words: topological descriptors, bond-additive descriptors, discrete Adriatic indices, software.

Izvleček

O JADRANSKIH INDEKSIH, SPEKTRALNIH LASTNOSTIH JADRANSKIH MATRIK IN PROGRAMSKA OPREMA ZA TOPOLOŠKE DESKRIPTORJE

V delu se v prvi vrsti ukvarjamo z vprašanji, ki se nanašajo na preučevanje topoloških indeksov.

Najprej predstavimo programski okvir MathChem, ki ga uporabljamo za izračun obstoječih in definiranje novih topoloških indeksov iz struktur molekulskih grafov. Podrobno opišemo različne vidike uporabe paketa MathChem in podamo primere.

Z uporabo paketa MathChem smo prišli do več domnev v povezavi z odprtimi problemi, večinoma povezanih z Jadranskim indeksom. V splošnem lahko rezultate predstavimo kot odgovore na sledeča vprašanja:

- Kaj so spodnje in zgornje meje za določen topološki indeks med grafi z dano lastnostjo?
- Kakšna je struktura pripadajočega ekstremalnega grafa?

Rešeni so naslednji problemi.

- Določili smo ekstremne vrednosti in ekstremalne grafe indeksa *inverse sum indeg index* v več razredih grafov (z danim številom vozlišč): razred vseh povezanih grafov, razred vseh dreves, razred vseh kemijskih grafov, razred vseh kemijskih dreves, razred vseh grafov z dano maksimalno stopnjo, razred vseh grafov z dano minimalno stopnjo, razred vseh grafov z danim številom listi in razred vseh dreves za danim številom listi. Podane je nekaj popolnih in nekaj delnih rešitev teh primerov.
- Določili smo ekstremne vrednosti in ekstremalne grafe indeksa *symmetric division deg index* v več razredih grafov (z danim številom vozlišč): razred vseh povezanih grafov, razred vseh dreves, razred vseh enocikličnih grafov, razred vseh kemijskih grafov, razred vseh grafov z dano minimalno stopnjo. Določili smo tudi maksimalno vrednost indeksa *symmetric division deg index* v razredu vseh grafov z dano maksimalno stopnjo.
- Določili smo spodnjo mejo za indeks *Zagreb M_1 index* dreves in enocikličnih grafov z določenim številom *independence number.* Opisane so tudi strukture pripadajočih ekstremalnih grafov.

Math. Subj. Class (2010): 05C05, 05C10, 05C40, 05C35, 68R10, 92E10.

Ključne besede: topološke deskriptorje, Jadranski indeksi, programska oprema.

Краткое содержание

ОБ АДРИАТИЧЕСКИХ ИНДЕКСАХ, СПЕКТРАЛЬНЫХ СВОЙСТВАХ АДРИАТИЧЕСКИХ МАТРИЦ И ПРОГРАММНОМ ОБЕСПЕЧЕНИИ ДЛЯ ТОПОЛОГИЧЕСКИХ ДЕСКРИПТОРОВ

Настоящая работа касается изучения топологических индексов, используемых в области математической химии.

В первой части мы представляем новый программный инструмент MathChem, цель которого — вычисление существующих и определение новых топологических индексов, основанных на структуре молекулярных графов. Мы описываем в деталях все аспекты использования MathChem и предоставляем исчерпывающую документтацию с примерами.

Используя MathChem, мы вывели множество гипотез для различных открытых проблем, по большей части связанных с дискретными Адриатическими индексами. В целом, результаты исследования могут быть представлены в виде ответов на следующие вопросы:

- Каковы верхние и нижние границы значений отдельных топологических индексов для графов с данными свойствами?
- Какова структура соответствующих экстремальных графов?

В частности, следующие проблемы были решены.

- Нахождение верхней и нижней границ значений и соответствующих экстремальных графов индекса *inverse sum indeg index* в следующих классах графов (с данным числом вершин): класс всех связных графов, класс деревьев, класс химических графов, класс химических деревьев, класс графов с данной максимальной степенью, класс графов с данной минимальной степенью, класс графов и деревьев с данной числом висячих вершин.

  Некоторые из проблем решены полностью, некоторые — частично.

- Нахождение верхней и нижней границ значений и соответствующих экстремальных графов индекса *isymmetric division deg index* в следующих классах графов (с данным числом вершин): класс всех связных графов, класс деревьев, класс графов с одним циклом, класс химических графов, класс графов с данной минимальной степенью.
- Мы даем нижню границу индекса *Zagreb M_1 index* для деревьев и графов с одним циклом с фиксированным числом *independence number*. Мы также описываем структуру соответствующих экстремальных графов.

Chapter 1

Introduction

The PhD Thesis contributes to the field of mathematical and computational chemistry by introducing a novel software tool *MathChem* together with some new mathematical results obtained with its help.

Molecular descriptors, being numerical functions of molecular structure, play a fundamental role in chemistry. They are used in QSAR and QSPR studies to relate biological or chemical properties of molecules to specific molecular descriptors, thus enabling prediction of properties of molecules based on their structure only and without their synthetization. Topological indices, being numerical functions of (usually hydrogen-suppressed) molecular graph, represent an important type of molecular descriptors. Themselves being graph invariants, topological indices do not consider information about molecular geometry, such as bond lengths, bond angles or torsion angles, but instead encode information on atom adjacencies and branching within a molecule. Perhaps the most well-known topological indices are the Wiener index, the Randić index, the Hosoya Z index, the Balaban J index and graph energy (for their definitions and basic properties see, e.g., [82]). Since computation of topological indices uses fewer resources than computation of those molecular descriptors that also take molecular geometry into account, topological indices have gained considerable popularity and many new topological indices have been proposed and studied in the mathematical chemistry literature in recent years.

Although the existing QSAR software (such as Dragon [53, 54], Molgen-QSPR [4, 44], GenerateMD [6], PowerMV [49], Molconn-Z [23], CODESSA [76], Chemical Descriptors Library [80], AZOrange [77], PaDEL-Descriptor [60, 97] or Chemistry Development Kit [79]) implements calculation of topological indices, the focus is usually put onto a handfull of well-known indices, while many topological indices of interest to mathematical chemists are simply discarded. With 4885 molecular descriptors implemented (noting that many of them are specialized variations of more general descriptors), among which more than a thousand may be considered as topological indices, Dragon [54] probably has the most extensive list of implemented topological indices, but even it does not provide topological indices such as the Laplacian energy or the incidence energy.

Further, QSAR software expects molecular graphs to arrive from a set of molecules, provided in one of chemical formats such as SMILES or Molfile. It is not easy (or even not possible) to use QSAR software for answering questions like: *Which* *chemical tree on 16 vertices and diameter four has largest graph energy?* Such extremal problems, while being the topic of many mathematical chemistry articles published in journals such as *MATCH Communications in Mathematical and in Computer Chemistry*, *Journal of Mathematical Chemistry* or *Croatica Chemica Acta*, are anyway not the type of problems that QSAR software is aimed at and, consequently, such software is of little use in solving them.

In order to resolve these issues and serve a better purpose to researchers in mathematical chemistry, we have devised MathChem so that:

- it implements a set of topological indices that well represents current research in mathematical chemistry literature;
- it can load molecular and ordinary graphs from both chemical sources and graph theoretical sources;
- it is not bounded to solve predefined types of problems only, and
- anyone can easily extend it with definitions of new topological indices.

From these reasons, MathChem is implemented as an open-source Python package. Although Python [70] is a programming language, it is based on minimalist phylosophy and with strong emphasis on readability of the code (which the reader will be able to experience in the rest of the chapter through examples of the MathChem use). Due to these qualities, Python has a short learning time and is well accepted in scientific community. An additional advantage is that MathChem can be used in conjuction with a large number of scientific software already implemented in Python, such as Sage, the open source mathematical environment [78], NetworkX, the high-productivity software for complex networks [38, 61], or SciPy, the open source software system for mathematics, science, and engineering [42].

In Chapter 3 we describe in details all important aspects of installing, using, and integrating MathChem. We also provide several examples of how MathChem can be used in practice.

Being an extremely flexible framework, MathChem allowed us to generate conjectures regarding the structure of extremal graphs with some prescribed properties and resolve some open problems appeared in the literature. Namely, in Chapter 4 we investigate the mathematical properties of *inverse sum indeg index (ISI index)*, the discrete Adriatic index, showed good predictive property of total surface area of octane isomers. Discrete Adriatic indices is a family of 148 bond-additive topological indices defined by D. Vukićević in 2010 [87].

Chapter 5 continues study of discrete Adriatic indices by establishing upper and lower bounds for *symmetric division deg index (SDD index)* in certain classes of graphs. More precisely, extremal values and extremal graphs of the SDD index is given in the class of all connected graphs, all trees, all unicyclic graphs, all chemical graphs, and all graphs with given minimum degree. Moreover, we determine maximum values of the symmetric division deg index in the class of all graphs with given maximum degree.

Despite the fact that the Zagreb indices (M_1 and M_2) are well studied there are several open problems regarding their extremal values and corresponding extremal graphs in certain classes of graphs with prescribed properties. One of such problem was formulated in the paper published by K. C. Das, K. Xu, I. Gutman in [*MATCH Commun. Math. Comput. Chem.* **70** (2013), 301–314.]. The authors asked for characterization of the extremal tree with minimal (first or second) Zagreb index among trees of order n and independence number α. In Chapter 6 we provide a solution of this problem for the first Zagreb index. Moreover, in section 6.2 we extend the result obtained for trees to the class of unicyclic graphs.

In Chapter 7 we illustrate MathChem usage by providing correlation analysis of spectral properties of *Adriatic matrices* and measured physical-chemical measured properties on the benchmarks sets [59] proposed by the International Academy of Mathematical Chemistry. We also try to find unexpected correlations between certain topological indices that are based on different graph properties by establishing networks of topological indices.

In Chapter 2 notions concerning the thesis are introduced together with the notation and some auxiliary results that are needed in the subsequent chapters. Throughout the thesis graphs are finite, simple and undirected.

The results of this PhD Thesis are published in the following articles:

- A. Vasilyev, D. Stevanović, Mathchem: a python package for calculating topological indices, *MATCH Commun. Math. Comput. Chem.* **71** (2014) 657–680.
- A. Vasilyev, Upper and lower bounds of symmetric division deg index, *Iran. J. Math. Chem.* accepted.
- A. Vasilyev, R. Darda, D. Stevanović, Trees of given order and independence number with minimal first Zagreb index, *MATCH Commun. Math. Comput. Chem.* accepted.
- J. Sedlar, D. Stevanović, A. Vasilyev, On the Inverse Sum Indeg Index, *Discrete Applied Mathematics* submitted.

Chapter 2

Background

2.1 Graphs

A *graph* or *undirected graph* G is an ordered pair $G = (V, E)$ where $V = V(G)$ is a set, whose elements are called *vertices*, and $E = E(G)$ is a set whose elements are 2-element subsets of V, called *edges.* The usual way to picture a graph is by drawing a dot for each vertex and joining two of these dots by a line if the corresponding two vertices form an edge. How these dots and lines are drawn is considered irrelevant. All that matters is the information which pairs of vertices form an edge and which do not. The *order* of a graph G is the cardinality of its vertex set $|V(G)|$. Graphs are *finite* or *infinite* according to their order.

A vertex v is *incident* with an edge e if $v \in e$. The two vertices incident with an edge are its *endvertices*, and are said to be *adjacent.* For adjacent vertices u and v in G, we write $u \sim v$ and denote the corresponding edge by uv. If $u \in V(G)$ then $N(u)$ denotes the set of neighbors of u.

Pairwise non-adjacent vertices or edges are called *independent.* More formally, a set of vertices or of edges is *independent* (or *stable*) if no two of its elements are adjacent. The *degree* (or *valency*) $d_G(v) = d(v)$ of a vertex v in G is the number of edges incident to the vertex v, that is $|N(v)|$. If each vertex of the graph has the same degree d, the graph is called a *regular graph of degree d* (or a *d-regular graph*). The *maximum degree* of G is denoted by Δ. The *minimum degree* of G is denoted by δ.

Two graphs G and H are equal if and only if they have the same vertex set and the same edge set. Although this is a perfectly reasonable definition, for most purposes the model of a relationship is not essentially changed if H is obtained from G just by renaming the vertex set. This motivates the following definition: Two graphs G and H are *isomorphic* if there is a bijection φ from $V(G)$ to $V(H)$ such that $x \sim y$ in G if and only if $\varphi(x) \sim \varphi(y)$ in H. We say that φ is an *isomorphism* from G to H. If G and H are isomorphic, then we write $G \cong H$. It is normally appropriate to treat isomorphic graphs as if they were equal.

A *walk* in a graph G is an alternating sequence of vertices and edges, beginning and ending with a vertex, where each vertex is incident to both the edge that precedes it and the edge that follows it in the sequence. A *path* in G is a sequence of vertices $(v_1, v_2, \ldots, v_n)$ such that v_1v_2, v_2v_3, $\ldots$, $v_{n-1}v_n$ are edges of G and the vertices v_i are all distinct. A closed path $(v_1, v_2, \ldots, v_n, v_1)$ in G is called a *cycle*. The *length* of a path or a cycle is the number of edges it contains. An *even* (*odd*) cycle is a cycle of even (odd) length. A *Hamiltonian path* in G is a path which contains each vertex of G. A *Hamiltonian cycle* in G is a cycle that contains each vertex of G. A graph is *hamiltonian* if it possesses a Hamiltonian cycle. The *girth* $g = g(G)$ of G is the length of the shortest cycle contained in the graph. By an n-cycle we shall always mean a cycle with n vertices.

The *distance* $d(u, v)$ between two vertices u and v in G is the number of edges in a shortest path connecting them. With $N_i(u)$ we denote the set of vertices at distance $i > 1$ from a vertex u. A graph is *connected* if any two of its vertices are linked by a path, otherwise the graph is *disconnected*. A graph G is said to be *bipartite* if $V(G)$ can be divided into two disjoint sets U and U' ($U, U' \subseteq V(G)$, $U \cup U' = V(G)$) such that every edge has one endvertex in U and one in U'; that is, there is no edge between two vertices in the same set. It is not difficult to prove that a graph is bipartite if and only if it contains no odd cycle. A graph without a cycle (that is, an *acyclic* graph) is called a *forest*. A connected forest is called a *tree*. The vertices of degree 1 in a tree are its *leaves*. Another name of the vertices of degree 1 in any graph are *pendant vertices*. Every non-trivial tree has at least two leaves.

A *subgraph* of a graph G is a graph Y such that $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. If $V(H) = Y(G)$, we say that H is a *spanning subgraph* of G. A k-regular spanning subgraph is called a *k-factor*. If $S \subseteq V(G)$, then the subgraph $G[S]$ of G whose edges are precisely the edges of G with both endvertices in S is called the *graph induced by S*. More precisely, $V(G[S]) = S$ and $E(G[S]) = \{\{u, v\} \mid u, v \in S\} \cap E(G)$. Let G be a graph, and H its subgraph. Then $G - H$ denotes a graph with vertex set $V(G)$ and edge set $E(X - H) = E(G) \setminus E(H)$.

2.2 Spectra of graphs

Spectral graph theory looks at the connection between the eigenvalues of a matrix associated with a graph and the corresponding structures of a graph. Let G be a graph of order n without multiple edges. The *adjacency matrix* A is a square matrix of order n whose uv-entry is 1 if there is an edge uv in G, and 0 otherwise. The *Laplacian* $L = D - A$ where D is the diagonal matrix of degrees and A is the adjacency matrix. The *signless Laplacian* $Q = D + A$ where D is the diagonal matrix of degrees and A is the adjacency matrix. The *normalized Laplacian* $L = D^{-1/2}LD^{-1/2}$ where D is the diagonal matrix of degrees and L is the Laplacian matrix. By convention when a vertex has degree 0 (i.e., is isolated) then the corresponding entry of $D^{-1/2}$ is 0.

The characteristic polynomial $det(xI - A)$ of the adjacency matrix A is called the *characteristic polynomial* of G, and is denoted by $P_G(x)$. The eigenvalues of A (i.e., the zeros of $det(xI - A)$), and the spectrum of A (which consists of the n eigenvalues) are also called the *eigenvalues* and the *spectrum* of G, respectively. The eigenvalues of G are usually denoted by $\lambda_1, \lambda_2, \ldots, \lambda_n$; they are real because A is symmetric.

We shall assume that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$. We also use the notation $\lambda_i = \lambda_i(G)$ for $i = 1, 2, \ldots, n$. The largest eigenvalue, i.e., λ_1, is called the *spectral radius* or *index* of G.

The *energy* of a graph is the sum of the absolute values of the eigenvalues of the adjacency matrix of the graph. If instead of adjacency matrix Laplacian or incidence matrix is used then such sum is called *Laplacian* or *incidence energy.* respectively.

2.3 Molecular descriptors

Quantitative structure–activity relationship (QSAR) is the final result of the process that starts with a suitable description of molecular structures and ends with some inference, hypothesis, and prediction on the behavior of molecules in environmental, biological, and physico-chemical systems in analysis. QSARs are based on the assumption that the structure of a molecule (for example, its geometric, steric, and electronic properties) must contain features responsible for its physical, chemical, and biological properties and on the ability to capture these features into one or more numerical descriptors. Related terms include *quantitative structure–property relationships (QSPR)* when a chemical property is modelled as the response variable.

The majority of the QSAR strategies aimed at building models are based on regression and classification methods, depending on the problem studied. For continuous properties, like most of the biological activities and physico-chemical properties, the typical QSAR/QSPR model is defined as

$$P = f(x_1, x_2, \ldots, x_p),$$

where P is the molecular property/activity, $x_1, \ldots, x_p$ are the p molecular descriptors, and f is a function representing the relationship between response and descriptors.

The *molecular descriptor* is the final result of a logic and mathematical procedure which transforms chemical information encoded within a symbolic representation of a molecule into an useful number or the result of some standardized experiment [82]. By this definition, the molecular descriptors are divided into two main categories: *experimental measurements*, such as *log P, molar refractivity, dipole moment, polarizability,* and, in general, physico-chemical properties, and *theoretical molecular descriptors*, which are derived from a symbolic representation of the molecule and can be further classified according to the different types of molecular representation.

2.3.1 Topological indices

In the fields of chemical graph theory, molecular topology, and mathematical chemistry, a *topological index* or *topological descriptor* is a type of a molecular descriptor that is calculated based on the molecular graph of a chemical compound.

Simple topological indices do not recognize double bonds and atom types (C, N, O etc.) and ignore hydrogen atoms ("hydrogen suppressed") and defined for connected undirected molecular graphs only.

Many descriptors are *bond-additive*, i.e. they can be presented as a sum of edge contributions. A bond additive descriptor has the following form:

$$\sum_{uv \in E} f(g(u), g(v))$$

where $g(u)$ are usually degrees or the sum of distances from u to all other vertices of G. Descriptors of this form that are mostly studied in the literature are:

- The most studied topological index is *Randić index* also known as the Connectivity index. The index was defined in study of characterization of molecular branching by Milan Randić [72] in 1975 as a sum of bond contributions in molecule $\sum_{uv \in E}(d_u d_v)^{-1/2}$. Where d_u and d_v are degrees of corresponding adjacent vertices of the molecular graph [2, 69, 73]. The index was generalized as the sum of the weights $\sum_{uv \in E}(d_u d_v)^{\alpha}$. Some bounds for the connectivity index of a chemical graph was given in [1]. And then sharp bounds for the General Randić index was found [46]. Various theoretical results was obtained for extremal values of the index [3, 5, 47, 67, 68], for different classes of graphs: trees [9, 66], chemical graphs [33, 45], benzenoid systems and phenylenes [71], acyclic conjugated molecules and trees with a given size of matching [50]. The characteristic of extremal chemical trees with minimum or maximum General Randić index was given in [48]. Maximum Randić index on Trees with k-pendant Vertices was studied in [98]. A spectral approach was applied by Rodríguez in [74].

- The graph invariant M_2 , known under the name *second Zagreb index* , equal to the sum of the products of the degrees of pairs of adjacent vertices of the respective (molecular) graph, was first considered in 1972 [36]. A number of results for M_2 – identities and inequalities, including relations between the M_2-index of a graph and its complement was given in [12]. There was established an upper bound for the sum of the squares of the degrees of a graph G [10]. Then the sharp upper and lower bounds on the second Zagreb index of unicyclic graphs with n vertices and k pendant vertices were obtained in [96]. Trees with maximal second Zagreb index and prescribed number of vertices of the given degree was studied in [88]. Then an upper bound on the first Zagreb index M_1 of graph G in terms of the number of vertices n, number of edges m, maximum vertex degree Δ_1, second maximum vertex degree Δ_2 and minimum vertex degree δ was obtained. Using this result a new upper bound was found on M_2 [14]. There are also other results on Zagreb indices [27, 62]

- *ABC index* was defined by Ernesto Estrada in his work dedicated to modelling the enthalpy of formation of alkanes. [26]. The fromal definition is the following: $ABC = \sum_{uv \in E} \sqrt{\frac{d_u + d_v - 2}{d_u d_v}}$. Where again d_u and d_v are degrees of corresponding adjacent vertices of the molecular graph. He continued study of the index in [25]. Extremal values of the index was considered in [7, 28]. Lower and upper bounds was obtained in [11] where characterization of graphs for which these bounds are best possible was given. Some sharp lower and upper bounds on ABC was studied in [30]. Further results on atom-bond connectivity index of trees are listed in [92]. Several properties of ABC are established in [13]. Upper bounds for this graph invariant using the number of vertices, the number of edges, the Randić connectivity indices, and the first Zagreb index were given in [102]. Novel upper bounds on two vertex-degree-based graph invariants, Randić index and ABC index were obtained in [39]. A few structural features of the trees with minimal ABC were determined by Ivan Gutman in [35].

- In the last years there are a lot of new indices appeared. *Augmented Zagreb index* was defined in [29] where the tight upper and lower bounds for chemical trees are obtained. Upper and lower bounds for various classes of connected graphs, and characterization the corresponding extremal graphs was established in [40]. The further analysis was continued in [89].

- *Sum-Connectivity index* was posed as a novel connectivity index and defined in [99]. This is the additive version of the connectivity index introduced in 1975 by Milan Randić [51]. The extremal graphs are characterized in [19]. Some properties especially lower and upper bounds in terms of other graph invariants for the general sum-connectivity index which generalizes both the ordinary sum-connectivity index and the first Zagreb index were reported in [100]. Then extremal values of the index were studiet for various graph classes: for unicyclic graphs with k pendent vertices [8], for molecular trees [21, 94], unicyclic graphs [20], unicyclic graphs of fixed maximum degree [22]. Comparison between the sum-connectivity index and product-connectivity index for benzenoid hydrocarbons was considered in [52]. Sum–connectivity matrix and sum-connectivity energy of (molecular) graphs was studied in [101].

- It was found that the *sum-Balaban index* is correlated well with some physicochemical properties and other topological indices of octanes and lower benzenoids [16]. The value of the index for class of dendrimers was computed in [91]. Various upper and lower bounds for the sum-Balaban index were established, and determined the trees with the largest, second-largest, and third-largest as well as the smallest, second-smallest, and third-smallest sum-Balaban indices among the n-vertex trees for $n \geq 6$ [93].

- *Geometric-Arithmetic index* is also based on the degree of the adjacent vertices. The index was introduced in [86]. First geometric–arithmetic index and atom-bond connectivity index was compared in [15]. Further analysis was done in [18].

- *Multiplicative Zagreb indices.* The analysis of the index for trees was given in [32]. The index was also mentioned by R. Todeschini in [81].

- *Multiplicative Sum Zagreb indices* are the ones of the most recent topological descriptors. Multiplicative versions of first Zagreb index was introduced in [24]. The characterization of the trees, unicylcic, and bicyclic graphs extremal (maximal and minimal) with respect to the multiplicative sum Zagreb index was done in [95].

2.3.2 Discrete Adriatic indices

In this thesis particular attention is paid to Adriatic indices that have been defined by D. Vukičević in [87]. *Discrete Adriatic indices* are the family of 148 bond-additive topological indices defined as follows.

$$Adr(G) = \sum_{uv \in E(G)} \gamma_j(\phi_{i,a}(p_u), \phi_{i,a}(p_v)),$$

where the variables and functions can take the following values:

$$p_v \in \{d_v, D_v\} \qquad i \in \{1,2,3\} \qquad j \in \{1,2,...,8\}$$

$$a \in \begin{cases} \{-1, -\frac{1}{2}, \frac{1}{2}, 1, 2\}, & \text{if } i = 2 \text{ and } j \in \{1,2,...,5\} \\ \{\frac{1}{2}, 1, 2\}, & \text{otherwise and } i \in \{1,2\} \\ \{\frac{1}{2}, 2\}, & \text{if } i = 3. \end{cases}$$

$$\begin{aligned} \gamma_1(x,y) =& x \cdot y \\ \gamma_2(x,y) =& x + y \\ \gamma_3(x,y) =& \begin{cases} \frac{1}{x+y}, & \text{if } x+y \neq 0; \\ 0, & \text{if } x+y = 0; \end{cases} \\ \gamma_4(x,y) =& |x-y| \\ \gamma_5(x,y) =& \begin{cases} \frac{1}{|x-y|}, & \text{if } x \neq y; \\ 0, & \text{if } x = y; \end{cases} \\ \gamma_6(x,y) =& \begin{cases} \frac{min\{x,y\}}{max\{x,y\}}, & \text{if } max\{x,y\} \neq 0; \\ 0, & \text{if } max\{x,y\} = 0; \end{cases} \\ \gamma_7(x,y) =& \begin{cases} \frac{max\{x,y\}}{min\{x,y\}}, & \text{if } min\{x,y\} \neq 0; \\ 0, & \text{if } min\{x,y\} = 0; \end{cases} \\ \gamma_8(x,y) =& \begin{cases} \frac{x}{y} + \frac{y}{x}, & \text{if } x,y \neq 0; \\ 0, & \text{otherwise.} \end{cases} \end{aligned}$$

$$\begin{aligned} \phi_{1,a}(x) =& \log^a(x), & a > 0; \\ \phi_{2,a}(x) =& x^a, & a \in \mathbb{R}\backslash\{0\}; \\ \phi_{3,a}(x) =& a^x, & a > 0. \end{aligned}$$

The naming convention for the Adriatic indices is the following.

1. The name of each variable Adriatic index consists of five parts. The third and the the fourth part form the same word.

2. The first part is determined by the choice of the function γ_j :

   - γ_1 corresponds to *Randić type*;
   - γ_2 corresponds to *sum*;

- γ_3 corresponds to *inverse sum*;
- γ_4 corresponds to *misbalance*;
- γ_5 corresponds to *inverse misbalance*;
- γ_6 corresponds to *min-max*;
- γ_7 corresponds to *max-min*;
- γ_8 corresponds to *symmetric division*;

3. The second part is determined by the value of $\phi_{i,a}$:

   - $\phi_{1,1/2}$ corresponds to *lor*;
   - $\phi_{1,1}$ corresponds to *lo*;
   - $\phi_{1,2}$ corresponds to *los*;
   - $\phi_{2,-1}$ corresponds to *in*;
   - $\phi_{2,-1/2}$ corresponds to *ir*;
   - $\phi_{2,1/2}$ corresponds to *ro*;
   - $\phi_{2,1}$ corresponds to no letters (the whole word is just the third part);
   - $\phi_{2,2}$ corresponds to *s*;
   - $\phi_{3,1/2}$ corresponds to *ha*;
   - $\phi_{3,2}$ corresponds to *two*;

4. The fourth part is determined by the choice of the invariant (d_v or D_v):

   - d_v corresponds to *deg*;
   - D_v corresponds to *di*;

5. The fourth part is word *index*.

Let us illustrate this by several examples:

- $\sum_{uv\in E(G)} \dfrac{1}{\frac{1}{d_u}+\frac{1}{d_v}}$ corresponds to *inverse sum indeg index*;
- $\sum_{uv\in E(G)} \left(\frac{\min\{d_u,d_v\}}{\max\{d_u,d_v\}}+\frac{\max\{d_u,d_v\}}{\min\{d_u,d_v\}}\right)$ corresponds to *symmetric division deg index*.

2.4 Sage: mathematics software system

Sage is a free open-source mathematics software system licensed under the GPL. It builds on top of many existing open-source packages: NumPy, SciPy, matplotlib, Sympy, Maxima, GAP, FLINT, R and many more. Sage is math software that supports research and teaching in algebra, geometry, number theory, cryptography, numerical computation, and related areas. Both the Sage development model and the technology in Sage itself are distinguished by an extremely strong emphasis on openness, community, cooperation, and collaboration.

The overall goal of Sage is to create a viable, free, open-source alternative to Maple, Mathematica, Magma, and MATLAB.

Sage's intended audience is mathematics students (from high school to graduate school), teachers, and research mathematicians. The aim is to provide software that can be used to explore and experiment with mathematical constructions in algebra, geometry, number theory, calculus, numerical computation, etc. Sage helps make it easier to interactively experiment with mathematical objects. Sage uses highly-optimized mature software like GMP, PARI, GAP, and NTL, and so is very fast at certain operations. Sage provides robust interfaces to most other computer algebra systems, including PARI, GAP, Singular, Maxima, KASH, Magma, Maple, and Mathematica. Sage is meant to unify and extend existing math software.

Chapter 3

MathChem package

In this chapter we describe MathChem, the open source Python package for calculating topological indices.

Structure of the chapter is as follows. In Section 3.1 we describe installation of MathChem, after which the basic structure of MathChem is discussed in Section 3.2. In Section 3.3 we discuss the different ways of inputting (molecular) graphs in MathChem, and in Section 3.4 we describe MathChem's properties, methods and topological indices it can calculate. Finally, Section 3.5 provides elaborate examples of MathChem use.

Results of this chapter are published in [84].

3.1 Installation

Mathchem package can be installed as a standard Python module or integrated within Sage environment. It is available for download, together with its source code, from its home page `http://mathchem.iam.upr.si/`. During development, MathChem was tested under Mac OS X with Python 2.7 and Sage 5.4. However, as it does not contain any compiled code, MathChem is independent of the operating system and can be used at any computer with Python interpreter installed.

3.1.1 Installing MathChem as a Python module

To install MathChem as a Python module:

1. Go to `http://mathchem.iam.upr.si/` and download the *MathChem for Python* zip archive.

2. Unpack the archive in a folder of your choice.

3. Open the terminal window and make sure you have administrator privileges.

4. Change to the folder (`cd`) where MathChem archive is unpacked, then further change to the module directory: `cd mathchem-package-master`

5. Issue the installation command: `python setup.py install`

Alternatively, if you are familiar with the `pip` tool, you can issue the terminal command

```
pip install mathchem
```

from within the folder where you unpacked MathChem archive file. The `pip` tool checks for dependencies and installs them first, if they are not present.

Mathchem depends on package NumPy [63] only (which may be preinstalled with Python).

3.1.2 Installing MathChem as a Sage module

Sage [78] is an open source mathematics software system, which combines many existing mathematics packages into a common Python-based environment, providing additional web-based interfaces to them through the concept of notebooks. Sage uses a separate instance of Python interpreter for its work, which means that packages installed as Python modules are not automatically available in Sage, but have to be installed separately. To install MathChem as a Sage module:

1. Go to `http://mathchem.iam.upr.si/` and download the *MathChem for Sage* spkg file.
2. Run Sage from the terminal window with the command to install a new package:

   ```
   sage -f spkg-filename
   ```

   where `spkg-filename` denotes the full path to and the name of the spkg file.

In case you have installed Sage as Sage.app on Mac OS X system, choose the option `Development->Reveal in Shell` from Sage menu in order to open the terminal window with the current directory positioned to the Sage folder, and then issue the installation command as `./sage -f spkg-filename`.

3.2 MathChem package structure

The MathChem package consists of two modules: *MathChem* and *Utilities*.

The *MathChem* module contains the *Mol* class, which is the central part of the package. The Mol class contains a representation of a molecular graph in the form of adjacency matrix, together with methods for calculating various graph invariant and topological indices (whose full list is given in Section 5).

The *Utilities* module contains a set of functions for importing molecular graphs from external files and for performing a batch processing over a set of files. The currently supported chemical file formats in MathChem package are MDL MOL format (.mol, .sdf) and Sybyl Mol2 format (.ml2, .mol2). At the moment, further chemical formats can be converted to these by using Open Babel, the open source chemistry toolbox [64,65], which is able to read, write and convert over 110 chemical file formats. This module also contains functions for retrieving structure data online from the NCI online database [41] by the compound name, NSC or CAS number. These functions are elaborated in more detail in the next section.

In order to start working with MathChem, one has to issue the command

```
import mathchem
```

either in Python or in Sage. After issuing it, you may work with MathChem functions during the whole session, so that it is not necessary to issue it again. Note, however, that we have put this command at the beginning of each example in this manuscript, simply to make the examples self-sufficient.

3.3 Input of molecular graphs

The input of molecular graphs in MathChem is possible by directly constructing a *Mol* object, by reading data from an external file or by downloading data from the NCI online database.

3.3.1 Constructing a molecular graph

The direct way to construct a molecular graph in MathChem is to create an empty *Mol* object and then to provide either its edge list or adjacency matrix as the argument to one of the methods `read_edgelist` or `read_matrix`. The following example illustrates both methods:

```
import mathchem

m = mathchem.Mol()
m.read_edgelist( [(1,2), (3,1), (2,3)] )

g = mathchem.Mol()
g.read_matrix( [[0,1,1],[1,0,1],[1,1,0]] )
```

Another direct way to initialize a *Mol* object in MathChem is by providing either a Graph6 or Sparse6 string, representing a molecular graph, as the argument to its constructor:

```
import mathchem
m = mathchem.Mol("GhCH?_")
```

The Brendan McKay's Graph6 format [58] represents the upper part of the adjacency matrix of a graph as a (0,1)-sequence, divides it into chunks of six bits and then translates them to a readable part of the ASCII code. For example, the Graph6 string

"GhCH?_" above represents a carbon skeleton of the 3,4-dimethylhexane (C_8H_{18}). Sparse6 format [58] uses the same basic principle of dividing data into six bit chunks and translating them to a readable part of the ASCII code, with the difference that Sparse6 format encodes the list of graph edges, which may use less space than the adjacency matrix in case of large, sparse graphs.

Graph6 is a popular format among graph theorists for creating collections of graphs—see, for example, the web pages of Brendan McKay [57] or Gordon Royle [75] for a number of collections that are available online. Further collections can be generated in Graph6 format by using, for example, *geng* and *genbg* tools from the Brendan McKay's package *nauty* [56], or the Brendan McKay and Gunnar Brinkmann's program *plantri* [55].

Still, instead of constructing each molecular graph directly from a Graph6 string, it is more advisable to read all graphs from a collection at once with one of the functions described in the following subsection.

3.3.2 Reading data from an external file

MathChem can read molecular graphs from several file formats, originating from chemical sources (MDL MOL and Sybyl Mol2) or graph theoretical sources (Graph6, Sparse6 and planar_code). Planar_code format is relatively similar to Graph6 and Sparse6 formats and its description may be found at the web pages [58] and [55].

Molecular graphs can be read from external files by using functions in Table 3.1. The first argument `fname` is an input file name, while the second optional argument `hydrogens` is a Boolean value indicating whether hydrogen atoms should be read into a molecular graph (`True`) or supressed (`False`, which is the default value). This argument is not present in functions reading molecular graphs from Graph6, Sparse6 and planar_code file formats, as the vertices of a molecular graphs are not labeled in these formats.

Input file format	Input function
MDL MOL (.sdf)	`read_from_sdf(fname [, hydrogens])`
MDL MOL (.mol)	`read_from_mol(fname [, hydrogens])`
Sybyl Mol2 (.ml2, .mol2)	`read_from_mol2(fname [, hydrogens])`
Graph6 (.g6)	`read_from_g6(fname)`
Sparse6 (.s6)	`read_from_s6(fname)`
Planar code (.plc)	`read_from_planar_code(fname)`

Table 3.1: Input file formats.

Further, as files in all these formats (except in MDL MOL .mol format) may contain multiple records, the corresponding functions read all records and return a list of *Mol* objects. MDL MOL .mol file does not support multiple records, so that `read_from_mol` returns a single *Mol* object. For example, the command

```
import mathchem
mols = mathchem.read_from_sdf("compounds.sdf", True)
```

reads all records in `compounds.sdf` and returns a list `mols` containing a separate *Mol* object for each record in the file. Due to the second argument `True`, the command reads hydrogen atoms into molecular graphs as well.

3.3.3 Processing large files

The functions for reading data from an external file from Table 3.1 keep all read data in internal memory as a list of *Mol* objects. In cases where the internal memory is insufficient to hold all data (for example, one wants to process several millions of structures), the function `batch_process` can be used. This function iteratively reads a single molecular graph from an external file, process the graph through a user-supplied function and writes the result to the output text file, before processing the next molecular graph from the external file. The function call has the format

```
batch_process(infile, file_format, outfile, user_function[, hydrogens])
```

with arguments being:

- `infile`—the input filename;
- `file_format`—a string description of the input file format. Allowed values are `"g6"`, `"sparse6"`, `"planar_code"`, `"sdf"` and `"mol2"`;
- `outfile`—name of the output text file that contains results of the `user_function`;
- `user_function`—name of the user-supplied function that takes a single *Mol* object as an argument, performs calculations on it and returns the result as a string, which is then written to `outfile`;
- `hydrogens`—an optional Boolean argument, indicating whether hydrogens should be suppresed (`False`) or included in the molecular graph (`True`).

Let us look at a simple example of batch processing:

```
import mathchem

def process(m):
   e = m.energy()
   le = m.energy("laplacian")
   return str(e) + "; " + str(le)

mathchem.batch_process("compounds.sdf", "sdf", "results.csv", process)
```

After importing MathChem package follows the definition of the function `process(m)`, which calculates the energy and the Laplacian energy of the *Mol* object `m` and returns a string containing these two values, separated with a semicolon. Note that Python uses indentation to identify blocks of code, so that there is no need to separately denote the end of the definition of `process(m)`—it is enough to start the next line of code (i.e., `mathchem.batch_process`) at the same position as the beginning of function definition (i.e., `def process(m):`).

The batch processing command then calls the `process` function for each structure in the file `"compounds.sdf"` and writes the resulting energy and Laplacian energy to the file `"results.csv"`. The output file represents a table in a simple CSV format (one line per structure) and can be loaded into a spreadsheet program for further processing.

3.3.4 Downloading data from the NCI online database

Functions indicated in Table 3.2 provide a simple interface for downloading structures from the NCI online database. These functions perform a search query to the database and return a list of *Mol* objects as a result.

Retrieval type	Retrieval function
By name	`read_from_NCI_by_name(name [, False])`
By CAS number	`read_from_NCI_by_CAS(num [, False])`
By NSC number	`read_from_NCI_by_NSC(num [, False])`

Table 3.2: Functions for retrieving structures from the NCI online database.

Function `read_from_NCI_by_name` retrieves all structures that have `name` as part of their name. So, it is enough to run `read_from_NCI_by_name("alkane")` to retrieve the set of all alkanes in the NCI database.

Function `read_from_NCI_by_CAS` retrieves a structure with a given CAS number.

Function `read_from_NCI_by_NSC` retrieves a structure with a given NSC number. Besides a single number, this function also allows the user to specify a set of numbers, such as `"55+65+75"`, or an interval, such as `"10-20"`.

For example, the following command retrieves all structures (hydrogens supressed) having NSC number between 1 and 1000:

```
import mathchem
mols = mathchem.read_from_NCI_by_NSC("1-1000")
```

Note, however, that if we now issue the command `len(mols)`, which returns the number of items in the list `mols`, the result would be only 993, since the NCI database has gaps among NSC numbers.

3.4 MathChem properties and methods

We describe here the properties of *Mol* objects, and list the methods that calculate a number of topological indices, including the recently introduced Adriatic indices [85, 87]. The MathChem package also contains methods that return various matrices corresponding to molecular graph and can calculate their eigenvalues, spectral moments and energies.

3.4.1 Basic properties

Table 3.3: Basic properties of *Mol* objects

Property	Class method	Return type	Description
Order	`order()` or `n()`	*Integer*	Number of vertices
Vertices	`vertices()`	*List*	of vertices from 0 to `order()-1`
Size	`size()` or `m()`	*Integer*	Number of edges
Edges	`edges()`	*List*	of edges as vertex pairs
Degree list	`degrees()` or `deg()`	*List*	of vertex degrees
Connectedness	`is_connected()`	*Boolean*	True if connected, False otherwise
Diameter	`diameter()`	*Integer*	The diameter
Eccentricity	`eccentricity()`	*List*	of vertex eccentricities

Table 3.3 contains the list of basic properties of the molecular graph contained in a *Mol* object. After the *Mol* object is constructed, the value of a given property is obtained by issuing command of the form *objectname.propertyname()*, as usual in object-oriented languages. For example, the code

```
import mathchem
m = mathchem.Mol("GhCH?_")
m.degrees()
```

returns

```
[1, 2, 3, 3, 2, 1, 1, 1]
```

3.4.2 Graph matrices and their spectral properties

Table 3.4: Molecular graph matrices

Matrix	Class method
Adjacency matrix	`adjacency_matrix()` or `A()`
Incidence matrix	`incidence_matrix()`
Laplacian matrix	`laplacian_matrix()` or `L()`
Signless Laplacian matrix	`signless_laplacian_matrix()` or `Q()`
Normalized Laplacian matrix	`normalized_laplacian_matrix()` or `NL()`
Distance matrix	`distance_matrix()` or `D()`
Resistance distance matrix	`resistance_distance_matrix()`
Reciprocal Distance matrix	`reciprocal_distance_matrix()`

The list of graph matrices that MathChem is able to calculate is given in Table 3.4. The list of their eigenvalues, sorted from the largest to the smallest, is returned by the function

```
m.spectrum(matrixname)
```

where `m` is the name of the *Mol* object and `matrixname` is one of the following:

- `"adjacency"` or shortly `"A"`;
- `"laplacian"` or shortly `"L"`;
- `"signless_laplacian"` or shortly `"Q"`;
- `"normalized_laplacian"` or shortly `"NL"`;
- `"distance"` or shortly `"D"`;
- `"resistance_distance"` or shortly `"RD"`;
- `"reciprocal_distance"`.

The adjacency matrix is the default value of the argument, so that it can be omitted and one can shortly write `m.spectrum()` instead of `m.spectrum("adjacency")`. For example, the code

```
import mathchem
m = mathchem.Mol("GhCH?_")
m.spectrum()
```

returns the adjacency spectrum

```
[2.095293985223914, 1.355674293978083, 0.7376403052281872, 0.4772599964740198,
-0.477259996474019, -0.737640305228187, -1.355674293978082, -2.09529398522391]
```

For any of these matrices, its spectral radius, defined as the maximum of the absolute values of its eigenvalues, its `k`-th spectral moment, defined as the sum of `k`-th powers of its eigenvalues, and its energy, defined as the absolute deviation of its eigenvalues from their mean value, can be calculated with the following functions

```
m.spectral_radius(matrixname)
m.spectral_moment(k, matrixname)
m.energy(matrixname)
```

where `matrixname` is one of the values listed above. Similarly as above, the value of `matrixname` can be omitted in case the user wants to calculate a spectral property of the adjacency matrix. For example, the code

```
import mathchem
m = mathchem.Mol("GhCH?_")
m.energy()
```

returns the (usual) graph energy

```
9.3317371618084071
```

The incidence matrix is not a square matrix in general, so that the incidence energy is defined as the sum of its singular values. It is calculated with the function

```
m.incidence_energy()
```

MathChem also contains the corresponding functions for calculating spectral properties of an arbitrary user-supplied matrix `matrix`, represented as a two-dimensional array:

```
mathchem.spectrum(matrix)
mathchem.spectral_radius(matrix)
mathchem.spectral_moment(k, matrix)
mathchem.energy(matrix)
```

For example, the code

```
matrix = [[1,0,1],[0,1,0],[0,1,1]]
mathchem.spectrum(matrix)
```

returns

```
[2.0, 1.0, 0.0]
```

We should add here that, for performance reasons, MathChem calculates invariants of a *Mol* object on demand and then saves the results for future use. Every *Mol* object has its own set of private variables which is used as a cache for this purpose. This way, MathChem avoids unnecessary recalculation of resource consuming data, such as matrices or their spectral properties. For example, suppose that we want to calculate two distance-based invariants, the diameter and the distance energy of a molecular graph:

```
import mathchem
m = mathchem.Mol("GhCH?_")
print m.diameter(), m.energy("distance")
```

Both of these functions need a distance matrix of the molecular graph, which is calculated internally during the first function call `m.diameter()` and then reused, without recalculation, in the second function call `m.energy("distance")`.

3.4.3 Topological indices

MathChem package implements most popular topological indices. The full list of implemented indices is given in Tables 3.5 and 3.6. If necessary, see [82] for definitions and further references.

Table 3.5: Topological indices

Topological index	Class method
The first Zagreb Index	`zagreb_m1_index()`
The second Zagreb Index	`zagreb_m2_index()`
Connectivity index (R(power))	`connectivity_index(power)`
Randić Index (R(-1/2))	`randic_index()`
Sum-Connectivity index	`sum_connectivity_index()`
Geometric-Arithmetic index	`geometric_arithmetic_index()`
Eccentric Connectivity Index	`eccentric_connectivity_index()`
Atom-Bond Connectivity Index (ABC)	`atom_bond_connectivity_index()`
Estrada Index (EE) of a graph matrix	`estrada_index(matrixname)`
Degree Distance (DD)	`degree_distance()`
Reverse Degree Distance (rDD)	`reverse_degree_distance()`
Molecular Topological Index (MTI)	`molecular_topological_index()`
Eccentric Distance Sum	`eccentric_distance_sum()`
Balaban J index	`balaban_j_index()`
Sum-Balaban index	`sum_balaban_index()`
Kirchhoff Index (Kf)	`kirchhoff_index()`
Wiener Index (W)	`wiener_index()`
Terminal Wiener Index (TW)	`terminal_wiener_index()`
Reverse Wiener Index (RW)	`reverse_wiener_index()`
Hyper-Wiener Index (WW)	`hyper_wiener_index()`
Harary Index (H)	`harary_index()`
Laplacian-like energy (LEL)	`LEL()`
The first Zagreb coindex	`zagreb_m1_coindex()`
The second Zagreb coindex	`zagreb_m2_coindex()`
log(Multiplicative Sum Zagreb index)	`multiplicative_sum_zagreb_index()`
log(Multiplicative P1 Zagreb index)	`multiplicative_p1_zagreb_index()`
log(Multiplicative P2 Zagreb index)	`multiplicative_p2_zagreb_index()`

For example, the following code

```
import mathchem
m = mathchem.Mol("GhCH?_")
print m.zagreb_m1_index(), m.zagreb_m2_index()
```

returns

```
30 31
```

Table 3.6: The useful discrete Adriatic indices.

Adriatic index	Definition	Class method
Randić-type lodeg index	$\sum_{uv\in E(G)} \ln(d_u)\cdot\ln(d_v)$	`randic_type_lodeg_index()`
Randić-type sdi index	$\sum_{uv\in E(G)} D_u^2 D_v^2$	`randic_type_sdi_index()`
Randić-type hadi index	$\sum_{uv\in E(G)} \dfrac{1}{2^{D_u+D_v}}$	`randic_type_hadi_index()`
sum lordeg index	$\sum_{uv\in E(G)} \sqrt{\ln d_u}+\sqrt{\ln d_v}$	`sum_lordeg_index()`
inverse sum lordeg index	$\sum_{uv\in E(G)} \dfrac{1}{\sqrt{\ln d_u}+\sqrt{\ln d_v}}$	`inverse_sum_lordeg_index()`
inverse sum indeg index	$\sum_{uv\in E(G)} \dfrac{d_u d_v}{d_u+d_v}$	`inverse_sum_indeg_index()`
misbalance lodeg index	$\sum_{uv\in E(G)} \lvert\ln d_u-\ln d_v\rvert$	`misbalance_lodeg_index()`
misbalance losdeg index	$\sum_{uv\in E(G)} \lvert\ln^2 d_u-\ln^2 d_v\rvert$	`misbalance_losdeg_index()`
misbalance indeg index	$\sum_{uv\in E(G)} \lvert\frac{1}{d_u}-\frac{1}{d_v}\rvert$	`misbalance_indeg_index()`
misbalance irdeg index	$\sum_{uv\in E(G)} \lvert\frac{1}{\sqrt{d_u}}-\frac{1}{\sqrt{d_v}}\rvert$	`misbalance_irdeg_index()`
misbalance rodeg index	$\sum_{uv\in E(G)} \lvert\sqrt{d_u}-\sqrt{d_v}\rvert$	`misbalance_rodeg_index()`
misbalance deg index	$\sum_{uv\in E(G)} \lvert d_u-d_v\rvert$	`misbalance_deg_index()`
misbalance hadeg index	$\sum_{uv\in E(G)} \lvert(\frac{1}{2})^{d_u}-(\frac{1}{2})^{d_v}\rvert$	`misbalance_hadeg_index()`
misbalance indi index	$\sum_{uv\in E(G)} \lvert\frac{1}{D_u}-\frac{1}{D_v}\rvert$	`misbalance_indi_index()`
min-max rodeg index	$\sum_{uv\in E(G)} \sqrt{\dfrac{\min\{d_u,d_v\}}{\max\{d_u,d_v\}}}$	`min_max_rodeg_index()`
max-min rodeg index	$\sum_{uv\in E(G)} \sqrt{\dfrac{\max\{d_u,d_v\}}{\min\{d_u,d_v\}}}$	`max_min_rodeg_index()`
min-max sdi index	$\sum_{uv\in E(G)} \left(\dfrac{\min\{D_u,D_v\}}{\max\{D_u,D_v\}}\right)^2$	`min_max_sdi_index()`
max-min deg index	$\sum_{uv\in E(G)} \dfrac{\max\{d_u,d_v\}}{\min\{d_u,d_v\}}$	`max_min_deg_index()`
max-min sdeg index	$\sum_{uv\in E(G)} \left(\dfrac{\max\{d_u,d_v\}}{\min\{d_u,d_v\}}\right)^2$	`max_min_sdeg_index()`
symmetric division deg index	$\sum_{uv\in E(G)} \dfrac{d_u^2+d_v^2}{d_u d_v}$	`symmetric_division_deg_index()`

In addition, MathChem implements all 148 discrete Adriatic indices, recently introduced by Vukičević and Gašperov [87] (see also [85]). The general definition of a discrete Adriatic index of a graph G is given by

$$Adriatic(G)=\sum_{uv\in E(G)}\gamma_j(\phi_{i,a}(p(u)),\phi_{i,a}(p(v))),$$

where γ_j is one of eight binary functions listed in [87], $a\in\{-1,-\frac{1}{2},\frac{1}{2},1,2\}$, $\phi_{i,a}(x)$ is either $\log^a(x)$ for $i=1, a>0$, x^a for $i=2$, or a^x for $i=3, a>0$, while $p(u)$ is either the degree of vertex u or the sum of distances from u to all other vertices of G. For any feasible combination of the parameters p,i,j,a above, the value of the corresponding Adriatic index for a molecular graph contained in `m` is obtained with

```
m.adriatic_index(p,i,j,a)
```

The list of all feasible combinations of the parameters p, i, j, a is obtained with

```
mathchem.all_adriatic()
```

Vukičević and Gašperov [87] also introduced naming convention for Adriatic indices, that is fully implemented in MathChem. Instead of using the `adriatic_index` method and write `m.adriatic_index(0, 2, 7, 0.5)`, for example, one can equivalenty use `m.max_min_rodeg_index()`. The name of the Adriatic index for a given parameter set can be obtained with

```
mathchem.adriatic_name(p,i,j,a)
```

Table 3.6 lists the names of twenty discrete Adriatic indices that are identified as useful for QSAR/QSPR studies in [87].

The use of these functions may be illustrated with the following code:

```
import mathchem
m = mathchem.Mol("GhCH?_")
for x in mathchem.all_adriatic():
    print mathchem.adriatic_name(*x), m.adriatic_index(*x)
```

Here, `mathchem.all_adriatic()` returns the list of all feasible parameter sets (represented as fourtuples), and the `for` command iterates `x` through this list. The construction `*x` "opens up" each fourtuple into four separate arguments, which are then used as arguments to MathChem functions. The result are the names and the values of all 148 discrete Adriatic indices calculated for the molecular graph in `m`:

```
Randic type lordeg 2.84389164788
Randic type lodeg 2.72994898165
Randic type losdeg 2.61649032574
sum lordeg 9.61910088844
sum lodeg 9.36426245425
...
```

3.5 More elaborate examples of MathChem use

We give here a few more elaborate examples of MathChem use, which illustrate both the power and the simplicity of the package, as well as the possibilities offered by joint use of MathChem with NetworkX or Sage.

3.5.1 Examples of integration with NetworkX and Sage

NetworkX [61] is a popular Python package aimed for creation, manipulation, and study of the structure, dynamics, and functions of complex networks. Sage [78] is a powerful open-source mathematics software system, aimed as a free alternative to commercial systems like Mathematica or MATLAB, which has an interactive web-based user interface and contains more than 100 mathematical packages, including NetworkX.

MathChem contains two functions which translate the molecular graph contained in a *Mol* object m into the graph formats used by Sage (g) and NetworkX (h), respectively:

```
g = m.sage_graph()
h = m.NX_graph()
```

On the other hand, if a graph g is provided in Sage format, the corresponding *Mol* object m may be constructed by using the function graph6_string() from Sage's Graph class:

```
m = mathchem.Mol(g.graph6_string())
```

Next, if a graph h is provided in NetworkX format, the corresponding *Mol* object m may be constructed by using the function edges() from NetworkX:

```
m = mathchem.Mol()
m.read_edgelist(h.edges())
```

For example, to list all independent sets of a molecular graph, one can use functions find_cliques and complement from NetworkX:

```
import mathchem
import networkx
m = mathchem.Mol("GhCH?_")
g = m.NX_graph()
list(networkx.find_cliques(networkx.complement(g)))
```

which returns

```
[[0, 4, 7, 2],
 [0, 4, 7, 6],
 [0, 5, 2, 7],
 [0, 5, 6, 3],
 [0, 5, 6, 7],
 [1, 6, 4, 7],
 [1, 6, 5, 3],
 [1, 6, 5, 7]]
```

In the next example, to find the matching polynomial of a molecular graph, one can use function `matching_polynomial` from Sage:

```
import mathchem
m = mathchem.Mol("GhCH?_")
g = m.sage_graph()
g.matching_polynomial()
```

which returns

```
x^8 - 7*x^6 + 13*x^4 - 7*x^2 + 1
```

Sage can also be used for visualization of molecular graphs:

```
import mathchem
m = mathchem.Mol("GhCH?_")
g = m.sage_graph()
g.show()
```

Resulting drawing is shown in Fig. 3.1.

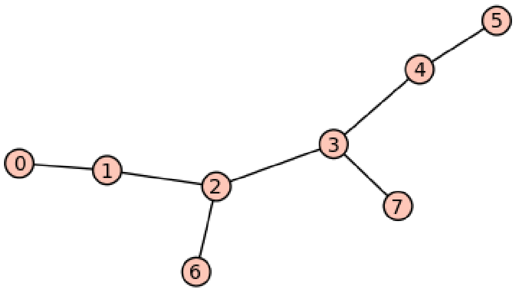


Figure 3.1: Molecular graph can be visualized with show() method from Sage.

MathChem can also be used to calculate topological indices for graphs created in Sage. The following example calculates Randić index of a random tree with 10 vertices:

```
import mathchem
g = graphs.RandomTree(10)
m = mathchem.Mol(g.graph6_string())
m.randic_index()
```

3.5.2 Correlation examples

We now give examples of creating bar charts, scatter plots and histograms for a list of molecular graphs. For this purpose, we use MathChem from within Sage (see Subsection 2.2 for installing MathChem as a Sage module). As a test bed, we use compounds from the NCI online database with NSC number from 1 to 5000.

Start Sage and import MathChem:

```
sage: import mathchem
```

To import all compounds with NSC number from 1 to 5000 in the NCI online database to the list `mols`, use:

```
sage: mols = mathchem.read_from_NCI_by_NSC("1-5000")
sage: len(mols)
4935
```

The actual number of retrieved records is 4935, because the NCI database has gaps in NSC numbers. The following code filters the list `mols` for connected molecular graphs:

```
sage: mols_c = filter(lambda m: m.is_connected(), mols)
sage: len(mols_c)
4800
```

Python's `filter` function iterates through every item of the list `mols`, checks whether it is a connected graph and if so appends the item to the new list `mols_c`. In the code above we also used Python's lambda-construct `lambda m: m.is_connected()` which allows to create small functions on the fly and make code shorter.

Now we calculate Randić index for every item of the list `mols_c` and put calculated values into a new list `ri`:

```
sage: ri = [m.randic_index() for m in mols_c]
```

The minimum and maximum entries of the list are obtained with functions `min` and `max`:

```
sage: print min(ri), max(ri)
1.0 42.1016302944
```

The bar chart of values in the list `ri` can be obtained with Sage's function `bar_chart`:

```
sage: bar_chart(ri)
```

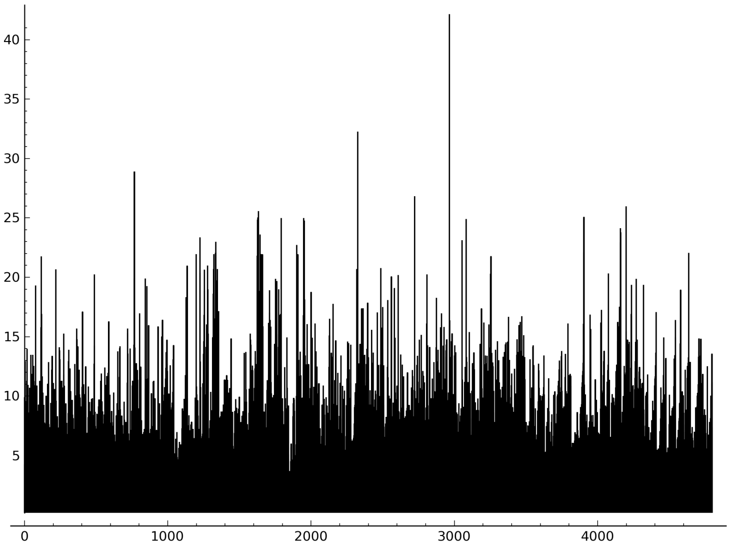


Figure 3.2: Bar chart of Randić index for connected NCI compounds with NSC numbers from 1 to 5000.

Resulting bar chart is shown in Fig. 3.2.

We can now explore correlation of Randić index with Harary index for these compounds. Let us calculate the Harary index as well:

```
sage: hi = [m.harary_index() for m in mols_c]
```

To get the scatter plot of values from the lists `ri` and `hi`, we use Sage's `scatter_plot` function. This function takes the list of pairs of values as its single argument. We can use Python's `zip` function to make such list of pairs out of two given lists:

```
sage: scatter_plot(zip(ri,hi))
```

Resulting scatter plot is shown in Fig. 3.3.

To get the histogram showing the distribution of orders of molecular graphs contained in the list `mols_c`, we first create the list containing the order of these graphs:

```
sage: orders = [m.order() for m in mols_c]
```

Then we create a new list that will contain number of molecular graphs for each different order. This list has to have one more element than the maximum order (as the list elements are indexed from 0) and the list elements are initially set to zeros:

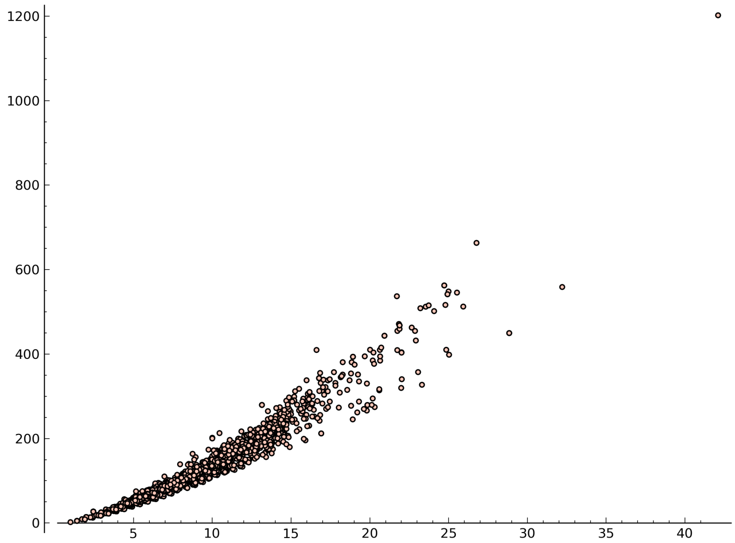


Figure 3.3: Scatter plot of Randić index versus the Harary index for connected NCI compounds with NSC numbers from 1 to 5000.

```
sage: hist_data = [0]*(max(orders)+1)
```

We now iterate through the list of orders and count appearances of each order:

```
sage: for i in orders: hist_data[i] += 1
```

To get the histogram of `hist_data`, use:

```
sage: bar_chart(hist_data).show(figsize=[10,5])
```

The part `.show(figsize=[10,5])` sets the size of the bar chart in inches. Resulting histogram is shown in Fig. 3.4.

3.5.3 Creating an interactive widget

Sage can be run either in console mode or in notebook mode, where the latter provides a flexible way for programming using its graphical user interface. While all examples given so far can work both in console and in notebook mode, the following example will specifically illustrate possibilities of Sage's notebook mode, by creating an interactive tool where one can select two topological indices from drop-down menus, while the tool automatically builds a scatter plot and calculates correlation between the indices.

Launch Sage as usual and type:

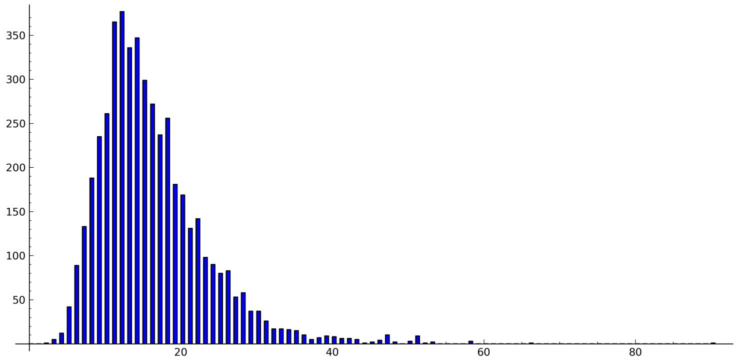


Figure 3.4: Histogram of orders of molecular graphs for connected NCI compounds with NSC numbers from 1 to 5000.

```
sage: notebook()
```

This command sets up a local web server and opens default browser with a main page located at `http://localhost:8080/home/admin/`. This page enables creation of separate worksheets, which serve to write and evaluate Sage programs.

Once again, we will use connected molecular graphs corresponding to compounds with NSC number from 1 to 5000 in the NCI online database: (see Section 6.2 for more detailed explanation):

```
sage: import mathchem
sage: mols = mathchem.read_from_NCI_by_NSC("1-5000")
sage: mols_c = filter(lambda m: m.is_connected(), mols)
```

The next command defines a list of topological indices to appear in drop-down menus:

```
sage: methods = ["order", "diameter", "energy", "incidence_energy",
      "randic_index", "zagreb_m1_index", "zagreb_m2_index",
      "eccentric_connectivity_index", "atom_bond_connectivity_index",
      "estrada_index", "eccentric_distance_sum", "reverse_degree_distance",
      "molecular_topological_index", "degree_distance", "balaban_j_index",
      "kirchhoff_index", "wiener_index", "harary_index", "LEL",
      "reverse_wiener_index", "hyper_wiener_index", "terminal_wiener_index",
      "randic_type_lodeg_index", "randic_type_sdi_index",
      "randic_type_hadi_index"]
```

Next we include ScyPy statistical library in order to use its linear regression methods:

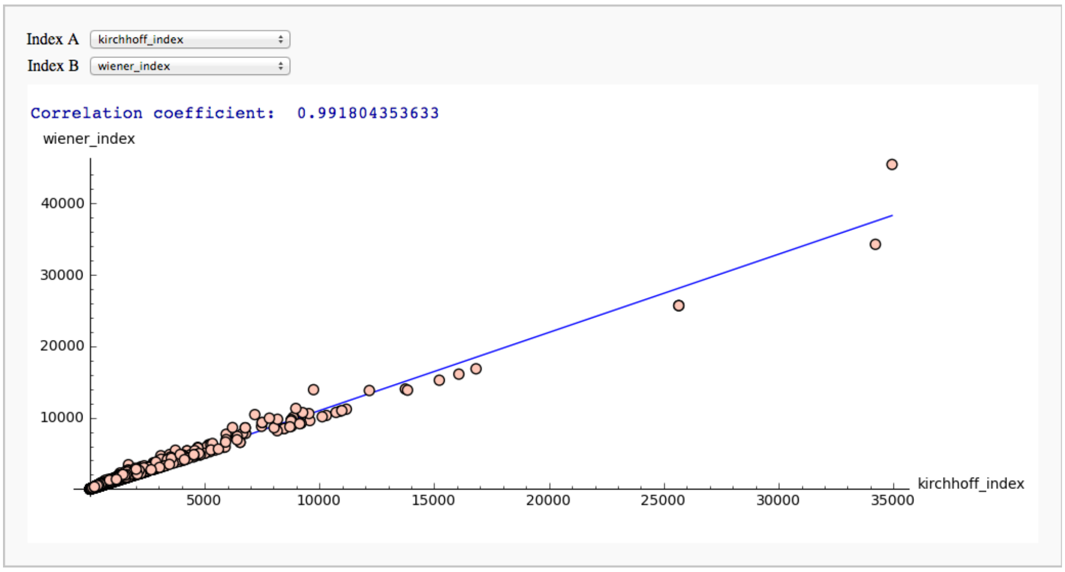
```
sage: import scipy.stats as stats
```

We are now ready to write an interactive tool:

```
@interact
def index_correlations(index_A = selector(methods,label="Index A"), \
    index_B = selector(methods,label="Index B")):
    data_A = [getattr(m, index_A)() for m in mols_c]
    data_B = [getattr(m, index_B)() for m in mols_c]
    data = zip(data_A, data_B)
    slope, inter, r, ttprob, stderr = stats.linregress(data)

    print "Correlation coefficient: ", r
    canvas = scatter_plot(data) + plot(slope*x+inter,min(data_A),max(data_A))
    canvas.show(figsize=[10,4], axes_labels=[index_A, index_B])
```

The function above is called automatically whenever its arguments are changing their values. Its arguments `index_A` and `index_B` are defined as visual selectors of all methods appearing in `methods`. Python construct `getattr(m, index_A)()` calls a method whose name is contained in `index_A` of the *Mol* object `m`. This is used in `for` loop, which then results in the values of selected topological index to be put in the list `data_A`, respectively `data_B`. The two list are then “zipped” to produce a list of pairs, after which linear regression is applied, with the results—scatter plot and the best fit line—visually presented in `canvas`. The look of the resulting tool is presented in Fig. 3.5.



Index A kirchhoff_index

Index B wiener_index

```
Correlation coefficient:  0.991804353633
```

Figure 3.5: Interactive widget.

3.6 Conclusion

We have described MathChem, a Python package for calculating topological indices, and provided examples of its joint use with other well known open-source products such as Sage or NetworkX. MathChem package does not solve problems out-of-the-box, but instead it provides a flexible and easily expandable framework for computational research in mathematical chemistry. All contributions or requests for implementation are welcome through MathChem's Github homepage: `https://github.com/hamster3d/Mathchem-package`.

The initial conjectures in chapters 4, 5, 6 were obtained with help of MathChem. The other two examples of MathChem usage are described in Chapter 7.

Chapter 4

Inverse Sum Indeg Index

Molecular descriptors, results of functions mapping molecule's chemical information into a number [82], have found applications in modeling many physicochemical properties in QSAR and QSPR studies [17, 43]. A particularly common type of molecular descriptors are those that are defined as functions of the structure of the underlying molecular graph, such as the Wiener index [90], the Zagreb indices [36], the Randić index [72] or the Balaban J-index [2]. Damir Vukičević and Marija Gašperov [87] observed that many of these descriptors are defined simply as the sum of individual bond contributions (actually, all previously mentioned descriptors except for the Wiener index). In order to study whether there are other possibly significant descriptors of this form, they have introduced a class of discrete Adriatic indices in [87], generally defined as

$$Adr(G) = \sum_{uv \in E(G)} f(g(p_u), g(p_v)),$$

where G is the molecular graph, $E(G)$ is the set of its bonds, p_u is either the degree d_u of a vertex $u \in V(G)$ or the sum D_u of distances from u to all other vertices in $V(G)$, while f and g are suitably chosen functions.

Among the 148 discrete Adriatic indices studied in [87], whose predictive properties were evaluated against the benchmark datasets of the International Academy of Mathematical Chemistry [59], 20 indices were selected as significant predictors of physicochemical properties. Graph-theoretical properties of one of these indices, the max-min rodeg index, were studied previously in [85]. In order to study graph-theoretical properties of other discrete Adriatic indices, we have implemented all of them in MathChem, an open source Python package for calculating topological indices [83, 84], and obtained a number of conjectures about their extremal values in several graph classes.

We study here the properties of the inverse sum indeg index, the descriptor that was selected in [87] as a significant predictor of total surface area of octane isomers and for which the extremal graphs obtained with the help of MathChem have a particularly simple and elegant structure. The inverse sum indeg index is defined as

$$ISI(G) = \sum_{uv \in E(G)} \frac{1}{\frac{1}{d_u} + \frac{1}{d_v}} = \sum_{uv \in E(G)} \frac{d_u d_v}{d_u + d_v}. \tag{4.1}$$

After establishing basic properties of the inverse sum indeg index in Section 4.1, in the following sections we determine extremal values and extremal graphs of the inverse sum indeg index in several classes of graphs (with given number of vertices): the class of all connected graphs, the class of all trees, the class of all chemical graphs, the class of all chemical trees, the class of all graphs with given maximum degree, the class of all graphs with given minimum degree, the class of all graphs with given number of pendant vertices, and the class of all trees with given number of pendant vertices. We give some complete and some partial solutions for these problems.

4.1 Preliminaries

We assume in the sequel that the graph G does not contain isolated vertices, although we allow the possibility that G may have several connected components.

The basic property of the inverse sum indeg index is that it is monotone with respect to addition of edges.

Lemma 4.1.1 *Let u and v be two nonadjacent vertices of graph G, and let $G+uv$ the graph obtained from G by adding edge uv to it. Then*

$$ISI(G) < ISI(G+uv).$$

Proof. Note first that for $x, y > 0$

$$\frac{(x+1)y}{(x+1)+y} - \frac{xy}{x+y} = \frac{y^2}{(x+y)(x+y+1)} > 0.$$

Now, let $u_1, \ldots, u_k$ be the neighbors of u in G for $k = d_u$, and let $v_1, \ldots, v_l$ be the neighbors of v for $l = d_v$. Then

$$\begin{aligned} ISI(G+uv) - ISI(G) &= \frac{(d_u+1)(d_v+1)}{d_u+d_v+2} \\ &+ \sum_{i=1}^{k} \left[\frac{(d_u+1)d_{u_i}}{(d_u+1)+d_{u_i}} - \frac{d_u\, d_{u_i}}{d_u+d_{u_i}} \right] \\ &+ \sum_{j=1}^{l} \left[\frac{(d_v+1)d_{v_j}}{(d_v+1)+d_{v_j}} - \frac{d_v\, d_{v_j}}{d_v+d_{v_j}} \right] \\ &> 0, \end{aligned}$$

where the first summand above is the (positive) contribution of edge uv in $G+uv$. ∎

The following corollary is an immediate consequence of previous lemma.

Corollary 4.1.2 *If a connected graph G itself is not a tree, then $ISI(T) < ISI(G)$ holds for any spanning tree T of G.*

The next result establishes useful bound on the inverse sum indeg index in terms of the numbers of vertices and edges of a graph.

Theorem 4.1.3 *If G is a graph with n vertices and m edges, then*

$$ISI(G) \geq 2m - n,$$

with equality if and only if G is a union of cycles.

Proof. Since the degree of any vertex $u \in V(G)$ is positive, we have

$$n = \sum_{u \in V(G)} 1 = \sum_{u \in V(G)} d_u \cdot \frac{1}{d_u} = \sum_{uv \in E(G)} \frac{1}{d_u} + \frac{1}{d_v} = \sum_{uv \in E(G)} \frac{d_u + d_v}{d_u d_v}.$$

Then

$$ISI(G) + n = \sum_{uv \in E(G)} \frac{d_u d_v}{d_u + d_v} + \frac{d_u + d_v}{d_u d_v}. \tag{4.2}$$

Since $x + \frac{1}{x} \geq 2$ for any positive x, with equality if and only if $x = 1$, we have from (4.2), by substituting x with $\frac{d_u d_v}{d_u + d_v}$ for each edge $uv \in E(G)$, that

$$ISI(G) + n \geq \sum_{uv \in E(G)} 2 = 2m.$$

Equality holds if and only if $d_u d_v = d_u + d_v$ for each edge $uv \in E(G)$ if and only if $d_u = 2$ for each $u \in V(G)$, *i.e.*, if and only if G is a union of cycles. ■

The next theorem gives a relation between the inverse sum indeg index and the first Zagreb index $M_1(G) = \sum_{u \in V(G)} d_u^2$.

Theorem 4.1.4 *For any graph G holds*

$$ISI(G) \leq \frac{M_1(G)}{4},$$

with equality if and only if G is a union of regular graphs.

Proof. Let $uv \in E(G)$ be an arbitrary edge in G (so that $d_u, d_v \neq 0$). From

$$0 \leq (d_u - d_v)^2$$

we get, after adding $4d_u d_v$ to both sides, that

$$4d_u d_v \leq (d_u + d_v)^2,$$

so that after division with $4(d_u + d_v)$, it follows that

$$\frac{d_u d_v}{d_u + d_v} \leq \frac{d_u}{4} + \frac{d_v}{4}.$$

Now we have

$$ISI(G) = \sum_{uv \in E(G)} \frac{d_u d_v}{d_u + d_v} \leq \sum_{uv \in E(G)} \frac{d_u}{4} + \frac{d_v}{4} = \sum_{u \in V(G)} \frac{d_u^2}{4} = \frac{M_1(G)}{4}.$$

Equality holds if and only if $d_u = d_v$ for each edge $uv \in E(G)$, *i.e.*, if and only if each component of G is a regular graph. ■

4.2 Minimum values of the inverse sum indeg index

Since the minimum value of the inverse sum indeg index is, by Corollary 4.1.2, necessarily obtained by some tree, we put focus first on the sets of trees.

Let S_n be the star on n vertices, a tree consisting of a central vertex adjacent to $n-1$ leaves.

Theorem 4.2.1 *If T is a tree with n vertices, then*

$$ISI(T) \geq n + \frac{1}{n} - 2,$$

with equality if and only if T is isomorphic to S_n.

Proof. Since $ISI(S_n) = (n-1)\frac{(n-1)\cdot 1}{(n-1)+1} = n + \frac{1}{n} - 2$, this theorem states that $ISI(S_n) \leq ISI(T)$ for any tree T, with equality if and only if $T \cong S_n$. The only tree with one, two or three vertices is exactly the star, so that the statement follows trivially for $n \leq 3$.

The proof for larger trees is by induction on n. The basis of induction for $n \leq 3$ is proved above, so that we may make an inductive hypothesis that, for some $n \geq 4$, $ISI(S_{n-1}) \leq ISI(T')$ holds for each $(n-1)$-vertex tree T' with equality if and only if $T' \cong S_{n-1}$.

Now, suppose that $T \ncong S_n$ is an arbitrary tree with n vertices. Let u be a pendant vertex of T, with v as its only neighbor. Since T has at least four vertices, v cannot be a leaf, so that $d_v \geq 2$. Further, since T is not isomophic to a star, we have that there exists at least one neighbor w of v in T with $d_w \geq 2$. Let $z_1, \ldots, z_{d_v-2}$ be the remaining neighbors of v in T, other than u and w.

Let $T' = T - u$. Since the edges of T, which are not incident to v, contribute equally to both $ISI(T)$ and $ISI(T')$, we have that

$$\begin{aligned} ISI(T) - ISI(T') &= \frac{d_v \cdot 1}{d_v + 1} + \left(\frac{d_w d_v}{d_w + d_v} - \frac{d_w(d_v - 1)}{d_w + d_v - 1}\right) \\ &+ \sum_{i=1}^{d_v-2} \left(\frac{d_{z_i} d_v}{d_{z_i} + d_v} - \frac{d_{z_i}(d_v - 1)}{d_{z_i} + (d_v - 1)}\right) \\ &= \frac{d_v \cdot 1}{d_v + 1} + f(d_w) + \sum_{i=1}^{d_v-2} f(d_{z_i}), \end{aligned}$$

where

$$f(x) = \frac{x d_v}{x + d_v} - \frac{x(d_v - 1)}{x + d_v - 1}.$$

The first derivative of $f(x)$ is equal to

$$f'(x) = \frac{d_v^2}{(x + d_v)^2} - \frac{(d_v - 1)^2}{(x + d_v - 1)^2} = \frac{x^2(2d_v - 1) + 2x d_v(d_v - 1)}{(x + d_v)^2(x + d_v - 1)^2}.$$

Since for $x > 0$ we have that $x^2(2d_v - 1) + 2x d_v(d_v - 1) > 0$, it follows that $f'(x) > 0$ and the function f is strictly increasing for $x > 0$. Together with $d_w \geq 2$ and $d_{z_i} \geq 1$, this implies that

$$\begin{aligned} ISI(T) - ISI(T') &\geq \frac{d_v \cdot 1}{d_v + 1} + f(2) + (d_v - 2)f(1) \\ &= 1 + \frac{2(d_v - 2)}{d_v(d_v + 1)(d_v + 2)}. \end{aligned} \tag{4.3}$$

Since $d_v \geq 2$, we have that

$$ISI(T) - ISI(T') \geq 1 > 1 - \frac{1}{n(n-1)} = ISI(S_n) - ISI(S_{n-1}),$$

from where it follows that

$$ISI(T) - ISI(S_n) > ISI(T') - ISI(S_{n-1}) \geq 0,$$

due to the inductive hypothesis.

This concludes the proof by induction and also shows that $ISI(T) = ISI(S_n)$ holds if and only if $T \cong S_n$. ■

Let us recall that a tree (a graph) is called *chemical* if the degree of each of its vertices is at most four.

Theorem 4.2.2 *If T is a chemical tree with n vertices, then*

$$ISI(T) \geq \begin{cases} n + \frac{1}{n} - 2 & \text{if } n \leq 5, \\ n - \frac{5}{3} & \text{if } n \geq 6, \end{cases}$$

with equality if and only if T is isomorphic to the star S_n if $n \leq 5$ and to the path P_n if $n \geq 6$.

Proof. For $n \leq 5$, the star S_n is a chemical tree and the result follows from Theorem 4.2.1.

We will prove the statement for larger values of n by induction. The basis of induction for $n \leq 5$ is proved above.

Let the statement of the theorem be proved for all chemical trees with less than n vertices for some $n \geq 6$, and let T be a chemical tree with n vertices.

Suppose first that T contains a vertex u of degree four. Then at least one neighbor of u has degree at least two. Denote the neighbors of u by u_1, u_2, u_3, u_4 and suppose, without loss of generality, that $d_{u_4} \geq 2$. Form trees T' and T'' from T by splitting the vertex u into two new, nonadjacent vertices u' and u'', such that u' is adjacent to u_1 and u_2 in T', while u'' is adjacent to u_3 and u_4 in T''. The edges of T that are not incident to u contribute equally to $ISI(T)$ and $ISI(T') + ISI(T'')$, so that

$$ISI(T) = ISI(T') + ISI(T'') + \sum_{i=1}^{4} \frac{4 \cdot d_{u_i}}{d_{u_i} + 4} - \frac{2 \cdot d_{u_i}}{d_{u_i} + 2}.$$

The function $p(x) = \frac{4x}{x+4} - \frac{2x}{x+2}$ is strictly increasing for $x > 0$, as $p'(x) = \frac{12x^2+32x}{(x+2)^2(x+4)^2} > 0$, so that from $d_{u_1}, d_{u_2}, d_{u_3} \geq 1$ and $d_{u_4} \geq 2$ it follows that

$$ISI(T) \geq ISI(T') + ISI(T'') + 3p(1) + p(2) = ISI(T') + ISI(T'') + \frac{11}{15}. \tag{4.4}$$

Let T' have n' vertices and T'' have n'' vertices. Then $3 \leq n', n'' < n$, so that the inductive hypothesis holds for both T' and T''. If $n' \geq 6$, then $ISI(T') \geq n' - \frac{5}{3}$ by the inductive hypothesis. If $n' \leq 5$, then T' cannot be a star, as it contains vertex u' of degree two, so that T' is either P_3, P_4, P_5 or a unique tree with degree sequence $3, 2, 1, 1, 1$, and $ISI(T') \geq n' - \frac{5}{3}$ for each of these cases as well. Inequality $ISI(T'') \geq n'' - \frac{5}{3}$ follows analogously. Taking into account that $n' + n'' = n + 1$, as the vertex u was split into two new vertices, (4.4) implies that

$$ISI(T) \geq \left(n' - \frac{5}{3}\right) + \left(n'' - \frac{5}{3}\right) + \frac{11}{15} = n - \frac{8}{5}.$$

Hence, if T contains a vertex of degree four, then $ISI(T) \geq n - \frac{8}{5} > n - \frac{5}{3}$.

Suppose next that T contains a vertex u of degree three. Then at least one neighbor of u has degree at least two. Denote the neighbors of u by u_1, u_2, u_3 and suppose, without loss of generality, that $d_{u_3} \geq 2$. Form trees T' and T'' from T by splitting the vertex u into two new, nonadjacent vertices u' and u'', such that u' is adjacent to u_1 and u_2 in T', while u'' is adjacent to u_3 in T''. The edges of T that are not incident to u contribute equally to $ISI(T)$ and $ISI(T') + ISI(T'')$, so that

$$ISI(T) = ISI(T') + ISI(T'') + \sum_{i=1}^{2} \left(\frac{3d_{u_i}}{d_{u_i} + 3} - \frac{2d_{u_i}}{d_{u_i} + 2} \right) + \left(\frac{3d_{u_3}}{d_{u_3} + 3} - \frac{d_{u_3}}{d_{u_3} + 1} \right). \tag{4.5}$$

The functions $q(x) = \frac{3x}{x+3} - \frac{2x}{x+2}$ and $r(x) = \frac{3x}{x+3} - \frac{x}{x+1}$ are strictly increasing for $x > 0$, as $q'(x) = \frac{5x^2+12x}{(x+3)^2(x+2)^2} > 0$ and $r'(x) = \frac{8x^2+12x}{(x+3)^2(x+1)^2} > 0$, so that from $d_{u_1}, d_{u_2} \geq 1$ and $d_{u_3} \geq 2$ we have that

$$ISI(T) \geq ISI(T') + ISI(T'') + 2q(1) + r(2) = ISI(T') + ISI(T'') + \frac{7}{10}. \tag{4.6}$$

Let T' have n' vertices and T'' have n'' vertices. Then $3 \leq n', n'' < n$, so that the inductive hypothesis holds for both T' and T''. Inequality $ISI(T') \geq n' - \frac{5}{3}$ follows as in the earlier case $d_u = 4$, as T' contains a vertex u' of degree two. If $n'' \geq 6$, then $ISI(T'') \geq n'' - \frac{5}{3}$ by the inductive hypothesis, so that (4.6) implies that

$$ISI(T) \geq \left(n' - \frac{5}{3}\right) + \left(n'' - \frac{5}{3}\right) + \frac{7}{10} = n - \frac{49}{30}. \tag{4.7}$$

For $n'' \leq 5$, if T'' is one of P_3, P_4, P_5 and a unique tree with degree sequence $3, 2, 1, 1, 1$, then $ISI(T'') \geq n'' - \frac{5}{3}$ and inequality (4.7) holds as well. However, if T'' is one of S_4 and S_5 (in which case $n'' = d_{u_3} + 1$ and $T'' \cong S_{d_{u_3}+1}$), then (4.5) implies that

$$\begin{aligned} ISI(T) &\geq ISI(T') + ISI(T'') + 2q(1) + r(d_{u_3}) \\ &\geq \left(n' - \frac{5}{3}\right) + \left(n'' + \frac{1}{n''} - 2\right) + \frac{1}{6} + r(d_{u_3}) \\ &= n - \frac{5}{2} + \frac{2d_{u_3}^2 + d_{u_3} + 3}{(d_{u_3} + 1)(d_{u_3} + 3)}. \end{aligned}$$

The function $s(x) = \frac{2x^2+x+3}{(x+1)(x+3)}$ is strictly increasing for $x > \frac{-3+6\sqrt{2}}{7}$, as $s'(x) = \frac{7x^2+6x-9}{(x+1)^2(x+3)^2} > 0$, so that from $d_{u_3} \geq 2$ we have that

$$ISI(T) \geq n - \frac{5}{2} + s(2) = n - \frac{49}{30}.$$

Hence, if T contains a vertex of degree three, then $ISI(T) \geq n - \frac{49}{30} > n - \frac{5}{3}$.

Finally, if T does not contain vertices of degrees three or four, then T is isomorphic to the path P_n and $ISI(T) = n - \frac{5}{3}$, finishing the proof of the inductive step. ■

Let $B_{n,k}$ be a tree obtained from the star S_{k+1} by identifying one of its leaves with a leaf of the path P_{n-k}. The tree $B_{n,k}$ has n vertices, among which there are k pendant vertices, and it is usually named the *broom* in the literature.

Theorem 4.2.3 *If T is a tree with n vertices and maximum vertex degree $\Delta \geq 2$, then*

$$ISI(T) \geq ISI(B_{n,\Delta}) = \begin{cases} n + \frac{1}{n} - 2, & \text{if } \Delta = n-1, \\ n - \frac{4}{3} - \frac{2\Delta}{(\Delta+1)(\Delta+2)}, & \text{if } \Delta < n-1, \end{cases} \tag{4.8}$$

with equality if and only if T is isomorphic to the broom $B_{n,\Delta}$.

Proof. We prove inequality (4.8) by double induction, first on the value of $\Delta \geq 2$ (the outer induction), and then on the value of $n \geq \Delta + 1$ (the inner induction).

The basis of the outer induction is $\Delta = 2$. As the path P_n is the only tree with maximum vertex degree two and $P_n = B_{n,2}$ by definition, inequality (4.8) holds for $\Delta = 2$ and all $n \geq \Delta + 1$.

Let us, therefore, make the outer inductive hypothesis that inequality (4.8) holds for all trees with maximum vertex degree $\Delta - 1$, for some $\Delta \geq 3$ (with equality for and only for the brooms $B_{n,\Delta-1}$). We will prove that inequality (4.8) then holds for all trees with maximum vertex degree Δ by the inner induction on $n \geq \Delta + 1$.

The basis of the inner induction are the cases $n = \Delta + 1$ and $n = \Delta + 2$. Unique tree with $\Delta + 1$ vertices and maximum vertex degree Δ is the star $S_{\Delta+1}$ and since $S_{\Delta+1} = B_{\Delta+1,\Delta}$ by definition, inequality (4.8) holds for $n = \Delta + 1$. Similarly, unique tree with $\Delta + 2$ vertices and maximum vertex degree Δ is the broom $B_{\Delta+2,\Delta}$, so that inequality (4.8) holds for $n = \Delta + 2$ as well.

Let us, therefore, make the inner inductive hypothesis that inequality (4.8) holds for all trees with $n - 1$ vertices with maximum vertex degree Δ, for some $n > \Delta + 2$ (with equality for and only for the broom $B_{n-1,\Delta}$).

Let T be a tree with n vertices and maximum vertex degree Δ. Since $n > \Delta + 2$, T cannot be a star and, therefore, there exists a pendant vertex u in T such that $T' = T - u$ also has maximum vertex degree Δ. Then the inner inductive hypothesis holds for T' and since $d_v \geq 2$ for the unique neighbor v of u, we have from (4.3)

$$\begin{aligned} ISI(T) &\geq ISI(T') + 1 + \frac{2(d_v - 2)}{d_v(d_v + 1)(d_v + 2)} \\ &\geq (n-1) - \frac{4}{3} - \frac{2\Delta}{(\Delta+1)(\Delta+2)} + 1. \end{aligned} \tag{4.9}$$

Equality in this chain of inequalities is attained only if T' satisfies equality in (4.8) and $d_v = 2$. The inner inductive hypothesis implies that T' is then isomorphic to the broom $B_{n-1,\Delta}$, while $d_v = 2$ (which holds for v as a vertex of T) implies that v is one of the leaves of $B_{n-1,\Delta}$. Closer inspection of the argument used to derive (4.3) further reveals that equality may hold in (4.9) only if the other neighbor of v in T has degree two, *i.e.*, if T itself is the broom $B_{n,\Delta}$. ■

With small modifications of the proof of Theorem 4.2.3, we get the following

Theorem 4.2.4 *If T is a tree with n vertices, among which there are $p \geq 2$ pendant vertices, then*

$$ISI(T) \geq ISI(B_{n,p}) = \begin{cases} n + \frac{1}{n} - 2, & \text{if } p = n-1, \\ n - \frac{4}{3} - \frac{2p}{(p+1)(p+2)}, & \text{if } p < n-1. \end{cases} \tag{4.10}$$

PROOF. We prove inequality (4.10) by double induction, first on the value of $p \geq 2$ (the outer induction), and then on the value of $n \geq p+1$ (the inner induction).

The basis of the outer induction is $p = 2$. As the path P_n is the only tree with exactly two leaves and $P_n = B_{n,2}$ by definition, inequality (4.10) holds for $p = 2$ and all $n \geq p+1$.

Let us, therefore, make the outer inductive hypothesis that inequality (4.10) holds for all trees with $p-1$ pendant vertices, for some $p \geq 3$. We will prove that inequality (4.10) then holds for all trees with p pendant vertices by the inner induction on $n \geq p+1$.

The basis of the inner induction are the cases $n = p+1$ and $n = p+2$. Unique tree with $p+1$ vertices, of which p are pendant vertices, is the star S_{p+1} and since $S_{p+1} = B_{p+1,p}$ by definition, inequality (4.10) holds for $n = p+1$. Similarly, unique tree with $p+2$ vertices, of which p are pendant vertices, is the broom $B_{p+2,p}$, so that inequality (4.10) holds for $n = p+2$ as well.

Let us, therefore, make the inner inductive hypothesis that inequality (4.10) holds for all trees with $n-1$ vertices, of which p are pendant vertices, for some $n > p+2$.

Let T be a tree with n vertices, of which p are pendant vertices. Let u be an arbitrary pendant vertex of T, with v as its only neighbor. Here $2 \leq d_v \leq p$: firstly, $2 \leq d_v$ because T has more than two vertices and, secondly, $d_v \leq p$ because each of d_v subtrees of T obtained by removing v contains at least one pendant vertex of T. Now, let $T' = T - u$. Since $n > p+2$, T cannot be a star, so that from (4.3):

$$ISI(T) \geq ISI(T') + 1 + \frac{2(d_v - 2)}{d_v(d_v+1)(d_v+2)}.$$

Tree T' has $n-1$ vertices, of which either p or $p-1$ are pendant vertices, depending on whether $d_v = 2$ or $d_v > 2$, respectively.

If $d_v = 2$, then T' has p pendant vertices and, by the inductive hypothesis of the inner induction, $ISI(T') \geq (n-1) - \frac{4}{3} - \frac{2p}{(p+1)(p+2)}$, so that

$$ISI(T) \geq n - \frac{4}{3} - \frac{2p}{(p+1)(p+2)}.$$

If $3 \le d_v \le p$, then T' has $p-1$ pendant vertices and, by the inductive hypothesis of the outer induction, $ISI(T') \ge (n-1) - \frac{4}{3} - \frac{2(p-1)}{p(p+1)}$, so that

$$ISI(T) \ge n - \frac{4}{3} - \frac{2(p-1)}{p(p+1)} + \frac{2(d_v-2)}{d_v(d_v+1)(d_v+2)}. \tag{4.11}$$

The function $f(x) = \frac{2(x-2)}{x(x+1)(x+2)}$ has the first derivative $f'(x) = \frac{-2(2x^3-3x^2-12x-4)}{x^2(x+1)^2(x+2)^2}$. The roots of $2x^3-3x^2-12x-4$ are, approximately, 3.42334, -0.37808 and -1.54526, so that $f'(x) < 0$ and $f(x)$ is strictly decreasing for $x > 3.42335$. Since also $f(3) = \frac{1}{30} = f(4)$, from $3 \le d_v \le p$ we conclude that $f(d_v) \ge f(p)$ (with equality either if $d_v = p$ or $d_v = 3, p = 4$). Therefore, from (4.11) and $f(d_v) \ge f(p)$ we have

$$ISI(T) \ge n - \frac{4}{3} - \frac{2(p-1)}{p(p+1)} + \frac{2(p-2)}{p(p+1)(p+2)} = n - \frac{4}{3} - \frac{2p}{(p+1)(p+2)}.$$

■

A careful reader will notice that the previous theorem does not fully characterize the case of equality in $ISI(T) \ge ISI(B_{n,p})$. The reason for this is that there exist trees that are not brooms and still have the same value of the inverse sum indeg index as the corresponding broom. They are obtained in the previous proof whenever $ISI(T') = ISI(B_{n-1,p-1})$ and $d_v = p \ge 3$ or $d_v = 3, p = 4$. One such example is shown in Fig. 4.1, while the remaining such graphs could be characterized with little extra effort.

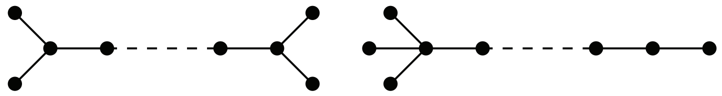


Figure 4.1: Two trees with the same value of the inverse sum indeg index, exactly one of which is a broom.

Previous results about trees may be directly extended to graphs, due to Corollary 4.1.2. Specifically, Theorems 4.2.1-4.2.4 give rise to the following

Corollary 4.2.5 *If G is a connected graph with n vertices, then*

$$ISI(G) \ge n + \frac{1}{n} - 2,$$

with equality if and only if G is isomorphic to S_n.

Corollary 4.2.6 *If G is a connected chemical graph with n vertices, then*

$$ISI(G) \ge \begin{cases} n + \frac{1}{n} - 2 & \text{if } n \le 5, \\ n - \frac{5}{3} & \text{if } n \ge 6, \end{cases}$$

with equality if and only if G is isomorphic to the star S_n if $n \le 5$ and to the path P_n if $n \ge 6$.

Corollary 4.2.7 *If G is a connected graph with n vertices and maximum vertex degree Δ, then*

$$ISI(G) \geq n - \frac{4}{3} - \frac{2\Delta}{(\Delta+1)(\Delta+2)}.$$

Corollary 4.2.8 *If G is a connected graph with n vertices, among which there are $p \geq 2$ pendant vertices, then*

$$ISI(G) \geq n - \frac{4}{3} - \frac{2p}{(p+1)(p+2)}.$$

The following theorem describes minimum value of the inverse sum indeg index for graphs with given minimum vertex degree.

Theorem 4.2.9 *Let G be a graph with n vertices and minimum vertex degree δ. Then*

$$ISI(G) \geq \begin{cases} n + \frac{1}{n} - 2, & \text{if } \delta = 1, \\ \frac{n\delta^2}{4}, & \text{if } \delta \geq 2 \text{ and } n\delta \text{ is even,} \\ \frac{n\delta^2}{4} + \frac{\delta}{2} + \frac{2\delta}{8(2\delta+1)}, & \text{if } \delta \geq 2 \text{ and } n\delta \text{ is odd.} \end{cases}$$

Proof. If $\delta = 1$, the statement follows from Corollary 4.2.5.

If $\delta \geq 2$, then the contribution of each edge $uv \in E(G)$ to $ISI(G)$ is at least $\frac{\delta}{2}$: from $d_u, d_v \geq \delta$ it follows that $(d_u - \frac{\delta}{2})(d_v - \frac{\delta}{2}) \geq \frac{\delta^2}{4}$, which is, after rearranging the terms and division by $d_u + d_v$, equivalent with

$$\frac{d_u d_v}{d_u + d_v} \geq \frac{\delta}{2}.$$

Therefore

$$ISI(G) \geq \frac{m\delta}{2} \geq \frac{n\delta^2}{4},$$

as $2m \geq n\delta$ holds in a graph in which every vertex has degree at least δ.

Note, however, that equality $2m = n\delta$ may hold only if $n\delta$ is even. If $n\delta$ is odd, then at least one vertex of G has degree at least $\delta+1$, so that there are at least $\delta+1$ edges whose contribution is at least $\frac{\delta(\delta+1)}{2\delta+1}$, and the total number of edges is at least $\frac{n\delta+1}{2}$. Hence, in such case

$$ISI(G) \geq (\delta+1)\frac{\delta(\delta+1)}{2\delta+1} + \left(\frac{n\delta+1}{2} - (\delta+1)\right)\frac{\delta}{2} = \frac{n\delta^2}{4} + \frac{\delta}{2} + \frac{2\delta}{8(2\delta+1)}.$$

■

4.3 Maximum values of the inverse sum indeg index

Our first result on the maximum value of the inverse sum indeg index is a direct corollary of Lemma 4.1.1 stating that the value of the inverse sum indeg index increases with addition of new edges. Namely, as each n-vertex graph G can be made into a complete graph by adding edges between all pairs of nonadjacent vertices, we obtain that

$$ISI(G) \le ISI(K_n) = \frac{n(n-1)^2}{4}$$

holds for each graph with n vertices.

Another useful auxiliary result is the following

Lemma 4.3.1 *Let G be a graph with m edges and the maximum degree Δ. Then*

$$ISI(G) \le \frac{m\Delta}{2},$$

with equality if and only if G is Δ-regular graph.

Proof. Since $d_u \le \Delta$ for each $u \in V(G)$, we have that

$$\frac{1}{\frac{1}{d_u}+\frac{1}{d_v}} \le \frac{1}{\frac{1}{\Delta}+\frac{1}{\Delta}} = \frac{\Delta}{2}.$$

Then, directly from (4.1), we have that

$$ISI(G) \le \frac{m\Delta}{2}.$$

The equality holds if and only if $d_u = \Delta$ for each $u \in V(G)$. ■

Previous lemma directly implies the following

Corollary 4.3.2 *If G is a graph with n vertices and maximum vertex degree Δ, then*

$$ISI(G) \le \begin{cases} \frac{n\Delta^2}{4}, & \text{if } n\Delta \text{ is even,} \\ \frac{(n-1)\Delta^2}{4} + \frac{\Delta(\Delta-1)^2}{2(2\Delta-1)}, & \text{if } n\Delta \text{ is odd.} \end{cases}$$

Proof. Since $2m$ is equal to the sum of the vertex degrees in a graph, the fact that the maximum vertex degree is Δ implies both that $m \le \frac{n\Delta}{2}$ and that the ISI contribution of each edge is at most $\frac{\Delta}{2}$. Then

$$ISI(G) \le \frac{m\Delta}{2} \le \frac{n\Delta^2}{4}.$$

Equality above is attained only if $m = \frac{n\Delta}{2}$ and G is Δ-regular graph, which is possible only if $n\Delta$ is even.

If $n\Delta$ is odd, then $m \le \frac{n\Delta-1}{2}$ and at least one vertex of G has degree at most $\Delta-1$, while the remaining vertices have degrees at most Δ. Hence, at most $\frac{(n-1)\Delta}{2}$ edges have the ISI contribution equal to $\frac{\Delta}{2}$, while the remaining $m - \frac{(n-1)\Delta}{2}$ edges have the ISI contribution at most $\frac{\Delta(\Delta-1)}{2\Delta-1}$. In total,

$$\begin{aligned} ISI(G) &\le \frac{(n-1)\Delta^2}{4} + \left(m - \frac{(n-1)\Delta}{2}\right)\frac{\Delta(\Delta-1)}{2\Delta-1} \\ &\le \frac{(n-1)\Delta^2}{4} + \frac{\Delta-1}{2}\frac{\Delta(\Delta-1)}{2\Delta-1}. \end{aligned}$$

■

When applied to chemical graphs, which have maximum vertex degree Δ at most four, previous corollary further implies

Corollary 4.3.3 *If G is a chemical graph with n vertices, then*

$$ISI(G) \le \begin{cases} \frac{n(n-1)^2}{4}, & \textit{if } n \le 4, \\ 4n & \textit{if } n \ge 5. \end{cases}$$

Proof. If $n \geq 5$, the upper bound follows directly from Corollary 4.3.2 and the fact that $\Delta \leq 4$. If $n \leq 4$, then no vertex of G can have degree equal to four, so that the maximum value of the inverse sum indeg index is obtained by the complete graph K_n, for which $ISI(K_n) = \frac{n(n-1)^2}{4}$. ■

Let $KD_{n,\delta}$ be the graph obtained from the complete graph K_{n-1} by adding to it a new vertex, adjacent to exactly δ vertices of K_{n-1}. Another corollary of Lemma 4.1.1 is the following

Corollary 4.3.4 *If G is a graph with n vertices and minimum vertex degree δ, then*

$$ISI(G) \leq ISI(KD_{n,\delta})$$

with equality if and only if G is isomorphic to $KD_{n,\delta}$.

Proof. Let u be the vertex of G having degree δ. As the part of G, induced by vertices different from u, can be made into the complete graph K_{n-1} by adding edges between all pairs of nonadjacent vertices, we immediately obtain that $ISI(G) \leq ISI(KD_{n,\delta})$, with equality if and only if no edges were added to G, *i.e.*, if and only if $G \cong KD_{n,\delta}$. ■

In order to characterize the graph with the maximum inverse sum indeg index among graphs with given number of pendant vertices, we need to define an auxiliary type of graphs. For $k \geq 1$ and the sequence of nonnegative integers $q_1, \dots, q_k$, the graph $H_{q_1,\dots,q_k}$ is obtained from the complete graph K_k on the vertex set $\{1, \dots, k\}$ by attaching q_i new pendant vertices to vertex i for each $i = 1, \dots, k$. Further, for given $k \geq 1$ and $p \geq 0$ let $KP_{k,p} = H_{q_1,\dots,q_k}$ where $q_1, \dots, q_k$ are chosen such that $\sum_{i=1}^{k} q_i = p$ and $q_i \in \{\lfloor \frac{p}{k} \rfloor, \lceil \frac{p}{k} \rceil\}$ for each $i = 1, \dots, k$.

Theorem 4.3.5 *If G is a graph with n vertices, among which there are $p \geq 0$ pendant vertices, then*

$$ISI(G) \leq ISI(KP_{n-p,p}),$$

with equality if and only if G is isomorphic to $KP_{n-p,p}$.

Proof. From Lemma 4.1.1 it is apparent that the inverse sum indeg index gets increased if one adds edges between all pairs of nonadjacent vertices in the part of G induced by nonpendant vertices. Therefore, the graph with the maximum value of the inverse sum indeg index is necessarily of type $H_{q_1,\ldots,q_k}$ for $k=n-p$ and some nonnegative integers $q_1,\ldots,q_k$ such that $\sum_{i=1}^{k} q_i = p$.

Let $H = H_{p_1,\ldots,p_k}$ have the maximum inverse sum indeg index among all graphs $H_{q_1,\ldots,q_k}$ such that $\sum_{i=1}^{k} q_i = p$. Note that the degree of the vertex i, $1 \leq i \leq k$, belonging to the complete part K_k of H is equal to $d_i = k-1+p_i$.

Suppose that $|p_i - p_j| \geq 2$ for some $j \neq i$ and, without loss of generality, suppose that $p_i+2 \leq p_j$. Let $H' = H_{p_1,\ldots,p_i+1,\ldots,p_j-1,\ldots,p_k}$ be the graph obtained by reattaching one of the pendant vertices from j to i. Since the edges of H, which are not adjacent to either i or j, contribute equally to both $ISI(H)$ and $ISI(H')$, we have that

$$\begin{aligned} ISI(H') - ISI(H) \quad = \quad & \left(\frac{d_i+1}{d_i+2} - \frac{d_j}{d_j+1}\right) + \left(\frac{(d_i+1)(d_j-1)}{d_i+d_j} - \frac{d_i d_j}{d_i+d_j}\right) \\ & +(p_j-1)\left(\frac{d_j-1}{d_j} - \frac{d_j}{d_j+1}\right) + p_i\left(\frac{d_i+1}{d_i+2} - \frac{d_i}{d_i+1}\right) \\ & + \sum_{k\neq i,j}\left(\frac{(d_j-1)d_k}{d_j-1+d_k} - \frac{d_j d_k}{d_j+d_k}\right) \\ & + \sum_{k\neq i,j}\left(\frac{(d_i+1)d_k}{d_i+1+d_k} - \frac{d_i d_k}{d_i+d_k}\right). \end{aligned}$$

This difference may be written more compactly using the function

$$f(x,y) = \frac{xy}{x+y} - \frac{(x-1)y}{x-1+y}$$

as

$$\begin{aligned} ISI(H') - ISI(H) \quad = \quad & \frac{d_i+1-d_j}{(d_i+2)(d_j+1)} + \frac{d_j-d_i-1}{d_i+d_j} \\ & -(p_j-1)f(d_j,1) + p_i f(d_i+1,1) \\ & - \sum_{k\neq i,j} f(d_j,d_k) + \sum_{k\neq i,j} f(d_i+1,d_k). \end{aligned}$$

For constant $y \geq 1$, the function $f(x,y)$ is strictly decreasing for $x > 0$ since

$$f'_x(x,y) = \frac{y^2}{(x+y)^2} - \frac{(y-1)^2}{(x+y-1)^2} = \frac{x^2(2y-1)+2xy(y-1)}{(x+y)^2(x+y-1)^2} > 0.$$

Therefore, from the initial assumption that $p_i+2 \leq p_j$ (which translates to $d_i+2 \leq d_j$ as $d_j - d_i = p_j - p_i$) we have that

$$f(d_i+1,d_k) > f(d_j,d_k) \text{ for each } k \neq i,j \text{ and } f(d_i+1,1) > f(d_j,1).$$

Hence

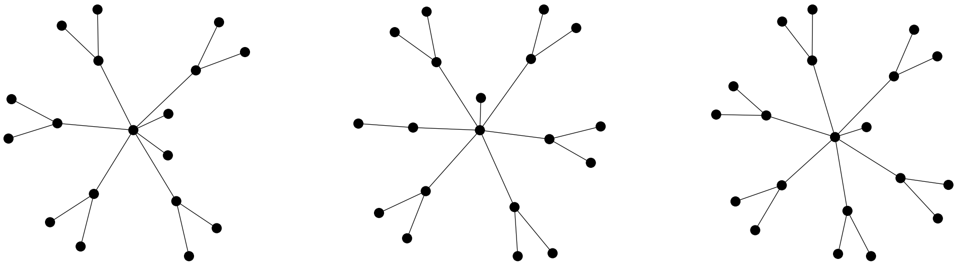
$$\begin{aligned} ISI(H') - ISI(H) \quad > \quad & \frac{d_i+1-d_j}{(d_i+2)(d_j+1)} + \frac{d_j-d_i-1}{d_i+d_j} - (p_j-p_i-1)f(d_j,1) \\ = \quad & (d_j-d_i-1)\left(\frac{1}{d_i+d_j} - \frac{1}{(d_i+2)(d_j+1)} - \frac{1}{d_j(d_j+1)}\right) \\ = \quad & (d_j-d_i-1)\frac{(d_j^2-d_i^2)+d_i(d_j^2-d_j-d_i-2)}{(d_i+d_j)(d_i+2)d_j(d_j+1)}. \end{aligned}$$

Since $d_j \geq d_i + 2$, we have that $d_j(d_j - 1) - d_i - 2 \geq (d_i + 2)d_i$, so that the difference $ISI(H') - ISI(H)$ is strictly positive, which is a contradiction to the maximality of H.

This contradiction shows that for each $i \neq j$ it has to hold $|p_i - p_j| \leq 1$, which is, due to the condition $\sum_{i=1}^{k} p_i = p$, equivalent to $p_i \in \{\lfloor \frac{p}{k} \rfloor, \lceil \frac{p}{k} \rceil\}$ for each $i = 1, \ldots, k$. Hence, the graph H with the maximum inverse indeg index is isomorphic to $KP_{k,p} = KP_{n-p,p}$. ■

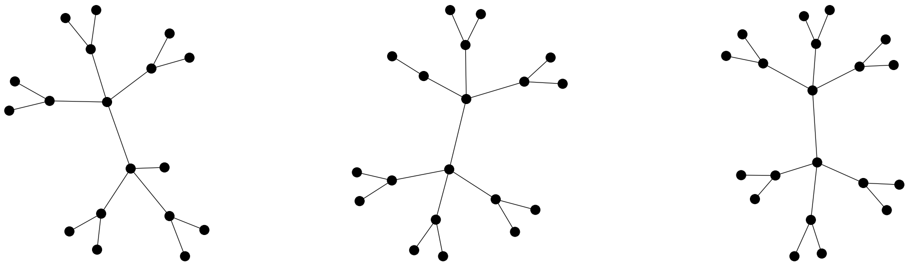
4.4 Open problems

We have to leave tasks of determining maximum inverse sum indeg index in the classes of trees and chemical trees as open problems. Experiments with MathChem with trees and chemical trees with up to 20 vertices suggest the emergence of a common structure—see extremal trees in Fig. 4.2 and extremal chemical trees in Fig. 4.3. However, in order to formulate sound conjectures on their structure, one would need to be able to process sets of (chemical) trees with much larger numbers of vertices, which is not easily attainable with computers at our disposal.



18 vertices 19 vertices 20 vertices

Figure 4.2: Trees on 18, 19 and 20 vertices having maximum inverse sum indeg index.



18 vertices 19 vertices 20 vertices

Figure 4.3: Chemical trees on 18, 19 and 20 vertices having maximum inverse sum indeg index.

Chapter 5

Symmetric Division Deg Index

This chapter deals with mathematical properties of the symmetric division deg index, the descriptor that was selected in [87] as a significant predictor of total surface area of polychlorobiphenyls (PCB) and for which the extremal graphs obtained with the help of MathChem have a particularly simple and elegant structure.

In the following sections we determine extremal values and extremal graphs of the symmetric division deg index in several classes of graphs (with given number of vertices): the class of all connected graphs, the class of all trees, the class of all unicyclic graphs, the class of all chemical graphs, the class of all graphs with given minimum degree. We also determine maximum values of the symmetric division deg index in the class of all graphs with given maximum degree.

The symmetric division deg index is defined as

$$SDD(G) = \sum_{uv \in E(G)} \left(\frac{\min\{d_u, d_v\}}{\max\{d_u, d_v\}} + \frac{\max\{d_u, d_v\}}{\min\{d_u, d_v\}} \right),$$

where d_i is the degree of vertex i.

Let us rewrite the definition of SDD index in the following way.

$$\begin{aligned} SDD(G) = \sum_{uv \in E(G)} \left(\frac{\min\{d_u, d_v\}}{\max\{d_u, d_v\}} + \frac{\max\{d_u, d_v\}}{\min\{d_u, d_v\}} \right) = \\ = \sum_{uv \in E(G)} \left(\frac{d_u}{d_v} + \frac{d_v}{d_u} \right) = \sum_{uv \in E(G)} \frac{d_u^2 + d_v^2}{d_u d_v} \end{aligned}$$

Let us denote by $\alpha(d_u, d_v)$ the individual contribution of an edge uv to SDD index.

$$\alpha(d_u, d_v) = \frac{d_u}{d_v} + \frac{d_v}{d_u} = \frac{d_u^2 + d_v^2}{d_u d_v}. \tag{5.1}$$

The definition of SDD index using α become more compact:

$$SDD(G) = \sum_{uv \in E(G)} \alpha(d_u, d_v). \tag{5.2}$$

5.1 Minimum values of the symmetric division deg index

We first establish a lower bound on the SDD index with respect to the number of edges.

The next theorem is the direct result of the fact, that the individual contribution $\alpha(d_u, d_v) = \frac{d_u}{d_v} + \frac{d_v}{d_u} \geq 2$.

Theorem 5.1.1 *Let G be a simple connected graph with m edges. Then*

$$SDD(G) \geq 2m.$$

■

Note that the equality holds if and only if $d_u = d_v$ for all $uv \in E(G)$, or alternatively, G is regular. The lower bound described in Theorem 5.1.1 can be achieved in the class of unicyclic graphs. In this case the corresponding extremal graph is the cycle C_n. The next theorem provides a lower bound on the SDD index in the class of all connected graphs with respect to the number of vertices.

Theorem 5.1.2 *Let G be a simple connected graph with $n \geq 3$ vertices. Then*

$$SDD(G) \geq 2n - 1.$$

The equality holds if and only if G is isomorphic to path P_n with n vertices.

Proof. Let us distinguish two cases:

Case 1: G is a tree.

Every tree has at least two pendant vertices. Each edge incident with that vertices contributes at least $\alpha(1, 2) = 2.5$. Each other edge contributes at least 2. Hence, $SDD(G) \geq 2(n-3) + 2 \cdot 2.5 = 2n - 1$. Moreover, the equality implies that G has exactly two pendant vertices, that is possible only if $G \cong P_n$.

It can be easily checked that $SDD(P_n) = 2n - 1$.

Case 2: G is not a tree.

G has at least n edges each of which contributes at least 2, hence

$$SDD(G) \geq 2n > 2n - 1.$$

■

The same lower bound holds for trees and chemical graphs since P_n belongs to these two graph classes.

The following theorem describes minimum value of the SDD index for graphs with given minimum vertex degree.

Theorem 5.1.3 *Let G be a graph with n vertices and minimum vertex degree δ. Then*

$$SDD(G) \geq n\delta.$$

The equality holds if and only if G is δ-regular graph.

Proof.

First, let us recall the well-known fact [85], that for any graph G without isolated vertices

$$\sum_{uv \in E(G)} \left(\frac{1}{d_u} + \frac{1}{d_v} \right) = n. \tag{5.3}$$

Then, let us consider the individual contribution of an edge uv to SDD index (5.1)

$$\begin{aligned}
\alpha(d_u, d_v) =& \frac{d_u}{d_v} + \frac{d_v}{d_u} = \left(\frac{1}{d_u} + \frac{1}{d_v} \right) (d_u + d_v) - 2 = \\
=& d_u \left(\frac{1}{d_u} + \frac{1}{d_v} \right) + d_v \left(\frac{1}{d_u} + \frac{1}{d_v} \right) - 2 = \\
=& \min\{d_u, d_v\} \left(\frac{1}{d_u} + \frac{1}{d_v} \right) + \max\{d_u, d_v\} \left(\frac{1}{d_u} + \frac{1}{d_v} \right) - 2 = \\
=& \min\{d_u, d_v\} \left(\frac{1}{d_u} + \frac{1}{d_v} \right) + \frac{\max\{d_u, d_v\}}{d_u} + \frac{\max\{d_u, d_v\}}{d_v} - 2 = \\
=& \min\{d_u, d_v\} \left(\frac{1}{d_u} + \frac{1}{d_v} \right) + \frac{\max\{d_u, d_v\}}{\min\{d_u, d_v\}} + 1 - 2 \geq \\
\geq& \min\{d_u, d_v\} \left(\frac{1}{d_u} + \frac{1}{d_v} \right) \geq \delta \left(\frac{1}{d_u} + \frac{1}{d_v} \right).
\end{aligned}$$

The equality holds if and only if $d_u = d_v = \delta$. Hence using (5.2) and (5.3),

$$SDD(G) = \sum_{uv \in E(G)} \alpha(d_u, d_v) \geq \sum_{uv \in E(G)} \delta \left(\frac{1}{d_u} + \frac{1}{d_v} \right) = n\delta,$$

with equality if and only if the degrees of all the vertices are equal to the minimum degree δ, or alternatively G is δ-regular. ■

5.2 Maximum values of the symmetric division deg index

Our first result on the maximum value of the SDD index is an upper bound with regards to the maximum degree Δ.

Theorem 5.2.1 *Let G be a simple connected graph with $n \geq 2$ vertices and maximum degree Δ. Then*

$$SDD(G) \leq n\Delta.$$

The equality holds if and only if G is Δ-regular graph.

Proof.

The theorem can be proved in exactly same the way as the previous one. Let us consider the individual contribution of an edge uv to SDD index and use the equation (5.3)

$$\begin{aligned}\alpha(d_u, d_v) =& d_u \left(\frac{1}{d_u} + \frac{1}{d_v}\right) + d_v \left(\frac{1}{d_u} + \frac{1}{d_v}\right) - 2 = \\ =& \max\{d_u, d_v\} \left(\frac{1}{d_u} + \frac{1}{d_v}\right) + \min\{d_u, d_v\} \left(\frac{1}{d_u} + \frac{1}{d_v}\right) - 2 = \\ =& \max\{d_u, d_v\} \left(\frac{1}{d_u} + \frac{1}{d_v}\right) + \frac{\min\{d_u, d_v\}}{d_u} + \frac{\min\{d_u, d_v\}}{d_v} - 2 = \\ =& \max\{d_u, d_v\} \left(\frac{1}{d_u} + \frac{1}{d_v}\right) + \frac{\min\{d_u, d_v\}}{\max\{d_u, d_v\}} + 1 - 2 \leq \\ \leq& \max\{d_u, d_v\} \left(\frac{1}{d_u} + \frac{1}{d_v}\right) \leq \Delta \left(\frac{1}{d_u} + \frac{1}{d_v}\right).\end{aligned}$$

The equality holds if and only if $d_u = d_v = \Delta$. Hence using (5.2) and (5.3),

$$SDD(G) = \sum_{uv \in E(G)} \alpha(d_u, d_v) \leq \sum_{uv \in E(G)} \Delta \left(\frac{1}{d_u} + \frac{1}{d_v}\right) = n\Delta,$$

with equality if and only if the degree of the all vertices is equal to the maximum degree, or alternatively G is Δ-regular. ∎

Now we can provide an upper bound on the SDD index in the class of all connected graphs and in the class of chemical graphs.

Corollary 5.2.2 *Let G be a simple connected graph with $n \geq 2$ vertices. Then*

$$SDD(G) \leq n(n-1).$$

The equality holds if and only if G is the complete graph K_n.

PROOF. It is clear, that $\Delta \leq n - 1$. From Theorem 5.2.1, it follows that:

$$SDD(G) \leq n\Delta \leq n(n-1).$$

The equality holds if and only if G is $(n-1)$-regular graph. The only such graph is the complete graph K_n. ∎

Corollary 5.2.3 *Let G be a chemical graph with $n \geq 2$ vertices. Then*

$$SDD(G) \leq 4n.$$

The equality holds if and only if G is 4-regular.

PROOF. The claim immediately follows from Theorem 5.2.1 assuming $\Delta = 4$. ∎

Let $KD_{n,\delta}$ be the graph obtained from the complete graph K_{n-1} by adding to it a new vertex, adjacent to exactly δ vertices in K_{n-1}. Another corollary of Theorem 5.2.1 is the following

Corollary 5.2.4 *If G is a graph with n vertices and minimum vertex degree δ, then*

$$SDD(G) \leq SDD(KD_{n,\delta})$$

with equality if and only if G is isomorphic to $KD_{n,\delta}$.

Proof. Let u be the vertex in G having degree δ. As the part of G, induced by vertices different from u, can be made into the complete graph K_{n-1} by adding edges between all pairs of nonadjacent vertices, we immediately obtain that $SDD(G) \leq SDD(KD_{n,\delta})$ by Corollary 5.2.2, with equality if and only if no edges were added to G, *i.e.*, if and only if $G \cong KD_{n,\delta}$. ■

Another important class of graphs is the class of trees. The next theorem shows that the star S_n has maximum value of the SDD index among all trees.

Theorem 5.2.5 *Let T be a tree with $n \geq 2$ vertices. Then*

$$SDD(T) \leq (n-1)^2 + 1.$$

The equality holds if and only if T is isomorphic to star S_n with n vertices.

Proof. In order to prove the upper bound it is sufficient to note that the contribution of each edge is at most $\alpha(1, n-1) = \frac{1}{n-1} + n - 1$. The only such graph is the star S_n. By summing up the contributions of all the edges we get

$$SDD(S_n) = \sum_{uv \in E(G)} \alpha(1, n-1) = (n-1)\left(\frac{1}{n-1} + n - 1\right) = (n-1)^2 + 1.$$

■

Unicyclic graphs are often considered in the field of mathematical chemistry. Let S_n^+ be the graph obtained from the star S_n by adding an edge that connects two pendant vertices.

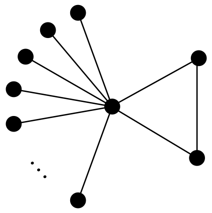


Figure 5.1: Graph S_n^+ whose value of the SDD index is maximal in the class of unicyclic graphs.

Theorem 5.2.6 *Let G be a unicyclic connected graph with $n \geq 3$ vertices. Then*

$$SDD(G) \leq \frac{n+1}{n-1} + (n-1)(n-2) + 2.$$

The equality holds if and only if G isomorphic to the graph S_n^+.

Proof. Let us distinguish two cases:

Case 1: G has a vertex of degree $n-1$.

It can be checked that S_n^+ is the only such graph with n vertices and that

$$SDD(S_n^+) = \frac{n+1}{n-1} + (n-1)(n-2) + 2.$$

Case 2: G does not have a vertex of degree $n-1$ and it has a cycle of length at least 3. Note that it is only possible when $n \geq 4$.

In this case the contribution of each edge in the cycle is at most $\alpha(2, n-2)$ and the contribution of each other edge is at most $\alpha(1, n-2)$, so it is sufficient to prove that:

$$3 \cdot \alpha(2, n-2) + (n-3) \cdot \alpha(1, n-2) < SDD(S_n^+).$$

Let us consider the function $f(n)$ defined as follows

$$f(n) = 3 \cdot \alpha(2, n-2) + (n-3) \cdot \alpha(1, n-2) - SDD(S_n^+) = \frac{(n+3)(n^2-4n+2)}{2(n^2-3n+2)}.$$

The roots of equality $f(n) = 0$ are $n \in \{-3; 2-\sqrt{2}; 2+\sqrt{2}\}$. It is easy to check that $f(n) < 0$ for $n \geq 4$.

■

Conclusion

The chapter presents initial results on SDD index, one of the 148 discrete Adriatic indices, providing upper and lower bounds for certain classes of graphs.

It is clear that SDD index is in a sense a (local) measure of irregularity. It could be profitable to learn more on its behavior with respect to edge deletion and addition and various "transplantation" type transformations.

Chapter 6

Trees of given order and independence number with minimal first Zagreb index

In this chapter we establish lower bound for the first Zagreb index among trees and unicyclic graphs with given order and independence number. Furthermore, we describe the structure of corresponding extremal graphs. The result obtained for trees is a partial solution of an open problem published by K. C. Das, K. Xu, I. Gutman in [*MATCH Commun. Math. Comput. Chem.* **70** (2013), 301–314.]

First Zagreb index defined by Gutman and Trinajstić in [36] is one of the most studied topological descriptors found its application in chemistry. The index is defined as [36, 37]

$$M_1(G) = \sum_{v \in V(G)} d_v^2.$$

Recently K. C. Das, K. Xu, and I. Gutman [34] established upper bounds for first, second Zagreb indices, and Harary index with respect to independence number. They posed several open problems, one of which is characterization of the extremal tree with minimal (first or second) Zagreb index among trees of order n and independence number α. In this chapter we provide a solution of this problem for the first Zagreb index. Moreover we extend the result obtained for trees to the class of unicyclic graphs.

We used MathChem, an open source Python package for calculating topological indices [83] to obtain conjectures regarding the structure of trees with minimal first Zagreb index among the all trees of given order n and independence number α. For $\alpha = \lceil n/2 \rceil$ the extremal graph is isomorphic to path P_n, and for $\alpha = n - 1$ the extremal graph is the star S_n. More interesting structures appear in case of $\lceil n/2 \rceil < \alpha < n - 1$. The figures 6.1 and 6.2 illustrate the stuctural diversity of the extremal graphs. Black vertices are the vertices belonging to the corresponding maximum stable set.

The extremal graphs can be described as a set of stars of almost equal size sharing a leaf with neighboring stars. *Almost equal* or *balanced* means that the size of stars differ by at most one. Every graph on Fig. 6.1 consists of three stars of size two and

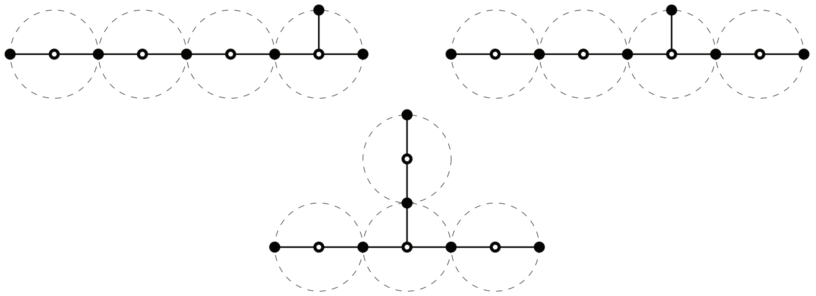


Figure 6.1: Three non-isomorphic trees with minimum value of $M_1 = 36$. $n = 10$, $\alpha = 6$.

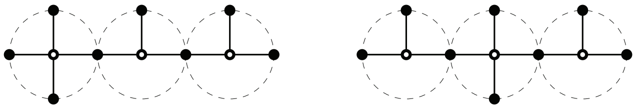


Figure 6.2: Two non-isomorphic trees with minimum value of $M_1 = 48$. $n = 11$, $\alpha = 8$.

one star of size three. In the same way, every graph on Fig. 6.2 is a chain of two stars of size three and one star of size four.

Let T be a tree with n vertices and the independence number α. Denote by S a maximal stable set in T. Thus $|S| = \alpha$. Denote by $\bar{S}$ all the rest vertices of T, or more formally $\bar{S} = V(T) \setminus S$. It is clear, that $|\bar{S}| = n - \alpha$. Let us define the graph $H_{n,\alpha}$.

Definition 6.0.7 *The graph $H_{n,\alpha}$ is a tree of order n and the independence number α such that its set $\bar{S}$ is independent, the degrees of the vertices in S are balanced and the degrees of the vertices in $\bar{S}$ are also balanced.*

It is easy to see, that the observed graphs satisfy the Definition 6.0.7. Indeed, the set S can be formed from all the leaves of the stars, and $\bar{S}$ from their central vertices.

The main result of this chapter is stated in the following theorem.

Theorem 6.0.8 *Let T be a tree with n vertices and the the independence number α, then*

$$M_1(T) \geq 4(n-1) - 2\alpha + (n-2+\alpha)\left\lfloor \frac{n-1}{n-\alpha} \right\rfloor - (n-\alpha)\left\lfloor \frac{n-1}{n-\alpha} \right\rfloor^2 \tag{6.1}$$

with equality if and only if $T \cong H_{n,\alpha}$.

6.1 Proof of the main result

K. C. Das in [10] and I. Gutman in [31] characterized graphs with n vertices and m edges having minimum value of the first Zagreb index.

Theorem 6.1.1 *[31] Among all the graphs with n vertices and m edges, for any $n \geq 1$ and $0 \leq m \leq n(n-1)/2$, the graphs in which the degree of any vertex is either $\lfloor 2m/n \rfloor$ or $\lceil 2m/n \rceil$ have minimum M_1.* ■

In other words, the degrees of all vertices in the extremal graph must be balanced.

Let us consider a tree T with n vertices and the independence number α, sets S and $\bar{S}$ as defined above. Denote by l the number of edges uv such that $u \in S$, $v \in \bar{S}$, and by k the number of edges between vertices in $\bar{S}$. Note, that $k + l = n - 1$. Our goal is to prove that minimum value of the first Zagreb index M_1 is attained by trees T with $k = 0$, or alternatively, $T \cong H_{n,\alpha}$.

Vertices in S are adjacent only to vertices in $\bar{S}$ and the total sum of their degrees is equal to l. Now, we want to minimize the sum of the squares of their degrees (i.e., the S-part of M_1). Without caring much for how the edges are placed around, the minimum possible value of the sum of the squares of degrees in S is obtained when the degrees of vertices in S are balanced. The same holds for vertices in $\bar{S}$ - except that the sum of their degrees is $l + 2k$.

Let us denote by d_i the degree of i-th vertex in S. Since $|S| = \alpha$, we can conclude that ether $d_i = \lfloor \frac{l}{\alpha} \rfloor$ or $d_i = \lceil \frac{l}{\alpha} \rceil$.

If $\alpha \mid l$ then all degrees are equal to l/α, or $l = \alpha \cdot q$. Now

$$\sum_{v \in S} d_v^2 = \alpha \cdot \frac{l^2}{\alpha^2} = \frac{l^2}{\alpha} = \alpha \cdot q^2.$$

Suppose, therefore, that $\alpha \nmid l$, and $l = \alpha \cdot q + r$, $0 < r < \alpha$. Then $d_v = q$ or $d_v = q + 1$ for each $v \in S$.

Since

$$\sum_{v \in S} d_v = l = \alpha \cdot q + r = (\alpha - r)q + r(q+1),$$

meaning that there are $(\alpha - r)$ vertices of degree q and r vertices of degree $(q + 1)$.

Then

$$\sum_{v \in S} d_v^2 \geq (\alpha - r)q^2 + r(q+1)^2 = \alpha \cdot q^2 + r(2q+1) \tag{6.2}$$

Having in mind, that $q = \lfloor \frac{l}{\alpha} \rfloor$ and $r = l - \alpha \cdot \lfloor \frac{l}{\alpha} \rfloor$, we have

$$\begin{aligned} \alpha \cdot q^2 + r(2q+1) =& \alpha \cdot \left\lfloor \frac{l}{\alpha} \right\rfloor^2 + \left(l - \alpha \cdot \left\lfloor \frac{l}{\alpha} \right\rfloor \right) \cdot \left(2 \left\lfloor \frac{l}{\alpha} \right\rfloor + 1 \right) \\ =& (2l - \alpha) \left\lfloor \frac{l}{\alpha} \right\rfloor - \alpha \cdot \left\lfloor \frac{l}{\alpha} \right\rfloor^2 + l. \end{aligned} \tag{6.3}$$

Combining left-hand side of (6.2) with right-hand side of (6.3), and remembering that $k + l = n - 1$ we get

$$\sum_{v \in S} d_v^2 \geq (2(n-1-k) - \alpha) \left\lfloor \frac{n-1-k}{\alpha} \right\rfloor - \alpha \left\lfloor \frac{n-1-k}{\alpha} \right\rfloor^2 + n - 1 - k. \tag{6.4}$$

Now we can use the same approach for $\bar{S}$. Since $|\bar{S}| = n - \alpha$ and $\sum_{v\in\bar{S}} d_v = n - 1 + k$, we have

$$\sum_{v\in\bar{S}} d_v^2 \geq (2(n-1+k)-(n-\alpha)) \left\lfloor \frac{n-1+k}{n-\alpha} \right\rfloor - (n-\alpha) \left\lfloor \frac{n-1+k}{n-\alpha} \right\rfloor^2 + n-1+k. \tag{6.5}$$

Now we can estimate the first Zagreb index, that is $M_1(T) = \sum_{v\in S} d_v^2 + \sum_{v\in\bar{S}} d_v^2$, by summing the inequalities (6.4) and (6.5),

$$\begin{aligned} M_1(T) \geq 2(n-1) &+ (2n-2k-\alpha-2) \left\lfloor \frac{n-1-k}{\alpha} \right\rfloor - \alpha \left\lfloor \frac{n-1-k}{\alpha} \right\rfloor^2 \\ &+ (n+2k+\alpha-2) \left\lfloor \frac{n-1+k}{n-\alpha} \right\rfloor - (n-\alpha) \left\lfloor \frac{n-1+k}{n-\alpha} \right\rfloor^2 . \end{aligned} \tag{6.6}$$

Let us denote the right-hand side of the inequality (6.6) by $f(n,\alpha,k)$. To prove the lower bound (6.0.8) for first Zagreb index it is sufficient to show, that for fixed n and α the minimum of the function $f(n,\alpha,k)$ is obtained for $k = 0$ for $k \in \{0, 1, ..., n-1-\alpha\}$.

Now, let us more carefully look into the function f. Since T is connected, there must be enough edges connecting S and $\bar{S}$. We have already referred to this number as $l = n-1-k$. In order to ensure connectivity $l \geq \alpha$, or $n-1-k \geq \alpha$, which implies $\frac{n-1-k}{\alpha} \geq 1$. On the other hand, the independence number $\alpha \geq n/2$ in trees and $\frac{n-1-k}{\alpha} < 2$. This implies equality $\lfloor \frac{n-1-k}{\alpha} \rfloor = 1$. Now, we can represent the function f in a more compact form

$$\begin{aligned} f(n,\alpha,k) =& 2(n-1) + 4n - 2\alpha - 4 - 2k \\ &+ (n+2k+\alpha-2) \left\lfloor \frac{n-1+k}{n-\alpha} \right\rfloor - (n-\alpha) \left\lfloor \frac{n-1+k}{n-\alpha} \right\rfloor^2 . \end{aligned}$$

Let $h(n,\alpha,k)$ be the variable part of the function f, that is

$$h(n,\alpha,k) = -2k + (n+2k+\alpha-2) \left\lfloor \frac{n-1+k}{n-\alpha} \right\rfloor - (n-\alpha) \left\lfloor \frac{n-1+k}{n-\alpha} \right\rfloor^2 .$$

Now, we want to minimize the function h in terms of k with fixed n and α. Denote by $t = \left\lfloor \frac{n-1+k}{n-\alpha} \right\rfloor$. Since $0 \leq k \leq n-1-\alpha$, it is easy to check, that ether $t = \left\lfloor \frac{n-1}{n-\alpha} \right\rfloor$ or $t = \left\lfloor \frac{n-1}{n-\alpha} \right\rfloor + 1$,

$$\left\lfloor \frac{n-1}{n-\alpha} \right\rfloor \leq \left\lfloor \frac{n-1+k}{n-\alpha} \right\rfloor \leq \left\lfloor \frac{n-2}{n-\alpha} \right\rfloor + 1 \leq \left\lfloor \frac{n-1}{n-\alpha} \right\rfloor + 1.$$

Let us find out for which k the function t remains constant:

$$\begin{aligned} t(n-\alpha) \leq n-1-k &\leq (t+1)(n-\alpha) - 1 \\ t(n-\alpha) - (n-1) \leq k &\leq (t+1)(n-\alpha) - n. \end{aligned}$$

Denote by $t_0 = \left\lfloor \frac{n-1}{n-\alpha} \right\rfloor$. Consider the first case, when $t = t_0$, that is possible for $0 \leq k \leq t_0(n-\alpha) - \alpha$. Now the function h is

$$h(n, \alpha, k) = 2k(t-1) + (n+\alpha-2)t - (n-\alpha)t^2.$$

Since $t \geq 1$, the function h is increasing in terms of k and minimum is obtained for $k = k_1 = 0$ which is equal to

$$h(n, \alpha, k_1) = t_0(n+\alpha-2) - (n-\alpha)t_0^2.$$

Consider another case, when $t = t_0 + 1$, that is possible for $t_0(n-\alpha) - \alpha + 1 \leq k \leq n - \alpha - 1$. Again, to obtain minimum of increasing h, we can use the smallest possible k. Thus the minimum of h is obtained for $k = k_2 = t_0(n-\alpha) - \alpha + 1$ which is equal to

$$h(n, \alpha, k_2) = 2kt_0 + (n+\alpha-2)(t_0+1) - (n-\alpha)t_0^2 - 2(n-\alpha)t_0 - (n-\alpha).$$

Eventually, we need to ensure, that $h(n, \alpha, k_2) \geq h(n, \alpha, k_1)$.

$$\begin{aligned} h(n, \alpha, k_2) - h(n, \alpha, k_1) &= ((t_0+1)(n-\alpha) - (n-1))t_0 + \alpha - 1 - (n-\alpha)t_0 \\ &= (n-\alpha)t_0^2 - (n-1)t_0 + \alpha - 1, \end{aligned}$$

that is equal to zero for $t_0 \in \{1, \frac{\alpha-1}{n-\alpha}\}$. Since $t_0 > \frac{\alpha-1}{n-\alpha}$, we can conclude, that $h(n, \alpha, k_2) \geq h(n, \alpha, k_1)$ and the minimum for the function h, as well as f, is obtained for $k = 0$ which yields the Theorem 6.0.8. ■

6.2 Unicyclic graphs

This section extends the result of Theorem 6.0.8 to the class of unicyclic graphs. A graph G is called *unicyclic* if it has equal number of edges and vertices. It is clear, that for $\alpha = \lfloor n/2 \rfloor$ the extremal graph is isomorphic to cycle C_n by Theorem 6.1.1 and $M_1(C_n) = 4n$.

Let G be a unicyclic graph with n vertices and the independence number α. Denote by S a maximal stable set in G. Thus $|S| = \alpha$. Denote by $\bar{S}$ all the rest vertices of G, i.e. $\bar{S} = V(T) \setminus S$. It is clear, that $|\bar{S}| = n - \alpha$. Let us define the graph $U_{n,\alpha}$.

Definition 6.2.1 *The graph $U_{n,\alpha}$ is a unicyclic graph of order n and the independence number α such that its set $\bar{S}$ is independent, the degrees of the vertices in S are balanced and the degrees of the vertices in $\bar{S}$ are also balanced.*

Again, the graphs satisfied the Definition 6.2.1 can be formed from joint stars such that, the set S consists of all the leaves of the stars, and $\bar{S}$ from their central vertices.

The following theorem is an analogical result of Theorem 6.0.8 applied to the class of unicyclic graphs.

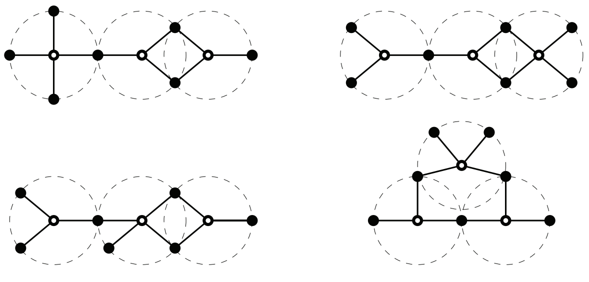


Figure 6.3: All non-isomorphic graphs $U_{10,7}$.

Theorem 6.2.2 *Let G be a unicyclic graph with n vertices and the the independence number $\alpha \geq n/2$, then*

$$M_1(G) \geq 4n - 2\alpha + (n+\alpha)\left\lfloor \frac{n}{n-\alpha} \right\rfloor - (n-\alpha)\left\lfloor \frac{n}{n-\alpha} \right\rfloor^2. \qquad (6.7)$$

with equality if and only if $G \cong U_{n,\alpha}$.

Proof.

Let us consider a unicyclic graph G with n vertices and the independence number α, sets S and $\bar{S}$ as defined above. As well as for trees we denote by l the number of edges uv such that $u \in S$, $v \in \bar{S}$, and by k the number of edges between vertices in $\bar{S}$. Note, that in case of unicyclic graphs $k + l = n$. Our goal is to prove that minimum value of the first Zagreb index M_1 is attained by graphs G with $k = 0$, or alternatively, $G \cong U_{n,\alpha}$.

Again, using Theorem 6.1.1, we presume that the degrees of the vertices in S and $\bar{S}$ are balanced separately. Then, taking into account that $k + l = n$, the inequality (6.4) for unicyclic graphs takes the following form:

$$\sum_{v\in S} d_v^2 \geq (2(n-k)-\alpha)\left\lfloor \frac{n-k}{\alpha} \right\rfloor - \alpha\left\lfloor \frac{n-k}{\alpha} \right\rfloor^2 + n - k.$$

Since $n - k \geq \alpha$ to have enough edges between S and $\bar{S}$, and $\alpha \geq n/2$, it is easy to check, that $\left\lfloor \frac{n-k}{\alpha} \right\rfloor = 1$. Thus the last inequality will have more simple form:

$$\sum_{v\in S} d_v^2 \geq 3(n-k) - 2\alpha. \qquad (6.8)$$

The inequality (6.5) for uncicyclic graphs become

$$\sum_{v\in \bar{S}} d_v^2 \geq (2(n+k)-(n-\alpha))\left\lfloor \frac{n+k}{n-\alpha} \right\rfloor - (n-\alpha)\left\lfloor \frac{n+k}{n-\alpha} \right\rfloor^2 + n + k. \qquad (6.9)$$

The sum of left-hand sides of the inequalities (6.8) and (6.9) is the value of the first Zagreb index of the graph G:

$$M_1(G) \geq 4n - 2k - 2\alpha + (n+2k+\alpha)\left\lfloor \frac{n+k}{n-\alpha} \right\rfloor - (n-\alpha)\left\lfloor \frac{n+k}{n-\alpha} \right\rfloor^2. \qquad (6.10)$$

Let function $h(n, \alpha, k)$ be the variable part of the right-hand side of the inequality (6.10), that is

$$h(n, \alpha, k) = -2k + (n + 2k + \alpha) \left\lfloor \frac{n+k}{n-\alpha} \right\rfloor - (n-\alpha) \left\lfloor \frac{n+k}{n-\alpha} \right\rfloor^2 .$$

In order to prove the theorem we need to show that the function $h(n, \alpha, k)$ attains its minimum for $k = 0$ with fixed n and α. This can be done in exactly the same way as we did it for trees. ■

Remark. Note, that the condition $\alpha \geq n/2$ in the statement of the Theorem 6.2.2 doesn't cover the case when n is odd and $\alpha = (n-1)/2$. In this case the extremal graph is the cycle C_n of the odd order that is not isomorphic to $U_{n,\alpha}$. However the lower bound (6.7) works for this case as well. Indeed, $\left\lfloor \frac{n}{n-\alpha} \right\rfloor = 1$ and we get $M_1(C_n) \geq 4n$.

Chapter 7

Applications of MatchChem

7.1 Correlation analysis of spectral properties of Adriatic matrices

The way the Adriatic indices are formed, provide us with definitions of so-called *Adriatic matrices*, whose (u, v)-entry equal to

$$a_{i,j} = \begin{cases} \gamma_j(\phi_{i,a}(p_u), \phi_{i,a}(p_v)), & \text{if } uv \in E(G), \\ 0 & \text{otherwise.} \end{cases},$$

I this section we pay attention to the spectral properties of the Adriatic matrices, in particular the spectral radius and the energy, and try to find out if these invariants may have practical application in mathematical chemistry.

The usual way to estimate predictive activity of topological descriptors is to take various molecular structures with known (measured) properties and check the correlation between them and the topological descriptors.

In this experiment we compare correlation between measured properties and classical topological indices, discrete Adriatic indices, and spectral properties of Adriatic matrices.

In order to promote the comparison of new and old molecular descriptors, evaluate their predictive ability and better understand their meaning, the International Academy of Mathematical Chemistry suggests the use of some benchmark data sets.

The IAMC dataset [59] includes the set of 18 octane isomers (C8), 82 polyaromatic hydrocarbons (PAH), 209 polychlorobiphenyls (PCB), and 22 Phenetylamines.

- Octane isomers (C8)
  The data set is constituted by 18 octane isomers (C8). The following properties are given:

  1. boiling point (BP)
  2. melting point (MP)
  3. heat capacity at T constant (CT)

4. heat capacity at P constant (CP)
5. Entropy (S)
6. density (DENS)
7. enthalpy of vaporization (HVAP)
8. standard enthalpy of vaporisation (DHVAP)
9. enthalpy of formation (HFORM)
10. standard enthalpy of formation (DHFORM)
11. motor octane number (MON)
12. molar refraction (MR)
13. acentric factor (AcenFac)
14. total surface area (TSA)
15. octanol-water partition coefficient (LogP)
16. molar volume (MV)

A set of 102 molecular descriptors (topological descriptors) is also given.

- Polyaromatic hydrocarbons (PAH)
  The data set is constituted by 82 polyaromatic hydrocarbons (PAH). The following properties are given:

  1. melting point (MP)
  2. boiling point (BP)
  3. octanol-water partition coefficient (LogP)

  A set of 112 molecular descriptors (topological descriptors) is also given.

- Polychlorobiphenyls (PCB)
  The data set is constituted by 209 polychlorobiphenyls (PCB). The following properties are given:

  1. melting point (MP)
  2. relative retention time (RTT)
  3. octanol-water partition coefficient (logP)
  4. total surface area (TSA)
  5. log Henry constant (logH)
  6. log water solubility (logSw)
  7. log water activity coefficient (logYw)
  8. relative enthalpy of formation (dHf)

  A set of 106 molecular descriptors (topological descriptors) is also given.

- Phenetylamines
  The data set is constituted by 22 phenetyl-amines with two substituent sites (Phenet). The following properties are given:

1. biological activity: log(1/C)

A set of 110 molecular descriptors (topological descriptors) is also given.

Definition 7.1.1 *Let G be a graph with n vertices, and let A be a matrix, whose (u, v)-entry defined as*

$$a_{i,j} = \begin{cases} \gamma_j(\phi_{i,a}(p_u), \phi_{i,a}(p_v)), & \textit{if } uv \in E(G), \\ 0 & \textit{otherwise.} \end{cases},$$

where $\gamma_j, \phi_{i,a}, p_u$ is taken from the definition of discrete Adriatic indices. Then A is Adriatic matrix.

For Adriatic matrices we will use the same naming convention as for Adriatic indices. For instance, the matrix, whose (u, v)-entry is

$$a_{u,v} = \frac{d_u d_v}{d_u + d_v},$$

is called *Inverse sum indeg matrix.*

Now, we can define *Adriatic energy* and *Adriatic spectral radius.*

Definition 7.1.2 *Let G be a graph with n vertices, and let $\lambda_1, \lambda_2, \ldots, \lambda_n$ be the eigenvalues of of the corresponding Adriatic matrix. Then the Adriatic energy of the graph is defined as:*

$$AE(G) = \sum_{i=1}^{n} |\lambda_i|.$$

Definition 7.1.3 *Let G be a graph with n vertices, and let $\lambda_1, \lambda_2, \ldots, \lambda_n$ be the eigenvalues of of the corresponding Adriatic matrix. Then the Adriatic spectral radius of the graph is defined as:*

$$A\rho(G) = \max_i(|\lambda_i|).$$

The following piece of Python code shows how to calculate these spectral invariants using MathChem:

```
# mols is array of mathchem.Mol instances from IAMC data
for m in mols:
    data = dict()
    for i in mathchem.all_adriatic():
        AM = m.adriatic_matrix(*i)
        spectrum = m.spectrum(AM)

        value = numpy.sum( map( lambda x: abs(x) ,spectrum))
        name = mathchem.adriatic_name(*i)+' matrix energy'
        data[name] = value

        value = max( map( lambda x: abs(x) ,spectrum))
        name = mathchem.adriatic_name(*i)+' matrix spectral radius'
        data[name] = value
```

We used *Sage* with *Mathchem* to perform a liner regression of the measured properties and topological descriptors, both provided by IAMC and calculated in MathChem, such as Adriatic indices and spectral invariants of Adriatic matrices.

In Table 7.1 the full names of descriptors can be found by their symbols. The names was taken from supporting documentation of benchmark sets provided by IAMC [59].

Table 7.1: Molecular descriptors symbols

Symbol	Description
X2	connectivity index chi-2
SPI	superpendentic index
X2v	valence connectivity index chi-2
X2sol	solvation connectivity index chi-2
PW3	path/walk 3 - Randic shape index
Pol	polarity number
TIE	E-state topological parameter
Har	Harary H index
Rww	reciprocal hyper-detour index
RHyDp	reciprocal hyper-distance-path index
X3sol	solvation connectivity index chi-3
X0v	valence connectivity index chi-0
ZM1V	first Zagreb index by valence vertex degrees
X2sol	solvation connectivity index chi-2
X3Av	average valence connectivity index chi-3
X2v	valence connectivity index chi-2
VDA	average vertex distance degree
D/Dr06	distance/detour ring index of order 6
UNIP	unipolarity
RDCHI	reciprocal distance Randic-type index
X3A	average connectivity index chi-3
AECC	average eccentricity

7.1.1 Octane isomers

Table 7.2 reflects the best-correlated descriptors for each property of octane isomers.

We can see, that spectral radius and energy of some Adriatic matrices correlate better with some properties, such as motor octane number (MON), total surface area (TSA), Entropy (S), boiling point (BP), melting point (MP), enthalpy of vaporization (HVAP), heat capacity at P constant (CP). The corresponding linear regression graphs are shown on Fig. 7.1-7.7.

7.1.2 Polyaromatic hydrocarbons

In table 7.3 the best-correlated descriptors for each property of polyaromatic hydrocarbons (PAH) are given.

In case of polyaromatic hydrocarbons, the spectral radius and energy of Adriatic matrices are not the best, but also showing good correlations ($R^2 > 0.9$) for octanol-water partition coefficient (LogP) and boiling point (BP).

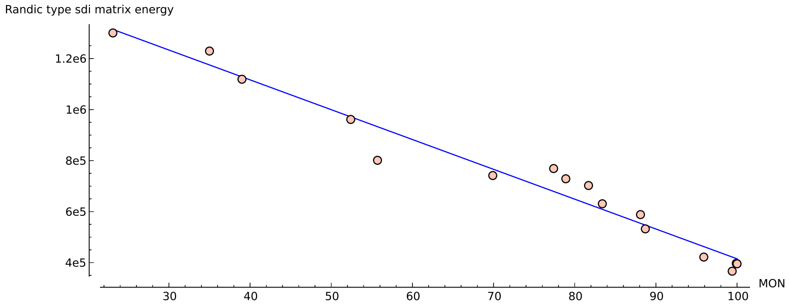
Table 7.2: Correlation coefficients R^2 for Octane isomers (C8)

Property	Best result	Second result	Third result
MON	0.964271396489 Randic type sdi energy	0.956903864728 Randic type sdi index	0.951462580403 inverse misbalance indi energy
LogP	0.364869638665 sum lordeg index	0.294103957398 SPI	0.249399637963 symmetric division losdeg energy
MV	0.897332230827 Randic type hadi index	0.88590628243 Randic type hadi spectral radius	0.885778175168 Randic type hadi energy
DENS	0.911799976992 Randic type hadi index	0.906408098904 Randic type hadi spectral radius	0.906278894665 Randic type hadi energy
TSA	0.853649022177 Randic type lordeg spectral radius	0.802226267328 symmetric division hadi energy	0.802226267328 symmetric division twodi energy
HFORM	0.832380555424 X2	0.832380555424 X2v	0.832380555424 X2sol
DHFORM	0.669178972828 PW3	0.646040716141 Pol	0.629541876164 inverse sum hadeg energy
S	0.92341559283 sum sdeg energy	0.916285902824 TIE	0.91241378821 Randic type twodeg index
AcentFac	0.992290358491 Har	0.990620387337 Rww	0.990620387337 RHyDp
BP	0.838735182594 inverse sum indeg energy	0.785519904272 symmetric division losdi energy	0.785054372859 inverse sum irdeg energy
MP	0.801449693557 Randic type hadi spectral radius	0.801286284279 Randic type hadi energy	0.773167789909 Randic type hadi index
HVAP	0.924667378131 min-max rodeg energy	0.907092308671 max-min rodeg index	0.906715335124 misbalance lodeg index
MR	0.979373453621 PW3	0.97727473637 Pol	0.941700449762 X3sol
DHVAP	0.967777024348 misbalance lodeg index	0.965899436997 min-max rodeg energy	0.965273474175 misbalance rodeg index
CP	0.698038673673 Randic type lordeg spectral radius	0.696538371227 misbalance di energy	0.681772954298 inverse sum di energy
CT	0.760107640486 Randic type lodeg index	0.758254172244 max-min rodi energy	0.745624250492 inverse sum lodi energy

7.1.3 Polychlorobiphenyls

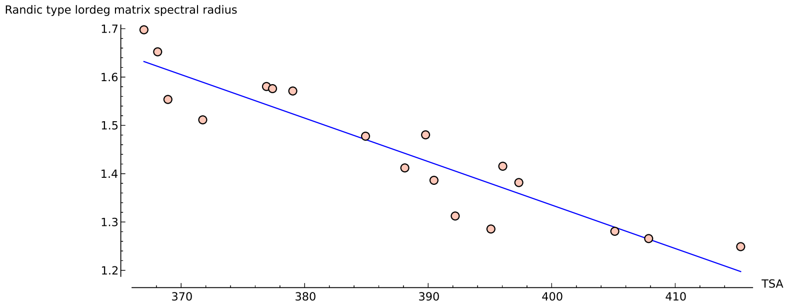
Table 7.4 reflects the best-correlated descriptors for each property of polychlorobiphenyls (PCB).

Melting point (MP) of polychlorobiphenyls can be predicted well by min-max sdi energy, as well as log water solubility (logSw) by misbalance lodeg energy, and relative retention time (RTT) by min-max sdi energy.



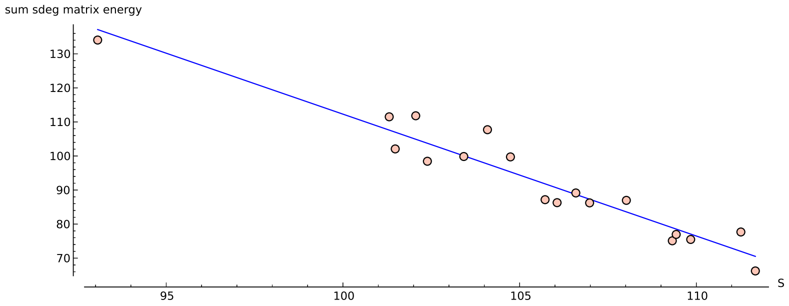
$$R^2 = 0.964271396489$$

Figure 7.1: Octanes. Correlation between MON and Randic type sdi energy



$$R^2 = 0.853649022177$$

Figure 7.2: Octanes. Correlation between TSA and Randic type lordeg spectral radius

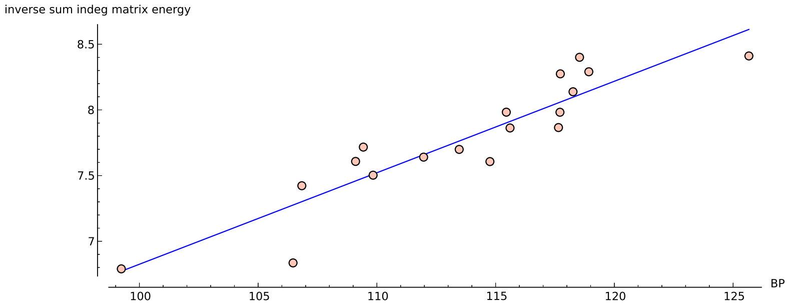


$$R^2 = 0.92341559283$$

Figure 7.3: Octanes. Correlation between S and sum sdeg energy

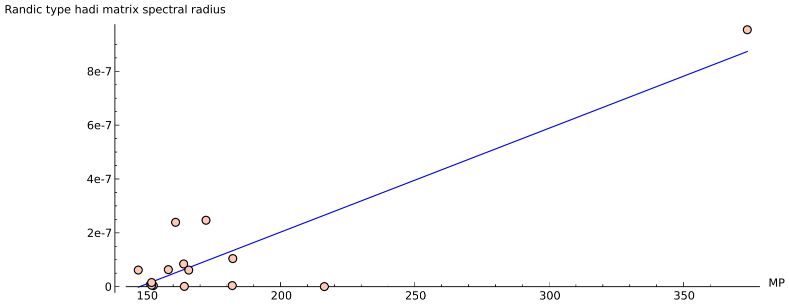
7.1.4 Phenetyl-amines with two substituent sites

It is seen from Table 7.5, that all the descriptors tested do not provide good predictive properties.



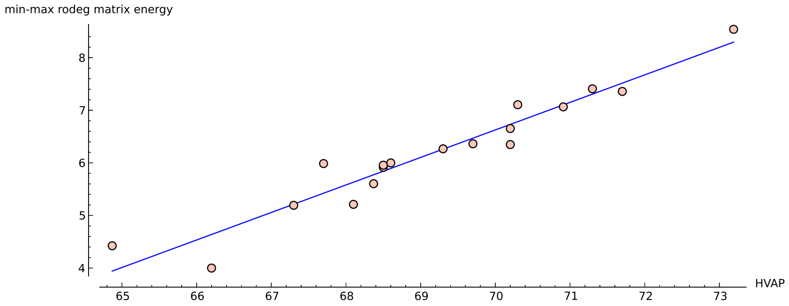
$R^2 = 0.838735182594$

Figure 7.4: Octanes. Correlation between BP and inverse sum indeg energy



$R^2 = 0.801449693557$

Figure 7.5: Octanes. Correlation between MP and Randic type hadi spectral radius



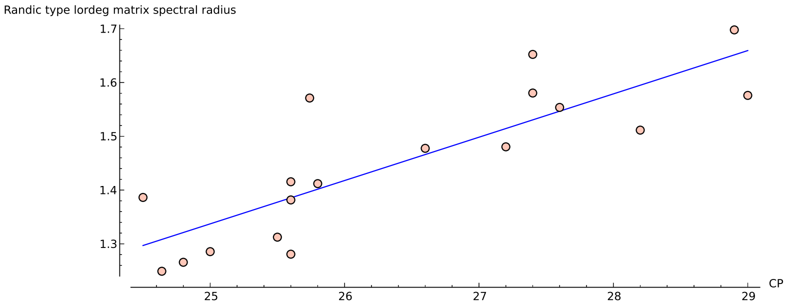
$R^2 = 0.924667378131$

Figure 7.6: Octanes. Correlation between HVAP and min-max rodeg energy

7.1.5 Conclusion

The results obtained during this numerical experiment make spectral radius and energy of selected Adriatic matrices to be reasonable candidates for further study. Below is the list of Adriatic spectral invariants that showed good predictive properties.

- Randic type sdi energy,
- inverse misbalance indi energy,



$$R^2 = 0.698038673673$$

Figure 7.7: Octanes. Correlation between CP and Randic type lordeg spectral radius

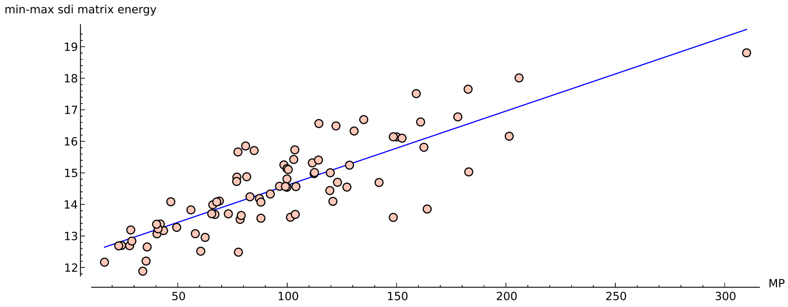
Table 7.3: Correlation coefficients R^2 for Polyaromatic hydrocarbons (PAH)

Property	Best result	Second result	Third result
LogP	0.941283981936 X0v	0.932065990453 inverse sum lordeg energy	0.921654022396 inverse sum lordeg index
BP	0.979780899895 ZM1V	0.977976230509 inverse sum rodeg index	0.97669122318 symmetric division rodeg energy
MP	0.780135415346 max-min losdeg index	0.774851462695 max-min losdeg energy	0.744237434401 X2sol

Table 7.4: Correlation coefficients R^2 for Polychlorobiphenyls (PCB)

Property	Best result	Second result	Third result
logP	0.930452747586 RDCHI	0.927960020417 MDDD	0.923317304111 sum irdi spectral radius
dHf	0.671321368225 X3Av	0.670853650743 min-max sdi spectral radius	0.617333571866 min-max di spectral radius
TSA	0.9971144917414 symmetric division deg index	0.996536763832 symmetric division sdeg index	0.99644602652 X2v
logSw	0.943960484608 VDA	0.943314601781 misbalance lodeg energy	0.943034176412 D/Dr06
logYw	0.827078391427 max-min deg index	0.827002806706 max-min sdeg index	0.826696299111 max-min rodeg index
MP	0.684116644585 min-max sdi energy	0.674406184195 UNIP	0.669406451698 min-max sdi index
RRT	0.965430956508 min-max sdi index	0.96146441158 min-max sdi energy	0.960018655635 RDCHI
logH	0.70572306396 X3A	0.703754615776 PW4	0.500485572168 X4Av

- Randic type lordeg spectral radius,
- sum sdeg energy,
- min-max rodeg energy,



$R^2 = 0.684116644585$

Figure 7.8: Polychlorobiphenyls. Correlation between MP and min-max sdi energy

Table 7.5: Correlation coefficients R^2 for Phenetyl-amines (Phenet).

Property	Best result	Second result	Third result
log(1/C)	0.574505978492 misbalance indi energy	0.570678247645 misbalance indi index	0.540490011576 AECC

- inverse sum lordeg energy,
- symmetric division rodeg energy,
- min-max sdi energy,
- misbalance lodeg energy.

7.2 Network of topological indices

In this section we investigate relations between topological indices of various nature. We establish networks of high correlations among topological indices in order to spot unexpected similarities between them. We will mostly pay attention to those topological indices that correlates well but formed by different graph properties.

We will perform the following numerical experiment. We take a class of graphs, let say all trees of order $n \in \{6, \dots, 19\}$, and calculate most of the topological indices allowed by MathChem for all graphs in that class. Then, for given class of graphs we can calculate the correlation coefficient for each pair of topological indices. Now, let's introduce the network of topological indices, which is the graph, whose vertices are topological indices adjacent if the correlation coefficient for them are greater then some threshold value.

In other words, the experiment follows the next steps:

1. Choose several classes of graphs.
2. Choose a set of graph invariants.
3. Calculate all invariants for all classes of graphs.

4. Calculate correlation coefficient for all pair of invariants grouped by graphs' order for each graph class.
5. Take minimum value of correlation coefficient among the groups.
6. Build network of highly correlated invariants for each graph class.
7. Analyse results.

In this experiment we consider the following classes of graphs:

- Connected graphs of order $n \in \{6, \dots, 8\}$.
- Trees of order $n \in \{6, \dots, 19\}$.
- Chemical graphs of order $n \in \{6, \dots, 11\}$.
- Unicyclic graphs of order $n \in \{7, \dots, 16\}$.

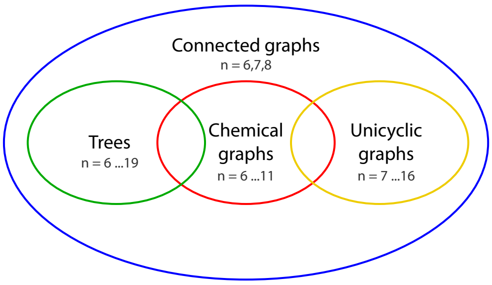


Figure 7.9: classes of networks of topological indices.

Complete list of graph invariants used in the experiment.

- *Trivial graph invariants:*
  order, size, diameter.
- *Classical topological indices:*
  first Zagreb index, second Zagreb index, first Zagreb coindex, second Zagreb coindex, augmented Zagreb index, sum connectivity index, geometric arithmetic index, eccentric connectivity index, Randić index, atom bond connectivity index, degree distance, reverse degree distance, molecular topological index, eccentric distance sum, Balaban J index, sum Balaban index, Kirchhoff index, Wiener index, terminal Wiener index, reverse Wiener index, hyper Wiener index, Harary index, Szeged index, revised Szeged index, multiplicative sum Zagreb index, multiplicative Π_2 Zagreb index, multiplicative Π_1 Zagreb index.
- *Spectral invariants*
  LEL (Laplacian like energy), incidence energy, HOMO-LUMO index; largest and smallest eigenvalues; fourth, sixth, eighth spectral moments; energy of adjacency, Laplacian, normalized Laplacian, distance, resistance distance, reciprocal distance, and Seidel matrices.

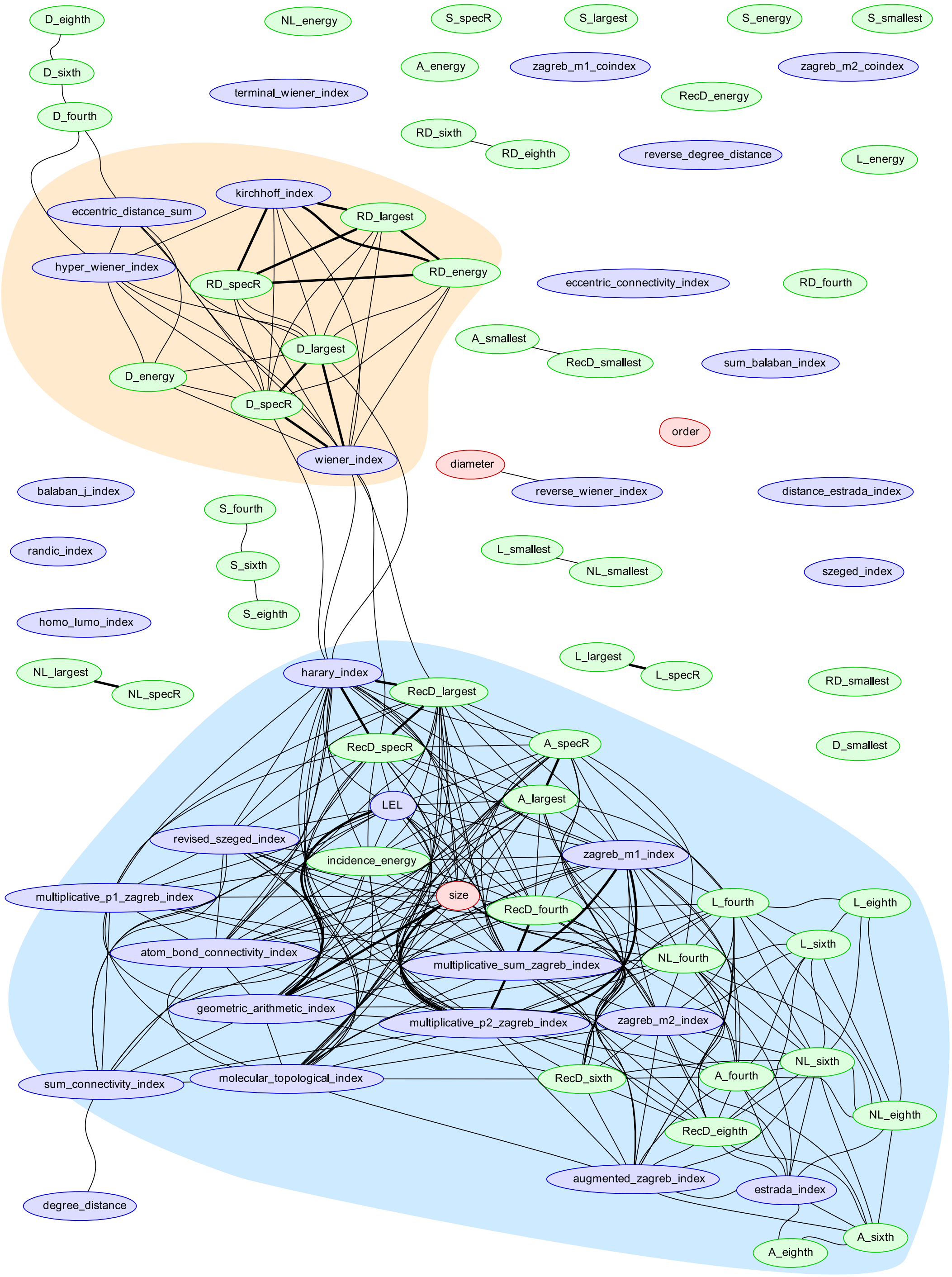


Figure 7.10: Network of topological indices. Connected graphs. $R^2 \geq 0.9$

The resulting networks of topological indices for class of connected graphs, trees, and unicyclic graphs are presented on Fig. 7.10, 7.11, 7.12. Strong lines between nodes mean that the correlation coefficient of corresponding topological indices ex-

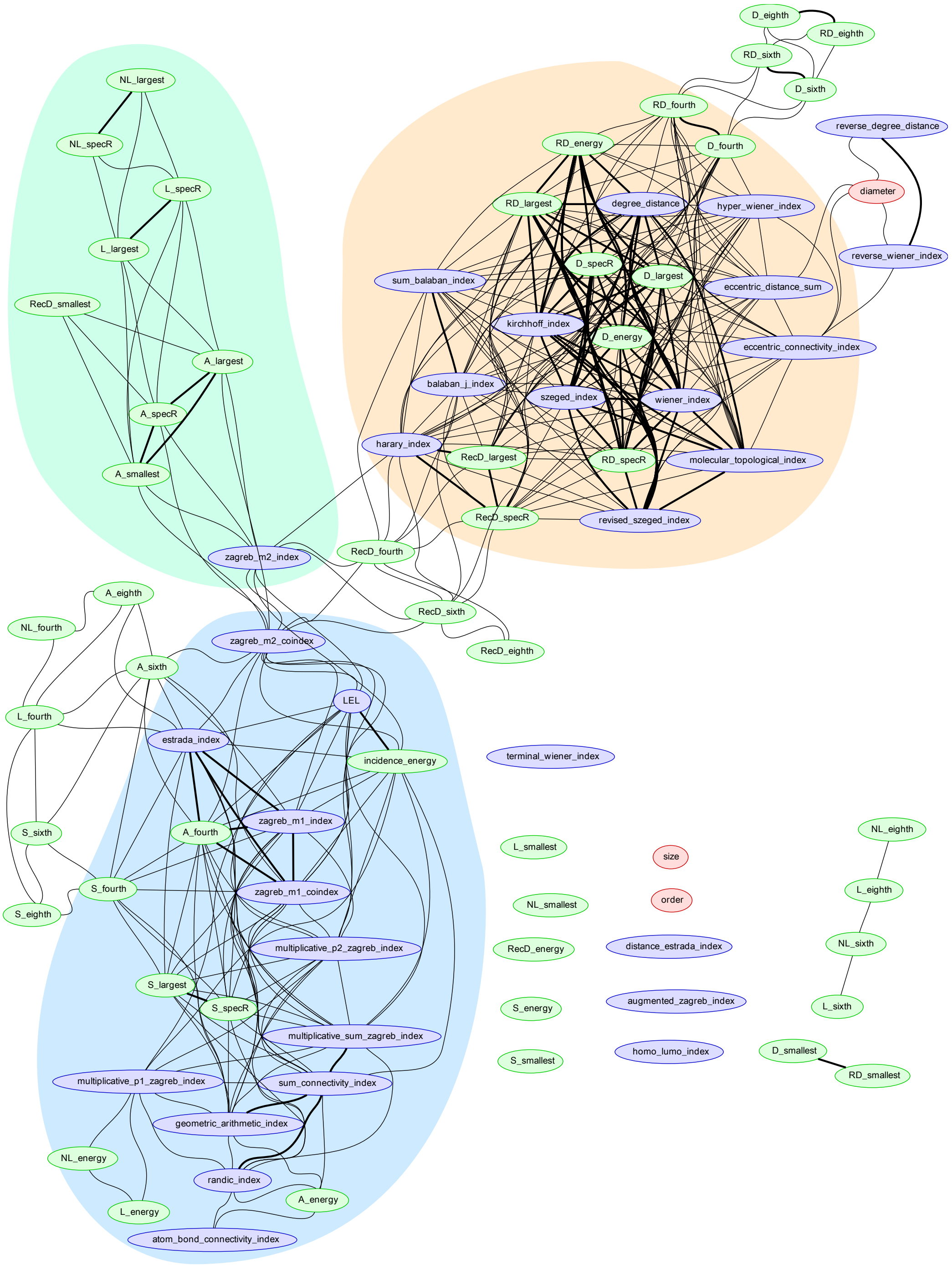


Figure 7.11: Network of topological indices. Trees. $R^2 \geq 0.9$

ceeds 0.99.

There are several clusters in the networks. They mostly consist of indices of the same nature. For instance, most indices based on degrees are adjacent with each other. However, it is more interesting to discover dependences between indices of

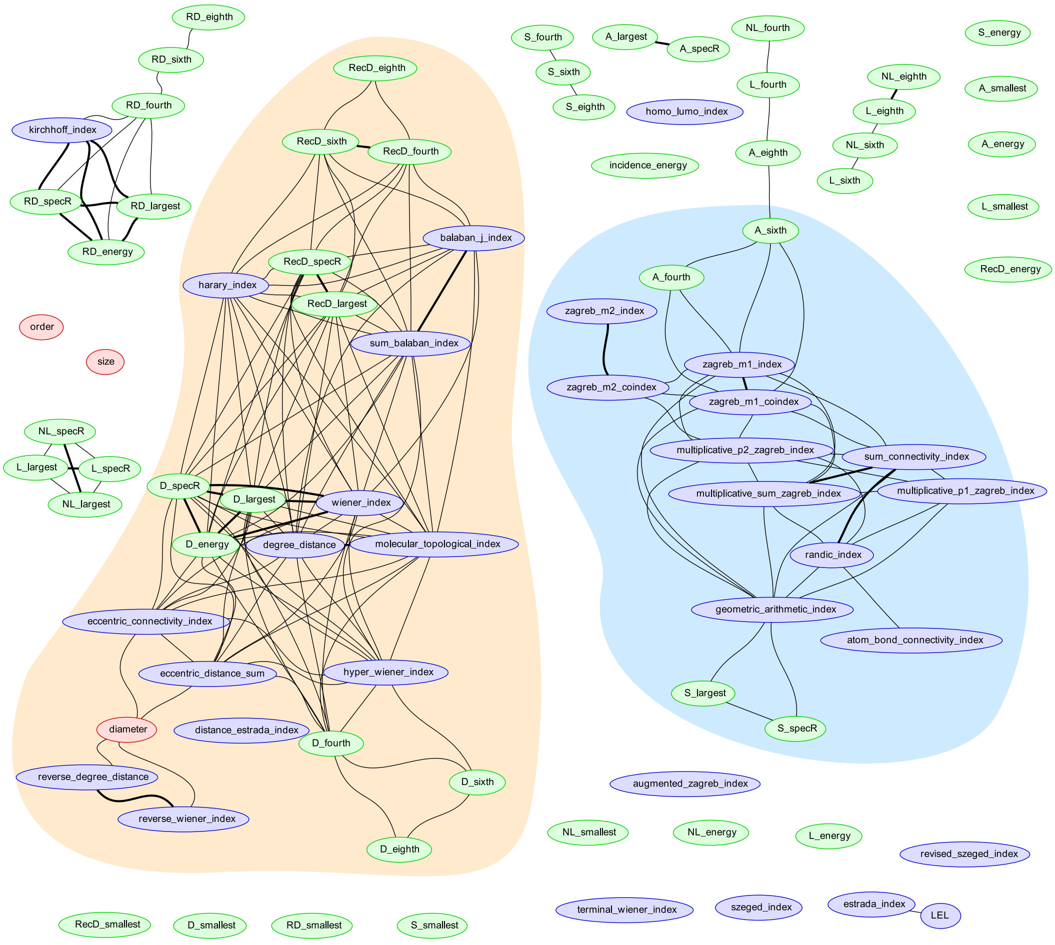


Figure 7.12: Network of topological indices. Unicyclic graphs. $R^2 \geq 0.9$

different nature. By analysing the resulting networks, one can find a number of candidates for further investigation.

Since the correlation coefficients were calculated for all the graphs of certain class with fixed order, the question about how the correlation coefficient depends on order, arise quite naturally.

We have selected several pairs of topological indices that are correlate well in certain graph class and have different nature. In the next subsections we describe the obtained results divided by classes of graphs where high correlation appeared.

7.2.1 Connected graphs

The typical picture in case when the correlation coefficient is high in the class of connected graphs, is that other subclasses (in our case trees, unicyclic and chemical graphs) do not show similar good correlation. Therefore, we assume that for further research a proper subset must be found. Fig. 7.13 illustrates low correlation among trees, unicyclic and chemical graphs for *molecular topological index* defined as $MTI(G) = \sum_{i=1}^{n} E_i$, where vector $E = (A + D) \cdot d$ and *incidence energy* defined as $IE(G) = \sum_{i=1}^{n} |\mu_i|$, where μ_i are the eigenvalues of $I(G) \cdot I(G)^T$, and $I(G)$ is the incidence matrix of graph G.

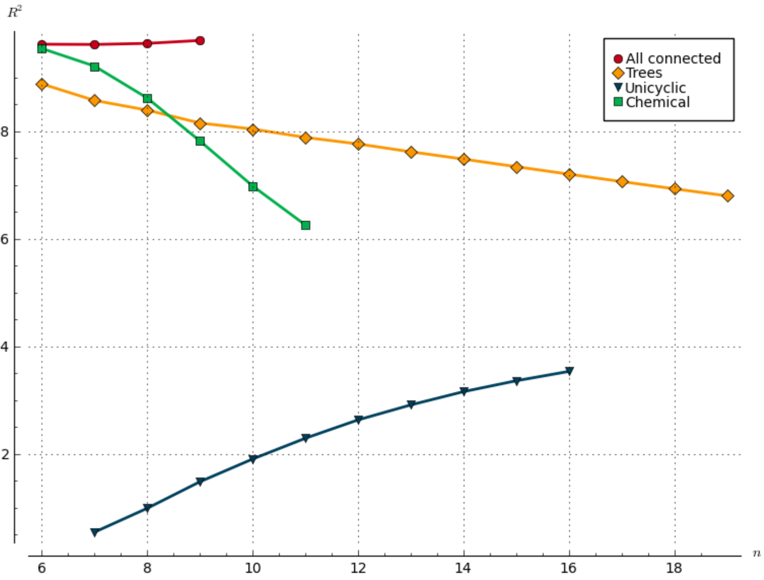


Figure 7.13: Correlation of MTI and incidence energy.

7.2.2 Trees and unicyclic graphs

In case of trees we selected three pairs of well-correlated topological indices. The first pair is *first Zagreb index* which is the sum of squares of all vertices degrees in graph and the *largest eigenvalue of Seidel matrix* $S(G) = J - I - 2A(G)$, where J is all-one diagonal matrix, I is identity matrix, and A is the adjacency matrix of graph G.

The second pair is *geometric arithmetic index* defined as $GA(G) = \sum_{ij \in E(G)} \frac{2\sqrt{d_i d_j}}{d_i + d_j}$ and *fourth spectral moment of Seidel matrix*. The corresponding diagrams are presented on Fig. 7.14, 7.15.

Note, that these two pairs of indices are also correlate well in the class of unicyclic graphs. However, more common classes of graphs – connected and chemical graphs remain completely uncorrelated.

The third pair is *multiplicative* Π_2 *Zagreb index* defined as $\Pi_2(G) = \prod_{i \in V(G)} d_i^2$ and *normalized Laplacian energy* that is the sum of absolute values of eigenvalues of normalized Laplacian matrix. It is clear from Fig. 7.16, that this pair of indices correlate well only in class of trees.

7.2.3 Chemical graphs

Chemical graphs, being the connected graphs in which each vertex has degree at most four, similarly to the class of connected graphs seems to be too common and

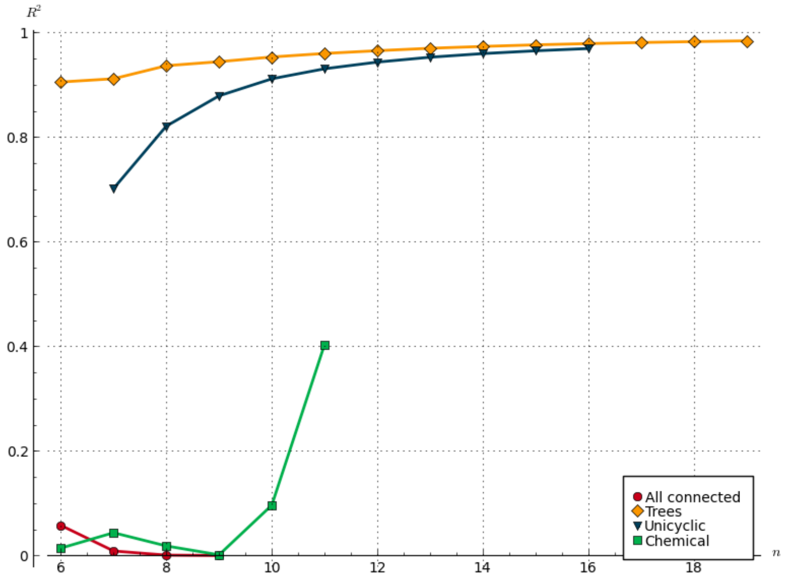


Figure 7.14: Correlation of first Zagreb index and the largest eigenvalue of Seidel matrix.

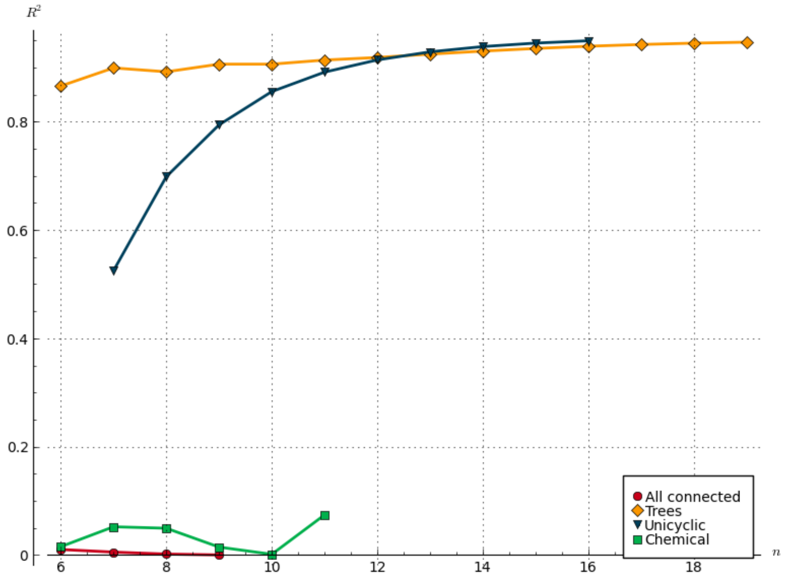


Figure 7.15: Correlation of geometric arithmetic index and fourth spectral moment of Seidel matrix.

non-specific to reveal unexpected correlations. As matter for further research, this class should be divided into more narrow subclasses.

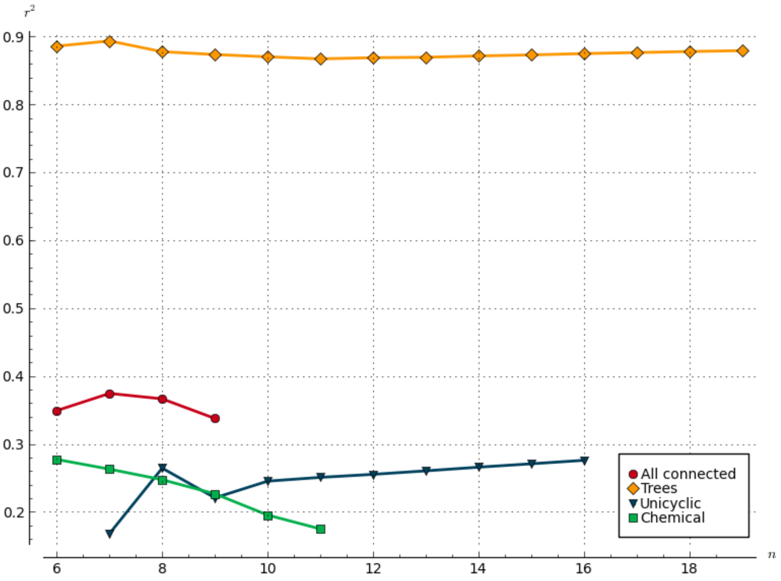


Figure 7.16: Correlation of multiplicative Π_2 Zagreb index and normalized Laplacian energy.

7.2.4 Conclusion

In this section we used MathChem to find several pairs of topological indices based on different properties of graphs that correlates well in various classes of graphs. We did not mentioned those pairs where the curve of correlation coefficient is above 0.9 but decreasing for greater graph order.

The experiment may be extended by introducing new graphs classes, such as bipartite graphs, bicyclic graphs, graphs with given maximum degree Δ, minimum degree δ, graphs with certain chromatic number, etc. On the other hand, new connections may arise by taking into account other topological descriptors, for instance, discrete Adriatic indices.

Chapter 8

Conclusion

In the PhD thesis we introduce Mathchem [84]. Mathchem is a free open-source Python package for calculating topological indices and other invariants of molecular graphs. Together with Sage [78] or SciPy [42] it can be used in QSAR/QSPR research. MathChem is a flexible framework that allows to easily define own topological indices, explore their properties and build conjectures.

Being written in Python and having simple structure, MathChem is clearly readable and easy to learn. In contrast to proprietary software with closed code, MathChem allows to have a total control on calculations and extend the package's functionality. Therefore the product represents a significant contribution to study of topological indices, complex network analysis, and graph theory in general.

Another large part of the PhD thesis is devoted to solving extremal problems of certain Adriatic indices [87] and the first Zagreb index [36]. The conjectures for those problems were obtained with help of MathChem. In particular, the following problems are solved. Upper and lower bounds of inverse sum indeg index of trees, connected graphs, graphs with given number of edges, trees with given number of pendant vertices, connected chemical graphs, graphs with given maximum degree Δ, graphs with given minimum degree δ. Upper and lower bounds of symmetric division deg index of all connected graphs, trees, unicyclic graphs, chemical graphs, graphs with given minimum degree δ. The smallest value of Zagreb M_1 index of trees and unicyclic graphs with fixed independence number α. All the bounds are given together with description of the corresponding extremal graphs.

Eventually, two examples of MathChem usage is given. We have extended default functionality of MathChem to define Adriatic Matrices and introduce Adriatic energy and spectral radius. We studied correlations between these spectral invariants and measured physicochemical properties on the benchmark sets provided by the International Academy of Mathematical Chemistry [59]. There was selected several well-correlated spectral invariants of certain Adriatic matrices that can be of interest of further research. Another example reveals some unexpected correlations between certain topological indices of the different nature. This can be a matter for future research as well.

Bibliography

[1] O. Araujo and J. De la Peña. Some bounds for the connectivity index of a chemical graph. *Journal of chemical information and computer sciences*, 38(5):827–831, 1998.

[2] A. Balaban. Highly discriminating distance-based numerical descriptor. *Chem. Phys. Lett.*, 89:399–404, 1982.

[3] B. Bollobabs and P. Erdos. Graphs of extremal weights. *Ars Combinatoria*, 50:225–234, 1998.

[4] J. Braun, M. Meringer, and C. Rücker. Molecular Structure Generation.

[5] G. Caporossi, I. Gutman, P. Hansen, and L. Pavlović. Graphs with maximum connectivity index. *Computational Biology and Chemistry*, 27(1):85–90, 2003.

[6] ChemAxon. GenerateMD.

[7] J. Chen and X. Guo. Extreme atom-bond connectivity index of graphs. *MATCH Commun. Math. Comput. Chem*, 65:713–722, 2011.

[8] J. Chen and S. Li. On the sum-connectivity index of unicyclic graphs with k pendent vertices. *Mathematical Communications*, 16(2):359–368, 2011.

[9] L. H. Clark and J. W. Moon. On the general randic index for certain families of trees. *Ars Combinatoria*, 54:223–236, 2000.

[10] K. C. Das. Sharp bounds for the sum of the squares of the degrees of a graph. *Kragujevac J. Math*, 25:31–49, 2003.

[11] K. C. Das. Atom-bond connectivity index of graphs. *Discrete Applied Mathematics*, 158(11):1181–1188, 2010.

[12] K. C. Das and I. Gutman. Some properties of the second zagreb index. 2004.

[13] K. C. Das, I. Gutman, and B. Furtula. On atom-bond connectivity index. *Chemical Physics Letters*, 511(4):452–454, 2011.

[14] K. C. Das, I. Gutman, and B. Zhou. New upper bounds on zagreb indices. *Journal of mathematical chemistry*, 46(2):514–521, 2009.

[15] K. C. Das and N. Trinajstić. Comparison between first geometric–arithmetic index and atom-bond connectivity index. *Chemical physics letters*, 497(1):149–151, 2010.

[16] H. Deng. On the sum-balaban index. *Communications in Mathematical and in Computer Chemistry/MATCH*, 66(1):273–284, 2011.

[17] J. Devillers and A. Balaban. *Topological Indices and Related Descriptors in QSAR and QSPR*. Taylor & Francis, 2000.

[18] T. Došlic, B. Furtula, A. Graovac, I. Gutman, S. Moradi, and Z. Yarahmadi. On vertex–degree–based molecular structure descriptors. *MATCH Commun. Math. Comput. Chem*, 66:613–626, 2011.

[19] Z. Du and B. Zhou. On sum-connectivity index of bicyclic graphs. *Bull. Malays. Math. Sci. Soc*, 2009.

[20] Z. Du, B. Zhou, and N. Trinajstić. Minimum general sum-connectivity index of unicyclic graphs. *Journal of mathematical chemistry*, 48(3):697–703, 2010.

[21] Z. Du, B. Zhou, and N. Trinajstić. On the general sum-connectivity index of trees. *Applied Mathematics Letters*, 24(3):402–405, 2011.

[22] Z. Du, B. Zhou, and N. Trinajstic. Sum-connectivity indices of trees and unicyclic graphs of fixed maximum degree. *arXiv preprint arXiv:1210.5043*, 2012.

[23] EduSoft. Molconn-Z.

[24] M. Eliasi, A. Iranmanesh, and I. Gutman. Multiplicative versions of first zagreb index. *Match-Communications in Mathematical and Computer Chemistry*, 68(1):217, 2012.

[25] E. Estrada. Atom–bond connectivity and the energetic of branched alkanes. *Chemical Physics Letters*, 463(4):422–425, 2008.

[26] E. Estrada, L. Torres, L. Rodríguez, and I. Gutman. An atom-bond connectivity index: Modelling the enthalpy of formation of alkanes. *Indian journal of chemistry. Sect. A: Inorganic, physical, theoretical & analytical*, 37(10):849–855, 1998.

[27] G. Fath-Tabar. Old and new zagreb indices of graphs. *MATCH Commun. Math. Comput. Chem.*, 65:79–84, 2011.

[28] B. Furtula, A. Graovac, and D. Vukičević. Atom–bond connectivity index of trees. *Discrete applied mathematics*, 157(13):2828–2835, 2009.

[29] B. Furtula, A. Graovac, and D. Vukičević. Augmented zagreb index. *Journal of mathematical chemistry*, 48(2):370–380, 2010.

[30] L. Gan, H. Hou, and B. Liu. Some results on atom–bond connectivity index of graphs. *MATCH Commun. Math. Comput. Chem*, 66:669–680, 2011.

[31] I. Gutman. Graphs with smallest sum of squares of vertex degrees. *Kragujevac J. Math*, 25:51–54, 2003.

[32] I. Gutman. Multiplicative zagreb indices of trees. *Bull. Soc. Math. Banja Luka*, 18:17–23, 2011.

[33] I. Gutman, O. Araujo, and D. A. Morales. Bounds for the randic connectivity index. *Journal of Chemical Information and Computer Sciences*, 40(3):593–598, 2000.

[34] I. Gutman, K. C. Das, and K. Xu. On zagreb and harary indices. *MATCH Commun. Math. Comput. Chem.*, 70:301–314, 2013.

[35] I. Gutman, B. Furtula, and M. Ivanovic. Notes on trees with minimal atom-bond connectivity index. *Match-Communications in Mathematical and Computer Chemistry*, 67(2):467, 2012.

[36] I. Gutman and N. Trinajstić. Graph theory and molecular orbitals. Total π-electron energy of alternant hydrocarbons. *Chem. Phys. Lett.*, 17:535–538, 1972.

[37] I. Gutman, N. Trinajstić, C. Wilcox Jr, et al. Graph theory and molecular orbitals. XII. acyclic polyenes. *The Journal of Chemical Physics*, 62(9):3399–3405, 1975.

[38] A. A. Hagberg, D. A. Schult, and P. J. Swart. Exploring network structure, dynamics, and function using NetworkX. In *Proceedings of the 7th Python in Science Conference (SciPy2008)*, pages 11–15, Pasadena, CA USA, 2008.

[39] B. Horoldagva and I. Gutman. On some vertex–degree–based graph invariants. *MATCH Commun. Math. Comput. Chem*, 65:723–730, 2011.

[40] Y. Huang, B. Liu, and L. Gan. Augmented zagreb index of connected graphs. *Match-Communications in Mathematical and Computer Chemistry*, 67(2):483, 2012.

[41] W.-D. Ihlenfeldt, J. H. Voigt, B. Bienfait, F. Oellien, and M. C. Nicklaus. Enhanced cactvs browser of the open nci database. *Journal of chemical information and computer sciences*, 42(1):46–57, 2002.

[42] E. Jones, T. Oliphant, P. Peterson, et al. *SciPy: Open source scientific tools for Python*, 2001.

[43] M. Karelson. *Molecular descriptors in QSAR/QSPR.* Wiley-Interscience, New York, 2000.

[44] A. Kerber, R. Laue, M. Meringer, and C. Rücker. MOLGEN-QSPR, a Software Package for the Study of Quantitative Structure Property Relationships. *MATCH Commun. Math. Comput. Chem.*, 51:187–204, 2004.

[45] X. Li, X. Wang, and B. Wei. On the lower and upper bounds for general randic index of chemical (n, m)-graphs. *MATCH Commun. Math. Comput. Chem*, 52:157–166, 2004.

[46] X. Li and Y. Yang. Sharp bounds for the general randic index. *MATCH Commun. Math. Comput. Chem*, 51:155–166, 2004.

[47] X. Li and J. Zheng. A unified approach to the extremal trees for different indices. *MATCH Commun. Math. Comput. Chem*, 54(1):195–208, 2005.

[48] X. Li and J. Zheng. Extremal chemical trees with minimum or maximum general randic index. *MATCH Commun Math Comput Chem*, 55(2):381–390, 2006.

[49] K. Liu, J. Feng, and S. S. Young. PowerMV: A Software Environment for Molecular Viewing, Descriptor Generation, Data Analysis and Hit Evaluation. *Journal of Chemical Information and Modeling*, 45(2):515–522, 2005.

[50] M. Lu, L.-z. Zhang, and F. Tian. On the randić index of acyclic conjugated molecules. *Journal of mathematical chemistry*, 38(4):677–684, 2005.

[51] B. Lučić, S. Nikolić, N. Trinajstić, B. Zhou, and S. Ivaniš Turk. Sum-connectivity index. *Mathematical Chemistry Monographs 8*, 2010.

[52] B. Lučić, N. Trinajstić, and B. Zhou. Comparison between the sum-connectivity index and product-connectivity index for benzenoid hydrocarbons. *Chemical Physics Letters*, 475(1):146–148, 2009.

[53] A. Mauri, V. Consonni, M. Pavan, and R. Todeschini. Dragon 6.

[54] A. Mauri, V. Consonni, M. Pavan, and R. Todeschini. Dragon software: an easy approach to molecular descriptors calculation. *MATCH Commun. Math. Comput. Chem.*, 56:237–248, 2006.

[55] B. McKay and G. Brinkmann. plantri and fullgen.

[56] B. McKay and A. Piperno. nauty and traces.

[57] B. D. McKay. Collections of graphs.

[58] B. D. McKay. graph6 and sparse6 graph formats.

[59] Milano Chemometrics & QSAR Research Group. Molecular Descriptors Dataset.

[60] National University of Singapore. PaDEL-Descriptor.

[61] NetworkX developer team. High-productivity software for complex networks.

[62] S. Nikolic, G. Kovacevic, A. Milicevic, and N. Trinajstic. The zagreb indices 30 years after. *Croatica chemica acta*, 76(2):113–124, 2003.

[63] NumPy developer team. NumPy.

[64] N. O'Boyle, M. Banck, C. James, C. Morley, T. Vandermeersch, and G. Hutchison. Open Babel: An open chemical toolbox. *Journal of Cheminformatics*, 3:33, 2011.

[65] Open Babel developer team. Open Babel: The Open Source Chemistry Toolbox.

[66] X. Pan, H. Liu, and J.-M. Xu. Sharp lower bounds for the general randić index of trees with a given size of matching. *Communications in Mathematical and in Computer Chemistry/MATCH*, 54(2):465–480, 2005.

[67] L. Pavlović. Graphs with extremal randić index when the minimum degree of vertices is two. *Kragujevac Journal of Mathematics*, (25):55–63, 2003.

[68] L. Pavlović and I. Gutman. Graphs with extremal connectivity index. *Novi Sad J. Math*, 31(2):53–58, 2001.

[69] L. Pogliani et al. From molecular connectivity indices to semiempirical connectivity terms: recent trends in graph theoretical descriptors. *Chemical reviews*, 100(10):3827–3858, 2000.

[70] Python Software Foundation. Python programming language—oficial website.

[71] J. Rada, O. Araujo, and I. Gutman. Randic index of benzenoid systems and phenylenes. *Croatica Chemica Acta*, 74(2):225–235, 2001.

[72] M. Randić. On characterization of molecular branching. *J. Am. Chem. Soc.*, 97(23):6609–6615, 1975.

[73] M. Randić. The connectivity index 25 years after. *Journal of Molecular Graphics and Modelling*, 20(1):19–35, 2001.

[74] J. Rodríguez. A spectral approach to the randić index. *Linear algebra and its applications*, 400:339–344, 2005.

[75] G. Royle. Collections of graphs.

[76] Semichem. CODESSA.

[77] J. Stålring, L. Carlsson, P. Almeida, and S. Boyer. AZOrange—High performance open source machine learning for QSAR modeling in a graphical programming environment. *J. Cheminform.*, 3:28, 2011.

[78] W. Stein. Sage: Open Source Mathematics Software.

[79] C. Steinbeck, Y. Han, S. Kuhn, O. Horlacher, E. Luttmann, and E. Willighagen. The Chemistry Development Kit (CDK): An Open-Source Java Library for Chemo- and Bioinformatics. *J. Chem. Inf. Comput. Sci.*, 43:493–500, 2003.

[80] V. Sykora. Chemical Descriptors Library (CDL).

[81] R. Todeschini, D. Ballabio, and V. Consonni. Novel molecular descriptors based on functions of new vertex degrees. 2010.

[82] R. Todeschini, V. Consonni, R. Mannhold, H. Kubinyi, and H. Timmerman. *Handbook of Molecular Descriptors.* Methods and Principles in Medicinal Chemistry. Wiley–VCH, 2008.

[83] A. Vasilyev and D. Stevanović. Mathchem: a python package for calculating topological indices.

[84] A. Vasilyev and D. Stevanović. Mathchem: a python package for calculating topological indices. *MATCH Commun. Math. Comput. Chem.*, 71:657–680, 2014.

[85] D. Vukičević. Bond additive modeling 2. mathematical properties of max-min rodeg index. *Croatica Chemica Acta*, 83(3):261–273, 2010.

[86] D. Vukičević and B. Furtula. Topological index based on the ratios of geometrical and arithmetical means of end-vertex degrees of edges. *Journal of mathematical chemistry*, 46(4):1369–1376, 2009.

[87] D. Vukičević and M. Gašperov. Bond Additive Modeling 1. Adriatic Indices. *Croatica Chemica Acta*, 83(3):243–260, 2010.

[88] D. Vukičević, S. M. Rajtmajer, and N. Trinajstić. Trees with maximal second zagreb index and prescribed number of vertices of the given degree. *Communications in Mathematical and in Computer Chemistry/MATCH*, 60(1):65–70, 2008.

[89] D. Wang, Y. Huang, and B. Liu. Bounds on augmented zagreb index. *Match-Communications in Mathematical and Computer Chemistry*, 68(1):209, 2012.

[90] H. Wiener. Structural determination of paraffin boiling points. *J. Am. Chem. Soc.*, 69:17–20, 1947.

[91] R. Wu and H. Deng. The sum-balaban index of a class of dendrimers. *Journal of Computational and Theoretical Nanoscience*, 9(10):1667–1669, 2012.

[92] R. Xing, B. Zhou, and Z. Du. Further results on atom-bond connectivity index of trees. *Discrete Applied Mathematics*, 158(14):1536–1545, 2010.

[93] R. Xing, B. Zhou, and A. Graovac. On sum-Balaban index. *Ars Combin.*, 104:211–223, 2012.

[94] R. Xing, B. Zhou, and N. Trinajstić. Sum-connectivity index of molecular trees. *Journal of mathematical chemistry*, 48(3):583–591, 2010.

[95] K. Xu and K. Das. Trees, unicyclic, and bicyclic graphs extremal with respect to multiplicative sum zagreb index. *Match-Communications in Mathematical and Computer Chemistry*, 68(1):257, 2012.

[96] Z. Yan, H. Liu, and H. Liu. Sharp bounds for the second zagreb index of unicyclic graphs. *Journal of Mathematical Chemistry*, 42(3):565–574, 2007.

[97] C. Yap. PaDEL-Descriptor: An open source software to calculate molecular descriptors and fingerprints. *J. Comput. Chem.*, 32:1466–1474, 2011.

[98] L.-Z. Zhang, M. Lu, and F. Tian. Maximum randić index on trees with k-pendant vertices. *Journal of mathematical chemistry*, 41(2):161–171, 2007.

[99] B. Zhou and N. Trinajstić. On a novel connectivity index. *Journal of mathematical chemistry*, 46(4):1252–1270, 2009.

[100] B. Zhou and N. Trinajstić. On general sum-connectivity index. *Journal of mathematical chemistry*, 47(1):210–218, 2010.

[101] B. Zhou and N. Trinajstić. On the sum–connectivity matrix and sum-connectivity energy of (molecular) graphs. *Acta Chim. Slov*, 57:513–517, 2010.

[102] B. Zhou and R. Xing. On atom-bond connectivity index. *Zeitschrift fur Naturforschung A-Journal of Physical Sciences*, 66(12):61, 2011.

Index

Adriatic spectral radius, 63

characteristic polynomial, 6
cycle, 5, 6
    Hamiltonian, 6

degree, 5
    maximum, 5
    minimum, 5
descriptor
    bond-additive, 7
    molecular, 7
    topological, 7
discrete Adriatic indices, 10

eigenvalues, 6
energy, 7
    Adriatic, 63
    incidence, 7
    Laplacian, 7

finite graph, 5
forest, 6

girth, 6
graph, 5
    bipartite, 6
    connected, 6
    disconnected, 6
    hamiltonian, 6
    regular, 5
graph isomorphism, 5

index
    ABC, 8
    augmented Zagreb, 9
    geometric-arithmetic, 9
    multiplicative Zagreb, 9
    Randić, 8
    second Zagreb, 8
    sum multiplicative Zagreb, 9
    sum-Balaban, 9
    sum-connectivity, 9
    topological, 7
index of graph, 7
inverse sum indeg index, 11

k-factor, 6

leave, 6

matrix
    adjacency, 6
    Adriatic, 3, 61
    Laplacian, 6
    normalized Laplacian, 6
    signless Laplacian, 6

order of a graph, 5

path, 5, 6
    Hamiltonian, 6

QSAR, 7
QSPR, 7

Sage, 11
set, 5
    independent, 5
    of edges, 5
    of vertices, 5
spectral radius, 7
spectrum, 6
subgraph, 6
    induced, 6
symmetric division deg index, 11

tree, 6

valency, 5
vertex, 5

pendant, 6

walk, 5

Povzetek v slovenskem jeziku

Molekularni zapis, tesno povezan s konceptom molekularne strukture, igra pomembno vlogo v znanstvenem raziskovanju kot teoretično jedro kompleksne mreže znanja.

Molekularni zapisi (oz. deskriptorji) temeljijo na več različnih teorijah kot so kvantna kemija, teorija informacij, organska kemija, teorija grafov ter drugih teorijah, ki so uporabljene pri modeliranju različnih lastnosti kemikalij na znanstvenih področjih kot so toksikologija, analitična kemija, fizikalna kemija ter medicinska, farmacevtska in okoljska kemija.

Kvantitativni strukturno-dejavnostni odnosi (Quantitative Structure-Activity Relationships) oziroma QSAR-ji so končni produkt procesa, ki se začne z ustreznim opisom molekulske strukture in se konča z nekimi ugotovitvami, hipotezami ter predvidevanji o obnašanju molekul v okoljskem, biološkem oziroma fizikalno-kemijskem modelu v analizi. QSAR-ji temeljijo na predpostavki, da mora struktura molekule (na primer njene geometrične, sterične in električne lastnosti) vsebovati značilnosti, ki so vzrok za njene fizikalne, kemijske ter biološke lastnosti, ter na naši sposobnosti opisati te značilnosti z enim ali več molekularnih zapisov. QSAR modeli predpostavljajo, da se lahko biološke aktivnosti (oz. določena lastnost, reaktivnost itd.) na novo zasnovane ali še netestirane kemikalije razbere iz molekulske strukture podobnih spojin, katerih aktivnost (oz. določena lastnost, reaktivnost itd.) je bila že določena [82].

V doktorski disertaciji si smo ogledali določene probleme (ki jih opisali v nadaljevanju) v na novo definiranih molekularnih zapisih.

Definirajmo graf $G = (V, E)$ kot par množic V in E, kjer je V množica vozlišč in E množica povezav grafa G. Ukvarjali se z *enostavnimi* grafi, kar pomeni, da ti ne bodo imeli zank (povezav, ki se začnejo ter končajo v istem vozlišču) ali večkratnih povezav med istima vozliščema.

Mnogi molekularni deskriptorji so sledeče oblike:

$$\sum_{uv \in E} f(g(u), g(v)),$$

kjer je f neka funkcija, $g(u)$ pa ponavadi predstavlja stopnjo ali vsoto razdalj od vozlišča u do preostalih vozlišč grafa G. Oglejmo si nekaj v literaturi najpogosteje proučevani deskriptorji te oblike.

- Najbolj proučevani topološki indeks je *Randić indeks*, poznan tudi kot Indeks povezanosti. Indeks je Milan Randić [72] leta 1975 definiral med študijem karakterizacije molekularnega razvejanja kot vsoto prispevanih vezi v molekuli

$\sum_{uv \in E} \frac{1}{\sqrt{d_u d_v}}$, kjer sta d_u in d_v stopnji odgovarjajočih sosednjih vozlišč v molekularnem grafu [2, 69, 73]. Indeks je bil posplošen na vsoto uteži $\sum_{uv \in E}(d_u d_v)^{\alpha}$. Nekatere meje za Indeks povezanosti kemijskega grafa so bile podane v [1]. Nato so bile najdene stroge meje za Posplošeni Randić indeks [46]. Doseženi so bili razni teoretični rezultati glede ekstremnih vrednosti indeksa [3, 5, 47, 67, 68] za različne razrede grafov: drevesa [9, 66], kemijske grafe [33, 45], sisteme benzenoidov in fenilenov [71], aciklične konjugirane molekule in drevesa z določeno velikostjo prirejanja [50]. Karakteristike skrajnega kemijskega drevesa z minimalnim ali maksimalnim Posplošenim Randić indeksom so bile podane v [48]. Maksimalni Randić indeks na drevesih s k listi je bil proučevan v [98]. Rodríguez je v [74] uporabil spektralni pristop.

- Invarianta grafu poznana pod imenom *drugi Zagreb indeks*, je enaka vsoti produktov stopenj parov sosednjih vozlišč pripadajočega (molekularnega) grafa in je bila prvič predstavljena leta 1972 [36]. Drugi Zagreb indeks ozna"cujemo kraj"se tudi z M_2. Mnogo ugotovitev o M_2, identitete, neenakosti, povezave med M_2-indeksom grafa ter njegovim komplementom, je bilo predstavljenih v [12]. Dokazana je bila zgornja meja za vsoto kvadratov stopenj grafa G [10]. Nato sta bili dokazani stroga zgornja in spodnja meja drugega Zagreb indeksa unicikličnih grafov na n vozliščih s k listi [96]. Drevesa z maksimalnim drugim Zagreb indeksom in določenim številom vozlišč podane stopnje so bila proučevana v [88]. Zatem je bila odkrita zgornja meja za prvi Zagreb indeks M_1 grafa G glede na število vozlišč n, število povezav m, maksimalno stopnjo vozlišča Δ_1, drugo maksimalno stopnjo vozlišča Δ_2 in minimalno stopnjo vozlišča δ. Z uporabo te ugotovitve je bila dokazana nova zgornja meja za M_2 [14]. O Zagreb indeksih obstajajo še ostala dognanja [27, 62].

- *ABC indeks* je definiral Ernesto Estrada v svojem delu, posvečenem modeliranju entalpije formacij alkanov [26]. Formalna definicija je sledeča: $ABC = \sum_{uv \in E} \sqrt{\frac{d_u + d_v - 2}{d_u d_v}}$, kjer sta ponovno d_u in d_v stopnji odgovarjajočih sosednjih vozlišč v molekularnem grafu. V [25] je nadaljeval s proučevanjem indeksa. Ekstremne vrednosti indeksa si je pobližje ogledal v [7, 28]. Zgornja in spodnja meja sta bili podani v [11], kjer je podana tudi karakterizacija grafov, pri katerih so te meje najboljše možne. Nekatere stroge spodnje in zgornje meje za ABC indeks so bile proučevane v [30]. Nadaljnji rezultati za indeks povezanosti atomskih vezi (atom-bond connectivity index – ABC index) so našteti v [92]. Številne lastnosti ABC indeksa so navedene v [13]. Zgornje meje za to invarianto grafa glede na število vozlišč, število povezav, Randićev indeks povezanosti in prvi Zagreb indeks so bile podane v [102]. Nove zgornje meje za dve invarianti grafov, bazirani na stopnji vozlišč (torej Randić indeks in ABC indeks), so bile predstavljene v [39]. Nekaj strukturnih značilnosti dreves z minimalnim ABC indeksom je dokazal Ivan Gutman v [35].

- V zadnjih letih so se pojavili mnogi novi indeksi. *Razširjen Zagreb indeks* je bil definiran v [29], kjer so dobljene tesne zgornje in spodnje meje za kemijska drevesa. Zgornje in spodnje meje za razne razrede povezanih grafov ter karakterizacija odgovarjajočih ekstremnih dreves so bile dokazane v [40]. S podrobnejšo analizo se je nadaljevalo v [89].

- *Indeks vsota-povezanost (degree distance index)* je bil predstavljen kot nov indeks povezanosti in je bil definiran v [99]. To je verzija indeksa povezanosti, ki ga je leta 1975 uvedel Milan Randić, vendar s seštevanjem [51]. Ekstremalni grafi so karakterizirani v [19]. Nekatere lastnosti, predvsem zgornje ter spodnje meje glede na druge invariante grafov za splošni indeks vsota-povezanost (ki posploši tako običajni indeks vsota-povezanost kot tudi prvi Zagreb indeks), so bile objavljene v [100]. Ekstremne vrednosti indeksa so bile proučene za različne razrede grafov: za grafe s k listi [8], za molekularna drevesa [21,94], unициклиčne grafe [20], unициклиčne grafe z določeno maksimalno stopnjo [22]. V [52] sta bila primerjana indeks vsota-povezanost in indeks produkt-povezanost za hidrokarbonske benzenoide. Matrika vsota-povezanost ter energija vsota-povezanost (molekularnega) grafa sta bili opisani v [101].

- Za *Sumativni Balaban indeks* je bilo ugotovljeno, da dobro korelira z določenimi fizikalno-kemijskimi lastnostmi in drugimi topološkimi indeksi oktanov in nižjih benzenoidov [16]. Vrednost indeksa za razred dendrimerov je bil izračunana v [91]. Dokazanih je bilo ve"c spodnjih in zgornjih mej, določena so bila drevesa z največjim, drugim največjim, tretjim največjim ter najmanjšim, drugim najmanjšim in tretjim najmanjšim sumativnim Balaban indeksom med drevesi na n vozliščih za $n \geq 6$ [93].

- *Geometrično-aritmetični indeks* prav tako temelji na stopnji sosednjih vozliče. Indeks je bil predstavljen v [86]. V [15] je bila narejena primerjava med prvim geometrično-aritmetičnim indeksom in ABC indeksom. Nadaljnja analiza je bila izpeljana v [18].

- *Multiplikativni Zagreb indeks.* Analiza indeksa za drevesa je bila podana v [32]. Indeks je omenil tudi R. Todeschini v [81].

- *Multiplikativni sumativni Zagreb indeksi* so eni izmed nazadnje uvedenih topoloških indeksov. Multiplikativna verzija prvega Zagreb indeksa je bila predstavljena v [24]. Karakterizacija dreves, uniciklov in biciklični grafovskih ekstremov (maksimalnih in minimalnih) glede na multiplikativni sumativni Zagreb indeks je bila podana v [95].

V disertaciji smo posebno pozornost posvetili Jadranskim indeksom. Množico 148 diskretnih *Jadranskih indeksov* je definiral D. Vukičević v [87]. Običajna oblika diskretnih Jadranskih indeksov je sledeča:

$$A(G) = \sum_{uv \in E} \gamma_j(\phi_{i,a}(p_u), \phi_{i,a}(p_v)),$$

kjer je p_v lahko ali stopanj vozlišča v, ali vsota razdalj od vozlišča v do vseh ostalih vozlišč grafa G. Funkciji γ_j in $\phi_{i,a}$ sta definirani v [87].

Vsi omenjeni indeksi s svojimi posplošenimi in razširjenimi verzijami so implementirani v paketu Matchem in so bili uporabljeni za analizo korelacije. Poleg tega smo v doktorski disertaciji uporabili ta paket, da pridobimo primere ekstremnih grafov raznih razredov grafov za številne indekse.

Na kratko predstavimo paket Matchem. Matchem je brezplačen odprtokodni Python paket za izračun topoloških indeksov in drugih invariant molekulskih grafov. Skupaj s Sage [78] in SciPy [42] ga lahko uporabimo v QSAR/QSPR raziskavah. Paket omogoča branje raznih kemijskih datotečnih formatov, prenos podatkov s spletne NCI baze podatkov [41], uvoz strukture grafov iz Sage in NetworkX [61] in zapis *graph6* ali *sparse6* niza. Za vse zahtevne izračune smo uporabili koncept lenega računanja. Ta pristop omogoča shranjevanje rezultatov in njihovo ponovno uporabo. Glavne lastnosti tega paketa so naštete odspodaj.

- Vsebuje večino pred kratkim definiranih topoloških indeksov.
- Deluje na več platformah.
- Odprtokoden.
- Zapisan je kot Python paket. Jezika Python se je enostavno priučiti in je zelo razširjen med znanstveniki.
- Je popolnoma zastonj ter brez doplačil ali preizkusnih dob.

Mathchem je bil razvit z uporabo odprto-kodnih paketov omenjenih v znanstvenih člankih kot so ScyPi [42], GraphViz, IPython, Network X [61], Sage [78], OpenBabel [64].

Doktorska disertacija je razdeljena na tri glavne dele. Prvi opisuje program Matchem, ki je bil razvit za potrebe te raziskave. Drugi je sestavljen iz teoretičnih rezultatov, do katerih smo prišli s pomočjo Matchema. Tretji del prikaže uporabo paketa Matchem v raziskavi topoloških indeksov korelacij.

Naši cilji so sledeči:

Ustvarjanje programske opreme za izračun topoloških indeksov

Naš cilj je predstaviti brezplačen odprto-kodni matematični paket, ki omogoča branje in razčlenitev kemijskih podatkov ter računanje topoloških indeksov z uporabo hitrih in pametnih algoritmov.

Aplikacija programskega okvira

Proučevanje korelacij nekaterih topoloških deskriptorjev. Iskanje množic grafov z ekstremnimi vrednostmi različnih indeksov.

Karakterizacija grafov z ekstremnimi vrednostmi določenih Jadranskih indeksov

Dokažemo domneve o določenih Jadranskih indeksih kot so *inverse sum indeg indeks*, *symmetric division deg index* za naslednje razrede grafov: povezani grafi, drevesa, enociklični grafi, kemijski grafi, kemijska drevesa, kemijski enociklični grafi in drevesa s predpisanim številom listov.

Poleg tega karakteriziramo drevesa z ekstremnimi vrednostmi *prvega Zagreb indeksa* z določenim številom *independence number* α.

Spektralne lastnosti Jadranskih matrik

Način, kako so Jadranskih indeksi skonstruirani, nam poda definicijo tako imenovanih Jadranskih matrik, katerih element (u, v) je enak

$$a_{u,v} = \begin{cases} \gamma_j(\phi_{i,a}(p_u), \phi_{i,a}(p_v)), & \text{če je } uv \in E(G), \\ 0 & \text{sicer.} \end{cases}$$

V doktorski disertaciji smo proučevali spektralne lastnosti Jadranskih matrik, če posebej spektralni radius ter energijo, ki ste zelo uporabna v matematični kemiji. Podoben pristop sta uporabila Zhou in Trinajstić za sumativno povezljivostne matrike [101].

Naš cilj je proučiti spektralni radij in energije 148 Jadranskih matrik. Najprej si ogledamo korelacijo IAMC podatkov, da vidimo, katere so matematično pomembne.

Declaration

I declare that this thesis does not contain any materials previously published or written by another person except where due reference is made in the text.

Alexander Vasilyev