Advanced Fractal Graph Theory and Applications

P. Tharaniya, G. Jayalalitha, Pethuru Raj, and B. Sundaravadivazhagan

A **Chapman & Hall** Book

CRC Press
Taylor & Francis Group

Advanced Fractal Graph Theory and Applications

This book explores the dynamic interplay between fractals and graph theory, two powerful mathematical tools with vast applications. It presents a strategic combination and the synergistic use of these disciplines to address real-world problems and challenges. This book begins with an introduction to the basic concepts of fractals and graph theory and then explores their applications in various domains, including natural phenomena modeling, scheduling, and network optimization.

This book:

- Illustrates the innovative ways in which fractals and graph theory can be combined, laying the groundwork for future applications across various industries
- Introduces the fundamental concepts and principles of both fractals and graph theory in detail, making the material accessible to a broad audience, including those new to these topics
- Explores practical applications in image processing, network optimization, social network analysis, and more, demonstrating the real-world impact of these mathematical tools
- Analyzes advanced techniques in graph theory, such as matching, domination, and coloring, with practical examples and case studies
- Highlights the latest research advancements in fractal graph theory, showcasing its potential for future developments and applications

This book is intended for students, researchers, and professionals in mathematics, computer science, engineering, and related fields.

Advanced Fractal Graph Theory and Applications

P. Tharaniya, G. Jayalalitha, Pethuru Raj,
and B. Sundaravadivazhagan

CRC Press
Taylor & Francis Group
Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business
A CHAPMAN & HALL BOOK

Designed cover image: ShutterStock

First edition published 2025
by CRC Press
2385 NW Executive Center Drive, Suite 320, Boca Raton FL 33431

and by CRC Press
4 Park Square, Milton Park, Abingdon, Oxon, OX14 4RN

CRC Press is an imprint of Taylor & Francis Group, LLC

ISBN: 9781032768106 (hbk)
ISBN: 9781032770697 (pbk)
ISBN: 9781003481096 (ebk)

DOI: 10.1201/9781003481096

Typeset in Times
by codeMantra

Contents

Preface

A fractal is a never-ending pattern. Fractals are infinitely complex patterns exhibiting self-similarity across different scales. They possess the innate ability to model complex physical processes and dynamic systems. The central principle of fractals is that a simple process, when repeated infinitely, can lead to highly complex outcomes.

Most fractals operate on the principle of a feedback loop, which is a process where the output of a system is fed back as input, influencing subsequent outputs. A simple operation is performed on a piece of data, which is then fed back into the system. This process is repeated multiple times, and the limit of this process is called the fractal. Fractals are predominantly self-similar, meaning that a part of the fractal is identical to the entire fractal itself.

The fractal dimension is a measure of a fractal object's complexity. It is a ratio that provides a statistical index of complexity, comparing how the detail in a fractal pattern changes with the scale at which it is measured.

Despite their complex and intriguing nature, fractals are surprisingly simple to create. They originate from a fundamental process and gradually become more intricate. Chaos theory also reflects this property, where simple processes can generate complex results. With the aid of high-performing computers, it is now possible to generate and decode fractals, presenting them in graphical representations that are easy to comprehend.

Amazingly, fractals are extremely simple to create and are found throughout nature. These repeating patterns range from the tiny branching of blood vessels and neurons to the branching of trees, lightning bolts, and river networks. Other examples include coastlines, mountains, clouds, seashells, and hurricanes. Abstract fractals, such as the Mandelbrot Set, can be generated by a computer that repeatedly calculates a simple equation. In this book, we begin with graph theory and fractal graph theory. In the third chapter, we will discuss fractal geometry, providing all the relevant theoretical and practical information to enrich our esteemed readers. The fourth chapter covers iterated function systems (IFSs), while the fifth chapter explains how to create fractals from IFSs. Fractals have a dazzling array of industrial applications, including data compression, image processing, computer graphics, and even the design of antennas and microchips, as illustrated in the sixth chapter. The seventh chapter addresses matching and its real-world applications, while the eighth chapter demystifies domination and its practical uses. The ninth chapter vividly describes the aspect of coloring and its corresponding applications. Healthcare is a promising domain for smartly applying advancements in fractal theory. The 11th chapter details how fractals are useful in circuit theory. The 12th chapter is dedicated to expounding on the advantages of fractals in architecture. Finally, the last chapter discusses fractal neural networks and their unique industrial use cases.

Authors

Dr. P. Tharaniya

In 2022, Dr. P. Tharaniya obtained her Ph.D. in the Department of Mathematics from VELS Institute of Science, Technology, and Advanced Studies, Chennai. Currently, she is working as an Assistant Professor in the Department of Mathematics at Rajalakshmi Institute of Technology, Chennai. She has over 14 years of teaching experience and more than 5 years of research experience. She has presented at more than ten international and national conferences and has attended numerous webinars, faculty development programs, and workshops related to graph theory. She has published 12 technical papers in journals and conferences. Her research interests include fractal graph theory, matching, domination, and coloring.

Dr. G. Jayalalitha

In 2011, Dr. G. Jayalalitha obtained her Ph.D. in Mathematics from Sri Chandrasekharendra Saraswathi Vishwa Mahavidyalaya. She is a lifetime member of the Indian Mathematical Society, MISTE, and the Association of Mathematics Teachers of India. Currently, she is a Professor in the Department of Mathematics at VELS Institute of Science, Technology, and Advanced Studies, Chennai. She has nearly 18 years of teaching and research experience. She has published nearly 120 articles and supervised 11 Ph.D. scholars. She has presented at more than ten international and national conferences and has served as a resource person and keynote speaker at over ten programs. Additionally, she is an advisory board member at VISTAS. Her papers have been cited nearly 158 times, and her h-index is 7. Her research interests include fractals, mathematical modeling, operations research, and graph theory.

Pethuru Raj, PhD
Vice President and Chief Architect
Edge AI Division, Reliance Jio Platforms Ltd.
Bangalore, India – 560 103.

Currently, Dr. Pethuru Raj works at Reliance Jio Platforms Ltd. (JPL), Bangalore. Previously, he worked at IBM Global Cloud Center of Excellence, Wipro Consulting Services, and Robert Bosch Corporate Research. He has over 22 years of experience in the IT industry and 8 years of research experience. He completed his CSIR-sponsored Ph.D. at Anna University, Chennai, and continued with UGC-sponsored postdoctoral research in the Department of Computer Science and Automation at the Indian Institute of Science, Bangalore. Subsequently, he was awarded two international research fellowships (JSPS and JST) to work as a research scientist for 3.5 years at two leading Japanese universities. He is a professional member of ACM and IEEE. His focus includes various digital transformation technologies such as the Internet of Things, artificial intelligence (AI), streaming data analytics, blockchain, digital twins, cloud-native computing, edge and serverless computing, reliability engineering, microservices architecture, event-driven architecture, and 5G/6G.

B. Sundaravadivazhagan, Ph.D., SMIEEE
Department of Information Technology
University of Technology and Applied Sciences-Al Mussanah, Oman

B. Sundaravadivazhagan obtained his Ph.D. in computer science from Anna University, Chennai, India. Currently, he is a faculty member in the Department of Information Technology at the University of Technology and Applied Sciences-Al Mussanah in Oman. He holds professional memberships in ISACA and ISTE and is a senior member of IEEE. His academic and research background spans more than 23 years at various institutions. He is currently working on two funded research projects for the Ministry of Higher Education Research Innovation in Oman through the TRC. His research interests include cybersecurity, AI, machine learning, deep learning, and cloud computing. He has published more than 75 technical articles in journals and conferences around the world.

1 Graph Theory – An Overview

1.1 INTRODUCTION

Mathematics plays a crucial role in various fields, such as the natural sciences, engineering, health, finance, and social sciences. The study of applied mathematics has led to the development of completely new fields of mathematics, including statistics and game theory. An arrangement that is fundamentally a collection of objects, some of which are "related" to one another, is called a graph. Graphs are commonly represented diagrammatically by a set of lines or curves that connect the edges and a set of dots or circles that represent the vertices. Graphs are one of the subjects taught in discrete mathematics. Among the key subjects covered in discrete mathematics is graph. It is established that directed graphs in which edges connect two vertices asymmetrically and undirected graphs in which edges connect two vertices symmetrically are different from one other. Graph theory, a branch of mathematics is dedicated to the study of graphs [1].

Graph theory is based on combination logic, and "graphics" are solely used to visualize data. Graph-theoretic models and applications typically involve definition and computational techniques provided by combinatorial mathematics and linear-algebra on the one hand, and linkages to the "real world" on the other hand (sometimes described in vivid graphical terms). Graph theory is fascinating to many because of this interaction [2]. Simple algorithms for planarity testing and graph drawing are presented in a section of graph theory that focuses on the graphical representation and drawing of graphs. However, this topic is addressed in a rather cursory manner; a deeper analysis would necessitate conclusions from more complex area such as curve theory and topology. This section also provides a succinct overview of matroids, a useful generalization that can be used in place of graphs.

Graph-theoretic findings and methods are typically not proven in a strictly combinatorial form; instead, they leverage the visualization opportunities provided by graphical presentations [3]. Depending on their structure, graphs can be characterized by a variety of attributes that they possess. These characteristics are defined in language peculiar to the field of graph theory. It is the number of edges in the shortest path connecting vertices U and V. Graph theory is applied across many engineering domains. For example, circuit connection design heavily relies on graph theory principles.

Topologies refer to the categories or configurations of connections. Star, bridge, series, and parallel topologies are a few types of topologies. The interactions between interconnected computers are governed by graph theory. Graphs are used to illustrate chemical and molecular structures of substances, DNA structures in living

DOI: 10.1201/9781003481096-1

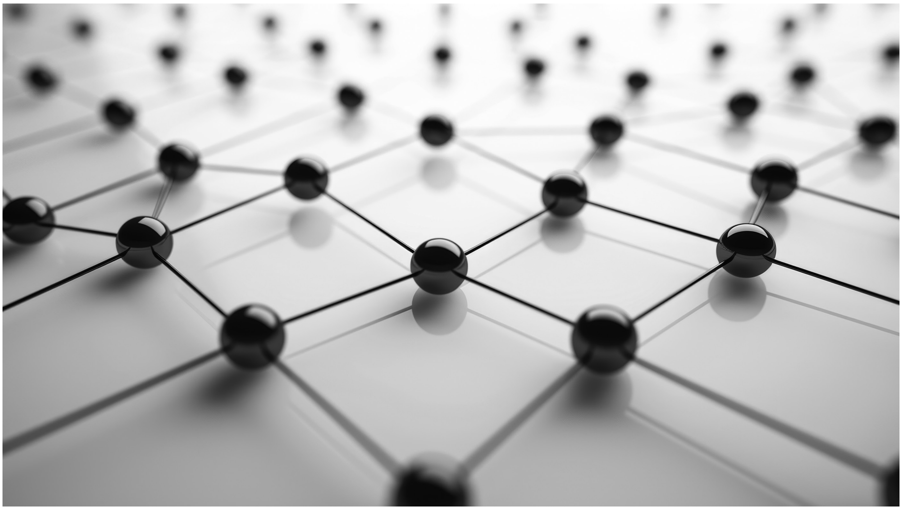

FIGURE 1.1 Example of graph.

things, and even grammar and language parsing trees. They can also illustrate routes between cities. A particular type of graph known as a tree can be used to represent hierarchically organized data, such as family trees.

1.2 DEFINITIONS

1.2.1 Graph

Graphs are mathematical structures made up of a set of nodes (also known as vertices) and a set of edges. They model pair-wise relationships between items in a given collection. In a plane, edges are represented as line segments connecting the vertices, which are depicted as points.

1.2.2 Directed and Undirected Graph

A graph with undirected edges is referred to as an undirected graph. A graph with directed edges is called a directed graph.

Figure 1.2 shows the example of a directed graph.

1.2.3 Connected Graph

A graph is considered to be linked if a path connects all its vertices. In a directed graph, if directed edge is transformed into an undirected edge, the directed graph is said to have weak connectivity. If at least one vertex in a simple directed graph can be adjacent with another vertex, the graph is said to be unilaterally connected. If both vertices in a directed graph are reachable from each other, the graph is said to be highly linked.

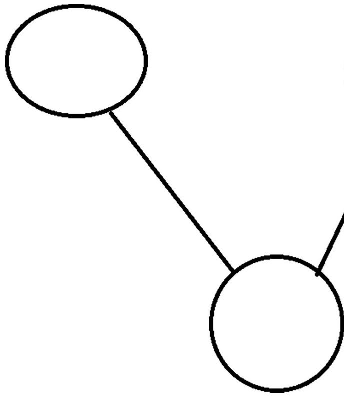

FIGURE 1.2 Directed graph and undirected graph.

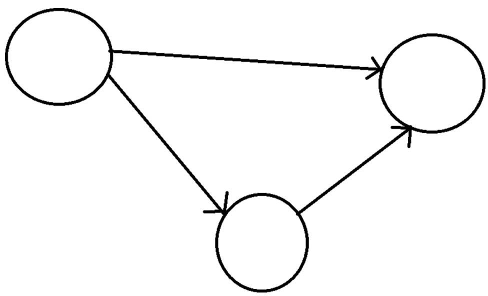

FIGURE 1.3 Weakly connected.

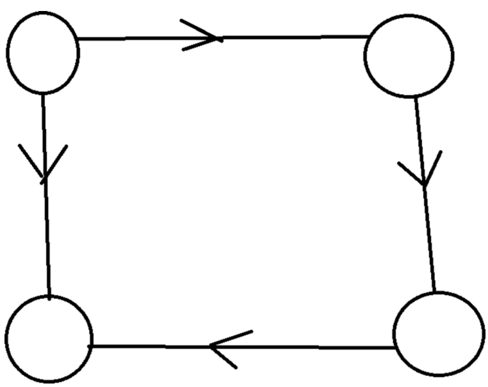

FIGURE 1.4 Unilaterally connected.

Figure 1.3 shows an example of a weakly connected graph, Figure 1.4 shows an example of a unilaterally connected, and Figure 1.5 shows an example of a strongly connected graph.

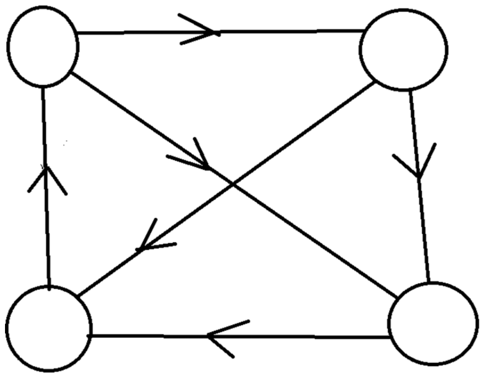

FIGURE 1.5 Strongly connected.

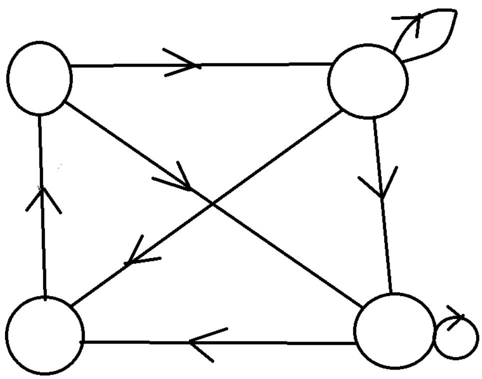

FIGURE 1.6 Loop.

1.2.4 Loop and Parallel Edges

A loop is made up of edges that are drawn from a vertex to itself. When two vertices are joined by more than one edge, the edges are referred to be parallel edges.

Figure 1.6 shows an example of a loop.

1.2.5 Simple Graph

A simple graph is defined as G= (V, E) if it has no loops and no multiple edges, or parallel edges.

Figure 1.7 shows an example of a simple graph.

1.3 EDGES AND VERTICES

A vertex, also referred to as a node (plural: vertices), is the fundamental building block of graphs. In an undirected graph, the structure consists of sets of vertices and edges, unordered pairs of vertices. In contrast, a directed graph is composed of

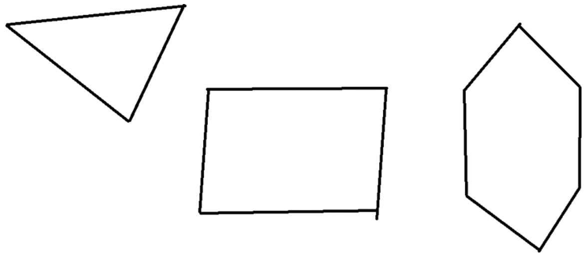

FIGURE 1.7 Simple graph.

ordered pairs of vertices called arcs. In graph diagrams, a line or arrow connecting two vertex points is commonly used to represent an edge. A vertex is usually represented as a circle with a label. The number of edges incident to a vertex is called the degree of the vertex. The number of outgoing edges is considered as the out-degree and the numbers of incoming edges is considered as the in-degree.

1.3.1 Degree of the Vertex

The degree of a vertex represented by the symbol (v) in a graph, is the number of edges incident to a vertex. In a directed graph, the out-degree, or the number of outgoing edges, is represented by $\delta+(v)$, while the in-degree, or the number of incoming edges, is represented by $\delta-(v)$.

1.3.2 Types of Vertices

A vertex with degree zero is called as isolated vertex. A vertex with one degree is called as a leaf vertex. A vertex with in-degree zero is called a source vertex, while a vertex with out-degree zero is called as sink vertex. When less than k vertices are removed from a graph, the remaining graph remains linked, which is known as a k-vertex-connected graph. If removing a vertex with all incident edges results in a subgraph with more linked elements is known as cut point. A cut edge, often called a bridge, is an edge that removed from a graph creates a new graph with more connected components. A cut set S satisfies the condition that S is a subset of E. A linked graph G becomes disconnected when its edges are removed. No proper subset of G satisfies this requirement.

1.3.3 Vertex Cover

An independent set is a set of vertices in which no two vertices are adjacent. A vertex cover is a set of vertices that contains at least one endpoint of each edge in the graph.

1.3.4 Vertex Space

The vertex space of a graph is a vector space having a set of basis vectors corresponding to the graph's vertices.

1.3.5 Vertex Transitive

A graph is vertex-transitive if it has symmetries that can map any vertex to any other vertex.

1.3.6 Labeled Vertex

A labeled vertex is a vertex that has additional information attached, making it distinguishable from other labeled vertices. Two graphs be said to be isomorphic only when the vertices between two graphs are matched based on equivalent labels. Unlabeled vertices are those that can be used in place of any other vertex in the network based on their adjacencies, without any additional information.

1.3.7 Edge Connectivity

In a connected graph, the minimum number of edges that must be removed is known as edge connectivity. λ(G) represents the edge connectivity of a connected graph G if G is a disconnected graph, linked graph G with a bridge has an edge connectivity of 1.

1.3.8 Vertex Connectivity

In a connected graph, the minimum number of vertices that must be removed is known as vertex connectivity. The vertex connectivity of a linked graph is represented by either V(G) or k(G).

i. E(G) = 0 if G is a disconnected graph.
ii. A linked graph G with a bridge has an edge connectivity of 1.
iii. Deleting a single vertex does not disconnect the entire graph k_n, but deleting n – 1 vertices reduces it to a simple graph. Hence, n – 1 = $k(k_n)$.
iv. A graph of order at least has one vertex connectivity if and only if it has a cut vertex.
v. The vertex connectivity of a path is one

1.4 TYPES OF EDGES

Numerous data types can be represented by networks. In biological networks, the nodes represent various entities (such as proteins or genes), while the edges provide information about the connections between these nodes. We will focus on the edges first. The type of edge information determines the type of analyses that can be carried out. Therefore it is helpful to identify the main type of edges that exist in networks. Undirected edges: Typically, directed edges are represented as arrows pointing from the origin vertex—also known as the tail of the arrow toward the destination vertex – also known as the head of the arrow. Because directed graphs do not impose the restrictive requirement of symmetry in the relationships described by the edges, they are considered the most general type of graph.

Figure 1.8 illustrates an example of directed edges.

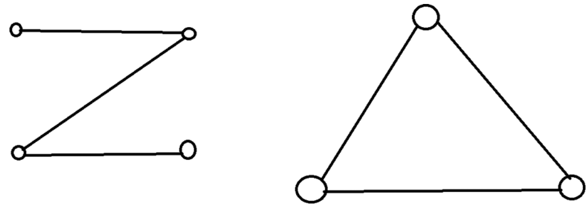

FIGURE 1.8 Undirected edges.

1.4.1 Directed Edges

In sparse networks directed graphs provide more information than similar undirected graphs. This implies that we likely to lose information if we treat a sparse directed graph as undirected. One relevant example is the construction of genealogical trees, where the relationship is "a child of" is significant. Undirected graphs are not inherently transitive, but they work well for relationships where the existence of connections is important. For instance, we can represent pedestrian pathways as an undirected graph if they allow travel in both directions.

1.4.2 Undirected Edges

Undirected edges transmit mutual understanding and connection in both directions, much like whispers between friends. They represent the soft dance of reciprocity, where power or hierarchy hold no sway over the free exchange of ideas. In a network, undirected edges embody the essence of reciprocal relationships, wherein communication is two-way and every node holds equal importance. They promote cooperation and harmony by enabling information, energy, or influence to flow effortlessly between connected nodes akin to an equal-opportunity dialogue. Undirected edges encourage the exploration of interconnectedness, nodes engage in both giving and receiving, resulting in an interdependent web that strengthens the network's structure. In the creative domain, undirected edges ignite the flow of inspiration and ideas, launching cooperative endeavours that transcend individual boundaries. As nodes interact and influence one another in a symphony of shared invention, they invite investigation and discovery. Ultimately, undirected edges serve as a reminder of the beauty of reciprocity and mutual respect, where relationships flourish in an environment equality and openness, allowing communication to flow freely. They subtle highlight the beauty of connection and the strength of teamwork in shaping the fabric of our shared experiences.

Figure 1.9 will provide an example of undirected edges.

1.4.3 Weighted Edges

Numerical values can be assigned to both directed and undirected edges. This is used to illustrate concepts such as gene-on-gene interacting, the quantitative difference in expression that one gene causes over another, or the degree of sequence similarity between two genes. Additionally weights based on edge centrality values and various other topological factors can be applied

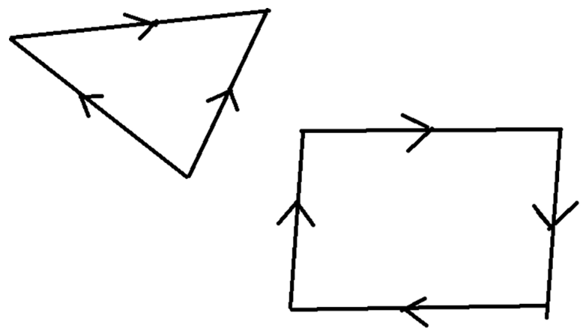

FIGURE 1.9 Directed edges.

1.5 FUZZY GRAPH

Graphs often do not accurately represent all systems due to uncertainty or ambiguity in the parameters of those systems. Crisp graphs and fuzzy graphs are structurally similar. However, fuzzy graphs are particularly important when there is uncertainty regarding vertices and / or edges. A fuzzy graph $\xi = (V, \sigma, \mu)$ is an algebraic structure consisting of a non-empty set V together with a pair of functions $\sigma : V \rightarrow [0,1]$ and μ: $V \times V \rightarrow [0,1]$ such that for all x, y $\in$ V, $\mu(x, y) \leq \sigma(x) \wedge \sigma(y)$ and μ is a symmetric fuzzy relation on σ. Here $\sigma(x)$ and $\mu(x, y)$ represent the membership values of the vertex x and of the edge (x, y) in ξ respectively. The fuzzy graph $\xi_1 = (V, \sigma_1, \mu_1)$ is called a fuzzy sub graph of $\xi = (V, \sigma, \mu)$ if $\sigma_1(x) \leq \sigma(x)$ for all x and $\mu_1(x, y) \leq \mu(x, y)$ for all edges (x, y), x, y $\in$ V.

1.5.1 Strongest Fuzzy Graph

For the fuzzy graph $\xi = (V, \sigma, \mu)$, an edge (x, y), where x, y $\in$ V is called strong if $1/2\,[\{\sigma(x) \wedge \sigma(y)\}] \leq \mu(x, y)$ and it is called weak otherwise. The strength of an edge (u, v) is denoted by $I(u, v) = \mu(u, v) / \sigma(u) \wedge \sigma(v)$.

The strength of a path is defined as min $\{\mu(x_{i1}, x_i), i=1,2,3.....,n\}$. In other words, the strength of a path is the weight (membership value) of the weakest arc of the path. The strength of connectedness between two nodes x and y is defined as the maximum of the strengths of all paths between x and y.

A function F defined on some set X with real or complex values is called bounded, if the set of its value is bounded [12]. In other words, there exists a real number M such that $|f(x)| \leq M$ for all x in X. If the sequence a_n is either monotone increasing or monotone decreasing, then a_n is said to be monotone. If the sequence a_n is monotone increasing and bounded above, then a_n converges. Likewise, if an is monotone decreasing and bounded below then a_n converges. This thesis explains the existence of fuzzy fractals in many fields. Fuzzy number, which is an extension of real numbers, has properties that can be related to the theory of numbers. It is widely used in engineering applications because of its suitability for representing uncertain information

1.6 HAMILTON GRAPH

A graph with a Hamiltonian cycle is known as a Hamiltonian graph. A Hamiltonian cycle visits each vertex in the graph exactly once, with the beginning and ending vertices being the same. In other words, a Hamiltonian cycle is a closed loop that circles each graph vertex precisely once. There is no straightforward prerequisite for a graph to be Hamiltonian. However, several well-known types of Hamiltonian graphs exist, such cycle graphs and complete graphs, which are graphs in which every pair of distinct vertices is connected by an edge. A path in a graph that visits each vertex exactly once is called a Hamiltonian path.

A graph is referred to as traceable if it has a Hamiltonian path but not necessarily a Hamiltonian cycle. Applications for Hamiltonian graphs can be found in many domains, including optimization, network architecture, and computer science. For instance, in the traveling salesman problem, finding the shortest path that makes exactly one stop in each city before returning to the starting point corresponds to identifying a Hamiltonian cycle in a weighted graph. As fundamental concepts in graph theory and combinatorial optimization, Hamiltonian graphs and cycles have a wide range of theoretical and practical applications. Despite the challenges in determining whether a graph is Hamiltonian, the study of Hamiltonian graphs remains a hot topic in computer science and mathematics research.

1.7 THE ORIGIN OF THE GRAPH THEORY

Graph theory delves into the examination of connections among objects, depicted through vertices and interconnecting lines known as edges. Such structures are commonly referred to as graphs. Originating in the 18th century, the evolution of graph theory can be traced through various mathematical problems and puzzles. A concise historical overview is provided below. The 18th-century quandary emerged in the city of Konigsberg (present-day Kaliningrad, Russia). The city, spanning both banks of the Pregel River, featured two sizable islands linked by seven bridges. The puzzle posed the question of whether one could stroll through the city, traversing each of the seven bridges precisely once, and ultimately returning to the initial point. Geography of Konigsberg: The city comprised four distinct areas, two riverbanks and two islands, linked by a total of seven bridges. The identified land masses were Kneiphof, Lomse, and the two riverbanks.

Bridge A: Connecting the two riverbanks
Bridge B: Connecting one of the riverbanks to Kneiphof.
Bridge C: Connecting Kneiphof to the other riverbank
Bridge D: Connecting Kneiphof to Lomse
Bridge E: Connecting Lomse to one of the riverbanks
Bridge F: Connecting Lomse to the other riverbank

The task involved discovering a path within the city that would traverse each bridge precisely once and lead back to the initial point. This challenge captivated the residents of Konigsberg and transformed into a widely embraced puzzle within the city.

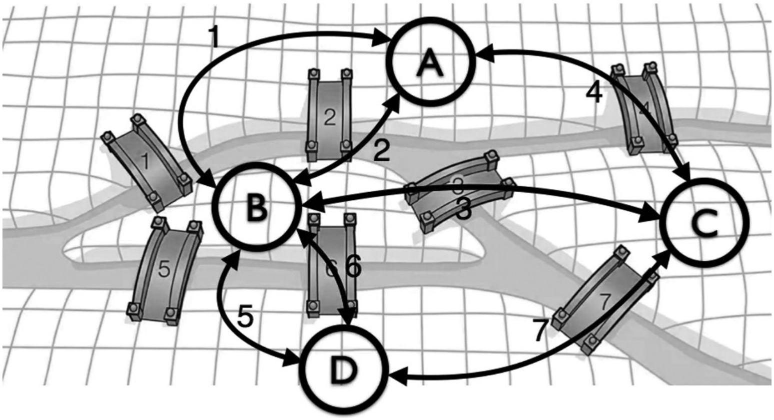

FIGURE 1.10 Konigsberg bridge.

Euler's Seven Bridges of Konigsberg (1736): Graph theory is commonly credited to Leonhard Euler, a Swiss mathematician, who in 1736 successfully tackled the renowned Seven Bridges of Konigsberg problem. Euler approached the challenge uniquely by transforming the city's layout into a graph. His abstraction proved beyond dispute that it is impossible to find a path across the city that crosses each bridge exactly once.

This ground-breaking solution served as the cornerstone for graph theory, and Euler's paper on the matter, titled "Solutio problematis ad geometriam situs pertinentis" (The Solution of a Problem Relating to the Geometry of Position), published in 1736, is widely regarded as the birth of the field. Euler's pioneering work introduced the fundamental concepts of graph theory, establishing it as a distinct and significant branch within mathematics. In 1736, Leonhard Euler published a significant paper titled that the solution of a Problem Relating to the Geometry of Position. This publication holds a pivotal place in the history of mathematics, as it established the groundwork for graph theory. This field, stemming from Euler's foundational work, has since been extensively applied across various disciplines. Background: The issue Euler tackled in his paper stemmed from a widely known puzzle in Konigsberg (present-day Kaliningrad, Russia). The puzzle revolved around the possibility of strolling through the city, crossing all bridges precisely once, and returning to the initial point. Euler approached this challenge by innovatively shifting the perspective, abstracting the city's physical layout into a mathematical structure.

Figure 1.10 illustrates the structure of the Konigsberg Bridge.

1.7.1 Euler's Approach

Euler represented the land masses (islands and riverbanks) as vertices and the bridges as edges. This abstraction led to the creation of a mathematical object that we now call a graph. **Vertices and Edges**: Euler assigned symbols to each land mass and bridge. The four land masses were represented by vertices labeled A, B, C, and D while the bridges were represented by edges. A represents one of the riverbanks, B represents Kneiphof (an island), C represents the other riverbank, and D represents Lomse (the second island).

Edges (Bridges):

1. AB: Bridge connecting the first riverbank (A) to Kneiphof (B)
2. AC: Bridge connecting Kneiphof (B) to the other riverbank (C)
3. BC: Bridge connecting Kneiphof (B) to Lomse (D)
4. CD: Bridge connecting Lomse (D) to the other riverbank (C)
5. DB: Bridge connecting Lomse (D) to the first riverbank (A)
6. BA: Bridge connecting Kneiphof (B) back to the first riverbank (A).
7. CA: Bridge connecting the other riverbank (C) back to the first riverbank (A)

Euler subsequently expressed the problem using the graph and investigated its characteristics, placing particular emphasis on Eulerian paths and circuits. This abstract portrayal enabled Euler to extend his solution, establishing the foundation for the emergence of graph theory. The specific graph used for the Königsberg scenario became a notable illustration in history, and Euler's observations regarding connectivity and paths within graphs significantly influenced the field of mathematics.

1.7.2 Network Analysis

Euler examined the connections between the vertices and edges, transforming the tangible problem into a concern about the structure of this conceptual graph. Eulerian Paths and Circuits: Euler introduced the notions of an Eulerian path and an Eulerian circuit. An Eulerian circuit is a closed path that includes each edge exactly once. Impossibility of the Desired Walk: Euler demonstrated that the challenge of discovering a route through Königsberg. Euler's publication marked the inception of graph theory as an independent field in mathematics. By introducing abstract structures and expressing the problem in terms of vertices and edges, Euler laid the foundation for a robust mathematical framework that extended beyond the confines of the Königsberg problem. This conceptualization empowered mathematicians to construct a comprehensive theory applicable to diverse problems concerning relationships and connectivity. Legacy: The Königsberg Bridge problem and Euler's resolution laid the groundwork for the establishment of graph theory as a formal discipline. Today, graph theory is an essential component of discrete mathematics, finding practical applications in computer science, network analysis, operations research, and various other fields. Euler's contributions are revered as a pivotal force in the history of mathematics, playing a foundational role in shaping contemporary graph theory. Kirochhoff's Matrix Tree Theorem:Gustav Kirchhoff employed matrices to depict graphs in his exploration of electrical networks. His Matrix Tree Theorem, introduced in 1847, offered a powerful method for enumerating spanning trees.

Figure 1.11 shows an example of the Konigsberg Bridge.

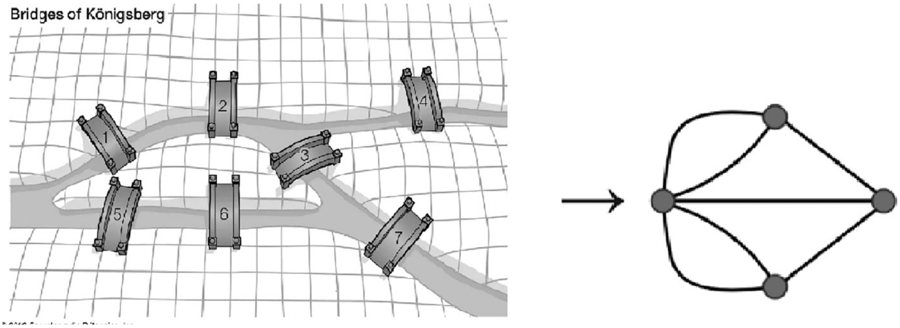

FIGURE 1.11 Euler solution to the Konigsberg problem.

1.8 GRAPH THEORY THROUGH THE CENTURIES

1.8.1 Graph Theory in 19th century

During the 19th century, mathematicians like August Möbius and Listing made significant contributions to topology, which encompasses the study of spaces and their properties. Graphs, particularly those representing surfaces, played a major role in enhancing the understanding of the properties associated with these spaces.

1.8.1.1 Networks and Transportation Planning

While Euler's contributions marked the inception of graph theory, the 19th century saw further progress and utilization of these principles. It is noteworthy that the term "graph theory" may not have been widely used during this era. Advancements in transportation, such as the establishment of railways and telecommunication networks, were particularly significant. Informal applications of graph theory concepts were evident in the planning and optimization of these networks. Engineers and planner, tasked with designing and organizing efficient routes and connections for transportation systems, likely employed graph theory principles, even if they were not explicitly identified as such.

1.8.1.2 Map coloring

The concept of map coloring, which is connected to graph theory, found practical applications in political geography. In the 19th century, the four-color theorem, a result from graph theory, answered the question of how to color every map using only four colors, ensuring that adjacent regions are colored distinctly. This theorem is significant for the practical aspects of creating maps and flags, as it allows for the design using a minimal color palette.

1.8.1.3 Electrical Network Analysis

In the 19th century, physicist Gustav Kirchhoff, contributed to the advancement of electrical circuit theory. Although Kirchhoff's laws are primarily associated with circuit analysis, his research explored the use of matrices to depict networks of electrical elements. The principles he employed connections to later concepts in graph theory, particularly in the analysis of network structures.

1.8.1.4 Combinatorics

While the specific term "graph theory" may not have been widely used in the 19th century, the foundational ideas of graph theory were actively explored and applied across various disciplines. The formalization and recognition of these ideas occurred in the 20th century, driven by the field's expansion and the development of specific terminology and notation

1.8.2 Graph Theory as a Formal Field (20th Century)

Although the specific term "graph theory" was not coined until this era, mathematicians in the 19th century explored into several mathematical concepts that are now associated with graph theory. During this period, mathematicians such as Frank Harary and Paul Erdős made notable contributions to the formalization and development of graph theory as a distinct field.

1.8.2.1 Map Coloring and the Four Color Theorem:

During the mid-19th century, mathematicians and cartographers engaged with the challenge of map coloring. The Four-Color Theorem, initially proposed by Francis Guthrie in 1852, emerged as a significant result in graph theory. This theorem posits that any map on a plane can be colored using only four colors, ensuring that adjacent regions do not share the same color. Although the proof of the theorem did not come until the 20th century, the problem and its investigation involved the examination of planar graphs and coloring.

1.8.2.2 Hamiltonian Paths and Circuits

Hamiltonian paths involve traversing each vertex exactly once, while Hamiltonian circuits are closed paths that include every vertex exactly once. Although Hamilton's primary focus was not explicitly on graphs, his ideas served as foundational elements for subsequent developments in graph theory. Mathematicians continued to investigate these concepts to solve problems related to the traversability of graphs and networks.

1.8.2.3 Networks and Mathematical Physics

In the 19th century, advancements were made in applying mathematics to the physical sciences. Although not explicitly utilizing graph theory, mathematicians like Gustav Kirchhoff employed principles of network theory in the context of electrical circuits, later establishing connections to graph theory. It is crucial to highlight the systematic development and formalization of graph theory. Mathematicians in the 19th century often delved into problems and concepts that laid the groundwork for subsequent developments in graph theory. The terminology and notation associated with graph theory were established in the decades that followed.

1.9 APPLICATIONS AND GROWTH OF GRAPH THEORY

With the increasing prevalence of computers, graph theory has found applications in computer science, operations research, network analysis, and various other fields. Researchers have developed algorithms and methods to address real-world problems using graph theory concepts. Today, graph theory constitutes a fundamental aspect of discrete mathematics with widespread applications across various disciplines, including computer science, social network analysis, biology, and logistics. The field continues to evolve through ongoing research, with new applications emerging regularly.

1.9.1 Graph Theory in Image Processing

In image processing, graph theory is frequently employed to depict and examine the connections among pixels or regions within an image. Utilizing graph-based representations offers a robust framework for capturing the structure of an image and extracting significant information.

The following provides a concise overview of the application of graph theory in image processing.

Figure 1.12 shows an example of a grayscale image and a binary image.

1.9.1.1 Graph Representation

The type of connectivity (e.g. four-connectivity or eight-connectivity) depends on the application.

1.9.1.2 Segmentation

Graph theory applied in image segmentation, aiming to divide an image into significant regions or objects. In the context of segmentation algorithms, the image is often represented as a graph, with each segment treated as a connected component within the graph.

1.9.1.3 Image Representation

Representing images as graphs involves assigning pixels to vertices and indicating relationships between neighboring pixels with edges. This representation facilitates the extraction of structural information from the image.

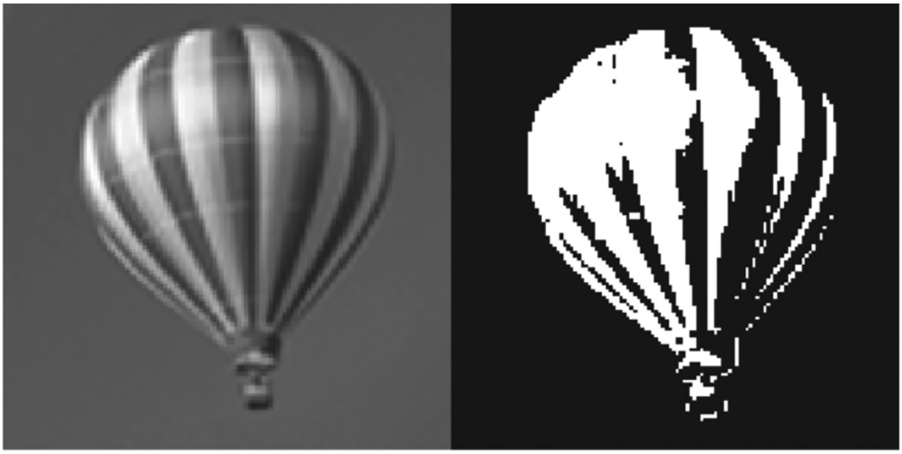

FIGURE 1.12 Grayscale image and binary image.

1.9.1.4 Graph Cuts

Utilizing graph cut algorithms, such as the min-cut/max-flow algorithm, is common in image segmentation. The image is graphically represented, and cuts in the graph correspond to segmenting the image into distinct regions. Min-cut algorithms assist in finding the optimal partitioning of the graph.

1.9.1.5 Image Denoising

Graph-based methods play a role in image denoising by treating noisy pixels as outliers in the graph. Employing graph-based filtering techniques allows for the effective identification and suppression of noisy pixels.

1.9.1.6 Graph-Based Filters

The application of graph filters to an image helps in smoothing or enhancing specific features. For instance, bilateral filtering can be implemented using graph structures to preserve edges while reducing noise.

1.9.1.7 Object Recognition

Graph theory aids in object representation and recognition within images. Graph-based models capture pixel relationships, enabling algorithms to identify patterns or shapes based on graph connectivity.

1.9.1.8 Graph Based Image Retrieval

Graph-based representations allow for indexing and retrieval of images based on similarity measures between graphs. This enables the retrieval of images with similar structures or patterns. In summary, graph theory serves as a versatile framework for various image processing tasks, enabling the modeling of complex relationships, segmentation, denoising, and the extraction of meaningful information.

Figure 1.13 shows an example of the process of filtered graph.

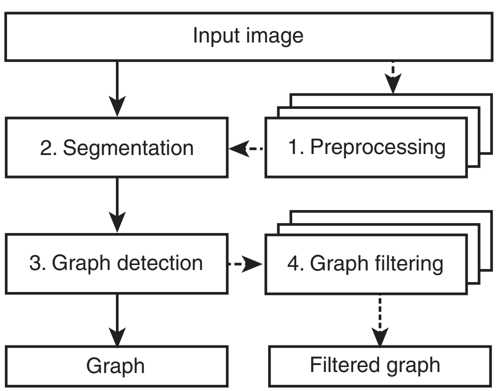

FIGURE 1.13 Segmentation, image representation, graph cuts, and graph based image retrieval.

1.9.2 Application of Network with Graph Theory

Graph theory plays a vital role in analyzing and optimizing diverse networks, with applications across different fields.

1.9.2.1 Social Networks

Community Detection: Identifying groups or communities within social networks based on connections between individuals.

Centrality Measures: Evaluating the importance of nodes within a social network using centrality measures.

1.9.2.2 Computer Networks Routing Algorithms

Designing and analyzing routing algorithms for optimal data transmission paths in computer networks.

1.9.2.3 Network Topology Design

It is Optimizing node layout and connectivity for efficiency and reliability in computer networks.

1.9.2.4 Financial Networks

Analyzing financial transactions and identifying patterns in transaction networks.

1.9.2.5 Wireless Networks

Designing and optimizing communication in networks without a fixed infrastructure.

1.9.2.6 Wireless Sensor Networks

Analyzing connectivity and coverage of sensor networks in various applications

1.9.2.7 Epidemiology Models

Modeling disease spread in populations using graph structures.

1.9.2.8 Internet of Things

It is designing efficient sensor networks for data collection and analysis in Internet of Things applications.

1.9.2.9 Transportation Networks

Route Planning: Determining the shortest or most efficient routes in road, rail, or air networks

1.9.2.10 Traffic Flow Analysis

Studying and optimizing traffic flow in road networks to minimize congestion.

1.9.2.11 Protein–Protein Interaction Networks

Analyzing protein interactions in biological systems to understand cellular processes.

1.9.2.12 Telecommunication Networks

Designing and analyzing the performance of telecommunication networks using graph theory.

1.9.3 Graph Theory with AI

Graph theory expresses intricate linkages and structures, making it a vital tool in many branches of artificial intelligence (AI). In AI, graph theory can be utilized in the following ways.

1.9.3.1 Data Representation

Various data structures, including social networks, biological networks, recommendation systems, and knowledge graphs, can be represented using graphs. Edges signify the connections between nodes, which represent entities. Understanding these relationships is essential for tasks such as social network analysis and recommendation systems in AI applications.

1.9.3.2 Algorithms for Searching

In AI, graph traversal algorithms such as depth-first search (DFS) and breadth-first search (BFS) are essential for navigating extensive state spaces. Pathfinding tasks, including route planning, solving puzzles, and making decisions in games, utilize these methods.

1.9.3.3 Optimization Issues

Optimization issues, such as minimal spanning tree and shortest path finding, employ graph techniques like Dijkstra's algorithm. AI leverages these algorithms for logistics planning, resource allocation, and route optimization.

Clustering and Classification: In AI applications like image segmentation, document clustering, and recommendation systems, related data points are grouped using graph-based clustering algorithms such as spectral clustering and community discovery techniques.

1.9.3.4 Deep Learning on Graphs

Graph neural networks (GNNs), which extend conventional neural networks, operate directly on data organized into graphs. In AI, GNNs have proven effective for tasks including graph classification, link prediction, and node classification, particularly in fields like recommendation systems, social network analysis, and drug discovery. Knowledge Representation and Reasoning: Knowledge graphs organize information using nodes and edges. In AI, knowledge graphs facilitate tasks such as inference, question answering, and semantic search. Algorithms for graph-based reasoning assist in drawing logical conclusions and generating new information from existing knowledge graphs. Fraud Detection and Anomaly Detection: Graph-based anomaly detection algorithms identify unusual behaviors or patterns in network traffic or financial transactions. These algorithms support cybersecurity and fraud detection by analyzing the graph structure and spotting abnormalities.

1.9.3.5 Natural Language Processing

Natural language processing (NLP) employs dependency graphs and semantic graphs to depict the syntactic and semantic relationships between words in a phrase. Graph-based NLP models use these representations for tasks such as semantic parsing, part-of-speech tagging, and named entity recognition. By utilizing graph theory, AI systems can model intricate relationships, make defensible judgments, and extract insightful knowledge from interconnected data, thereby enabling the development of more intelligent and practical solutions across various fields.

1.9.4 Graph Theory in Agriculture

Graph theory can be applied in agriculture in several ways to increase productivity, manage resources, and optimize operations. Here are a few examples:

1.9.4.1 Crop Planning and Rotation

Fields and crops can be represented as nodes and edges, respectively, in graphs. Graph algorithms can optimize crop rotation schedules to promote soil health, avoid pest infestations, and maximize yields over time by analyzing the relationships and compatibility between various crops.

1.9.4.2 Irrigation Network Optimization

Graphs with nodes representing fields or irrigation points and edges representing water pipes or channels can model agricultural irrigation systems. Graph algorithms can efficiently distribute water to different crops based on their water requirements, optimize the irrigation network structure, and reduce water waste.

1.9.4.3 Supply Chain Management

Production facilities, distribution hubs, retail stores, and transportation routes can be represented as nodes in graphs, which simulate supply chains in agriculture. By analyzing the supply chain graph, agricultural enterprises can ensure timely delivery of produce to markets, minimize transportation costs, and improve transportation routes.

1.9.4.4 Management of Pests and Diseases

The spread of pests and diseases in agricultural fields can be represented graphically. Graph algorithms can forecast the growth of infestations and suggest targeted interventions, such as pesticide treatments or quarantine measures, to limit outbreaks by examining the connectivity between various fields and the movement patterns of pests or pathogens.

1.9.4.5 Farm Management System

Graph databases are useful for storing and displaying intricate interactions among various components of a farm management system, including fields, crops, workers, equipment, and environmental conditions. Through graph database queries, farmers can gain insights into crop performance, profitability, and resource use, facilitating data-driven decision-making and improving farm operations

Overall, graph theory offers powerful tools and methods for deciphering complex agricultural systems, streamlining processes, and enhancing the sustainability and productivity of farming practices.

1.9.5 Application of Graph Theory in Industries

Graph theory can represent intricate linkages and structures, making it applicable across various industries. Its applications in the industry are as follows:

1.9.5.1 Network Analysis

In sectors such as logistics, transportation, and telecommunications, graphs are used to simulate different types of networks. Graph algorithms enable capacity planning, routing, and optimization of network infrastructure. For instance, in telecommunications, graph theory enhances bandwidth efficiency, reduces signal interference, and optimizes communication network topology.

1.9.5.2 Management of Supply Chains

Supply chains can be visualized as graphs, where nodes represent various production and distribution phases, and edges signify the movement of products or information between them. Graph algorithms ensure timely product delivery to clients, reduce transportation costs, and optimize supply chain logistics.

1.9.5.3 Social Network Analysis

Graph theory is frequently applied in social network analysis to examine the connections between people or entities across various sectors, including marketing, finance, and healthcare. In social networks, graph algorithms assist in identifying communities, recognizing key nodes, and predicting user behavior [4].

1.9.5.4 Recommendation Systems

In recommendation systems, user-item interactions are modeled using graphs. Nodes represent users and items, while edges depict interactions or connections between them. Graph-based recommendation algorithms analyze relationships between users and items to generate personalized recommendations [6].

1.9.5.5 Fraud Detection

Graph theory is used in fraud detection to analyze relationships between entities, such as customers, transactions, and accounts. In sectors like banking, insurance, and e-commerce, graph algorithms help identify suspicious patterns, prevent financial losses, and detect fraudulent activities.

1.9.5.6 Drug Development

Chemical compounds and their interactions can be visualized as graphs, with nodes representing atoms or molecules and edges representing chemical bonds or interactions. Graph-based algorithms optimize drug candidates for safety and efficacy, aiding drug discovery and development by studying chemical structures and predicting drug-target interactions [5].

1.9.5.7 Cybersecurity

Computer and device networks can be represented as graphs, with nodes signifying individual devices and edges representing communication links or routes. Cybersecurity applications utilize graph-based algorithms to detect malicious activity, identify anomalies, and protect against cyber threats such as malware, phishing, and network attacks.

1.9.5.8 Semantic Web and Knowledge Graphs

In Semantic Web and knowledge graph technologies, data and information are represented as graphs of connected entities and relationships, based on graph theory.

Graph-based knowledge representation facilitates advanced search, data integration, and semantic reasoning in sectors like publishing, e-commerce, and healthcare.

In conclusion, graph theory provides powerful methods and tools for simulating, assessing, and refining intricate systems and networks across diverse sectors, promoting increased productivity, creativity, and decision-making.

1.9.6 Graph Theory in Ornaments

Graph theory has multiple applications in decoration design and analysis.

1.9.6.1 Pattern Generation

Graph theory is a useful tool for creating complex ornamental patterns and designs. In a graph, edges signify connections or links between nodes, which represent individual elements or motifs. By employing techniques such as recursive subdivision or random walks, designers can produce ornament patterns that are both visually appealing and distinctive.

1.9.6.2 Symmetry Analysis

Many decorations rely heavily on symmetry. Graph theory provides tools for examining the symmetry characteristics of ornamental patterns. Graphs that have nodes for motif locations and edges for symmetry transformations can illustrate symmetrical patterns. By analyzing the symmetries inherent in the graph structure, designers can comprehend and control the symmetry characteristics of ornaments.

1.9.6.3 Tessellation and Tiling

The tessellations and tilings of geometric shapes serve as the foundation for many ornaments. The fundamental structure of tessellations can be represented using graph theory, where nodes stand for tiles and edges represent adjacency interactions between them. Techniques such as graph coloring and minimal spanning trees enable designers to create tessellations with the uniformity and visual appeal they desire.

1.9.6.4 Fractal Ornaments

Self-similar patterns with intricate detail at various scales are known as fractals. Fractal ornament patterns can be modeled and analyzed using graph theory. Graphs with nodes representing elements at different scales and edges showing transformations of self-similarity can illustrate fractal patterns. By utilizing graph-based fractal algorithms such as recursive subdivision or iterative refinement, designers can produce visually striking fractal ornaments.

1.9.6.5 Analysis of Complexity

Graph theory offers methods for evaluating the complexity of an ornament design. An ornament's complexity can be assessed using graph-based metrics like graph entropy, degree distribution, or density. By examining these metrics, designers can evaluate ornament designs for visual richness, intricate detail, and aesthetic appeal

Overall, graph theory provides valuable methods and insights for the analysis and design of ornaments, empowering designers to create aesthetically pleasing and structurally sound decorative patterns.

1.9.7 Chemical Graph Theory

Within the field of mathematical chemistry, chemical graph theory applies concepts from graph theory to the study of molecular structure. In this field, molecules are represented as graphs, with chemical bonds depicted as edges (connecting lines between nodes) and atoms as vertices (nodes). The goal of chemical graph theory is to use these graph representations of molecules to investigate and understand various features of molecules.

This involves examining molecular graphs for patterns and themes that may indicate specific chemical properties or behaviors, assessing molecular connectivity, determining molecular symmetry, predicting molecular stability, and analyzing molecular reactions. The application of graph theory in chemistry facilitates the development of computational techniques and algorithms to tackle complex chemical problems, such as designing new compounds, predicting their properties, and understanding their interactions. It provides a robust mathematical framework for comprehending the composition and behavior of molecules.

1.9.8 Graph Theory with Chemical Components

Graph theory can be applied in various ways to evaluate and understand chemical components. Figure 1.14 illustrates the chemical formula of an alkane molecule.

1.9.8.1 Molecular Structure Analysis

In accordance with graph theory, atoms can be represented as vertices and bonds as edges in a graph. Important structural characteristics, such as ring systems, branching patterns, and overall symmetry, can be deduced by examining the topology and atomic connectivity of the molecular graph.

1.9.8.2 Isomer Enumeration

Isomers are compounds that have distinct structural configurations but share the same chemical formula. Graph theory can be employed to generate all non-isomorphic graphs corresponding to valid molecular structures, thereby listing all possible isomers of a given chemical formula.

1.9.8.3 Substructure Looking

This method involves searching for specific molecular fragments or patterns within larger chemical structures using graph theory. Substructure searching is essential in chemical informatics, drug discovery, and chemical synthesis planning.

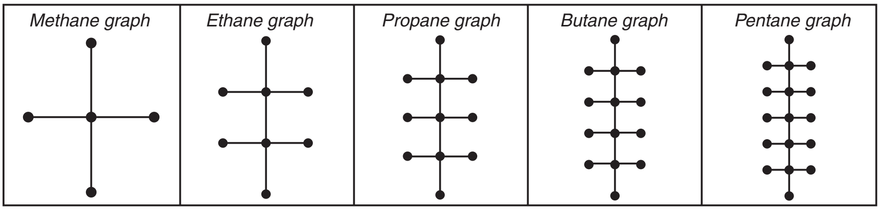

FIGURE 1.14 The structural formula of an alkane molecule with topological index.

1.9.8.4 Molecular Descriptors

These numerical representations, known as graph-based molecular descriptors, are derived from the molecular graph. They represent various topological and structural characteristics of molecules and are utilized in virtual screening, molecular modeling, and studies of quantitative structure–activity relationships (SARs).

1.9.8.5 Networks of Chemical Reactions

Molecular graph transformations can be used to illustrate chemical reactions. By visualizing the network of chemical reactions as a graph, researchers can analyze reaction pathways, identify key intermediates, and understand the kinetics of chemical processes.

1.9.8.6 Chemical Similarity and Clustering

The similarities between molecules are calculated based on their graph representations using graph-based techniques. Similarity metrics derived from graph theory are employed in scaffold hopping in drug development, molecular similarity searching, and grouping molecules based on structural similarity. Overall, graph theory provides a flexible framework for dissecting the structural, topological, and dynamic features of chemical constituents, thereby shedding light on their characteristics, interactions, and behaviors.

1.9.9 Carbon Structures with Graph Theory

Since carbon is a key element in organic chemistry, graph theory is often utilized to explore carbon structures. The following is an application of graph theory to carbon structures: An alkane graph is a tree where the edges represent carbon-carbon or hydrogen-carbon bonds in an alkane, and the vertices represent atoms. An alkane is defined as an acyclic saturated hydrocarbon, which is a molecule made up of carbon and hydrogen atoms arranged in a tree structure with only one carbon-carbon bond. Figure 1.15 shows an example of carbon structures.

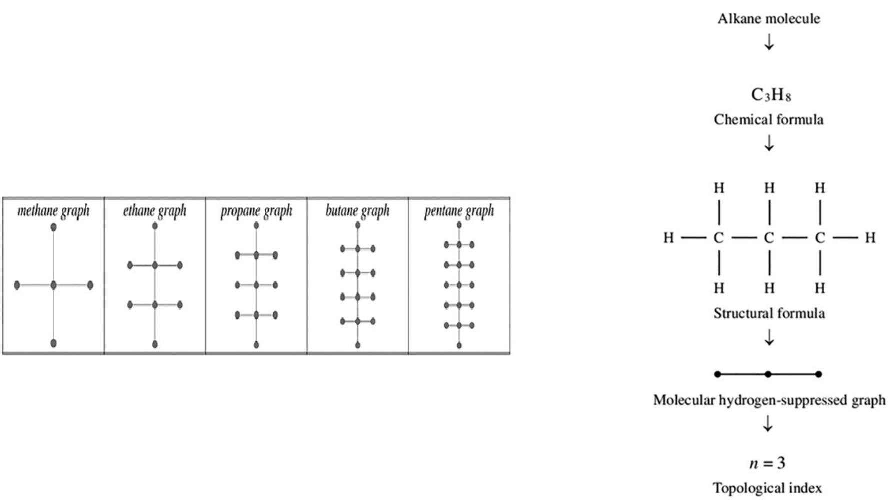

FIGURE 1.15 Example of carbon structures.

1.9.9.1 Representation as Graphs

Chemical bonds (such as single, double, or triple bonds) between carbon atoms are the edges of graphs that depict carbon structures, such as organic compounds. This format enables simplified viewing and study of intricate molecular structures.

1.9.9.2 Topological Analysis

Carbon structures can be analyzed topologically using graph theory. This entails examining the relationships between carbon atoms, recognizing rings and cycles in the molecular graph, and describing the molecule's overall architecture.

1.9.9.3 Isomer Enumeration

Graph theory can be applied to list and categorize isomers of carbon compounds. Isomers are compounds that have distinct structural configurations but the same chemical formula. A given molecular formula can be used to generate and analyze all potential non-isomorphic graphs, which can help detect and classify various types of isomers, including constitutional isomers, stereoisomers, and tautomers.

1.9.9.4 Aromaticity and Conjugation

Graph theory can be used to study aromatic carbon structures, such as benzene and its derivatives. Analyzing the connectivity and resonance structures of the molecular graph can provide insights into aromaticity, which is associated with the presence of conjugated pi electron systems. The stability and reactivity of aromatic rings can be understood and identified with the aid of graph-based techniques.

1.9.9.5 Substructure Searching

Substructure searching in carbon structures utilizes graph theory. Graph matching algorithms are employed to look for specific structural motifs or patterns within larger chemical networks. These patterns include functional groups, substituents, and reaction sites, and are represented as subgraphs. Substructure searches are crucial in chemical informatics, drug discovery, and SAR research.

1.9.9.6 Molecular Descriptors

Graph-based descriptors measure the structural and topological characteristics of carbon compounds. These descriptors are valuable for predicting the physical, chemical, and biological properties of organic molecules, as they capture details about molecular size, shape, branching, and symmetry. In conclusion, graph theory provides a robust framework for examining the composition, characteristics, and reactivity of carbon compounds, enhancing our understanding of the intricate and diverse field of organic chemistry.

1.9.10 Graph Theory with Arts

There are interesting uses for graph theory in many artistic fields. This is how it relates to the arts: Graph theory is frequently employed by artists to create network art, where nodes and edges represent objects and their relationships, respectively.

These networks can depict conceptual links as well as social interactions and communication patterns. Network art explores themes of interconnectivity, intricacy, and emergence and can take various forms, such as interactive installations, visualizations, and installations.

1.9.10.1 Algorithmic Art

Algorithmic art applies mathematical concepts to produce artistic forms and patterns, drawing inspiration from graph-based algorithms. Creative use of algorithms for traversing graphs, such as BFS or DFS, can lead to the creation of complex visual compositions. Artists can investigate the aesthetic properties of graph topologies—such as symmetry, balance, and rhythm—to produce visually striking works of art.

1.9.10.2 Generative Art

Ideas from graph theory often inspire generative art, which employs algorithms to autonomously create artwork. Artists can utilize graph-based procedural generation techniques to create a diverse range of shapes, textures, and patterns. For example, graph coloring algorithms can generate intricate color patterns, while graph grammars can be employed to produce organic shapes or architectural structures.

1.9.10.3 Data Visualization

Artists commonly use data visualization techniques to explore and convey abstract themes, with graph theory being fundamental to the presentation of large datasets. By utilizing graph layout methods and representing data as graphs, artists can create visually engaging representations that communicate information in an intuitive and aesthetically pleasing manner. Data-driven artworks frequently address contemporary themes like social networks, urban dynamics, and environmental trends

1.9.10.4 Interactive Installations

Interactive artworks often incorporate graph theory concepts to create captivating and immersive experiences. Artists can develop interactive exhibits that allow users to manipulate graph topologies through digital or physical interfaces. These installations invite viewers to actively engage with the artwork and influence its evolution while exploring ideas such as collaboration, communication, and emergence.

1.9.10.5 Mathematical Art

Graph theory serves as a rich source of inspiration for artists who work with graph structures to investigate their visual complexity and beauty. Graph-theoretic concepts such as fractals, tessellations, and graph embedding can inspire artists as they create intricate drawings, sculptures, or digital artworks. Mathematical art highlights the intrinsic beauty of mathematical structures, celebrating the interaction of geometry, topology, and aesthetics. In conclusion, graph theory provides a rich environment for artistic inquiry, fostering a variety of artistic expressions that combine mathematical precision with aesthetic awareness. By bridging the gap between art and science, graph theory enhances our understanding of the expressive potential of artistic practice and the structural complexity of networks. Figure 1.16 shows an example of art created using graph theory.

Graph Theory – An Overview

Bondy, J. A. , and Murty, U. S. R. , Graph Theory with Applications, North-Holland, New York (1976).
Bresar, B. , On Vizing's conjecture, discussiones mathematicae, Graph Theory, Vol. 21(1), pp. 5–11 (2001).
Cockayne, E. J. , Gamble, B. , and Shepherd, B. , An upper bound for the kdomination number of a graph, Journal of Graph Theory, Vol. 9(4), pp 533–534 (1985).
West, D. B. , Introduction to Graph Theory, Vol. 2, Prentice Hall, Upper Saddle River, NJ, pp. 1–150 (2001).
Kumar, R. , and Kant, R. , Graph theory, importance and scope, International Global Journal for Research Analysis, Vol. 4(7), pp. 365–367 (2015).
Frankl, P. , and Rödl, V. , Near perfect coverings in graphs and hypergraphs, European Journal of Combinatorics, Vol. 6(4), 317–326 (1985).

Fractal Graph Theory

Andres, J. , and Rypka, M. , Fuzzy fractals and hyperfractals, Fuzzy Sets and Systems, Vol. 300, pp. 40–56 (2016).
Barnsely, M. F. , Fractals Everywhere, 2nd ed., Academic Press, Boston, MA (1993).
Berry, M. V. , Lewis, Z. V. , and Nye, J. F. , On the Weierstrass Mandelbrot fractal function, Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences, Vol. 370(1743), pp. 459–484 (1980).
Falconer, K. J. , Techniques in the Fractal Geometry, John Wiley & Sons, Ltd, Chichester (1997).
Garg, A. , A review on natural phenomenon of fractal geometry, International Journal of Computer Applications, Vol. 975, p. 8887 (2014).
Telcs, A. , Spectra of graphs and fractal dimensions, Journal of Theoretical Probability, Vol. 8, pp. 77–96 (1995).

Fractals and Fractal Geometry

Garg, A. , A review on natural phenomenon of fractal geometry, International Journal of Computer Applications, Vol. 975, p. 8887 (2014).
Barnsely, M. F. , Fractals Everywhere, 2nd ed., Academic Press, Boston, MA (1993).
Mandelbrot, B. , The Fractal Geometry of Nature, Vol. 1, WH Freeman, New York (1982).
Falconer, K. J. , The Geometry of Fractal Sets, Cambridge University Press, Cambridge (1985).
Kigmi, J. , Analysis on Fractals, Cambridge University Press, Cambridge (2011).
Falconer, K. J. , Techniques in the Fractal Geometry, John Wiley & Sons, Ltd, Chichester (1997).

Classical Iterated Function System

Barnsley, M. F. , and Demko, S. , Iterated function systems and the global construction of fractals, Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences, Vol. 399(1817), pp. 243–275 (1985).
Ezhumalai, A. , Ganesan, N. , and Balasubramaniyan, S. , An extensive survey on fractal structures using iterated function system in patch antennas, International Journal of Communication Systems, Vol. 34(15), p. e4932 (2021).
Yu, Z. G. , Anh, V. , and Lau, K. S. , Iterated function system and multifractal analysis of biological sequences, International Journal of Modern Physics B, Vol. 17(22n24), pp.

4367–4375 (2003).
Singh, S. L. , Prasad, B. , and Kumar, A. , Fractals via iterated functions and multifunctions, Chaos, Solitons & Fractals, Vol. 39(3), pp. 1224–1231 (2009).
Garg, A. , Negi, A. , Agrawal, A. , and Latwal, B. , Geometric modelling of complex objects using iterated function system, International Journal of Scientific & Technology Research, Vol. 3(6), p. 6 (2014).
Hart, J. C. , Fractal image compression and recurrent iterated function systems, IEEE Computer Graphics and Applications, Vol. 16(4), pp. 25–33 (1996).
Kominek, J. , Advances in fractal compression for multimedia applications, Multimedia Systems, Vol. 5(4), pp. 255–270 (1997).
Vrscay, E. R. , Iterated function systems: Theory, applications and the inverse problem, In: J. Bélair and S. Dubuc (eds.), Fractal Geometry and Analysis, Springer, Netherlands, Dordrecht, pp. 405–468 (1991).

Fractals from Iterated Function Systems

Riehl, J. R. , and Hespanha, J. P. , Fractal graph optimization algorithms, In Proceedings of the 44th Industrial of Electrical and Electronics Engineers Conference on Decision and Control, Seville, Spain, pp. 2188–2193 (2005).
Kumar, R. , and Kant, R. , Graph theory, importance and scope, International Global Journal for Research Analysis, Vol. 4(7), pp. 365–367 (2015).
Ore, O. , Theory of Graphs, Vol. 38, American Mathematical Society Colloquium Publications, Providence, RI (1962).
Lai, C. , On the size of graphs with all cycle having distinct length, Discrete Mathematics, Vol. 122(3), pp. 363–364 (1993).
Krön, B. , Growth of self-similar graphs, Journal of Graph Theory, Vol. 45(3), pp. 224–239 (2004).

Application of Fractals in Various Fields

Garg, A. , A review on natural phenomenon of fractal geometry, International Journal of Computer Applications, Vol. 975, p. 8887 (2014).
Ankenbrand, T. , and Tomassini, M. , Multivariate time series modelling of financial markets with artificial neural networks, In: Artificial Neural Nets and Genetic Algorithms, Springer, Vienna, pp. 257–260 (1995).
Telcs, A. , Spectra of graphs and fractal dimensions, Journal of theoretical Probability, Vol. 8, pp. 77–96 (1995).
Andres, J. , and Rypka, M. , Fuzzy fractals and hyperfractals, Fuzzy Sets and Systems, Vol. 300, pp. 40–56 (2016).
Mandelbrot, B. , The Fractal Geometry of Nature, Vol. 1, WH Freeman, New York (1982).
Bohdalova, M. , and Gregus, M. , Markets, Information and Their Fractal Analysis, E-Leader, CASA, New York, pp. 1–8 (2010).
Blackledge, J. M. , Evans, A. K. , and Turner, M. J. , Fractal Geometry: Mathematical Methods, Algorithms, Applications, Elsevier (2002).
Berry, M. V. , Lewis, Z. V. , and Nye, J. F. , On the Weierstrass Mandelbrot fractal function, Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences, Vol. 370(1743), pp. 459–484 (1980).

Matching and Its Applications in Real Life

Anitha, A. , Tharaniya, P. , Senthil, S. , and Jayalalitha, G. , Analyzation of maximum matching and maximal matching in various graphs, International Journal of Pharmaceutical Research, Vol. 12(3), pp. 778–782 (2020).

Gutman, I. , Some analytic properties of the independence and matching polynomials, Match, Vol. 28, pp. 139–150 (1992).

Suguna, R. , Tharaniya, P. , and Jayalalitha, G. , Maximum matching in Sierpenski arrow head curves, Journal of Advanced Research in Dynamical & Control Systems, Vol. 12(03–Special Issue), pp. 1380–1385 (2020).

Rabin, M. O. , and Vazirani, V. V. , Maximum matchings in general graphs through randomization, Journal of Algorithms, Vol. 10, pp. 557–567 (1989).

Domination and Its Applications in Real Life

Bollobas, B. , and Cockayne, E. J. , Graph Theoretic Parameters concerning domination, independence and irredundance, Journal of Graph Theory, Vol. 3(3), pp. 241–249 (1979).

Bondy, J. A. , and Murty, U. S. R. , Graph Theory with Applications, North-Holland, New York (1976).

Bresar, B. , Henning, M. A. , and Rall, D. F. , Paired-domination of Cartesian products of graphs and rainbow domination, Electronic Notes in Discrete Mathematics, Vol. 22, pp. 233–237 (2005).

Fomin, F. V. , and Dimitrios, M. , Dominating sets in planar graphs: Branch-width and exponential speed-up, Society for Industrial and Applied Mathematics Journal on Computing, Vol. 36(2), pp. 281–309 (2006).

Cockayne, E. J. , Gamble, B. , and Bruce, S. , An upper bound for the kdomination number of a graph, Journal of Graph Theory, Vol. 9(4), pp. 533–534 (1985).

Bange, D. W. , Barkauskas, A. E. , Host, L. H. , and Slater, P.J. , Efficient near domination of grid graphs, Congressus Numerantium, Vol. 58, pp. 83–92 (1987).

Coloring and Its Applications in Real Life

Ore, O. , The Four Color Problem, Academic Press, New York (1967).

Guptha, P. , and Sikhwa, O. , A study of vertex-edge coloring techniques with application, International Journal of Core Engineering & Management, Vol. 1(2), pp. 27–32 (2014).

Balakrishnan, R. , Wilson, R. J. , and Sethuraman, G. , Graph Theory and Its Applications, Narosa Publications (2015).

Falconer, K. J. , The Geometry of Fractal Sets, Cambridge University Press, Cambridge, 1985.

Guichard, D. R. , No-hole k tuple, (r+1) distant colorings, Discrete Appied Mathematics, Vol. 64(1), pp. 87–92 (1996).

Gyarfas, A. , and Lehel, J. , Online and first fit coloring of graphs, Graph Theory, Vol. 12(2), pp. 217–227 (1988).

Harary, F. , Graph Theory, Addison Wesley Publishing Company, Reading, MA (1969).

Mahapatra, T. , Ghorai, G. , and Pal, M. , Fuzzy fractional coloring of fuzzy graph with its application, Journal of Ambient Intelligence and Humanized Computing, Vol. 11(11), pp. 5771–5784 (2020).

Fractals in Medicine

Huang, K. , and Yan, H. , Offline signature verification based on geometric feature extraction and neural network classification, Pattern Recognition, Vol. 30(1), pp. 9–17 (1997).
Cohn, D. , Optimal systems I: The vascular system, Bulletin of Mathematical Biophysics, Vol. 16(1), pp. 59–74 (1954).
Wagner, C. D. , and Persson, P. B. , Chaos in the cardiovascular system: An update, Cardiovascular Research, Vol. 40(2), pp. 257–264 (1998).
Mandelbrot, B. B. , The Fractal Geometry of Nature, W.H. Freeman, San Francisco, CA, p. 460 (1982).
Hastings, H. M. , and Sugihara, G. , Fractals, A User's Guide for the Natural Sciences, Oxford University Press, New York (1993).
Hutchinson, J. , Fractals and self-similarity, Indiana University Mathematics Journal, Vol. 30(5), pp. 713–747 (1981).
Mandlebrot, B. B. , How long is the coast of Britain? Statistical self similarity and fractional dimension, Science, Vol. 156, pp. 636–638 (1967).
Caldwell, C. B. , Moran, E. L. , and Bogoch, E. R. , Fractal dimension as a measure of altered trabecular bone in experimental inflammatory arthritis. Journal of Bone Minerals Research, Vol. 13(6), pp. 978–985 (1998).
Ott, E. , Chaos, Cambridge University Press, New York (1988).
Baish, J. W. , and Jain, R. K. , Fractals and cancer, Cancer Research, Vol. 60, pp. 3683–3688 (2000).
Fierraz, J. , Ortega-Garcia, J. , Ramos, D. , and Rodriguez, J. G. , HMM based on-line signature verification: Feature extraction and signature modeling, Pattern Recognition Letters, Vol. 28(16), pp. 2325–2334 (2007).
Zook, J. M. , and Iftekharuddin, K. M. , Statistical analysis of fractal-based brain tumor detection algorithms, Magnetic Resonance Imaging, Vol. 23(5), 671–678 (2005).

Fractals in Circuit Theory

Coltman, J. W. , The specification of imaging properties by response to a sine wave input, JOSA, Vol. 44(6), pp. 468–471 (1954).
Corbin, M. J. , and Butler, G. F. , MulTiSIM: An object-based distributed framework for mission simulation, Simulation Practice and Theory, Vol. 3(6), pp. 383–399 (1996).
Falconer, K. J. , Techniques in Fractal Geometry, Wiley, Chichester, p. 3 (1997).
Mahata, S. , Maiti, A. , and Maiti, C. K. , Cost-effectiveweb-based electronics laboratory using NI MultiSim, LabVIEW and ELVIS II, In 2010 International Conference on Technology for Education, pp. 242–243, IEEE (2010). https://doi.org/10.1109/T4E.2010.5550110.
Mandelbrot, B. B. , Fractal geometry: What is it, and what does it do? Proceedings of the Royal Society of London A: Mathematical and Physical Science, Vol. 143(1864), pp. 3–16 (1989).
Xian, F. , and Lai, X. Z. , Based on multisim simulation digital logic experiment teaching reform. Research and Exploration in Laboratory, Vol. 38(9), pp. 228–232, 297 (2019).
Zhengdong, X. L. , et al., Application of multisim simulation software in teaching of analog electronic technology, Journal of Physics: Conference Series, Vol. 1544(1), p. 012063 (2020). https://doi.org/10.1088/1742–6596/1544/1/012063.

Fractals in Architecture

Abboushi, B. , Elzeyadi, I. , Taylor, R. , and Sereno, M. , Fractals in architecture: The visual interest, preference, and mood response to projected fractal light patterns in interior spaces, Journal of Environmental Psychology, Vol. 61, pp. 57–70 (2019).
Fleron, J. F. , A note on the history of the Cantor set and Cantor function, Mathematics Magazine, Vol. 67(2), pp. 136–140 (1994).

Joye, Y. , Fractal architecture could be good for you, Nexus Network Journal, Vol. 9, pp. 311–320 (2007).
Kitchley, J. L. , Fractals in architecture, Architecture and Design-New Delhi, Vol. 20(3), pp. 42–51 (2003).
Lorenz, W. E. , Andres, J. , and Franck, G. , Fractal aesthetics in architecture, Applied Mathematics & Information Sciences, Vol. 11(4), pp. 971–981 (2017).

Fractal Neural Networks and the Industrial Use Cases

Roberto, G. F. , et al., Fractal Neural Network: A New Ensemble of Fractal Geometry and Convolutional Neural Networks for the Classification of Histology Images, https://www.sciencedirect.com/science/article/abs/pii/S0957417420308563.
Larsson, G. , Maire, M. , and Shakhnarovich, G. , FractalNet: Ultra-Deep Neural Networks without Residuals, https://arxiv.org/abs/1605.07648v4.
Narasimhan, R. , Fractal Networks for AI, https://www.linkedin.com/pulse/fractal-networks-take-ai-one-step-closer-natural-narasimhan/
Sohl-Dickstein, J. , The Boundary of Neural Network Trainability is Fractal, https://arxiv.org/pdf/2402.06184v1.
Zuev, S. , et al., Fractal Neural Networks, https://ieeexplore.ieee.org/document/9698649.