Title	重み付き木の定数時間列挙とその応用
Author(s)	銭, 夢沢
Citation	
Issue Date	2025-09
Type	Thesis or Dissertation
Text version	ETD
URL	http://hdl.handle.net/10119/20079
Rights	
Description	Supervisor: 上原 隆平, 先端科学技術研究科, 博士

Abstract

The enumeration problem is a fundamental issue across various disciplines, requiring the output of all elements from a specified set without duplicates or omissions. In graph theory, the enumeration problem of the graph class always restricted by the number of vertices for graphs. Since graph structures form the backbone of many domains within computer science, including algorithms, network theory, computational biology, and data modeling, the enumeration of a specific graph class is an important problem in graph theory and combinatorics. In general, enumeration algorithms are designed individually for each specific graph class. Our motivation is to present a comprehensive framework for graph enumeration that is applicable to various graph classes. To address this, we introduce a modular two-phase framework that is both expressive and widely applicable, capable of exhaustively and non-redundantly enumerating graphs under various structural constraints.

The heart of this framework lies in the block-cutpoint tree (BC-tree). Tree structures hold a central place due to their simplicity and power in representing hierarchical relationships, dependencies, and decompositions. Of particular importance are BC-trees, which are tree representations of connected undirected graphs where the nodes correspond to either blocks (biconnected components) or cutpoints (vertices whose removal increases the number of connected components).

A BC-tree reflects the global structure of a connected graph in terms of its cutpoints and blocks. Given their relevance in areas such as network analysis and graph decomposition, the ability to generate all BC-trees of a fixed size under structural constraints is an essential foundational step toward the enumeration of wider graph families. We propose the first constant time enumeration algorithm for the class of BC-trees. The key feature of this algorithm is its ability to enumerate each satisfied BC-tree exactly once, without duplication, and with a constant time delay per output. The algorithm operates by reverse search, which defines a parent-child relationship among a target set of BC-trees and traverses this implicit family tree of all solutions without constructing it explicitly. Theoretical analysis guarantees the algorithm's correctness and output sensitivity. Previous studies have not thoroughly studied efficient enumeration algorithms for BC-trees, despite their theoretical and practical significance.

To further enrich this framework and address the enumeration problems for wider graph classes, we introduce an enumeration algorithm for rooted vertex-weighted trees (referred to as weighted trees). A weighted tree is a rooted tree where each vertex is

assigned a non-negative integer weight. For the enumeration problem of weighted trees, the tree structure and the total weight are fixed. This model arises in practical domains such as hierarchical clustering, compressed data structures, and frequency-based modeling. In our framework, weighted trees can be used, for example, to determine the number of vertices across the blocks in a graph corresponding to a BC-tree.

The weighted tree enumeration algorithm also uses the reverse search technique and is capable of enumerating all possible weight assignments that meet a specified structure and total weight condition with a constant time delay per tree. The design of the algorithm carefully handles the partitioning of integer weights for subtrees, especially for isomorphic subtrees, while keeping parent-child relationships on weight sequences. The enumeration process avoids redundant generation and guarantees exhaustiveness. Through theoretical analysis, we show that the algorithm is efficient and scalable and that it can be smoothly integrated into the graph enumeration framework to support weighted graph generation. Our framework thus becomes a powerful and flexible tool for the generation of graph classes with specific constraints, such as the vertex number of the graph.

Then, we propose the notion of the enumeration framework. The framework consists of two phases:

The first phase involves enumerating all possible BC-trees that represent the skeletons of the graphs in the target graph class. More specifically, a BC-tree captures the connectivity structure among blocks within a connected graph. This step generates the abstract structure that underlies the target graphs.

In the second phase, for each BC-tree obtained, we enumerate all possible graphs belonging to the target class that correspond to it. It leverages the fact that each undirected connected graph corresponds uniquely to a BC-tree, which effectively prevents redundancy in the enumeration. In other words, the output contains no isomorphism. This phase ensures comprehensive coverage of the target graph class while maintaining efficiency.

As described above, to realize this enumeration framework, the problem can be decomposed into two subproblems: the enumeration of BC-trees and the enumeration of vertex-weighted trees corresponding to each BC-tree. The results of these two subproblems allow us to construct a unified and efficient enumeration algorithm for a wide range of graph classes.

Then, we show instances for the application of the enumeration framework. In this framework, the first phase leverages the constant time BC-tree enumeration algorithm. The second phase depends on the specific graph class being targeted—such as block graphs and cactus graphs—and involves the generation of concrete block structures that satisfy class-specific constraints. For instance, in the case of block graphs, each block must be a clique, while for cactus graphs, each block must be a cycle or an edge. The realization phase may also include conditions on cutpoint positions, block sizes, etc.

We outline the main contributions of this dissertation below. First, we present the first known constant time enumeration algorithm for BC-trees under the constraint of the number of vertices, providing a foundational tool for graph decomposition and reconstruction. Second, we propose the first constant amortized time enumeration algorithm for weighted trees with a fixed weight sum and a tree structure, and demonstrate how it enhances the expressive power and flexibility of our graph enumeration framework. Third, we propose a notion of a two-phase enumeration framework for graph classes that leverages BC-trees as structural skeletons and supports a wide variety of graph classes by varying the block realization rules.

This research significantly advances the understanding of enumeration problems in graph theory and opens new avenues for practical applications. The combination of efficient enumeration techniques and a modular framework enables the systematic study and enumeration of complex graph classes. Potential applications include network analysis (where reliability and redundancy depend on structural decomposition), bioinformatics (where molecular structures are often represented as block graphs or trees), chemistry (for enumerating feasible compound structures), and artificial intelligence (for structured model space exploration). The proposed methods also have implications for database design, compressed representation of hierarchical data, and algorithmic testing through exhaustive instance generation.

In summary, this dissertation proposes the first known constant time enumeration algorithm for BC-trees and the first known constant amortized time enumeration algorithm for weighted trees. It also proposes the notion of a brand-new framework for enumeration through weighted BC-trees, setting the stage for future research into more general and expressive enumeration frameworks.

Keywords: graph enumeration, reverse search, BC-tree, rooted vertex-weighted tree, enumeration framework