## Streszczenie

Homomorfizmem z grafu $G$ w graf $H$ nazywamy funkcję ze zbioru wierzchołków $G$ w zbiór wierzchołków $H$ zachowującą krawędzie. Homomorfizmy w naturalny sposób uogólniają kolorowania grafów; poprawne $k$-kolorowanie grafu $G$ możemy interpretować jako homomorfizm z $G$ w graf pełny $K_k$ i na odwrót. Dla ustalonego grafu $H$ problem homomorfizmu grafów $\textsc{Hom}(H)$ przyjmuje jako instancję graf $G$ i odpowiada na pytanie, czy istnieje homomorfizm z $G$ w $H$.

W niniejszej rozprawie badamy, jak złożoność problemu homomorfizmu grafów (i jego listowej wersji) zależy od struktury instancji. W ogólności, przy standardowych założeniach teorii złożoności, nie istnieją algorytmy rozwiązujące NP-zupełne przypadki problemu $\textsc{Hom}(H)$ istotnie szybciej niż metodą brute-force. Nie wyklucza to jednak możliwości rozwiązania problemu w czasie wielomianowym lub podwykładniczym względem rozmiaru instancji, jeśli nałożymy pewne ograniczenia na klasę tych instancji. Istnienie tego typu algorytmów dla problemu kolorowania zostało przebadane z użyciem wielu metod; w poniższej pracy analizujemy, które z nich można uogólnić na problem homomorfizmu.

Interesują nas dwa rodzaje ograniczeń. W pierwszej kolejności przyjrzymy się klasom instancji, które można otrzymać poprzez ograniczenie pewnego strukturalnego parametru grafu – w naszej pracy będzie to szerokość drzewowa oraz szerokość klikowa. Wiadomo, że jeśli ograniczymy szerokość drzewową lub klikową instancji, wówczas problem homomorfizmu można rozwiązać w czasie wielomianowym. W niniejszej pracy analizujemy zaś dokładnie, jak wygląda zależność funkcji złożoności od parametru. W drugiej części pracy rozważamy klasy instancji, które można zdefiniować poprzez zabranianie ustalonego grafu jako podgrafu indukowanego. Pokazujemy, dla jakich klas, które można otrzymać w ten sposób, istnieją algorytmy rozwiązujące $\textsc{Hom}(H)$ (oraz wariant listowy) istotnie szybciej niż metoda brute-force. Uzupełniamy te wyniki, pokazując dolne ograniczenia przy założeniu standardowych hipotez z teorii złożoności.

Chociaż główna motywacja niniejszej rozprawy wywodzi się z teorii złożoności, warto podkreślić, że przedstawione twierdzenia są wynikiem analizy kombinatorycznych własności grafów oraz ich homomorfizmów, oraz głębokiego zrozumienia ich struktury.



**Słowa kluczowe:** homomorfizmy grafów, listowe homomorfizmy, dziedziczne klasy grafów, szerokość klikowa, szerokość drzewowa, Hipoteza o czasie wykładniczym, Silna hipoteza o czasie wykładniczym

## Abstract

A homomorphism from a graph $G$ to a graph $H$ is an edge-preserving mapping from the vertex set of $G$ to the vertex set of $H$. Graph homomorphisms are a wide and natural generalization of graph colorings; each proper $k$-coloring of a graph $G$ is a homomorphism from $G$ to the complete graph $K_k$ and vice versa. For a fixed graph $H$, the graph homomorphism problem $\textsc{Hom}(H)$ takes a graph $G$ as an instance and asks whether there exists a homomorphism from $G$ to $H$.

The leading objective of this dissertation is to study how the complexity bounds of the graph homomorphism problem (and its list generalization) depend on the structural properties of the instances. In general, the existence of algorithms solving NP-complete cases of $\textsc{Hom}(H)$ significantly faster than brute force is unlikely under standard complexity assumptions. However, it is still possible to find algorithms working in time polynomial, or at least subexponential, in the size of the input, if we put additional assumptions on the class of instances. Many such phenomena were observed in the case of graph colorings. In this work we examine to which extent they can be generalized to graph homomoprhisms.

We are interested in two kinds of restrictions. First, we investigate the classes of instances that can be obtained by bounding a structural parameter of a graph, i.e., treewidth or clique-width. The homomorphism problems can be solved in polynomial time in classes of bounded treewidth/clique-width, and here we aim to understand the optimal dependence on the parameter. Second, we consider classes of graphs that can be obtained by excluding a fixed graph as an induced subgraph, and show for which such classes the algorithms solving $\textsc{Hom}(H)$ (and its list variant) significantly faster than the standard brute-force approach exists. We complement these results with lower bounds, based on standard complexity theory hypotheses.

We emphasize that while our primary interest comes from the perspective of computational complexity, the presented algorithmic results are usually a consequence of combinatorial properties of graphs and homomorphisms, that require a deep understanding of their structure.



**Keywords:** graph homomorphisms, list homomorphisms, hereditary graph classes, clique-width, treewidth, Exponential Time Hypothesis, Strong Exponential Time Hypothesis