Optimality program in graph homomorphism problems

Paweł Rzążewski

With the development of computer science, many fast algorithms solving fundamental problems (both theoretical and applied) have been developed. However, there is a large family of natural problems, that are very simple to define, but we still cannot solve them efficiently. What is even worse, we cannot also prove that fast algorithms for these problems do not exist. The general way to formulate hardness results is: if we could solve a certain problem efficiently, then we could also solve many other problems, for which efficient algorithms were unknown so far. This way any progress in solving our problem would lead to a big breakthrough in computer science.

Because of these reasons, the algorithmic community is putting much effort in investigating the borderline between "easy" and "hard" problems carefully, in hope to gain a better understanding what properties of particular problems can be used in design of efficient algorithms.

Many such problems, also coming from real-life applications, can be conveniently expressed as graphs problems. Suppose for example that we are given a set of objects, and some pairs of these object are in conflict. The aim is to partition the set into the smallest possible number of subsets, such that each of the subsets is conflict-free. In the language of graphs, we have a set of vertices, representing objects, and two vertices are joined by an edge if and only if they are in conflict. We want to partition the vertex set into the minimum number of subsets, so that no two vertices in one subset are adjacent. Such a "conflict-free" partition is called a *proper coloring* of a graph G, see Fig. 1.

Figure 1: Two graphs with their proper colorings with the minimum possible number of colors.

The notion of proper colorings can be generalized to the so-called H-colorings, where apart from the input graph G, we are additionally given a graph H, which describes possible relations between colors. We can think of vertices of H as colors, and we ask for a coloring of vertices of G, so that each pair of adjacent vertices in G must be mapped to a pair of adjacent vertices in H, see Fig. 2. Observe that if H is the graph with k pairwise adjacent vertices, then any H-coloring is exactly a proper coloring using at most k colors.

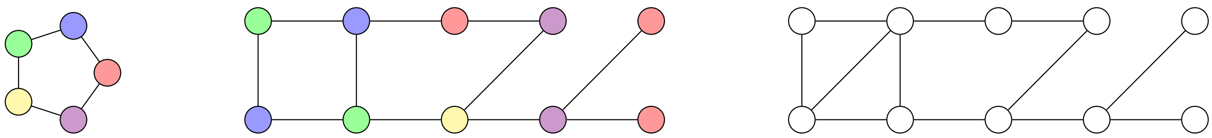

Figure 2: A graph H, an H-coloring of a graph, and a graph that has no H-coloring.

It turns out that many well-known and natural graph problems can be expressed as finding some special types of H-colorings for various graphs H. Due to this fact, investigating H-colorings might shed new light on some well-studied problems: a wider perspective could exhibit some similarities between the problems, that look very different at the first glance.

Even for small graphs H, e.g., a triangle, the problem of determining whether an H-coloring of a given graph G exists is computationally hard. However, the problem might get significantly easier, if we only consider graphs G of some special type, e.g. that do not contain some specific structure.

The aim of the project is to carefully investigate the complexity of finding some variants of H-colorings, for various graphs H. We want to understand the restrictions that must be imposed on the input graph, in order to obtain a problem that can be solved efficiently. We are especially interested in classifying all graphs H, for which H-colorings can be found efficiently in given graph classes.