1 Research project objectives

In this project we introduce the family of *(Approximate) Election Isomorphism* problems that aim at analysing the distance between elections. Our goal is to seek algorithms for these problems, including exact, approximate and heuristic approaches, analyse the properties of the problem, and test its effectiveness at solving certain practical problems.

By an election, we mean a set of candidates and a collection of voters where each voter has some preferences over the candidates. Different ballots can be used, depending on the type of elections. For example *ordinal ballots* allow voters to rank all the candidates according to their preferences, while *approval ballots* allow voters only to select a set of approved candidates (the rest of the candidates are regarded as disapproved). In our research, we will focus on elections with ordinal ballots only.

In the Election Isomorphism problem, we are given two elections with the same numbers of candidates and voters, and we ask if there are one-to-one correspondences between the candidates and between the voters from these elections so that they transform the first election into the second one. If it is possible to rename the candidates and the voters in this way, this means that these two elections are isomorphic. We will refer to the problem of deciding if two elections are isomorphic as the *Exact Election Isomorphism* problem. However, in most cases we expect this kind of renaming to be impossible to achieve and, in such situations, we would like to compute the distance between the elections. We refer to this more general problem as *Approximate Election Isomorphism* (in fact this is a whole family of problems, parametrized by the exact measure of distance between elections).

2 Work Plan

Our first task regards theoretical analysis of the Election Isomorphism problem and seek algorithms with provable guarantees (approximation and FPT algorithms). Within this task, our objective is to understand the complexity of the Election Isomorphism problem. In the second task, we will focus on designing practical algorithms for Election Isomorphism (including heuristics) and on implementing all our algorithms. After the implementation we will move to the third task which is fine-tuning and testing of the algorithms(in particular, we plan to evaluate the running times of our algorithm and the quality of solutions provided by approximation and heuristic algorithms). In the forth task will evaluate our algorithms within applications, we will try them on practical problems, where we expect the ability of computing distances between elections to be useful. Within the final task, we will test some alternative approaches.

3 Justifications of the project's value

We believe that the (Approximate) Election Isomorphism family of problems is interesting for the following reasons:

1. First of all, these are interesting problems in themselves. Indeed, they are very similar in spirit to the famous Graph Isomorphism problem, but the structure of ordinal elections is such that it is easy to provide a simple polynomial-time algorithm for the Exact Election Isomorphism problem (due to this fact, in this project we will focus mostly on the Approximate Election Isomorphism). On the other hand, for approval-based elections the Election Isomorphism problem is at least as hard as Graph Isomorphism since Graph Isomorphism problem can be reduced to it.
2. Second, we believe that the (Approximate) Election Isomorphism problem will be helpful in understanding elections generated according to various distributions of votes (for example, it is interesting to measure the difference between elections where each vote is drawn uniformly at random and elections where there is intricate structure of correlations, such as in Euclidean elections; our preliminary results show that we may expect some surprising similarities).
3. Third, studying (Approximate) Election Isomorphism problem will provide us with tools for analysing PrefLib—a popular library of preferences, which has not been analysed holistically yet. In particular, we aim to prepare a map of PrefLib, showing which data sets there are similar to each other and which are not.