Lower Bounds in the Preprocessing and Query Phases of Routing Algorithms

Colin White
Carnegie Mellon University

crwhite@cs.cmu.edu

Abstract. In the last decade, there has been a substantial amount of research in finding routing algorithms designed specifically to run on real-world graphs. In 2010, Abraham et al. showed upper bounds on the query time in terms of a graph’s highway dimension and diameter for the current fastest routing algorithms, including CONTRACTION HIERARCHIES, TRANSIT NODE ROUTING, and HUB LABELING. In this paper, we show corresponding lower bounds for the same three algorithms. We also show how to improve a result by Milosavljević which lower bounds the number of shortcuts added in the preprocessing stage for CONTRACTION HIERARCHIES. We relax the assumption of an optimal contraction order (which is NP-hard to compute), allowing the result to be applicable to real-world instances. Finally, we give a proof that optimal preprocessing for HUB LABELING is NP-hard. Hardness of optimal preprocessing is known for most routing algorithms, and was suspected to be true for HUB LABELING.

1 Introduction

The problem of finding shortest paths in road networks has been well-studied in the last decade, motivated by the application of computing driving directions. Although Dijkstra’s algorithm runs in small polynomial time, for applications involving continental-sized road networks, Dijkstra’s algorithm is simply not fast enough. There have been many different approaches to find algorithms that specifically run fast on real-world graphs.

Most recent innovations involve a two-stage algorithm: a preprocessing stage and a query stage. The preprocessing stage runs once and can spend hours calculating data. Then the query stage uses this data to find shortest paths very fast, often several orders of magnitude faster than Dijkstra’s algorithm for a continental query. Once the preprocessing stage is completed, the users can run as many queries as they want. For a query between two nodes $s$ and $t$ (an $s$–$t$ query), the algorithm returns $\text{dist}(s, t)$, the cost of the shortest path between $s$ and $t$. Most algorithms can also return the vertices on the shortest path using an extra data structure.

The current fastest routing algorithm on real-world graphs is HUB LABELING [2], which achieves a speedup of six orders of magnitude over Dijkstra’s algorithm. The TRANSIT NODE ROUTING algorithm is second-fastest, and requires
an order of magnitude less space than HUB LABELING. CONTRACTION HIERARCHIES is also a fast routing algorithm, which was state of the art in 2008. For a comprehensive overview of the best routing algorithms, see [6].

Until recently, it was known that these algorithms performed very well on real-world maps, but there were no theoretical guarantees. In fact, it is not hard to construct specific graphs for which these algorithms perform no faster than Dijkstra’s algorithm. So, an interesting theoretical question is to find properties present in all real-life graphs that explain why these algorithms work so well.

With this motivation in mind, Abraham et al. defined the notion of highway dimension [1], intuitively, the extent to which all shortest paths are hit by at least one of a small set of access nodes. Although it is too computationally intensive to calculate the exact highway dimension for a continental road map, there is evidence that the highway dimension $h$ is at most polylogarithmic in the number of vertices. It is conjectured that real-world routing networks always have low highway dimension, based on experimental evidence [3]. Abraham et al. were able to prove strong upper bounds on the query times in terms of highway dimension and diameter $d$ for four of the fastest routing algorithms: HUB LABELING, CONTRACTION HIERARCHIES, TRANSIT NODE ROUTING, and REACH.

1.1 Our Results

In this paper, we are interested in finding lower bounds for the current state-of-the-art routing algorithms. We show tight or near-tight bounds on the runtime for HUB LABELING, CONTRACTION HIERARCHIES, and TRANSIT NODE ROUTING.

Our lower bounds may facilitate proving better guarantees of these algorithms, or provide intuition for new routing algorithms, if one can find differences between the graphs we use and real world instances. For example, the graphs we use have low highway dimension, but they do not have small separators and are nonplanar, so perhaps there is a way to modify HUB LABELING to take this into account.

We show a tight lower bound for HUB LABELING, the fastest routing algorithm to date [6]. For CONTRACTION HIERARCHIES and TRANSIT NODE ROUTING, the definition of highway dimension in the lower bound versus upper bound is slightly different (because of a recent redefinition by Abraham et al.), so we cannot quite say the bounds are tight.

We can also use our analysis to generalize a known result by Milosavljević, which lower bounds the number of shortcut edges in the preprocessing stage of CONTRACTION HIERARCHIES [12]. This result assumes an optimal contraction order which is NP-hard to compute [7]. So for real-world instances, we rely on using contraction orders based on heuristics. We show how to relax the assumption about the contraction order, which means the result can be applied to real-world instances.

We also contribute a hardness result for optimal preprocessing of HUB LABELING. In 2010, Bauer et al. established hardness for optimal preprocessing for a variety of the best routing algorithms, including CONTRACTION HIERARCHIES and TRANSIT NODE ROUTING. In this paper, we show that in HUB LABELING