On Approximating a Geometric Prize-Collecting Traveling Salesman Problem with Time Windows
Extended Abstract

Reuven Bar-Yehuda\(^1\), Guy Even\(^2\), and Shimon (Moni) Shahar\(^3\)

\(^1\) Computer Science Dept., Technion, Haifa 32000, Israel. reuven@cs.technion.ac.il
\(^2\) Electrical-Engineering, Tel-Aviv Univ., Tel-Aviv 69978, Israel. guy@eng.tau.ac.il
\(^3\) Electrical-Engineering, Tel-Aviv Univ., Tel-Aviv 69978, Israel. moni@eng.tau.ac.il

Abstract. We study a scheduling problem in which jobs have locations. For example, consider a repairman that is supposed to visit customers at their homes. Each customer is given a time window during which the repairman is allowed to arrive. The goal is to find a schedule that visits as many homes as possible. We refer to this problem as the Prize-Collecting Traveling Salesman Problem with time windows (TW-TSP).

We consider two versions of TW-TSP. In the first version, jobs are located on a line, have release times and deadlines but no processing times. A geometric interpretation of the problem is used that generalizes the Erdős-Szekeres Theorem. We present an $O(\log n)$ approximation algorithm for this case, where $n$ denotes the number of jobs. This algorithm can be extended to deal with non-unit job profits.

The second version deals with a general case of asymmetric distances between locations. We define a density parameter that, loosely speaking, bounds the number of zig-zags between locations within a time window. We present a dynamic programming algorithm that finds a tour that visits at least $OPT/density$ locations during their time windows. This algorithm can be extended to deal with non-unit job profits and processing times.

1 Introduction

We study a scheduling problem in which jobs have locations. For example, consider a repairman that is supposed to visit customers at their homes. Each customer is given a time window during which the repairman is allowed to arrive. The goal is to find a schedule that visits as many homes as possible. We refer to this problem as the Prize-Collecting Traveling Salesman Problem with time windows (TW-TSP).

Previous Work. The goal in previous works on scheduling with locations differs from the goal we consider. The goal in previous works is to minimize the makespan (i.e. the completion time of the last job) or minimize the total waiting...
time (i.e. the sum of times that elapse from the release times till jobs are served). Tsitsiklis \[T92\] considered the special case in which the locations are on a line. Tsitsiklis proved that verifying the feasibility of instances in which both release times and deadlines are present is strongly NP-complete. Polynomial algorithms were presented for the cases of (i) either release times or deadlines, but not both, and (ii) no processing time. Karuno et. al. \[KN98\] considered a single vehicle scheduling problem which is identical to the problem studied by Tsitsiklis (i.e. locations on a line and minimum makespan). They presented a 1.5-approximation algorithm for the case without deadlines (processing and release times are allowed). Karuno and Nagamochi \[KN01\] considered multiple vehicles on a line. They presented a 2-approximation algorithm for the case without deadlines. Augustine and Seiden \[AS02\] presented a PTAS for single and multiple vehicles on trees with a constant number of leaves.

Our results. We consider two versions of TW-TSP. In the first version, TW-TSP on a line, jobs are located on a line, have release times, deadlines, but no processing times. We present an $O(\log n)$ approximation algorithm for this case, where $n$ denotes the number of jobs. Our algorithm also handles a weighted case, in which a profit $p(v)$ is gained if location $v$ is visited during its time window.

The second version deals with a general case of asymmetric distances between locations (asymmetric TW-TSP). We define a density parameter that, loosely speaking, bounds the number of zig-zags between locations within a time window. We present a dynamic programming algorithm that finds a tour that visits at least $OPT/density$ locations during their time windows. This algorithm can be extended to deal with non-unit profits and processing times.

Techniques. Our approach is motivated by a geometric interpretation. We reduce TW-TSP on a line to a problem called MAX-MONOTONE-TOUR. In MAX-MONOTONE-TOUR, the input consists of a collection of slanted segments in the plane, where the slope of each segment is 45 degrees. The goal is to find an $x$-monotone curve starting at the origin that intersects as many segments as possible. MAX-MONOTONE-TOUR generalizes the longest monotone subsequence problem \[ES35\]. A basic procedure in our algorithms involves the construction of an arc weighted directed acyclic graph and the computations of a max-weight path in it \[F75\]. Other techniques include interval trees and dynamic programming algorithms.

Organization. In Section 2 we formally define TW-TSP. In Section 3 we present approximation algorithms for TW-TSP on a line. We start with an $O(1)$-approximation algorithm for the case of unit time-windows and end with an $O(\log n)$-approximation algorithm. In Section 4 we present algorithms for the non-metric version of TW-TSP.