Combining Approximation Algorithms for the Prize-Collecting TSP

Michel X. Goemans

Abstract

We present a 1.91457-approximation algorithm for the prize-collecting travelling salesman problem. This is obtained by combining a randomized variant of a rounding algorithm of Bienstock et al. [2] and a primal-dual algorithm of Goemans and Williamson [5].

1 Introduction

In the prize-collecting travelling salesman problem (PC-TSP), we are given a vertex set $V$ (with $|V| = n$), a metric $c$ on $V \times V$ (i.e. $c$ satisfies (i) $c_{ij} = c_{ji} \geq 0$ for all $i, j \in V$ and (ii) triangle inequality: $c_{ij} + c_{jk} \geq c_{ik}$ for all $i, j, k \in V$), a special vertex $r \in V$ (the depot), penalties $\pi : V \to \mathbb{R}_+$, and the goal is to find a cycle $T$ with $r \in V(T)$ such that

$$c(T) + \pi(V \setminus V(T))$$

is minimized, where $c(T) = \sum_{(i,j) \in T} c_{ij}$, $\pi(S) = \sum_{i \in S} \pi_i$, and $V(T)$ denotes the vertices spanned by $T$.

The first constant approximation algorithm for PC-TSP was given by Bienstock et al. [2]. It is based on rounding the optimum solution to a natural LP relaxation for the problem, and provides a performance guarantee of 2.5. Goemans and Williamson [5] have designed a primal-dual algorithm based on the same LP relaxation, and this gives a 2-approximation algorithm for the problem. In 1998, Goemans [4] has shown that a simple improvement of the algorithm of Bienstock et al. gives a guarantee of $2.055 \cdots = \frac{1}{1-e^{-2/3}}$. Recently, Archer et al. [1] are the first to break the barrier of 2 and provide an improvement of the primal-dual algorithm of Goemans and Williamson; their performance guarantee is 1.990283. In this note, we show that by combining the rounding algorithm of Bienstock et al. and the primal-dual algorithm of Goemans and Williamson, we can obtain a guarantee of $1.91456 \cdots = \frac{1}{1-\frac{2}{3}e^{-2/3}}$. The analysis uses the technique in [4] together with an improved analysis of the primal-dual algorithm as observed in [3] and used in Archer et al. [1].

*MIT Department of Mathematics. goemans@math.mit.edu. Supported by NSF contract CCF-0829878 and by ONR grant N00014-05-1-0148.
2 Combining Approximation Algorithms

We start by briefly reviewing the rounding result of Bienstock et al. [2]. Consider a classical LP relaxation of PC-TSP:

\[
\begin{align*}
\text{Min} & \quad \sum_{e \in E} c_e x_e + \sum_v \pi(v)(1 - y_v) \\
\text{subject to:} & \quad x(\delta(v)) = 2y_v \quad v \in V \setminus \{r\} \\
& \quad x(\delta(S)) \geq 2y_v \quad S \subset V, r \notin S, v \in S \\
& \quad 0 \leq x_e \leq 1 \quad e \in E \\
& \quad 0 \leq y_v \leq 1 \quad v \in V \\
& \quad y_r = 1,
\end{align*}
\]

where \( E \) denotes the edge set of the complete graph on \( E \). For conciseness, we use \( c(x) + \pi(1 - y) \) to denote the objective function of this LP. Let \( x^*, y^* \) be an optimum solution of this LP relaxation, and let \( LP = c(x^*) + \pi(1 - y^*) \) denote its value. Bienstock et al. [2] show the following (based on the analysis of Christofides’ algorithm due to Wolsey [8] and Shmoys and Williamson [7]).

**Proposition 1** (Bienstock et al.). Let \( 0 < \gamma \leq 1 \) and let \( S(\gamma) = \{v : y^*_v \geq \gamma\} \). Let \( T_\gamma \) denote the cycle on \( S(\gamma) \) output by Christofides’ algorithm when given \( S(\gamma) \) as vertex set. Then:

\[
c(T_\gamma) \leq \frac{3}{2\gamma} c(x^*).
\]

The 2.5-approximation algorithm can then be derived by setting \( \gamma = \frac{3}{5} \) since we get \( c(T_{3/5}) \leq \frac{5}{2} c(x^*) \) and \( \pi(V \setminus S(3/5)) \leq \frac{5}{2} \pi(1 - y^*) \). In [4], we have shown that one can get a better performance guarantee by taking the best cycle output over all possible values of \( \gamma \); notice that this leads to at most \( n - 1 \) different cycles.

The primal-dual algorithm in [5] constructs a cycle \( T \) and a dual solution to the linear programming relaxation above such that their values are within a factor 2 of each other, showing a performance guarantee of 2 since the value of any dual solution is a lower bound on \( LP \). Chudak, Roughgarden and Williamson [3] (see their Theorem 2.1) observe that the analysis of [5] actually shows a stronger guarantee on the penalty side of the objective function, namely that the cycle \( T \) returned satisfies:

\[
c(T) + \left(2 - \frac{1}{n - 1}\right) \pi(V \setminus V(T)) \leq \left(2 - \frac{1}{n - 1}\right) \text{LP}.
\]

This increased factor on the penalty side is exploited in Archer et al. [1], and this motivated the result in this note. Suppose now that we apply the primal-dual algorithm to an instance in which we replace the penalties \( \pi(\cdot) \) by \( \pi'(\cdot) \) given by

\[
\pi'(v) = \frac{1}{2 - 1/(n - 1)} \pi(v).
\]

Thus, [1] implies that the cycle \( T \) returned satisfies:

\[
c(T) + \pi(V \setminus V(T)) \leq \left(2 - \frac{1}{n - 1}\right) \text{LP'},
\]

where

\[
\text{LP'} = c(T) + \pi'(V \setminus V(T)) \leq \left(2 - \frac{1}{n - 1}\right) \text{LP}.
\]

This result is the one we use in this note.
where \( LP' \) denotes the LP value for the penalties \( \pi'(\cdot) \). As the optimum solution \( x^*, y^* \) of LP (with penalties \( \pi(\cdot) \)) is feasible for the linear programming relaxation with penalties \( \pi'(\cdot) \), we derive that the cycle \( T_{pd} \) output satisfies:

\[
c(T_{pd}) + \pi(V \setminus V(T_{pd})) = c(T_{pd}) + \left(2 - \frac{1}{n-1}\right)\pi'(V \setminus V(T_{pd}))
\leq \left(2 - \frac{1}{n-1}\right)LP'
\leq \left(2 - \frac{1}{n-1}\right)(c(x^*) + \pi'(1 - y^*))
= \left(2 - \frac{1}{n-1}\right)c(x^*) + \pi(1 - y^*).
\]

Summarizing:

**Proposition 2.** The primal-dual algorithm applied to an instance with penalties \( \pi'(\cdot) \) given by (2) outputs a cycle \( T_{pd} \) such that

\[
c(T_{pd}) + \pi(V \setminus V(T_{pd})) \leq 2c(x^*) + \pi(1 - y^*).
\]

We claim that the best of the algorithms given in Propositions 1 and 2 gives a better than 2 approximation guarantee for PC-TSP.

**Theorem 3.** Let

\[
H = \min(\min_{\gamma}(c(T_\gamma) + \pi(V \setminus V(\gamma))), c(T_{pd}) + \pi(V \setminus V(T_{pd}))).
\]

Then

\[
H \leq \alpha(c(x^*) + \pi(1 - y^*)) = \alpha LP,
\]

where \( \alpha = \frac{1}{1 - \frac{3}{4}e^{-1/3}} < 1.91457 \).

As mentioned earlier, the minimum in the theorem involves only \( n \) different algorithms as we need only to consider values \( \gamma \) equal to some \( y^*_v \).

**Proof.** We construct an appropriate probability distribution over all the algorithms involved such that the expected cost of the solution produced is at most \( \alpha(c(x^*) + \pi(1 - y^*)) \).

First, assume that we select \( \gamma \) randomly (according to a certain distribution to be specified). Then, by Proposition 1 we have that

\[
E[c(T_\gamma)] \leq \frac{3}{2} E \left[\frac{1}{\gamma}\right]c(x^*),
\]

while the expected penalty we have to pay is

\[
E[\pi(V \setminus V(\gamma))] = \sum_{v \in V} Pr[\gamma > y^* (v)]\pi(v).
\]


Thus, the overall expected cost is:

\[ E[c(T_\gamma) + \pi(V \setminus V(\gamma))] \leq \frac{3}{2} E \left[ \frac{1}{\gamma} \right] c(x^*) + \sum_{v \in V} Pr[\gamma > y^*(v)]\pi(v). \] (4)

Assume now that \( \gamma \) is chosen uniformly between \( a = e^{-1/3} = 0.71653 \cdots \) and 1. Then,

\[ E \left[ \frac{1}{\gamma} \right] = \int_a^1 \frac{1}{1-a} \frac{1}{x} dx = -\ln(a) + \frac{1}{1-a} = \frac{1}{3(1-a)} = \frac{1}{3(1-e^{-1/3})}, \]

and

\[ Pr[\gamma > y] = \begin{cases} \frac{1-y}{1-a} & a \leq y \leq 1 \\ \frac{1}{1-a} & 0 \leq y \leq a. \end{cases} \]

Therefore, (4) becomes:

\[ E[c(T_\gamma) + \pi(V \setminus V(\gamma))] \leq \frac{1}{2(1-e^{-1/3})} c(x^*) + \frac{1}{1-e^{-1/3}} \pi(1-y^*). \] (5)

Suppose we now select, with probability \( p \), the primal-dual algorithm as given in Proposition 2 or, with probability \( 1-p \), the rounding algorithm with \( \gamma \) chosen randomly according to \( \gamma \sim U[e^{-1/3}, 1] \). From (5) and Proposition 2 we get that the expected cost \( E^* \) of the resulting algorithm satisfies:

\[ E^* \leq \left(2p + (1-p)\frac{1}{2(1-e^{-1/3})}\right) c(x^*) + \left(p + (1-p)\frac{1}{1-e^{-1/3}}\right) \pi(1-y^*). \]

Choosing \( p = (1-p)\frac{1}{2(1-e^{-1/3})} \), i.e. \( p = \frac{1}{3-2e^{-1/3}} \), we get

\[ E^* \leq 3p(c(x^*) + \pi(1-y^*)) = 3pLP. \]

Therefore, the best of the algorithms involved outputs a solution of cost at most \( 3pLP = \alpha LP \) where

\[ \alpha = \frac{1}{1-3/2e^{-1/3}} < 1.91457. \]

One can show that the probability distribution given in the proof is optimal for the purpose of this proof; this is left as an exercise for the reader.

Theorem 3 shows that the linear programming relaxation of PC-TSP has an integrality gap bounded by 1.91457; in contrast, the result of Archer et al. [1] does not imply a better than 2 bound on the integrality gap.

As a final remark, if we replace Christofides’ algorithm with an algorithm for the symmetric TSP that outputs a solution within a factor \( \beta \) of the standard LP relaxation for the TSP then the approach described in this note gives a guarantee of

\[ \frac{1}{1-\frac{1}{\beta}e^{1-2/\beta}} \]

for PC-TSP.
References

[1] A. Archer, M. Bateni, M. Hajiaghayi and H. Karloff, “Improved Approximation Algorithms for Prize-Collecting Steiner Tree and TSP”, Proceedings of the 50th Annual Symposium on Foundations of Computer Science, 2009.

[2] D. Bienstock, M.X. Goemans, D. Simchi-Levi and D. Williamson, “A Note on the Prize Collecting Traveling Salesman Problem”, Mathematical Programming, 59, 413–420, 1993.

[3] F.A. Chudak, T. Roughgarden, and D.P. Williamson, “Approximate k-MSTs and k-Steiner trees via the primal-dual method and Lagrangean relaxation”, Mathematical Programming, 100, 411–421, 2004.

[4] M.X. Goemans, “The Prize-Collecting TSP Revisited”, talk at the SIAM Discrete Mathematics conference, Toronto, Canada, July 1998.

[5] M.X. Goemans and D.P. Williamson, “A General Approximation Technique for Constrained Forest Problems”, SIAM Journal on Computing, 24, 296–317, 1995.

[6] D.B. Shmoys. ”Using linear programming in the design and analysis of approximation algorithms: two illustrative problems”, in: Approximation Algorithms for Combinatorial Optimization, Lecture Notes in Computer Science 1444 (K. Jansen and J. Rolim, eds.), Springer, Berlin, 15–32, 1998.

[7] D.B. Shmoys and D.P. Williamson, “Analyzing the Held-Karp TSP lower bound: A monotonicity property with applications”, Operations Research Letters, 35, 281–285, 1990.

[8] L.A. Wolsey, “Heuristic analysis, linear programming and branch-and-bound”, Mathematical Programming Study, 13, 121–134, 1980.