Upper Bounds on ATSP Neighborhood Size

Gregory Gutin* Anders Yeo†
Department of Computer Science
Royal Holloway, University of London
Egham, Surrey, TW20 0EX, UK

Abstract

We consider the Asymmetric Traveling Salesman Problem (ATSP) and use the definition of neighborhood by Deineko and Woeginger (see Math. Program. 87 (2000) 519-542). Let $\mu(n)$ be the maximum cardinality of polynomial time searchable neighborhood for the ATSP on n vertices. Deineko and Woeginger conjectured that $\mu(n) < \beta(n-1)!$ for any constant $\beta > 0$ provided P$\neq$NP. We prove that $\mu(n) < \beta(n-k)!$ for any fixed integer $k \geq 1$ and constant $\beta > 0$ provided NP$\not\subseteq$P/poly, which (like P$\neq$NP) is believed to be true. We also give upper bounds for the size of an ATSP neighborhood depending on its search time.

Keywords: ATSP, TSP, exponential neighborhoods, upper bounds.

1 Introduction, Terminology and Notation

We consider the *Asymmetric Traveling Salesman Problem* (ATSP): given a weighted complete directed graph, $(\overleftrightarrow{K}_n, c)$, where n is the number of vertices and c is the weight function from the arc set of $\overleftrightarrow{K}_n$ to the set of reals, find a hamiltonian cycle of minimum total weight. Below we call a hamiltonian cycle a *tour* and $c(a)$ the *cost* of a for an arc a of $\overleftrightarrow{K}_n$. For a tour T, its cost $c(T)$ is the sum of the costs of its arcs. Observe that $\overleftrightarrow{K}_n$ contains $(n-1)!$ hamiltonian cycles, i.e., the ATSP on n vertices has $(n-1)!$ tours.

Local search heuristics are among the main tools to compute near optimal tours in large instances of the ATSP in relatively short time, see e.g.

*Corresponding author. E-mail: gutin@cs.rhul.ac.uk

†This paper was written when AY worked in BRICS, Department of Computer Science, University of Aarhus, Denmark. E-mail: anders@cs.rhul.ac.uk

Cirasella, Johnson, McGeoch and Zhang [7]. In many cases the neighborhoods used in the local search algorithms are of polynomial cardinality. One may ask whether it is possible to have larger, exponential size, neighborhoods for the ATSP such that the best tour in such a neighborhood can be computed in polynomial time. Fortunately, the answer to this question is positive. (This question is far from being trivial for some generalizations of the TSP, e.g. Deineko and Woeginger [8] conjecture that for the quadratic assignment problem there is no exponential neighborhood "searchable" in polynomial time.)

Sarvanov and Doroshko [21, 22] and Gutin [10] were the first to introduce exponential neighborhoods for the ATSP. In particular, they independently showed the existence of $(n/2)!$-size neighborhood for the ATSP with n vertices. In this neighborhood, the best tour can be computed in $O(n^3)$ time, i.e., asymptotically in at most the same time as a complete iteration of 3-OPT, which finds the best tour among only $\Theta(n^3)$ tours. For more recent work on exponential neighborhoods for Symmetric and Asymmetric TSP, see e.g. [2, 5, 6, 9, 11, 17, 18] an informative survey paper [8], and a chapter [14]. Local search algorithms based on exponential neighborhoods were implemented in some of those papers with encouraging results, see especially Balas and Simonetti [2].

We adapt the definition of a neighborhood for the ATSP due to Deineko and Woeginger [8]. Let P be a set of permutations on $\{1, 2, \ldots, n\}$. Then the *neighborhood* (with respect to P) of a tour $T = x_1 x_2 \ldots x_n x_1$, $x_1 = 1$, is defined as follows:

$$N_P(T) = \{x_{\pi(1)} x_{\pi(2)} \ldots x_{\pi(n)} x_{\pi(1)} : \ \pi \in P\}.$$

The above definition of a neighborhood is somewhat restrictive (in particular, this definition implies that the neighborhood of every tour is of the same cardinality, $|P|$), but reflects the very important "shifting" property of neighborhoods which distinguishes them from arbitrary sets of tours. Another important property usually imposed on a neighborhood $N(T)$ of a tour T is that the best among tours of $N(T)$ can be computed in time $p(n)$ polynomial in n. This is necessary to guarantee an efficient local search. Neighborhoods satisfying this property are called *polynomially searchable* or, more precisely, *$p(n)$-searchable.*

Not much is known so far on the maximum cardinality $\mu(n)$ of polynomial time searchable neighborhood for the ATSP on n vertices. The above mentioned result implies that $\mu(n) \geq (n/2)!$. This was slightly improved in

[11] to $\mu(n) = \Omega(\frac{e^{\sqrt{n/2}}\lfloor n/2\rfloor!}{n^{1/4}})$. Deineko and Woeginger [8] conjectured that there exists a constant $\alpha > \frac{1}{2}$ such that $\mu(n) \geq (\alpha n)!$. They also conjectured that $\mu(n) < \beta(n-1)!$ for any positive constant β provided P$\neq$NP. In Section 2 we prove that $\mu(n) < \beta(n-k)!$ for any constant $\beta > 0$ and fixed integer k provided NP$\not\subseteq$P/poly.

P/poly is a well-known complexity class in structural complexity theory, see e.g. [3], and it is widely believed that NP$\not\subseteq$P/poly for otherwise, as proved in the well-known paper by Karp and Lipton [15], it would imply that the so-called polynomial hierarchy collapses on the second level, which is thought to be very unlikely. The idea that defines P/poly is that, for each input size n, one is able to compute a polynomial-sized "key for size n inputs". This is called the "advice for size n inputs". It is allowed that the computation of this "key" may take time exponential in n (or worse). P/poly means solvable in polynomial time (in input size n) / given the poly-sized general advice for inputs of size n. For formal definitions of P/poly and related nonuniform complexity classes, consult [3].

Notice that the above mentioned result from Section 2 reflects the fact that neighborhoods are quite special sets of tours. Indeed, it was shown in [12, 19, 20] that there are sets of tours of cardinality at least $(n-2)!$, for which the best tour can be found in time $O(n^3)$. This result was further improved in [13].

A very useful upper bound is given in [8] of the size of ATSP neighborhood depending on the time $t(n)$ required for its search (in other words, $t(n)$ is the minimum time required to find the best tour in the neighborhood). However, that bound is not valid for $t(n) \leq n/2$ (see a remark after Corollary 3.3). We correct and improve the bound of [8] in Section 3. The upper bounds imply that, if we are ready to invest only linear time, $O(n)$, in the search of the neighborhood, then the neighborhood size is bounded from above by $2^{O(n)}$. (Notice that $(n/2)! = 2^{\Theta(n\log n)}$ and $(n-1)! = 2^{\Theta(n\log n)}$.)

2 Upper Bounds for Polynomial Time Searchable Neighborhoods

Let S be a finite set and let $\mathcal{F}$ be a family of subsets of S ($\mathcal{F}$ may have several copies of the same subset of S). Suppose that $\mathcal{F}$ is a *cover* of S, i.e., $\cup\{F: F \in \mathcal{F}\} = S$. The well-known *covering problem* is to find a cover of S containing the minimum number of sets in $\mathcal{F}$. While the following *greedy covering algorithm* (GCA) does not always produce a cover with minimum number of sets, GCA finds asymptotically optimal results for some wide classes of families, see e.g. [16]. GCA starts by choosing a set F in $\mathcal{F}$ of maximum cardinality, deleting F from $\mathcal{F}$ and initiating a "cover" $\mathcal{C} = \{F\}$. Then GCA deletes the elements of F from every remaining set in $\mathcal{F}$ and chooses a set H of maximum cardinality in $\mathcal{F}$, appends it to $\mathcal{C}$ and updates $\mathcal{F}$ as above. The algorithm stops when $\mathcal{C}$ becomes a cover of S. The following lemma have been obtained independently by several authors, see Proposition 10.1.1 in [1].

Lemma 2.1 *Let $|S| = s$, let $\mathcal{F}$ contain f sets, and let every element of S be in at least δ sets of $\mathcal{F}$. Then the cover found by GCA is of cardinality at most $1 + f(1 + \ln(\delta s/f))/\delta$.*

Using this lemma we can prove the following:

Theorem 2.2 *Let $\mathcal{T}$ be the set of all tours of the ATSP on n vertices. For every fixed integer $k \geq 1$ and constant $\beta > 0$, unless NP$\subseteq$ P/poly, there is no set Π of permutations on $\{1, 2, \ldots, n\}$ of cardinality at least $\beta(n-k)!$ such that every neighborhood $N_\Pi(T)$, $T \in \mathcal{T}$, is polynomial time searchable.*

Proof: Assume that, for some $k \geq 1$ and $\beta > 0$, there exists a set Π of permutations on $\{1, 2, \ldots, n\}$ of cardinality at least $\beta(n-k)!$ such that every neighborhood $N_\Pi(T)$, $T \in \mathcal{T}$, is polynomial time searchable. Let $\mathcal{N} = \{N_\Pi(T) : T \in \mathcal{T}\}$. Consider the covering problem with $S = \mathcal{T}$ and $\mathcal{F} = \mathcal{N}$. Observe that $|S| = (n-1)!$ and family $\mathcal{F}$ contains $(n-1)!$ neighborhoods. To see that every tour is in at least $\delta = \beta(n-k)!$ neighborhoods of $\mathcal{N}$, consider a tour $Y = y_1 y_2 \ldots y_n y_1$ and observe that for every $\pi \in \Pi$,

$$Y \in N_\Pi(y_{\pi^{-1}(1)} y_{\pi^{-1}(2)} \cdots y_{\pi^{-1}(n)} y_{\pi^{-1}(1)}).$$

By Lemma 2.1 there is a cover $\mathcal{C}$ of S with at most $O(n^k \ln n)$ neighborhoods from $\mathcal{N}$. Since every neighborhood in $\mathcal{C}$ is polynomial time searchable and $\mathcal{C}$ contains only polynomial number of neighborhoods, we can construct the best tour in polynomial time provided $\mathcal{C}$ is found. Note that $\mathcal{C}$ depends only on n, and not on the instance of the ATSP, so the ATSP must be in P/poly. Since the ATSP is NP-hard, we conclude that NP$\subseteq$ P/poly. $\square$

3 General Upper Bounds

It is realistic to assume that the search algorithm spends at least one unit of time on every arc of $\overleftrightarrow{K}_n$ that it considers. We use this assumption in the rest of this paper.

It is worth noting that the results of this section are valid for a much more general definition of neighbourhood.

For a digraph or tour H, $V(H)$ $(A(H))$ denotes the vertex (arc) set of H. In the proof of the following theorem we use the operation of arc contraction. For an arc $a = (x, y)$ in $(\overleftrightarrow{K}_n, c)$, the *contraction* of a results in a complete digraph with vertex set $V' = V(\overleftrightarrow{K}_n) \cup \{v_a\} - \{x, y\}$ and cost function c', where $v_a \notin V(\overleftrightarrow{K}_n)$, such that the cost $c'(u, w)$, for $u, w \in V'$, is defined by $c(u, x)$ if $w = v_a$, $c(y, w)$ if $u = v_a$, and $c(u, w)$, otherwise. The above definition has an obvious extension to a set of arcs; for more details, see [4]. For a digraph or tour H, $A(H)$ denotes the arc set of H.

Theorem 3.1 *Let N_n be an ATSP neighborhood that can be searched in time $t(n)$. Then $|N_n| \le \max_{1 \le n' \le n}(t(n)/n')^{n'}$.*

Proof: Let $D = (\overleftrightarrow{K}_n, c)$ be an instance of the ATSP and let H be the tour that our search algorithm returns, when run on D. Let E denote the set of arcs in D, which the search algorithm actually examine; observe that $|E| \le t(n)$ by the assumption above. Let F be the set of arcs in H that are not examined in the search, and let G denote the set of arcs in $D - A(H)$ that are not examined in the search.

We first prove that every arc in F must belong to each tour of N_n. Assume that there is a tour $H' \in N_n$ that avoids an arc $a \in F$. If we assign to a a very large cost, H' becomes cheaper than H, a contradiction.

Similarly, we prove that no arc in G can belong to a tour in N_n. Assume that an $a \in G$ and a is in a tour $H' \in N_n$. By making a very cheap, we can ensure that $c(H') < c(H)$, a contradiction.

Now let D' be the digraph obtained by contracting the arcs in F and deleting the arcs in G, and let n' be the number of vertices in D'. Note that every tour in N_n corresponds to a tour in D' and, thus, the number of tours in D' is an upper bound on $|N_n|$. In a tour of D', there are at most $d^+(i)$ possibilities for the successor of a vertex i, where $d^+(i)$ is the out-degree of i in D'. Hence we obtain that

$$|N_n| \leq \prod_{i=1}^{n'} d^+(i) \leq \left(\frac{1}{n'}\sum_{i=1}^{n'} d^+(i)\right)^{n'} \leq \left(\frac{t(n)}{n'}\right)^{n'},$$

where we applied the arithmetic-geometric mean inequality. □

Corollary 3.2 *Let N_n be an ATSP neighborhood that can be searched in time $t(n)$. Then $|N_n| \leq \max\{e^{t(n)/e}, (t(n)/n)^n\}$, where e is the basis of natural logarithms.*

Proof: Let $U(n) = \max_{1 \leq n' \leq n}(t(n)/n')^{n'}$. By differentiating $f(n') = (t(n)/n')^{n'}$ with respect to n' we can readily obtain that $f(n')$ increases for $1 \leq n' \leq t(n)/e$, and decreases for $t(n)/e \leq n' \leq n$. Thus, if $n \leq t(n)/e$, then $f(n')$ increases for every value of $n' < n$ and $U(n) = f(n) = (t(n)/n)^n$. On the other hand, if $n \geq t(n)/e$ then the maximum of $f(n')$ is for $n' = t(n)/e$ and, hence, $U(n) = e^{t(n)/e}$. □

It follows from the proof of Corollary 3.2 that

Corollary 3.3 *For $t(n) \geq en$, we have $|N_n| \leq (t(n)/n)^n$.*

Note that the restriction $t(n) \geq en$ is important since otherwise the bound of Corollary 3.3 can be invalid. Indeed, if $t(n)$ is a constant, then for n large enough the upper bound implies that $|N_n| = 0$, which is not correct since there are neighborhoods of constant size that can be searched in constant time: consider a tour T, delete three arcs in T and add three other arcs to form a new tour T'. Clearly, the best of the two tours can be found in constant time by considering only the six arcs mentioned above. Notice that this observation was not taken into account in [8], where the bound $(2t(n)/n)^n$ was claimed. That bound is therefore invalid for $t(n) \leq n/2$.

Corollary 3.2 immediately implies that linear-time algorithms can be used only for neighborhoods of size at most $2^{O(n)}$. This answers a question from [11]. Using Corollary 3.2, it is also easy to show the next corollary, which is of interest due to a "matching" result in [11]: For every $\beta > 1$ there is an $O(n^\beta)$-searchable neighborhood of size $2^{\Theta(n \log n)}$.

Corollary 3.4 *The time required to search an ATSP neighborhood of size $2^{\Theta(n \log n)}$ is $\Omega(n^\alpha)$ for some constant $\alpha > 1$.*

Acknowledgments We would like to thank Mike Fellows for his "polynomial" advice on P/poly, Santosh Kabadi for useful discussions, and referees for some useful recommendations. The research of GG was partially supported by a grant from EPSRC.

References

[1] A.S. Asratian, T.M.J. Denley and R. Häggkvist, Bipartite Graphs and Their Applications, Univ. Press, Cambridge, 1998.

[2] E. Balas and N. Simonetti, Linear time dynamic programming algorithms for some new classes of restricted TSP's: a computational study, INFORMS J. Comput. 13 (2001) 56–75.

[3] J.L. Balcazar, J. Diaz and J. Gabarro, Structural Complexity I, EATCS Monographs on Theoretical Computer Science, 2nd Ed., Springer-Verlag, Berlin, 1995.

[4] J. Bang-Jensen and G. Gutin, Digraphs: Theory, Algorithms and Apllications, Springer-Verlag, London, 2000.

[5] R.E. Burkard, V.G. Deineko and G.J. Woeginger, The travelling salesman problem and the PQ-tree, Math. Oper. Res. 23 (1998) 613–623.

[6] J. Carlier and P. Villon, A new heuristic for the traveling salesman problem, RAIRO Oper. Res. 24 (1990) 245–253.

[7] J. Cirasella, D.S. Johnson, L.A. McGeoch and W. Zhang, The asymmetric traveling salesman problem: algorithms, instance generators and tests, Lect. Notes Comput. Sci. 2153 (2001) 32–59.

[8] V.G. Deineko and G.J. Woeginger, A study of exponential neighbourhoods for the traveling salesman problem and the quadratic assignment problem, Math. Prog. Ser. A 87 (2000) 519-542.

[9] F. Glover and A.P. Punnen, The traveling salesman problem: New solvable cases and linkages with the development of approximation algorithms, J. Oper. Res. Soc. 48 (1997) 502–510.

[10] G. Gutin, On an approach to solving the traveling salesman problem, In Proc. The USSR Conference on System Res., Moscow, 184–185, 1984 (in Russian).

[11] G. Gutin, Exponential neighbourhood local search for the traveling salesman problem, Comput. OR 26 (1999) 313–320.

[12] G. Gutin and A. Yeo, Polynomial approximation algorithms for the TSP and the QAP with factorial domination number, Discrete Appl. Math., in print.

[13] G. Gutin and A. Yeo, TSP tour domination and Hamilton cycle decomposition of regular digraphs, Oper. Res. Lett. 28 (2001) 107–111.

[14] G. Gutin, A. Yeo and A. Zverovitch, Exponential Neighborhoods and Domination Analysis for the TSP. In *The Traveling Salesman Problem and its Variations* (G. Gutin and A. Punnen, eds.), to be published by Kluwer.

[15] R.M. Karp and R.J. Lipton, Some connections between nonuniform and uniform complexity classes, In Proc. 12th ACM Annual Symp. on Theory of Computing (STOC'80), 302–309, L.A., California, 1980.

[16] N.N. Kuzjurin, Asymptotic investigation on the problem of covering, Problemy Kybernetiky 37 (1980) 19–56 (in Russian).

[17] C.N. Potts and S.L. van de Velde, Dynaserach - Iterative local improvement by dynamic programming: Part I, The traveling salesman problem, Tech. Report, University of Twente, The Netherlands, 1995.

[18] A.P. Punnen, The traveling salesman problem: new polynomial approximation algorithms and domination analysis, J. Inform. Optim. Sci. 22 (2001) 191–206.

[19] A.P. Punnen and S. Kabadi, Domination analysis of some heuristics for the asymmetric traveling salesman problem, to appear in Discrete Appl. Math.

[20] V.I. Sarvanov, The mean value of the functional of the assignment problem, Vestsi Akad. Navuk BSSR Ser. Fiz. -Mat. Navuk no. 2 (1976) 111–114 (in Russian).

[21] V.I. Sarvanov and N.N. Doroshko, The approximate solution of the traveling salesman problem by a local algorithm that searches neighborhoods of exponential cardinality in quadratic time, In Software: Algorithms and Programs, Math. Institute of Belorussian Acad. of Sci., Minsk, no. 31, 8–11 (1981) (in Russian).

[22] V.I. Sarvanov and N.N. Doroshko, Approximate solution of the traveling salesman problem by a local algorithm with scanning neighbourhoods of factorial cardinality in cubic time. In Software: Algorithms and Programs, Math. Institute of Belorussian Acad. of Sci., Minsk, no. 31, 11–13 (1981) (in Russian).