# From Symmetry to Asymmetry: Generalizing TSP Approximations by Parametrization

Lukas Behrendt[1], Katrin Casel[1(✉)], Tobias Friedrich[1], J. A. Gregor Lagodzinski[1], Alexander Löser[1], and Marcus Wilhelm[2]

[1] Hasso Plattner Institute, University of Potsdam, Potsdam, Germany
{lukas.behrendt,alexander.loeser}@student.hpi.de,
{katrin.casel,tobias.friedrich,gregor.lagodzinski}@hpi.de
[2] Karlsruhe Institute of Technology, Karlsruhe, Germany
marcus.wilhelm@kit.edu

**Abstract.** We generalize the tree doubling and Christofides algorithm to parameterized approximations for ATSP. The parameters we consider for the respective generalizations are upper bounded by the number of *asymmetric distances*, which yields algorithms to efficiently compute good approximations also for moderately asymmetric TSP instances. As generalization of the Christofides algorithm, we derive a parameterized 2.5-approximation, where the parameter is the size of a vertex cover for the subgraph induced by the asymmetric distances. Our generalization of the tree doubling algorithm gives a parameterized 3-approximation, where the parameter is the minimum number of asymmetric distances in a minimum spanning arborescence. Further, we combine these with a notion of symmetry relaxation which allows to trade approximation guarantee for runtime. Since the two parameters we consider are theoretically incomparable, we present experimental results which show that generalized tree doubling frequently outperforms generalized Christofides with respect to parameter size.

**Keywords:** Parameterized approximation · Stability of approximation · TSP vs. ATSP

## 1 Introduction

The ubiquitous traveling salesman problem asks for a shortest round trip through a given set of cities. Its relation to the Hamiltonian cycle problem does not only imply NP-hardness, but also implies that efficient approximation is impossible for unrestricted instances, which is why distances are usually assumed to satisfy the triangle inequality. This restriction to *metric* instances is one of the most extensively studied problems in combinatorial optimization, yet its approximability prevails as an active research area. Despite the breakthrough by Svensson


E. Bampis and A. Pagourtzis (Eds.): FCT 2021, LNCS 12867, pp. 53–66, 2021.
https://doi.org/10.1007/978-3-030-86593-1_4

et al. [32], particularly the difference between symmetric and asymmetric distances remains rather poorly understood. In this paper we employ the tools of parameterized complexity as a new approach to explicitly study the effects of asymmetry on the approximability of the metric traveling salesman problem.

### 1.1 Motivation

Symmetric distance, meaning that traveling from A to B has the same cost as traveling from B to A, is certainly the most common assumption to the metric traveling salesman problem. In fact, it is so common that the name (metric) traveling salesman problem (TSP) is usually associated with this symmetric version, while the more general case is explicitly referred to as *asymmetric* (ATSP).

It appears as if symmetry plays a vital role in view of approximations. For TSP it was known for over 40 years that a $\frac{3}{2}$-approximation is possible with the famous algorithm of Christofides [10] (or Christofides-Serdyukov, see [4]). Recently, Karlin et al. [22] showed a randomized approximation with an expected ratio of $\frac{3}{2} - \varepsilon$ for a small constant $\varepsilon > 0$. For ATSP, Svensson et al. [32] answered the longstanding open question for the existence of a constant factor approximation in the affirmative. Although the current state of the art was very recently established by Traub and Vygen [33] with the ratio of $22 + \varepsilon$, this still leaves a significant gap between the positive results for TSP and ATSP, whereas the currently known lower bounds by Karpinski et al. [23] of $\frac{123}{122}$ for TSP and $\frac{75}{74}$ for ATSP do not indicate such a vast difference. This raises the question of how symmetry truly affects approximability.

The assumption of symmetry does not seem very natural. A study by Martínez Mori and Samaranayake [28] shows that road networks exhibit asymmetry even when only the lengths of the shortest paths are considered. Phenomena like road blocks, one-way streets and rush hour can result in unbounded violations of symmetry while the triangle inequality remains satisfied. In comparison, restricting to distances that satisfy the triangle inequality is a reasonable assumption in all scenarios where visiting cities more than once is acceptable. Finding a shortest tour that visits each city *at least* once translates to metric TSP by taking the shortest path metric, also called *metric closure*.

The *asymmetry factor* is the maximum ratio between the length of the shortest paths from $A$ to $B$ and $B$ to $A$ over all cities $A, B$. The investigation in [28] revealed that most asymmetries are insignificantly small. With these few but existing significant asymmetries in mind, we consider spending exponential time with respect to some measure of the degree of asymmetry. Our basic objective is to salvage the approximability of TSP for ATSP by allowing this increase in runtime. Formally, we give parameterized approximations (see e.g., [26]), which means a guaranteed performance ratio and a runtime of the form $poly(n)f(k)$, where $f$ is an arbitrary function, $n$ is the size of the instance and $k$ is a measure for asymmetry. This approach aims to offer efficiency for instances of low asymmetry and to improve our understanding of the challenges asymmetric distances pose to the design of approximation algorithms.

### 1.2 Our Results

We derive parameterized approximations based on the Christofides and the tree doubling algorithm with respective suitable parameters. Both parameters under study are bounded by the number of *asymmetric distances*, i.e., pairs of vertices $(u, v)$ for which traveling from $u$ to $v$ is cheaper than traveling from $v$ to $u$. Further, we combine these parameters with the asymmetry factor $\Delta$ [28] in the sense that they treat distances with asymmetry factor $\Delta \leq \beta$ for some $\beta \geq 1$ as symmetric, which shrinks both parameters to consider only the more severe $\beta$-*asymmetries* (distance from $v$ to $u$ is at least $\beta$ times the distance from $u$ to $v$). In particular, we derive parameterized approximations with

- ratio $\frac{7}{4} + \frac{3}{4}\beta$ for parameter $k$ = size of a vertex cover for the subgraph induced by the $\beta$-asymmetric distances (*generalized Christofides*);
- ratio $2 + \beta$ for parameter $z$ = minimum number of $\beta$-asymmetric distances in a minimum spanning arborescence (*generalized tree doubling*).

For $\beta = 1$, we prove the ratio of 2.5 to be tight for generalized Christofides. The lack of such a tightness result and further observations lead us to conjecture that generalized tree doubling is actually a 2-approximation for $\beta = 1$. Since the two parameters $k$ and $z$ are theoretically incomparable, we conduct experiments which show that generalized tree doubling frequently outperforms generalized Christofides with respect to parameter size.

The paper is organized as follows. In Sect. 3 we generalize the Christofides algorithm. Our main result, the more elaborate generalized tree doubling algorithm, is presented in Sect. 4. In Sect. 5 we give the combination with the asymmetry factor and Sect. 6 describes our experimental results. For the full version of this extended abstract, see [3].

### 1.3 Related Work

Conceptually, our approach can be seen as a study of *stability* with respect to asymmetry in the framework of *stability of approximation* by Böckenhauer et al. [6]. Probably the most extensively studied stability measure for (A)TSP is the $\beta$-triangle inequality, also called *parameterized* triangle inequality, which refers to the requirement $c(u, v) \leq \beta(c(u, w) + c(w, v))$ for all $u, v, w \in V$ with $u \neq v \neq w$. For ATSP with $\beta$-triangle inequality, the $\frac{1}{2(1-\beta)}$-approximation derived by Kowalik and Mucha [25] for $\beta \in (\frac{1}{2}, 1)$ improves upon a series of previous results [5,9,34] and is also known to be tight with respect to the cycle cover relaxation as lower bound. For TSP, the survey of Klasing and Mömke [24] gives a summary of the known results with $\beta$-triangle inequality.

Martínez Mori and Samaranayake [28] showed that the Christofides algorithm is $\frac{3}{2}$-stable with respect to the asymmetry factor, meaning that it can be used to compute a $\frac{3}{2}\Delta$-approximation for instances with asymmetry factor at most $\Delta$.

So far, there are only a few parameterized approximations for (variations of) TSP. Marx et al. [27] consider ATSP on a restricted graph class called $k$-nearly-embeddable. They derive approximations where the ratio and the runtime depend on structural parameters of the given instance. A true parameterized approximation for a TSP type problem is given by Böckenhauer et al. in [7] for deadline TSP, a generalization of TSP where some cities have to be reached by the tour within a given deadline. They give a 2.5-approximation that requires exponential time only with respect to the number of cities with deadline.

Another interesting approach to invest moderate exponential time is given by Bonnet et al. in [8]. They derive a routine that allows to compute for any $r \leq n$ a $\log r$-approximation for ATSP that requires time $\mathcal{O}^*(2^{\frac{n}{r}})$.

## 2 Preliminaries

Throughout the paper, instances of ATSP are always simple complete directed graphs denoted by $G = (V, A, c)$ with non-negative cost function $c$ on $A$. For $u, v \in V$, $(u, v)$ denotes the *arc* from $u$ to $v$ and $c(u, v)$ denotes its cost. For an arc $(u, v) \in A$ we call $(v, u) \in A$ the *opposite* arc. To refer to the connections between vertices without regarding any directedness, for an arc $(u, v) \in A$ and its opposite arc, we call $\{u, v\}$ an *arc-pair* or simply *link*. Links can be thought of like edges in an undirected graph. If the cost function $c$ satisfies the triangle inequality, i.e., $c(u, v) \leq c(u, w) + c(w, v)$ for all $u, v, w \in V$, we call $G$ *metric*. If the graph is not clear from context, we use $V[G]$ and $A[G]$ to denote the vertices and arcs of $G$, respectively.

In a not necessarily complete graph $G'$, a *trail* is a sequence of vertices where each vertex is equal to or has an arc to its successor. A *path* is a trail containing no vertex twice. *Circuit* and *cycle* denote a trail and a path where the last vertex has an arc to the first vertex, respectively. We denote a trail by $v_1, \dots, v_n$ and a circuit by $(v_1, \dots, v_n)$. A *tour* of $G'$ is a cycle that visits each vertex of $G'$.

If $G$ is metric, every trail can be turned into a path visiting the same vertices via a *metric shortcut* without increasing the cost, where metric shortcut means removing multiple occurrences of each vertex. All tours in $G$ are valid ATSP solutions, and we use $c^*(G)$ to denote the cost of an optimal solution for $G$.

For $G = (V, A, c)$ we denote the vertex-induced subgraph of $V' \subseteq V$ by $G[V']$, the arc-induced subgraph of $A' \subseteq A$ by $G[A']$ and also the link-induced subgraph of a set of links $E$ by $G[E]$. Slightly abusing notation, $G[V']$ and $G[A']$ then also inherit the weights of $G$. Further, for a subgraph $G'$ of $G$, we use $c(G')$ to denote the sum of all arc costs in $G'$. We observe:

**Lemma 1.** *Let $G$ be a metric graph and $V' \subseteq V$. Then, $G[V']$ is metric as well.*

**Lemma 2.** *Let $G$ be a metric graph and $V' \subseteq V$. Then, $c^*(G[V']) \leq c^*(G)$.*

We also use one other transformation we call *minor*. Here, $G'$ is a *minor* of $G$ if there is a series of *contractions* which, starting from $G$, result in $G'$. A contraction of $(u, v)$ replaces $u$ and $v$ with a single vertex $uv$ and sets $c(w, uv) = \min\{c(w, u), c(w, v)\}$ and $c(uv, w) = \min\{c(u, w), c(v, w)\}$ for all $w \in V \setminus \{u, v\}$.

## 3 Generalized Christofides Algorithm

The Christofides algorithm [10] is a polynomial approximation for TSP with performance ratio $\frac{3}{2}$. On instance $G$ it first computes a minimum spanning tree $T$

for $G$ and then adds a minimum cost perfect matching $M$ on the vertices $V'$ of odd degree in $T$. The resulting subgraph is connected and each vertex has an even degree, so it is possible to compute an Eulerian cycle for it, which is a circuit of cost $c(T) + c(M)$ that visits all vertices. Metric shortcuts turn this circuit into a tour. Since taking every second edge in an optimal tour for $G[V']$ gives a perfect matching for the vertices of odd degree, the edges in $M$ have a cost of at most $\frac{1}{2}c^*(G[V']) \leq \frac{1}{2}c^*(G)$. Together with the bound of $c^*(G)$ on the cost of $T$, this proofs the approximation ratio of $\frac{3}{2}$.

Regarding ATSP, the most dire problem of this approach is that combining $T$ and $M$ to an Eulerian circuit is impossible if some arcs point in the wrong direction, and it is unclear how to restrict $T$ and $M$ accordingly while keeping the relation of their cost to the optimum value. Due to this conceptual problem, we use a reduction to a TSP instance for which the Christofides algorithm can be applied. Observe that such a reduction cannot simply be designed by brute-force guessing the correct set of asymmetries in an optimal solution; fixing a subset of arcs to be in a solution cannot be modeled as an undirected instance. The design of our algorithm is instead based on a simple structural insight that allows the use of the Christofides algorithm on a symmetric subgraph.

We first explain an easier variant of the algorithm. The idea is to divide the graph into an asymmetric and a symmetric subgraph. For $G = (V, A, c)$ we define the set of *asymmetric links* by $E_a = \{\{u, v\} \mid u, v \in V, c(u, v) \neq c(v, u)\}$ and the set of *asymmetric* and *symmetric vertices* by $V_a = \{v \in V \mid \{u, v\} \in E_a \text{ for some } u \in V\}$, and $V_s = V \setminus V_a$, respectively.

We define the asymmetric subgraph by $G[V_a \cup \{v\}]$, where $v$ is an arbitrary vertex in $V_s$, and the symmetric one by $G[V_s]$. Note that tours through both subgraphs can be merged at the overlap in $v$ and turned into a tour of the whole graph with metric shortcuts. Combining in this way an exact solution for $G[V_a \cup \{v\}]$ and a $\frac{3}{2}$-approximate solution for $G[V_s]$, computed by the Christofides algorithm, overall yields a parameterized $\frac{5}{2}$-approximation with parameter $|V_a|$.

To improve this, consider a vertex cover $VC$ of $G[E_a]$. The complement of $VC$ forms an independent set in $G[E_a]$, implying that $G$ contains no asymmetric links between vertices in $V_s \cup (V_a \setminus VC)$. This can be exploited to consider the smaller structural parameter $z$, the size of a vertex cover in $G[E_a]$. The improved algorithm uses a vertex cover $VC$ in $G[E_a]$, selects a vertex $v \in V_s$ and considers $G[VC \cup \{v\}]$ as the asymmetric and $G[V \setminus VC]$ as the symmetric subgraph.

Using a simple $\mathcal{O}(m + 2^z z^2)$ algorithm (e.g. branching on the $k^2$-kernel as discussed in the introduction of [11] for "Bar Fight Prevention") for the minimal vertex cover for $G[E_a]$, and the dynamic programming algorithm by Held and Karp [17] for ATSP on $G[VC \cup \{v\}]$ in $\mathcal{O}(2^z z^2)$ yields the following.

**Theorem 1.** *Metric ATSP can be $\frac{5}{2}$-approximated in $\mathcal{O}(n^3 + 2^z z^2)$ where $z$ is the size of a minimum vertex cover of the subgraph induced by all asymmetric links.*

Instead of exact algorithms for the vertex cover for $G[E_a]$ and the solution on $G[VC \cup \{v\}]$, we can also use approximations. A 2-approximation for vertex cover and the $\frac{2}{3}\log n$-approximation of Feige and Singh [14] on $G[VC \cup \{v\}]$ yields the following interesting result.

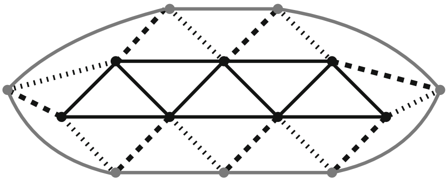

**Fig. 1.** $G_k$ for $k = 7$: Black and gray links are symmetric with cost 2, dotted links are symmetric with cost 1. Dashed links are asymmetric, with cost 1 from gray to black vertex and cost 2 from black to gray vertex.

**Corollary 1.** *Metric ATSP can be* $(\frac{2}{3}\log x + \frac{3}{2})$*-approximated in polynomial time, where* $x = \min(2z + 1, |V_a|)$*,* $V_a$ *is the set of asymmetric vertices and* $z$ *is the size of a minimum vertex cover for the subgraph induced by all asymmetric links.*

This improves upon the approximation ratio of $\frac{2}{3}\log n$ if $\frac{x}{n} < 2^{-\frac{9}{4}}$, meaning that $G[VC \cup \{v\}]$ only contains a sufficiently small fraction of the vertices. We note that the result of Asadapour et al. [2] gives a polynomial $(8\log(z)/\log\log(z) + \frac{3}{2})$-approximation, which is asymptotically stronger but less suitable for the instances with small values of $z$ we are interested in.

Further, note that one can also use any approximation for TSP (not just the Christofides algorithm) for the symmetric subgraph and obtain an $(\alpha + 1)$-approximation for ATSP from any $\alpha$-approximation for TSP.

It remains to see if this approach can be improved. Aiming for a smaller parameter seems difficult as this would not split off a symmetric subgraph. Regarding a possible improvement of the ratio, one might hope to salvage the ratio of $\frac{3}{2}$ for TSP, obtained by the Christofides algorithm, for ATSP. However, such an improvement requires a different algorithmic strategy as the ratio in Theorem 1 is asymptotically tight, which can be shown as follows.

We define a family of graphs $G_k$ for $k \in \mathbb{N}$, $k > 2$ such that the approximation ratio converges to 2.5 for increasing $k$, Fig. 1 describes $G_7$. The black zig-zag pattern is the textbook example for the tightness of the Christofides algorithm. The idea is that the gray vertices build the minimum vertex cover such that the black zig-zag pattern becomes the symmetric instance. The gray cycle is then the asymmetric subgraph and solving it exactly yields a tour of cost $2k$. Together with the approximation on the symmetric subgraph, which converges to $3k$, this results in a tour of length $5k$. As the optimal tour takes the dotted and dashed links in the cheaper direction and has cost $2k$, we deduce that 2.5 is asymptotically tight for Theorem 1.

## 4 Generalized Tree Doubling Algorithm

One other widely known approximation for TSP is the tree doubling algorithm. It computes a minimum spanning tree (MST) and doubles every edge in it to

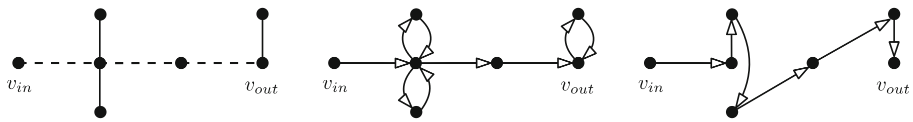


**Fig. 2.** Exemplary construction for a suitable path $\chi_i$. Left: spanning tree with $P_i$ dashed; Middle: trail through partially doubled edges; Right: resulting path.

ensure the existence of an Eulerian circuit. Since the circuit uses every MST edge exactly twice, it is twice as expensive as the tree, which itself is at most as expensive as the optimum tour. Thus, transforming the circuit with metric shortcuts gives a 2-approximation. To adapt this approach to ATSP we use a *minimum spanning arborescence* (MSA) as the directed variant of an MST. Tree doubling then runs into trouble when the cost of an opposite arc is arbitrarily higher than the direction contained in the MSA. These arcs are the core of the problem and hence our basis to generalize the tree doubling algorithm.

Formally, we call $(u, v) \in A$ a *one-way arc* in $G = (V, A, c)$ if $c(u, v) < c(v, u)$. In a nutshell, our algorithm removes all one-way arcs from an MSA, computes a tour for each resulting connected component by an altered tree doubling routine and uses exponential time in the number of removed one-way arcs to connect these subtours to a solution for the whole graph. For a best runtime, we hence want to keep the number of one-way arcs in the starting MSA as small as possible. For our parametrization, we formally define $k$ to be the minimum number of one-way arcs in an MSA for $G$.

At first glance, it might seem that finding an MSA with $k$ one-way arcs is a difficult task. However this can be accomplished by searching for an MSA with the altered weight function $c'$ defined by $c'(e) = |V|c(e) + 1$ if $e$ is a one-way arc, and $c'(e) = |V|c(e)$, otherwise. Trying every possible root vertex with this altered weight function, and the Chu–Liu/Edmonds algorithm [13] with Fibonacci heaps [15] to compute the MSA, yields the following result.

**Lemma 3.** *Let $G$ be a metric ATSP instance, then an MSA of $G$ with a minimum number of one-way arcs can be computed in $\mathcal{O}(n^3)$.*

With this best MSA, we can describe our generalized tree doubling algorithm. Let $T$ be the MSA for $G$ computed with Lemma 3, and let $T_1, \ldots, T_{k+1}$ be the connected components in the graph created by deleting all $k$ one-way arcs from $T$. We construct a graph $M$ by contracting each set of vertices $V[T_i]$ to one vertex $v_i^M$ with our notion of contraction to a minor. This results in $V[M] = \{v_1^M, \ldots, v_{k+1}^M\}$ and for all $v_i^M, v_j^M \in V[M]$ with $i \neq j$, $c(v_i^M, v_j^M) = \min(\{c(t_i, t_j) \mid t_i \in V[T_i], t_j \in V[T_j]\})$.

**Lemma 4.** *Let $G$ be a metric ATSP instance and $M$ be minor of $G$, then $c^*(M) \leq c^*(G)$.*

Since $M$ only contains $k + 1$ vertices, we brute-force an optimal tour $\tau'$ for $M$. It remains to extend $\tau'$ to a tour of $G$. Consider a vertex $v_i^M$ in $M$ (which corresponds to the component $T_i$) and assume w.l.o.g. that in $\tau'$ it is preceded by $v_{i-1}^M$

and precedes $v_{i+1}^M$. Further, let $(v_{out}^{T_{i-1}}, v_{in}^{T_i})$ and $(v_{out}^{T_i}, v_{in}^{T_{i+1}})$ be the cheapest arc between $T_{i-1}$ and $T_i$, and $T_i$ and $T_{i+1}$, respectively. The goal is to find a path $\chi_i$ that starts in $v_{in}^{T_i}$, ends in $v_{out}^{T_i}$, and spans all vertices in $T_i$ (formally a solution to $s$-$t$-path TSP for $G[T_i]$ with $s = v_{in}^{T_i}$ and $t = v_{out}^{T_i}$). Replacing $v^{T_i}$ in $\tau'$ by $\chi_i$ for each $i$ turns $\tau'$ into a tour. However, the cost of $\chi_i$ has to be bounded.

Such a path $\chi_i$ through $T_i$ can be found by adapting the tree doubling algorithm. We treat $T_i$ as undirected and double all its edges that are not on the shortest path $P_i$ from $v_{in}^{T_i}$ to $v_{out}^{T_i}$. The resulting graph contains an Eulerian trail from $v_{in}^{T_i}$ to $v_{out}^{T_i}$, which is turned into a path by metric shortcuts ensuring that $v_{in}^{T_i}$ and $v_{out}^{T_i}$ remain start and end node, see Fig. 2 for an example. Observe that we cannot use any of the better approximations for $s$-$t$-path TSP, such as [19], since the subgraph induced by the vertices in $T_i$ is not necessarily completely symmetric. Further, even if this was possible, the only information we can use to compare the tour through $T_i$ with the optimum for the whole graph $G$ are the arcs from the MSA, which in the worst case always results in a ratio of 2.

For the cost of $\chi_i$, note that it contains for each arc $(u, v)$ in $T_i$ at most both $(u, v)$ and $(v, u)$. Since there are no one-way arcs in $T_i$, any opposite arc is at most as expensive as the original arc in $T_i$. Consequently, the cost of $\chi_i$ is at most twice the cost of the arcs in $T_i$ and the sum of all $\chi_i$ is at most $2c^*(G)$. In combination with the cost of at most $c^*(G)$ for $\tau'$, this yields:

**Theorem 2.** *Metric ATSP can be 3-approximated in $\mathcal{O}(2^k \cdot k^2 + n^3)$, where $k$ is the minimum number of one-way arcs in a minimum spanning arborescence.*

Contrary to the approach in Sect. 3, we cannot plug in some approximation to find a good tour $\tau'$ for $M$ to derive something like Corollary 1. Note that the minor $M$ is not necessarily metric since contractions do not preserve the triangle inequality. Still, one might ask if $M$, as minor of a metric graph, has useful structural properties. However, the following result discourages such ideas.

**Lemma 5.** *Let $G$ be a complete, directed graph with cost function $c$. Then, there exists a complete, metric graph $\hat{G}$ of which $G$ is a minor.*

Computing a tour for $M$ is related to the *generalized traveling salesman problem* (GTSP) which can be tracked back to publications of Henry-Labordère and Saksena [18,31]. Given a partition of the cities into $r$ sets, GTSP asks for a minimum cost tour containing (at least) one vertex from each of the $r$ sets. Unfortunately, there are no known efficient ways to solve or approximate GTSP. However, we observe that using an optimal GTSP tour for the vertex sets corresponding to $T_1, \ldots, T_{k+1}$ instead of the tour through $M$ still yields a 3-approximation. In fact, this remains true even if we fix one arbitrary city for each set, which yields a graph $M'$ that is just an induced subgraph and hence metric. For this simplified approach, the ratio 3 is indeed asymptotically tight.

Aside from the fact that we did not find a tight example for Theorem 2, seeing that the choice of any arbitrary vertex still yields a 3-approximation causes us to conjecture that our more sophisticated generalization of the tree doubling algorithm is in fact a 2-approximation. Proving such a ratio however requires an exploitable connection between the cost for the paths $\chi_i$ and the cost of $\tau'$.

## 5 Trading Approximation Quality for Runtime

In real life, we expect instances with many small asymmetries which have little impact but lead to relatively large parameter values. Therefore, ignoring asymmetric links where both directions have similar cost and trading some approximation quality for running time yields an intriguing perspective. As a formal way to describe moderate asymmetry, we use the asymmetry factor of Martínez Mori and Samaranayake [28] as introduced in Sect. 1.3. Since $\Delta$ is commonly used for the maximum degree, and we want to describe variable restrictions of the asymmetry factor, we use $\beta$ instead. For $\beta \geq 1$ we call a link $\{u, v\}$ or arc $(u, v)$ $\beta$-*symmetric* if $\frac{1}{\beta} \leq \frac{c(u,v)}{c(v,u)} \leq \beta$, otherwise it is called $\beta$-*asymmetric*. We show that our algorithms support a quality-runtime trade-off with respect to $\beta$.

### 5.1 Relaxed Generalized Christofides Algorithm

For a given $\beta$ we modify the algorithm presented in Sect. 3 by treating every $\beta$-symmetric link as symmetric. This results in parametrization by the vertex cover of the subgraph induced by all $\beta$-asymmetric links. We denote this parameter by $z_\beta$. Since the $\beta$-symmetric subgraph is not completely symmetric, the Christofides algorithm cannot be directly used. Martínez Mori and Samaranayake [28] showed that it is $\frac{3}{2}$-stable by replacing every link with an undirected edge and assigning it the cost of the more expensive direction. Combined with the arguments used for Theorem 1, this gives a parameterized $(\frac{3}{2}\beta + 1)$-approximation for parameter $z_\beta$. This can be improved by turning the $\beta$-symmetric subgraph symmetric by assigning the cost of the cheaper direction. Although this may not yield a metric graph, it suffices that the original graph is metric to prove that the Christofides algorithm yields a good solution.

**Theorem 3.** *For any $\beta \geq 1$, metric ATSP can be $(\frac{3}{4}\beta + \frac{7}{4})$-approximated in $\mathcal{O}(n^3 + 2^{z_\beta} z_\beta^2)$ where $z_\beta$ is the size of a minimum vertex cover of the subgraph induced by all $\beta$-asymmetric links.*

### 5.2 Relaxed Generalized Tree Doubling Algorithm

For the generalized tree doubling algorithm, we define a $\beta$-*one-way arc* as a one-way arc that is $\beta$-asymmetric. We denote by $k_\beta$ the minimum number of $\beta$-one-way arcs in an MSA. Note that the strategy in Lemma 3 can also be used to find an MSA with $k_\beta$ $\beta$-one-way arcs. The generalization of Theorem 2 is straightforward, instead of deleting all one-way arcs we only delete $\beta$-one-way arcs. This results in fewer components and a smaller graph $M_\beta$. The drawback is a change to the cost analysis: so far, we considered the component trees $T_i$ to be symmetric. Now for every $(u, v) \in A[T_i]$, the opposite arc $(v, u)$ can be up to $\beta$ times as expensive. The adjusted tree doubling algorithm for the path through $T_i$ uses every arc in $T_i$ and its opposite arc at most once, which in total costs at most $(1 + \beta)c^*(G)$. Combined with the cost of at most $c^*(G)$ for an optimum tour through $M_\beta$, this yields:

**Theorem 4.** *For any $\beta \geq 1$, metric ATSP can be $(2+\beta)$-approximated in $\mathcal{O}(2^{k_\beta} k_\beta^2 + n^3)$ where $k_\beta$ is the minimum number of $\beta$-one-way arcs in an MSA.*

# 6 Experimental Results

To test the practical viability of our algorithms, we implemented them in their relaxed form (see Sect. 5) to also observe their behavior when certain asymmetries are ignored. We evaluated on the asymmetric graphs from the TSPLIB collection [29], the standard benchmark for TSP solvers, and on a set of specific ATSP instances extracted from road networks by Rodríguez and Ruiz [30].

## 6.1 Implementation Details

Our implementation (available on GitHub[1]) is written in *Python 3*, except for the vertex cover solver, which is written in *Java*. We used the Python library *NetworkX* [16] for graph manipulation, the C++ library *Lemon* [12] for computing MSAs, and *Concorde* [1] for solving TSP exactly. Since *Concorde* is a TSP solver, we transformed the ATSP instances into TSP instances with the transformation presented by Jonker and Volgenant [20, 21].

We note that the runtime of our implementations is incomparable to state of the art ATSP solvers. Among others, the reason is Python's inherently low performance and the inefficiency of solving ATSP with Concorde. However, this is of no importance for our evaluation of approximation ratio, parameter size, and the proof of concept.

## 6.2 Experiments

In the TSPLIB there are 19 asymmetric instances ranging from 17 up to 443 vertices. As some of the instances are not metric, we computed the metric closure of each graph. The instances' names contain the number of vertices (e.g., *ftv33*) and similar names indicate similar properties. For example, instances starting with *rbg* have relatively high symmetry and a high number of zero-cost arcs. Contrasting that, the instances with prefix *ftv* contain little symmetry, but most asymmetric links are only moderately asymmetric. Most instances are rather small, with only 6 of the instances having more than 70 vertices. We ignored the instance *br17* as its metric closure is completely symmetric.

For each TSPLIB instance we executed each algorithm five times with different values for $\beta$ and recorded the value of the parameter as well as the approximation ratio. Starting with $\beta = 1$ (which corresponds to 100% of the asymmetric links), we raised the value of $\beta$ each step, reducing the number of asymmetric links treated as asymmetric to a quarter of the previous experiment. Some instances include many zero-cost arcs, so there is no value of $\beta$ ignoring those. We considered zero-cost arcs to have a small positive cost (set to 0.1) when calculating the asymmetry factor, thus treating links with a small additive error

[1] https://github.com/Blaidd-Drwg/atsp-approximation.

**Table 1.** Experimental results on TSPLIB instances with percentage of asymmetric links that were treated as asymmetric shown in the column header. Each cell contains parameter value and approximation factor, separated by a slash (trivial parameter value 0 omitted in 0% column). Superiority in the sense of smaller kernel or better approximation ratio is highlighted with bold font.

| | Generalized Christofides algorithm | | | | | Generalized tree doubling algorithm | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | 100% | 25% | 6.25% | 1.56% | 0% | 100% | 25% | 6.25% | 1.56% | 0% |
| ft53 | 53/**1.00** | 29/1.54 | 13/1.70 | 6/1.69 | **1.72** | **45**/1.08 | **25/1.36** | **6/1.42** | **1/1.57** | 1.97 |
| ft70 | 69/1.02 | 34/1.24 | 12/1.26 | 7/1.41 | **1.24** | **64**/1.02 | **27/1.13** | **4/1.20** | **2/1.21** | 1.28 |
| ftv170 | 155/1.17 | 123/1.38 | **97**/1.57 | **64**/1.85 | 2.37 | **108/1.14** | **107/1.14** | 103/**1.21** | 75/**1.46** | **1.81** |
| ftv33 | 29/**1.12** | 19/1.45 | 11/**1.43** | 5/1.56 | **1.33** | **19**/1.34 | **16/1.34** | 11/1.44 | **2/1.23** | 1.50 |
| ftv35 | 32/**1.07** | 21/1.51 | 12/1.55 | 6/1.49 | **1.38** | **23**/1.15 | **17/1.23** | **11/1.47** | **2/1.28** | 1.58 |
| ftv38 | 33/**1.13** | 23/1.38 | 12/**1.43** | 7/1.47 | **1.39** | **23**/1.24 | **18/1.33** | 12/1.54 | **3/1.30** | 1.62 |
| ftv44 | 40/**1.09** | 32/**1.38** | 19/1.46 | 10/1.56 | **1.54** | **32**/1.24 | **25**/1.41 | **18/1.41** | **7/1.50** | 1.79 |
| ftv47 | 44/**1.05** | 32/1.47 | 19/1.66 | 13/1.65 | 1.66 | **35**/1.09 | **30/1.16** | 19/**1.34** | **9/1.38** | **1.58** |
| ftv55 | 49/**1.13** | 38/1.44 | **23**/1.57 | 15/1.65 | **1.84** | **37**/1.20 | **32/1.26** | 25/**1.34** | **12/1.58** | 2.00 |
| ftv64 | 57/1.11 | 46/1.46 | **30**/1.66 | 18/1.73 | 1.72 | **50/1.10** | **43/1.15** | 31/**1.29** | **14/1.71** | **1.45** |
| ftv70 | 63/**1.11** | 50/1.43 | **32**/1.64 | 20/1.72 | 1.96 | **53**/1.26 | **47/1.14** | 33/**1.21** | **16/1.57** | **1.51** |
| kro124p | 99/1.11 | 86/1.30 | 65/1.36 | 40/1.41 | **1.24** | **81/1.06** | **70/1.13** | **57/1.20** | **34/1.28** | 1.37 |
| p43 | 15/1.01 | 6/1.01 | 0/1.01 | 0/1.01 | 1.01 | **0**/1.01 | **0**/1.01 | 0/1.01 | 0/1.01 | 1.01 |
| rbg323 | **148/1.02** | 59/**1.17** | 43/**1.19** | 18/1.30 | 1.34 | 235/1.09 | **22**/1.27 | **6**/1.27 | **0**/1.30 | **1.30** |
| rbg358 | **108/1.01** | 47/**1.13** | 27/**1.15** | 22/**1.14** | **1.18** | 232/1.03 | **39**/1.14 | **18**/1.19 | **13**/1.20 | 1.22 |
| rbg403 | 125/**1.01** | 41/**1.12** | 11/1.26 | 11/1.26 | **1.17** | **113**/1.05 | **30**/1.14 | **0/1.24** | **0/1.24** | 1.24 |
| rbg443 | 138/**1.00** | 43/**1.14** | 12/1.24 | 12/1.24 | **1.15** | **127**/1.04 | **32**/1.17 | **0**/1.24 | **0**/1.24 | 1.24 |
| ry48p | 47/1.20 | 37/1.40 | 23/1.46 | 11/1.47 | **1.16** | **28/1.10** | **22/1.14** | **11/1.24** | **5/1.29** | 1.21 |

as symmetric in case of these otherwise undauntedly asymmetric one-way arcs of cost 0. Note that we did not alter the instance, but only used these additive errors for relaxation decisions. Finally, $\beta$ was set to $\infty$, such that the graph is treated as completely symmetric. This results in the non-generalized versions of the tree doubling and Christofides algorithm. The results are shown in Table 1.

The second dataset contains 450 ATSP instances based on travel distances between random points sampled across different regions and cities in Spain. The graphs in this dataset have between 50 and 500 vertices. On average 98.8% of the links are asymmetric (std. dev. 1.08%) and no graph contains arcs of cost zero. Most links are however only slightly asymmetric: denoting by *asymmetry factor* the relative difference between the cost of a links more expensive arc and its opposite arc, the mean asymmetry factor is 3.55% on average over all graphs (std. dev. 0.040%). The median asymmetry factor is 1.32% (std. dev. 1.56%) on average. There are however also links with large asymmetry factor. The highest asymmetry factor is 15.0 (std. dev. 58.8) on average. Overall this makes the graphs in the second dataset very relevant to the algorithms we present. Unfortunately, due to computational constraints and the size of the dataset and the graphs therein, we could only determine the values of the parameters and not the cost of all optimal tours and the obtained approximation ratio. Figure 3 presents the relative value of $z$ and $k$ for different values of $\beta$.

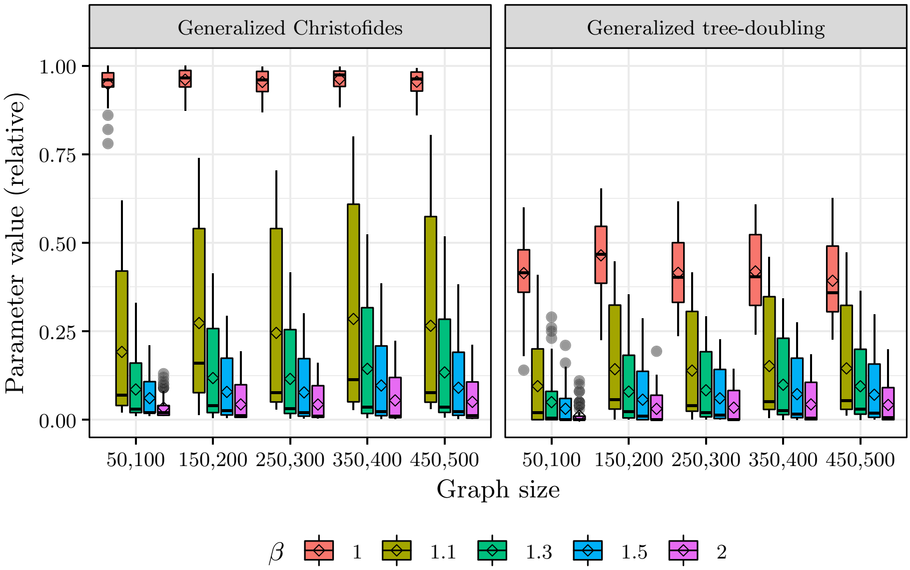


**Fig. 3.** Parameter values relative to graph size for generalized Christofides and tree doubling algorithms for different values of $\beta$. Each box spans the second and third quartile of the data and whiskers extend for 1.5 inter-quartile-ranges. The median is marked as a line, the mean as a rhombus and outliers as disks.

### 6.3 Evaluation

First, we note that most graphs in the TSPLIB contain very little symmetry. This leads to large parameter values for $\beta = 1$, i.e., only some graphs with more than 10% symmetry have parameter values below 50% of the graph size. Still, we observe that the approximation factor is always far below the upper bound, never exceeding even 2.0. Also, we see that interpolating $\beta$ to reduce the number of relevant asymmetric links produces a valuable trade-off between approximation quality and parameter value. Comparing both algorithms, we observe that on the majority of instances and values for $\beta$ the generalized tree doubling algorithm produces smaller parameter values.

This can also be observed on the instances of the second dataset, which we consider to be more representative of realistic inputs. We want to highlight that the parameters are significantly smaller than the size of the input graphs even for small values of $\beta$. E.g., for the generalized tree doubling algorithm with $\beta = 1.1$ the median relative parameter value over all instances is 0.045. It also seems that the relative size of the parameters is stable for different input sizes.

These results underline the practicality of our approach, especially with regards to the parameter values obtained by choosing a suitable $\beta$.

## References


1. Applegate, D., Bixby, R., Cook, W., Chvátal, V.: On the solution of traveling salesman problems. Doc. Math. **111**, 645–656 (1998)
2. Asadpour, A., Goemans, M.X., Madry, A., Gharan, S.O., Saberi, A.: An O(log n/log log n)-approximation algorithm for the asymmetric traveling salesman problem. Oper. Res. **65**(4), 1043–1061 (2017). https://doi.org/10.1287/opre.2017.1603
3. Behrendt, L., Casel, K., Friedrich, T., Lagodzinski, J.A.G., Löser, A., Wilhelm, M.: From symmetry to asymmetry: generalizing TSP approximations by parametrization. CoRR abs/1911.02453 (2019). http://arxiv.org/abs/1911.02453
4. van Bevern, R., Slugina, V.A.: A historical note on the 3/2-approximation algorithm for the metric traveling salesman problem. Hist. Math. **53**, 118–127 (2020). https://www.sciencedirect.com/science/article/pii/S0315086020300240
5. Bläser, M., Manthey, B., Sgall, J.: An improved approximation algorithm for the asymmetric TSP with strengthened triangle inequality. J. Discret. Algorithms **4**(4), 623–632 (2006). https://doi.org/10.1016/j.jda.2005.07.004
6. Böckenhauer, H., Hromkovic, J., Klasing, R., Seibert, S., Unger, W.: Towards the notion of stability of approximation for hard optimization tasks and the traveling salesman problem. Theor. Comput. Sci. **285**(1), 3–24 (2002). https://doi.org/10.1016/S0304-3975(01)00287-0
7. Böckenhauer, H., Hromkovic, J., Kneis, J., Kupke, J.: The parameterized approximability of TSP with deadlines. Theor. Comput. Sci. **41**(3), 431–444 (2007). https://doi.org/10.1007/s00224-007-1347-x
8. Bonnet, É., Lampis, M., Paschos, V.T.: Time-approximation trade-offs for inapproximable problems. J. Comput. Syst. Sci. **92**, 171–180 (2018). https://doi.org/10.1016/j.jcss.2017.09.009
9. Chandran, L.S., Ram, L.S.: Approximations for ATSP with parametrized triangle inequality. In: Alt, H., Ferreira, A. (eds.) STACS 2002. LNCS, vol. 2285, pp. 227–237. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-45841-7_18
10. Christofides, N.: Worst-case analysis of a new heuristic for the travelling salesman problem. Technical report 388, Graduate School of Industrial Administration, Carnegie Mellon University (1976)
11. Cygan, M., et al.: Parameterized Algorithms. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-21275-3
12. Dezső, B., Jüttner, A., Kovács, P.: LEMON - an open source C++ graph template library. Electron. Notes Theor. Comput. Sci. **264**(5), 23–45 (2011). https://doi.org/10.1016/j.entcs.2011.06.003
13. Edmonds, J.: Optimum branchings. J. Res. Natl. Bur. Stan. Sect. B Math. Math. Phys. **71B**(4), 233 (1967)
14. Feige, U., Singh, M.: Improved approximation ratios for traveling salesperson tours and paths in directed graphs. In: Charikar, M., Jansen, K., Reingold, O., Rolim, J.D.P. (eds.) APPROX/RANDOM -2007. LNCS, vol. 4627, pp. 104–118. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-74208-1_8
15. Gabow, H.N., Galil, Z., Spencer, T.H., Tarjan, R.E.: Efficient algorithms for finding minimum spanning trees in undirected and directed graphs. Combinatorica **6**(2), 109–122 (1986). https://doi.org/10.1007/BF02579168
16. Hagberg, A., Schult, D., Swart, P.: Exploring network structure, dynamics, and function using NetworkX. In: Proceedings of the 7th Python in Science Conference, pp. 11–15 (2008). http://conference.scipy.org/proceedings/SciPy2008/paper_2/

17. Held, M., Karp, R.M.: A dynamic programming approach to sequencing problems. J. Soc. Ind. Appl. Math. **10**(1), 196–210 (1962)
18. Henry-Labordère, A.L.: The record balancing problem: a dynamic programming solution of a generalized traveling salesman problem. RAIRO **B-2**, 43–49 (1969)
19. Hoogeveen, J.A.: Analysis of Christofides' heuristic: some paths are more difficult than cycles. Oper. Res. Lett. **10**(5), 291–295 (1991). https://doi.org/10.1016/0167-6377(91)90016-I
20. Jonker, R., Volgenant, T.: Transforming asymmetric into symmetric traveling salesman problems. Oper. Res. Lett. **2**(4), 161–163 (1983)
21. Jonker, R., Volgenant, T.: Transforming asymmetric into symmetric traveling salesman problems: erratum. Oper. Res. Lett. **5**(4), 215–216 (1986)
22. Karlin, A., Klein, N., Gharan, S.O.: A (slightly) improved approximation algorithm for metric TSP. In: Khuller, S., Williams, V.V. (eds.) Proceedings of the STOC 2021, pp. 32–45. ACM (2021). https://doi.org/10.1145/3406325.3451009
23. Karpinski, M., Lampis, M., Schmied, R.: New inapproximability bounds for TSP. J. Comput. Syst. Sci. **81**(8), 1665–1677 (2015). https://doi.org/10.1016/j.jcss.2015.06.003
24. Klasing, R., Mömke, T.: A modern view on stability of approximation. In: Böckenhauer, H.-J., Komm, D., Unger, W. (eds.) Adventures Between Lower Bounds and Higher Altitudes. LNCS, vol. 11011, pp. 393–408. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-98355-4_22
25. Kowalik, Ł, Mucha, M.: Two approximation algorithms for ATSP with strengthened triangle inequality. In: Dehne, F., Gavrilova, M., Sack, J.-R., Tóth, C.D. (eds.) WADS 2009. LNCS, vol. 5664, pp. 471–482. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-03367-4_41
26. Marx, D.: Parameterized complexity and approximation algorithms. Comput. J. **51**(1), 60–78 (2008). https://doi.org/10.1093/comjnl/bxm048
27. Marx, D., Salmasi, A., Sidiropoulos, A.: Constant-factor approximations for asymmetric TSP on nearly-embeddable graphs. In: Proceedings of APPROX/RANDOM 2016, pp. 16:1–16:54. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2016). https://doi.org/10.4230/LIPIcs.APPROX-RANDOM.2016.16
28. Mori, J.C.M., Samaranayake, S.: Bounded asymmetry in road networks. Sci. Rep. **9**(11951), 1–9 (2019). https://doi.org/10.1038/s41598-019-48463-z
29. Reinelt, G.: TSPLIB–a traveling salesman problem library. INFORMS J. Comput. **3**(4), 376–384 (1991). https://doi.org/10.1287/ijoc.3.4.376
30. Rodríguez, A., Ruiz, R.: The effect of the asymmetry of road transportation networks on the traveling salesman problem. Comput. Oper. Res. **39**(7), 1566–1576 (2012). https://doi.org/10.1016/j.cor.2011.09.005
31. Saksena, J.P.: Mathematical model of scheduling clients through welfare agencies. Comput. Oper. Res. J. **8**, 185–200 (1970)
32. Svensson, O., Tarnawski, J., Végh, L.A.: A constant-factor approximation algorithm for the asymmetric traveling salesman problem. J. ACM **67**(6), 37:1-37:53 (2020). https://doi.org/10.1145/3424306
33. Traub, V., Vygen, J.: An improved approximation algorithm for ATSP. In: Proceedings of STOC 2020, pp. 1–13. ACM (2020). https://doi.org/10.1145/3357713.3384233
34. Zhang, T., Li, W., Li, J.: An improved approximation algorithm for the ATSP with parameterized triangle inequality. J. Algorithms **64**(2–3), 74–78 (2009). https://doi.org/10.1016/j.jalgor.2008.10.002