Technische Universität Berlin

FAKULTÄT II
MATHEMATIK UND
NATURWISSENSCHAFTEN
Institut für Mathematik

A New Bound on the Length of Minimum Cycle Bases

by

Christian Liebchen

TU Berlin, Institut für Mathematik, Sekr. MA 6-1
Strasse des 17. Juni 136, D-10623 Berlin, Germany
liebchen@math.tu-berlin.de

Romeo Rizzi

Università degli Studi di Udine
Dipartimento di Matematica ed Informatica
Via delle Scienze, 208, I-33100 Udine, Italy

No. 2006/32

A New Bound on the Length of Minimum Cycle Bases *

Romeo Rizzi [a] Christian Liebchen [b,*]

[a] *Università di Udine*

[b] *Technische Universität Berlin, Institut für Mathematik, Combinatorial Optimization and Graph Algorithms, Straße des 17. Juni 136, D-10623 Berlin, Germany*

Abstract

For any weighted graph we construct a cycle basis of length $O(W \cdot \log n \log\log n)$, where W denotes the sum of the weights of the edges. This improves the upper bound that was obtained only recently by Elkin et al. (2005) by a logarithmic factor. From below, our result has to be compared with $\Omega(W \cdot \log n)$, being the length of the minimum cycle bases (MCB) of a class of graphs with large girth.

We achieve this bound by not restricting ourselves to strictly fundamental cycle bases—as it is inherent to the approach of Elkin et al.—but rather also considering weakly fundamental cycle bases in our construction. This way, we can take profit of some nice properties of Hierarchically Partitioned Metrics (HPM) as they have been introduced by Bartal (1998).

1 Introduction

We consider a simple undirected 2-vertex connected graph $G = (V, E)$. A non-negative weight function w may be defined on the edges of G. We denote the sum of the weights of all the edges by W. A *circuit* C of G is a connected subgraph of G, in which each vertex has either degree two or zero. Sometimes, we may refer to C only through its edge set. The weight $w(C)$ of a circuit C is defined as the sum of the weights of its edges. The *incidence vector* $\gamma_C \in \{0,1\}^E$ of a circuit C is the characteristic vector of its edge set.

$\star$ Matheon

* Corresponding Author.
Email address: liebchen@math.tu-berlin.de (Christian Liebchen).

Cycle Bases. The *cycle space* $\mathcal{C}(G)$ of G is the linear vector subspace of $\mathrm{GF}(2)^E$ which is spanned by the incidence vectors of the circuits of G. It is well known that $\nu := \dim(\mathcal{C}(G)) = |E|-|V|+1$, or $\nu = m-n+1$ with m and n defined as usually.

A *cycle basis* $\mathcal{B}$ of G is a set of ν circuits of G which span $\mathcal{C}(G)$. The weight $w(B)$ of a cycle basis $\mathcal{B}$ is defined as $w(B) = \sum_{C\in B} w(C)$. In the Minimum Cycle Basis (MCB) Problem, we want to find a cycle basis of minimum weight. These are sought in many applications, see [Hor87] and references therein.

For some spanning tree T of G and some non-tree edge $e \in E \setminus T$, we denote by $C_T(e)$ the unique circuit in $T \cup \{e\}$, which is called the *fundamental circuit* induced by e with respect to T. It is well known that the set of all the fundamental circuits w.r.t. a spanning tree T is a cycle basis; these cycle bases have special and often desirable properties, and are called *strictly fundamental*. A cycle basis is called *weakly fundamental* if its circuits can be labeled as $C_1, C_2, \ldots, C_\nu$ such that

$$C_i \setminus (C_1 \cup \cdots \cup C_{i-1}) \neq \emptyset, \quad \text{for all } i = 2, \ldots, \nu. \tag{1}$$

Let $e_i \in C_i \setminus (C_1 \cup \cdots \cup C_{i-1})$. Notice that $\{e_1, \ldots, e_\nu\}$ are the co-tree arcs with respect to some spanning tree $T \subset E$. In particular the circuit C_1 is a fundamental circuit with respect to that spanning tree T.

Metrics. The *metric* d_G that is associated with a weighted connected graph (G, w) is defined by the shortest-path distances in G with respect to w. The *diameter* $\mathrm{diam}(G)$ of (G, w) is defined as

$$\mathrm{diam}(G) := \max\{d_G(u, v) : \; u, v \in V\}. \tag{2}$$

Let $V' \subset V$ and consider the induced subgraph $G[V']$ of G. The *strong internal diameter of* V' is $\mathrm{diam}(G[V'])$. In contrast, the *weak diameter of* V' is the maximum distance *in* G between two vertices in V' ([KRS01]). A metric d on a set of elements V is said to *dominate* another metric d' on V if for all $u, v \in V$ we have $d'(u, v) \leq d(u, v)$.

Related Work. Deo et al. ([DKP82]) conjecture that every unweighted graph has a strictly fundamental cycle basis of length $O(n^2)$. Since strictly fundamental cycle bases specialize general cycle bases, their conjecture may even serve as the first estimate on the length of an MCB. Still for unweighted graphs, Horton ([Hor87]) introduced a heuristic for computing a cycle basis of length $O(n^2)$. Later, it was shown in [Lie03] that the resulting cycle basis is always weakly fundamental. As the requirement of *strict* fundamentality is relaxed here, this result can only serve as a partial answer to Deo et al.'s conjecture.

Alon et al. ([AKPW95]) gave a more direct answer to Deo et al.'s conjecture. They prove that every weighted graph has a strictly fundamental cycle basis of length $O(W \cdot \exp(O(\sqrt{\log n \log\log n})))$. Recently, this was improved by Elkin et al. ([EEST05]) to only $O(W \cdot \log^2 n \log\log n)$. It is known that there exist graphs with large girth such that the length of their MCB is $\Omega(W \cdot \log n)$, see [Bol78]. Hence, for general graphs the bound of Elkin et al. could be improved asymptotically by at most $\log n \log\log n$.

Bartal ([Bar98]) investigated a problem which—at least at first sight—could appear not being related too much to cycle bases of graphs. He approximates a general metric by a tree metric, which is defined on some auxiliary graph, on a so-called k-Hierarchically Well-Separated Tree (k-HST). More precisely, the resulting tree metric dominates the initial metric and over-estimates it—in the deterministic case on average—by a factor of at most $\alpha(n) \in O(\log n \log\log n)$. Later, by introducing further techniques, this bound was decreased to only $O(\log n)$ ([FRT03]).

Contribution. We develop a way to profit from Bartal's techniques in the context of the Minimum Cycle Basis Problem. More precisely, based on the 3-HST that Bartal introduced, we construct for a weighted 2-vertex connected undirected graph a weakly fundamental cycle basis of length $O(W \cdot \log n \log\log n)$. Hereby, we improve the previously known best upper bound on the length of general minimum cycle bases by a factor of $\log n$. Also, since there are graphs of large girth which show that no general upper bound can drop below $O(W \cdot \log n)$, we conclude that our construction is almost tight.

This paper is organized as follows. In Section 2 we present the two main objects (Hierarchical Partition Metrics (HPM) and k-HSTs) that Bartal introduced when computing approximate tree metrics, together with their major properties. Based on 3-HSTs, in Section 3 we finally present the algorithm for computing a weakly fundamental cycle basis of length $O(W \cdot \log n \log\log n)$. In an appendix we consider it helpful to provide a detailed exposition of a procedure originally proposed in [Bar98] to derive a k-HST from a HPM.

2 Techniques and Main Idea

Following the presentation of our algorithm will be most convenient to the reader, when having in mind the following simple—though useful—property of weakly fundamental cycle bases.

Remark 1 *One way to construct a weakly fundamental cycle basis $\mathcal{B}$ is to define some spanning tree F and provide an order of the co-tree edges $E \setminus F$, say $e_1, \ldots, e_\nu$, where $F = e_{\nu+1}, \ldots, e_m$. Then, the basic circuit C that we associate with a co-tree edge $e_i \in E \setminus F$ has to contain e_i and may use some edges of the tree F and some of the co-tree edges $e_1, \ldots, e_{i-1}$. But C must not contain any of the edges $e_{i+1}, \ldots, e_\nu$.*

The key to our algorithm for constructing a short weakly fundamental cycle basis will thus be the order in which we process the edges of G. And this order will be dictated by a decomposition of G, the k-HST. These objects were introduced by Bartal ([Bar98]). Since their construction is based on a HPM, here we not only present k-HST and its properties, but also HPM.

Definition 2 *A* hierarchical partition metric *(HPM) of a weighted connected graph (G, w) is a length function d_{HPM} defined on the edges of G according to the following recursive process: Let $C \subseteq E$ be a cut in G. For every $e \in C$, we require that $d_{HPM}(e) \geq diam(G)$, and d_{HPM} forms a HPM over each connected component of $G \setminus C$ recursively.*

Theorem 3 here below is just a specialized formulation of one of Bartal's main results, as best suited for our purpose.

Theorem 3 (specialized formulation of Thm 13 in [Bar98]) *There exists a function $\alpha(n) \in \Theta(n \log n \log\log n)$ such that every weighted connected graph (G, w) admits an HPM d_{HPM} dominating d_G and such that*

$$\sum_{\{u,v\} \in E} d_{HPM}(u, v) \leq \alpha(n) \cdot W. \tag{3}$$

The metric d_{HPM} can be constructed in polynomial time.

To end with a more gentle presentation of our algorithm, we do not work directly on HPMs. Rather, as a kind of intermediate step, we derive from an HPM a so-called k-hierarchically well-separated tree (k-HST).

Where U is a subset of V, we denote by $G[U]$ the subgraph of G induced by U. When tracing down the recursive process in Definition 2, one produces smaller and smaller connected induced subgraphs of G until she gets down to single nodes in V. These connected subgraphs are called *clusters* and their node sets form a nested family $\mathcal{S}$ having V as its maximal element and all the singletons in V as minimal elements.

We resort on the standard way of representing such a laminar family $\mathcal{S}$ by a rooted tree T: the nodes of T are the sets in $\mathcal{S}$ and there is an arc (U, U') between two sets U and U' in $\mathcal{S}$ iff $U' \subseteq S$ and $U' \subseteq U'' \subseteq S$ holds for no U'' in $\mathcal{S}$. Notice that V is the root of T and the leaves of T are in 1,1-correspondence with the nodes in V. More in general, each node of T is the disjoint union of its children. Such a tree is called a V*-tree*. In a V-tree for a pair of vertices $u \neq v$ we consider their *least common ancestor* $\Lambda(u, v) \in V(T)$, being the vertex on the unique uv-path in T that is closest to the root of T. Its associated cluster in G is just $G[\Lambda(u, v)]$.

Definition 4 *A* k-hierarchically well-separated tree (k-HST) *of a weighted connected graph* (G, w) *is a* V*-tree* T *with weights* c *on the arcs, with the following properties:*

(1) $c(U, U_1) = c(U, U_2)$ *for any two arcs of* T *with a common tail;*
(2) $c(U', U'') \leq \frac{1}{k} c(U, U')$ *for any two subsequent arcs* (U, U') *and* (U', U'') *in* T*;*
(3) for each node U *of* T*, the induced graph* $G[U]$ *is connected (what we call a cluster).*

A k-HST (T, c) induces a metric d_{HST} over V, where $d_{HST}(u, v)$ is the distance between u and v in (T, c), i.e. $d_{HST}(u, v) := c(P_{u,v})$, where $P_{u,v}$ is the unique u, v-path in T. We only consider k-HSTs where d_{HST} dominates $d_{G,w}$. Clearly, for such k-HSTs, the metric d_{HST} is an HPM. We are interested in k-HSTs of low stretch, that is, we need the condition in Theorem 3 to hold for d_{HST}. In [Bar98], Bartal also indicated how to obtain such a k-HST from an HPM as in Theorem 3 by loosing only a factor of 4 to be placed in front of α.

Theorem 5 (stated in [Bar98] for weak diameters) *Given a weighted graph* (G, w) *and a HPM* d_{HPM} *over* (G, w)*, there exists a polynomial-time algorithm to construct a* k*-HST* T *with metric* d_{HST} *that dominates the strong internal diameters of the subgraphs of* (G, w) *that are induced by* T*, and which exceeds* d_{HPM} *by a factor of at most* $\frac{k^2}{k-1}$*, i.e.*

$$d_{G[\Lambda(u,v)]}(u, v) \leq d_{HST}(u, v) \leq \frac{k^2}{k-1} \cdot d_{HPM}(u, v), \ \forall u, v \in V(G). \tag{4}$$

Although the main idea for proving Theorem 5 has already been sketched in [Bar98], we consider it useful to present in detail the construction of a k-HST in an appendix.

In the algorithm that we are about to present in the next section we will process the clusters of T bottom-up, i.e. we will only start working in a cluster when we finished working in any of its descendents.

3 Computing a Short Weakly Fundamental Cycle Basis

Now we present our Algorithm 1 that for a given weighted graph (G, w) of total weight W constructs a weakly fundamental cycle basis of objective value at most

$$(16 \cdot \alpha(n) + 1) \cdot W, \quad \alpha \in O(\log n \log \log n), \tag{5}$$

where α refers to Bartal's constant in Theorem 3.

The first line of Algorithm 1 is a call to a procedure MAKE-3-HST. We assume procedure MAKE-3-HST performs the following tasks: (1) compute a hierarchical partition metric d_{HPM} holding the properties as stated in Theorem 3; and (2) derive from the metric d_{HPM} a 3-hierarchically well-separated tree T and the corresponding metric d_{HST} holding the properties as in (4). It was shown in [Bar98] how both (1) and (2) could be effectively and efficiently performed. As for (2), this can be done using the algorithm in the appendix, just a rewriting of the procedure first described in [Bar98].

Algorithm 1 resorts on a further external procedure: we assume that where (G, w) is a weighted graph and u is a node of G, then DIJKSTRA$((G, w), u)$ computes the u-rooted shortest path tree $S = e_1, e_2, \ldots, e_{n-1}$ in (G, w); it is assumed that the edges of S are given in the same order as they would be put into S by the classical Dijkstra's algorithm ([Dij59]).

After the 3-hierarchically well-separated tree T and the corresponding metric d_{HST} have been obtained, the actual iterative construction of the circuits to be put in the basis $\mathcal{B}$ can start. As anticipated in Remark 1, we will accompany the construction of $\mathcal{B}$ by also computing a spanning tree F of G plus an ordering of the non-tree edges $E(G) \setminus F$ of G. Thus, in our iterative process we start with $\mathcal{B} := \emptyset$ and $F := \emptyset$, and collect in the set Z all the edges that were already processed according to the ordering that we are about to define. Observe that Z will contain tree edges *and* non-tree edges. When we process an edge e (and thus add e to Z), this has one of the two possible results: Either $F \cup \{e\}$ is cycle-free, then we add e to F. Or $F \cup \{e\}$ contains some cycle, then we add to $\mathcal{B}$ a circuit through e which only uses edges in Z, and which is sufficiently short. For simplicity, you may think of a shortest circuit through e that only uses edges in Z.

We now give a description of the order in which we are going to process the edges of G. Processing the edges will be grouped according to the clusters of T. We will color a node $U \in V(T)$ *green*, if all the edges of its corresponding cluster $G[U]$ have been processed, i.e. $E(G[U]) \subseteq Z$. Otherwise, U will have to remain colored *red*. At the start of the algorithm we may thus color in green all the leaves of T, and the algorithm terminates when the root of T finally becomes green. The order by which we pass through the clusters of the k-HST will be bottom-up, i.e. we may only work in a node $U \in V(T)$, if all its descendents are colored green. Hence, when we start working in $G[U]$, some edges may have already been processed while working in clusters that correspond to descendents of U in T. The other edges, i.e. $E(G[U]) \setminus Z$, are called *U-proper*. We process all the U-proper edges in a specific order which we specify in the very next paragraph. After doing so, we may color U green.

The order in which we process the U-proper edges of a cluster $G[U]$ is as follows. We first compute Dijkstra's shortest path tree $S \subseteq E(G[U])$ in this cluster, rooted at some arbitrary vertex u of U. During this procedure, whenever a U-proper edge e gets added to S, we process this edge. Second, after all the edges of S have been processed, or $S \subseteq Z$, we process—in arbitrary order—all the U-proper edges in $E(G[U]) \setminus S$.

In practice, we suggest to compute the basic circuits which we add in Step 24 of Algorithm 1 as the shortest circuits through e in $Z \cup \{e\}$. In general, this would lead to shorter bases. However, our asymptotic analysis would not improve.

For the analysis, we define for a cluster $G[U]$

$$\Delta(U) := \mathrm{diam}_{\mathrm{HST}}(G[U]) := \max\{d_{\mathrm{HST}}(u, v),\ u, v \in U\}.$$

Since, according to (4), d_{HST} is dominating the strong internal diameter of the cluster, we find that

$$\Delta(U) \geq \mathrm{diam}(G[U]). \tag{6}$$

Moreover, for any two vertices $v_1, v_2 \in V(G)$ for which $\Lambda(v_1, v_2) = U$, we may bound $\Delta(U)$ as follows,

$$\Delta(U) \leq \frac{k}{k-1} \cdot d_{\mathrm{HST}}(v_1, v_2). \tag{7}$$

Lemma 6 *Let U be the cluster that is currently processed by Algorithm 1. Each circuit C that we add to the basis $\mathcal{B}$ in Step 18 as the shortest circuit through $e_i \in \{v_1, v_2\} \in S$ in Z has weight $w(C)$ at most*

$$w(C) \leq \frac{7}{2} \cdot d_{HST}(v_1, v_2). \tag{8}$$

Proof. Here, $e_i \notin Z$, and thus $\Lambda(v_1, v_2) = U$. Hence we will first identify a bound on $w(C)$ in terms of $\Delta(U)$, and finally apply (7) for $k = 3$. The bound on $w(C)$ is obtained by identifying a (possibly different) cycle C' in Z and with $e_i \in C'$, and establishing $w(C') \leq \frac{5}{2} \cdot \Delta(U)$. The cycle C' will be constructed out of two subpaths of S plus a shortest path within a subcluster of U.

We assume w.l.o.g. that v_1 is closer to the root u of the shortest-path tree S in $G[U]$. Denote by U_2 the child of U in T that contains v_2.

Algorithm 1 `wfcb`

Require: A weighted input graph (G, w).

Ensure: A weakly fundamental cycle basis $\mathcal{B}$ of G with $w(\mathcal{B}) = O(W \cdot \log n \log\log n)$.

$(T, d_{\mathrm{HST}}) :=$ MAKE-2-HST(G, w); // uses `hpm2hst`, see appendix
$\mathcal{B} := \emptyset$;
$F := \emptyset$; // becomes a spanning tree of G related to $\mathcal{B}$ as in Remark 1
$Z := \emptyset$; // the set of edges already considered. In particular, $F \subseteq Z$ and each edge in $Z \setminus F$ belongs to some circuit in $\mathcal{B}$.
Color with *green* all the leaves of T, and with *red* all other nodes;
while some node of T is *red* **do**
  Let U be any *red* node of T whose children are all *green*;
  Color U with *green*;
  Let u be an arbitrary vertex of $G[U]$;
  $S = \{e_1, \ldots, e_s\} :=$ DIJKSTRA$((G[U], w), u)$; // Recall that $G[U]$ is connected
  for $i = 1$ to s **do**
    if $e_i \in Z$ **then**
      // void — e_i already processed in a subcluster of U.
    else if $F \cup \{e_i\}$ is cycle-free **then**
      $F := F \cup \{e_i\}, \quad Z := Z \cup \{e_i\}$;
    else
      // $e_i \notin Z$, and $F \cup \{e_i\}$ contains some cycle
      Add to $\mathcal{B}$ the shortest circuit through the edge e_i that only uses edges in Z;
      $Z := Z \cup \{e_i\}$;
    end if
  end for
  // By now we have $S \subseteq Z$.
  for all $e \in E(G[U]) \setminus Z$ **do**
    Add to $\mathcal{B}$ the unique circuit in $S \cup \{e\}$
    $Z := Z \cup \{e\}$;
  end for
  // Here we have $E(G[U]) \subseteq Z$.
end while

We are in the situation where $F \cup \{e_i\}$ contains some circuit Q with $e_i \in Q$. Consider the cut $X \subset E(G[U])$ that corresponds to $(U_2, U \setminus U_2)$. Observe that $e_i \in X$. As every cycle has even intersection with any cut, and since $\emptyset \neq \{e_i\} \in Q \cap X$, we know that $|Q \cap X| \geq 2$, and in particular $|F \cap X| \geq 1$.[1]

But since edges in X have precisely one endpoint in U_2, they cannot be contained in any subcluster of U and thus are processed during U's iteration of

[1] One could even show by induction that in fact $|F \cap X| = 1$.

the **while**-loop. Hence, each element of $F \cap X$ is part of the shortest-path tree S in U.

We compose the not necessarily simple cycle C' out of the following three parts: First, the unique path $P_{ue_i} \subset S$ from u to e_i. Second, one path $P_{uf} \subset S$ from u to some edge f such that $P_{uf} \cap X = \{f\} \neq \{e_i\}$, see Figure 1 for the location of e, f, u, and U_2. As both these paths are subpaths of the shortest-path tree S

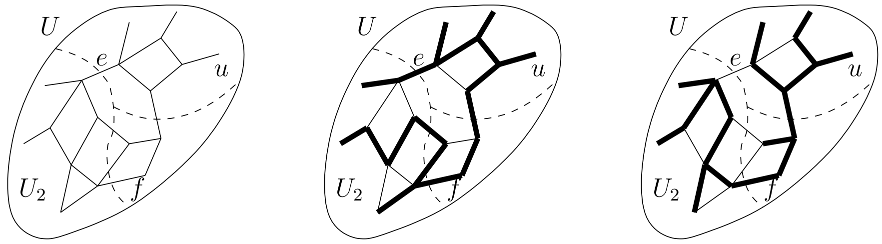

Fig. 1. Comparing for some cluster U (on the left) the u-rooted shortest-path tree S (in the middle) to the growing spanning tree F of the input graph (on the right)

in $G[U]$, for $P \in \{P_{ue_i}, P_{uf}\}$ we know that $w(P) \leq \mathrm{diam}(G[U]) \overset{(6)}{\leq} \Delta(U)$.

The third and last part of the cycle C' is selected as a shortest path P_{e_if} in $G[U_2]$ between the corresponding endpoints of e_i and f. Such a path exists because any cluster is connected, and P_{e_if} only uses edges in Z because it stays within the cluster U_2 all of whose edges were already processed, cf. the comment in Line 27 of Algorithm 1.

To bound the length of P_{e_if}, recall the second property of a k-HST and consider the subgraph $G[U_2]$ of G: Our bound $\Delta(U_2)$ on the strong internal diameter of $G[U_2]$ is by (at least) a factor of k smaller than the corresponding bound $\Delta(U)$ on the strong internal diameter of $G[U]$. As we chose T to be a 3-HST, we finally obtain $w(P_{e_if}) \leq \mathrm{diam}(G[U_2]) \leq \Delta(U_2) \leq \frac{\Delta(U)}{3}$. In total, this reads

$$w(C) \leq w(C') \leq w(P_{ue_i}) + w(P_{uf}) + w(P_{e_if}) \leq 2 \cdot \Delta(U) + \frac{\Delta(U)}{3} = \frac{7}{3} \cdot \Delta(U).$$

□

Lemma 7 *Let U be the cluster that is currently processed by Algorithm 1. Each circuit C that we add to the basis $\mathcal{B}$ in Step 24 for some non-tree edge $e = \{v_1, v_2\} \in E(G[U]) \setminus S$ has weight $w(C)$ at most*

$$w(C) \leq 3 \cdot d_{HST}(v_1, v_2) + w(e). \tag{9}$$

Proof. We identify a cycle C' that respects the length-bound in (9), but which is already a superset of the circuit C that we add to the basis in Step 24. Again, from $e = \{v_1, v_2\} \notin Z$ we conclude that $U = \Lambda(v_1, v_2)$. Let P_1 be the unique uv_1-path in S (thus being a shortest path), and P_2 be the unique uv_2-path in S. By defining $C' := P_1 \cup P_2 \cup \{e\}$, we obtain

$$\begin{aligned} w(C) \leq w(C') \leq w(P_1) + w(P_2) + w(e) &\overset{(6)}{\leq} 2 \cdot \Delta(U) + w(e) \\ &\overset{(7)}{\leq} 3 \cdot d_{\mathrm{HST}}(v_1, v_2) + w(e). \end{aligned}$$

□

Theorem 8 *Every weighted graph (G, w) with total edge weight W admits a weakly fundamental cycle basis of length at most $W \cdot O(\log n \log \log n)$. Such a basis can be computed in polynomial time.*

Proof. We apply Algorithm 1 to construct a weakly fundamental cycle basis $\mathcal{B}$. The length $w(B)$ of $\mathcal{B}$ is the sum of the weights of all the basic circuits which we add in Steps 18 and 24 of the algorithm. For every non-tree edge $e = \{v_1, v_2\} \in E(G) \setminus F$ according to Lemma 6, Lemma 7, and Equation (4), the length $w(C)$ of the circuit C that is added while processing e is bounded by

$$w(C) \leq w(e) + \frac{7}{2} \cdot d_{\mathrm{HST}}(v_1, v_2) \overset{(4)}{\leq} w(e) + \frac{63}{4} \cdot d_{\mathrm{HPM}}(v_1, v_2) \leq w(e) + 16 \cdot d_{\mathrm{HPM}}(v_1, v_2).$$

By summing this bound over all edges—although only non-tree edges with respect to F actually count—we bound $w(B)$ from above with

$$\sum_{\{u,v\} \in E} w(e) + 16 \cdot d_{\mathrm{HPM}}(u, v) \overset{\text{Thm.3}}{\leq} (16 \cdot \alpha(n) + 1) W, \ \alpha \in O(\log n \log \log n), \quad (10)$$

where $\alpha(n)$ refers to Bartal's constant in Theorem 3. □

The fact that we are omitting the explicit constants in our statement of Theorem 8 reflects that our focus is on the asymptotic improvement of the result by Elkin et al. ([EEST05]), rather than fine-tuning these constants.

4 Conclusions

After upper bounds of $O(n^2)$ (for unweighted graphs) and $O(W \cdot \log^2 n \log \log n)$ on the weights of Minimum Cycle Bases (MCB), we provide a polynomial-time algorithm that computes a weakly fundamental cycle basis of weight $O(W \cdot \log n \log \log n)$. Given the fact that there exist classes of graphs of large girth whose MCB has length $\Omega(W \cdot \log n)$, we consider our general bound fairly close to being asymptotically tight.

References

[AKPW95] Noga Alon, Richard M. Karp, David Peleg, and Douglas B. West. A graph-theoretic game and its application to the k-server problem. *SIAM J. Comput.*, 24(1):78–100, 1995.

[Bar98] Yair Bartal. On approximating arbitrary metrices by tree metrics. In *STOC*, pages 161–168, 1998.

[Bol78] Béla Bollobás. *Extremal Graph Theory.* Academic Press, 1978.

[Dij59] E.W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959.

[DKP82] Narsingh Deo, M.S. Krishnomoorthy, and G.M. Prabhu. Algorithms for generating fundamental cycles in a graph. *ACM Transactions on Mathematical Software*, 8(1):26–42, 1982.

[EEST05] Michael Elkin, Yuval Emek, Daniel A. Spielman, and Shang-Hua Teng. Lower-stretch spanning trees. In Harold N. Gabow and Ronald Fagin, editors, *STOC*, pages 494–503. ACM, 2005.

[FRT03] Jittat Fakcharoenphol, Satish Rao, and Kunal Talwar. A tight bound on approximating arbitrary metrics by tree metrics. In *STOC*, pages 448–455. ACM, 2003.

[Hor87] Joseph Douglas Horton. A polynomial-time algorithm to find the shortest cycle basis of a graph. *SIAM Journal on Computing*, 16(2):358–366, 1987.

[KRS01] Goran Konjevod, R. Ravi, and F. Sibel Salman. On approximating planar metrics by tree metrics. *Inf. Process. Lett.*, 80(4):213–219, 2001.

[Lie03] Christian Liebchen. Finding short integral cycle bases for cyclic timetabling. In Giuseppe Di Battista and Uri Zwick, editors, *ESA*, volume 2832 of *Lecture Notes in Computer Science*, pages 715–726. Springer, 2003.

Appendix

In this appendix we present in detail a construction that Bartal ([Bar98]) sketched to derive a k-HST T from a given HPM, preserving its average length-bound up to a constant factor. A detailed analysis of this construction establishes the following theorem.

Theorem 4 (stated in [Bar98] for weak diameters) *Let (G, w) be a weighted graph and d_{HPM} a HPM over (G, w). There exists a polynomial-time algorithm to construct a k-HST T with metric d_{HST} that dominates the strong internal diameters of the subgraphs of (G, w) that are induced by T, and which exceeds d_{HPM} by a factor of at most $\frac{k^2}{k-1}$, i.e.*

$$d_{G[\Lambda(u,v)]}(u, v) \overset{Lem.9}{\leq} d_{HST}(u, v) \overset{Lem.10}{\leq} \frac{k^2}{k-1} \cdot d_{HPM}(u, v),\ \forall u, v \in V(G). \quad (4)$$

To derive a k-HST T from an HPM, consider the V-tree $T(\mathrm{HPM})$ that corresponds immediately to the input-HPM. The key operation is to contract certain arcs of $T(\mathrm{HPM})$ whenever the diameter of a cluster in $T(\mathrm{HPM})$ would not comply with a certain threshold-value. These threshold-values are designed in particular to ensure Property 2 in Definition 4, and thus essentially depend on the depth of the corresponding node in $T(\mathrm{HPM})$. In the end, these threshold-values are used to define the lengths of the arcs in T, and thus the metric d_{HST} for G.

To each cluster $G[U]$ for which we copy a node from $T(\mathrm{HPM})$ into the k-HST T, we associate a threshold value $c(U) \in \{\frac{\mathrm{diam}(G)}{k^j} :\ j = 0, 1, 2, \dots\}$ and we ensure that

$$\frac{c(U)}{k} < \mathrm{diam}(G[U]) \leq c(U), \quad \forall U \in V(T). \quad (11)$$

We set the cost $c(U, U')$ of an arc (U, U') as $\frac{c(U)}{2}$. In $T(\mathrm{HPM})$ it could appear that the diameter of a child U' of U might not decrease sufficiently, in order to comply with Property 2 in Definition 4, i.e. $\mathrm{diam}(U') \geq \frac{c(U)}{k}$. In such a situation we continue deriving T from $T(\mathrm{HPM})$ as follows: do *not* introduce any node for the cluster $G[U']$ in T, simply contract the arc (U, U'), and hereby consider the children of U' in $T(\mathrm{HPM})$ as candidates to become direct children of U in T.

In most detail, we derive a k-HST T from an HPM by invoking the following initial call for the recursive Algorithm 2 `hpm2hst`

$$\texttt{hpm2hst}(T(\mathrm{HPM}), V(G), \mathrm{diam}(G)),$$

where $T(\text{HPM})$ is the V-tree that corresponds to the input-HPM.

Algorithm 2 `hpm2hst`

Require: A V-tree $T(\text{HPM})$ that encodes an HPM over (G, w);
a non-empty subset U of the vertices of $V(G)$;
a threshold-value $c(U)$ for which $\frac{c(U)}{k} < \text{diam}(G[U]) \leq c(U)$

Ensure: A k-HST d_{HST} defined on a new rooted V-tree T

1: Add a new artificial node for U to T
2: **if** $|U| > 1$ **then**
3: Let $\mathcal{P}$ be the children of U in $T(\text{HPM})$
4: **while** $\mathcal{P} \neq \emptyset$ **do**
5: Let $U' \in \mathcal{P}$ and remove U' from $\mathcal{P}$
6: **if** $\text{diam}(G[U']) \leq \frac{c(U)}{k}$ **then**
7: $i := 1$
8: **if** $\text{diam}(G[U']) \leq \frac{c(U)}{k^2}$ **then**
9: // strong internal diameter of U' too small − adjust i to meet precondition
10: Let i be such that $\frac{c(U)}{k^{i+1}} < \text{diam}(G[U']) \leq \frac{c(U)}{k^i}$
11: **end if**
12: $T' :=$ `hpm2hst`$(T(\text{HPM}), U', \frac{c(U)}{k^i})$ // i ensures precondition for U'
13: Link T' with U by adding an arc (U, U') to T with length $c(U, U') := \frac{1}{2}c(U)$
14: **else**
15: // strong internal diameter of U' too large − contract $T(\text{HPM})$-arc (U, U')
16: Add the children of U' in $T(\text{HPM})$ to $\mathcal{P}$
17: **end if**
18: **end while**
19: **end if**
20: **return** T

Recall that the leaves of the output tree T of `hpm2hst` are in $1,1$-correspondence with the vertices of G. Hence, for any two vertices $u, v \in V(G)$ we define the metric $d_{\text{HST}}(u, v)$ over G as the sum of the costs of the arcs on the unique undirected path P in T which connects their counterparts in T. The fact that T is indeed a k-HST can be seen fairly easily: Property 3 in Definition 4 is inherited from HPM, and Properties 1 and 2 are ensured by Lines 13 and 12 in Algorithm 2 `hpm2hst`, respectively.

Lemma 9 *For all $u, v \in V(G)$, $d_{HST}(u, v) \geq d_{G[\Lambda(u,v)]}(u, v)$.*

Proof. The unique undirected path between u and v in T contains precisely two arcs of cost $\frac{c(\Lambda(u,v))}{2}$. Together with the definition of $c(\Lambda(u, v))$ this yields

$$
\begin{aligned}
d_{\mathrm{HST}}(u,v) \;&\geq\; 2 \cdot \frac{c(\Lambda(u,v))}{2} \\
&=\; c(\Lambda(u,v)) \\
&\overset{(11)}{\geq}\; \mathrm{diam}(G[\Lambda(u,v)]) \\
&\overset{(2)}{\geq}\; d_{G[\Lambda(u,v)]}(u,v).
\end{aligned}
$$

□

Lemma 10 *For all $u, v \in V(G)$, $d_{HST}(u,v) \leq \frac{k^2}{k-1} \cdot d_{HPM}(u,v)$.*

Proof. On the one hand, the maximum cost of an arc on the unique undirected path between u and v in T is $\frac{c(\Lambda(u,v))}{2}$. By Property 2 in Definition 4 for the subpath P_u (joining $\Lambda(u,v)$ with u) we conclude that

$$
c(P_u) \leq \frac{c(\Lambda(u,v))}{2} \cdot \sum_{i=0}^{\infty} \left(\frac{1}{k}\right)^i \leq \frac{c(\Lambda(u,v))}{2} \cdot \frac{k}{k-1}, \quad \text{or}
$$

$$
d_{\mathrm{HST}}(u,v) \leq c(\Lambda(u,v)) \cdot \frac{k}{k-1}. \tag{12}
$$

On the other hand, by the definition of $\Lambda(u,v)$, T, and HPM, we know that

$$
\frac{1}{k} \cdot c(\Lambda(u,v)) \overset{(11)}{\leq} \mathrm{diam}(G[\Lambda(u,v)]) \leq d_{\mathrm{HPM}}(u,v),
$$

which concludes the proof, when combined with (12). □

Composing (4) and (7) indicates that it becomes interesting to us to minimize the function $f(x) = \frac{x^3}{(x-1)^2}$ over $(1, \infty)$. Here, its unique minimum is attained at $x_{\min} = 3$. Hence the metric d_{HST} that we define through our k-HST preserves the length bounds of the input metric d_{HPM} rather well, when choosing $k = 3$.

Reports from the group

"Combinatorial Optimization and Graph Algorithms"

of the Department of Mathematics, TU Berlin

2006/32 *Romeo Rizzi and Christian Liebchen:* A New Bound on the Length of Minimum Cycle Bases

2006/24 *Christian Liebchen and Sebastian Stiller:* Delay Resistant Timetabling

2006/08 *Nicole Megow and Tjark Vredeveld:* Approximation Results for Preemptive Stochastic Online Scheduling

2006/07 *Ekkehard Köhler and Christian Liebchen and Romeo Rizzi and Gregor Wünsch:* Reducing the Optimality Gap of Strictly Fundamental Cycle Bases in Planar Grids

2006/05 *Georg Baier and Thomas Erlebach and Alexander Hall and Ekkehard Köhler and Heiko Schilling:* Length-Bounded Cuts and Flows

2005/30 *Ronald Koch and Martin Skutella and Ines Spenke :* Maximum k-Splittable Flows

2005/29 *Ronald Koch and Ines Spenke :* Complexity and Approximability of k-Splittable Flows

2005/28 *Stefan Heinz and Sven O. Krumke and Nicole Megow and Jörg Rambau and Andreas Tuchscherer and Tjark Vredeveld:* The Online Target Date Assignment Problem

2005/18 *Christian Liebchen and Romeo Rizzi:* Classes of Cycle Bases

2005/11 *Rolf H. Möhring and Heiko Schilling and Birk Schütz and Dorothea Wagner and Thomas Willhalm:* Partitioning Graphs to Speed Up Dijkstra's Algorithm.

2005/07 *Gabriele Di Stefano and Stefan Krause and Marco E. Lübbecke and Uwe T.Zimmermann:* On Minimum Monotone and Unimodal Partitions of Permutations

2005/06 *Christian Liebchen:* A Cut-based Heuristic to Produce Almost Feasible Periodic Railway Timetables

2005/03 *Nicole Megow, Marc Uetz, and Tjark Vredeveld:* Models and Algorithms for Stochastic Online Scheduling

2004/37 *Laura Heinrich-Litan and Marco E. Lübbecke:* Rectangle Covers Revisited Computationally

2004/35 *Alex Hall and Heiko Schilling:* Flows over Time: Towards a more Realistic and Computationally Tractable Model

2004/31 *Christian Liebchen and Romeo Rizzi:* A Greedy Approach to Compute a Minimum Cycle Bases of a Directed Graph

2004/27 *Ekkehard Köhler and Rolf H. Möhring and Gregor Wünsch:* Minimizing Total Delay in Fixed-Time Controlled Traffic Networks

2004/26 *Rolf H. Möhring and Ekkehard Köhler and Ewgenij Gawrilow and Björn Stenzel:* Conflict-free Real-time AGV Routing

2004/21 *Christian Liebchen and Mark Proksch and Frank H. Wagner:* Performance of Algorithms for Periodic Timetable Optimization

2004/20 *Christian Liebchen and Rolf H. Möhring:* The Modeling Power of the Periodic Event Scheduling Problem: Railway Timetables — and Beyond

2004/19 *Ronald Koch and Ines Spenke:* Complexity and Approximability of k-splittable flow problems

2004/18 *Nicole Megow, Marc Uetz, and Tjark Vredeveld:* Stochastic Online Scheduling on Parallel Machines

2004/09 *Marco E. Lübbecke and Uwe T. Zimmermann:* Shunting Minimal Rail Car Allocation

2004/08 *Marco E. Lübbecke and Jacques Desrosiers:* Selected Topics in Column Generation

2003/050 *Berit Johannes:* On the Complexity of Scheduling Unit-Time Jobs with OR-Precedence Constraints

2003/49 *Christian Liebchen and Rolf H. Möhring:* Information on MIPLIB's timetab-instances

2003/48 *Jacques Desrosiers and Marco E. Lübbecke:* A Primer in Column Generation

2003/47 *Thomas Erlebach, Vanessa Kääb, and Rolf H. Möhring:* Scheduling AND/OR-Networks on Identical Parallel Machines

2003/43 *Michael R. Bussieck, Thomas Lindner, and Marco E. Lübbecke:* A Fast Algorithm for Near Cost Optimal Line Plans

2003/42 *Marco E. Lübbecke:* Dual Variable Based Fathoming in Dynamic Programs for Column Generation

2003/37 *Sándor P. Fekete, Marco E. Lübbecke, and Henk Meijer:* Minimizing the Stabbing Number of Matchings, Trees, and Triangulations

2003/25 *Daniel Villeneuve, Jacques Desrosiers, Marco E. Lübbecke, and François Soumis:* On Compact Formulations for Integer Programs Solved by Column Generation

2003/24 *Alex Hall, Katharina Langkau, and Martin Skutella:* An FPTAS for Quickest Multicommodity Flows with Inflow-Dependent Transit Times

2003/23 *Sven O. Krumke, Nicole Megow, and Tjark Vredeveld:* How to Whack Moles

2003/22 *Nicole Megow and Andreas S. Schulz:* Scheduling to Minimize Average Completion Time Revisited: Deterministic On-Line Algorithms

2003/16 *Christian Liebchen:* Symmetry for Periodic Railway Timetables

2003/12 *Christian Liebchen:* Finding Short Integral Cycle Bases for Cyclic Timetabling

762/2002 *Ekkehard Köhler and Katharina Langkau and Martin Skutella:* Time-Expanded Graphs for Flow-Dependent Transit Times

761/2002 *Christian Liebchen and Leon Peeters:* On Cyclic Timetabling and Cycles in Graphs

752/2002 *Ekkehard Köhler and Rolf H. Möhring and Martin Skutella:* Traffic Networks and Flows Over Time

739/2002 *Georg Baier and Ekkehard Köhler and Martin Skutella:* On the k-splittable Flow Problem

736/2002 *Christian Liebchen and Rolf H. Möhring:* A Case Study in Periodic Timetabling

Reports may be requested from:

Sekretariat MA 6–1
Fakultt II – Institut fr Mathematik
TU Berlin
Straße des 17. Juni 136
D-10623 Berlin – Germany

e-mail: klink@math.TU-Berlin.DE

Reports are also available in various formats from

`http://www.math.tu-berlin.de/coga/publications/techreports/`

and via anonymous ftp as

`ftp://ftp.math.tu-berlin.de/pub/Preprints/combi/Report-`*number*`-`*year*`.ps`