Minimum Uniform Gap Labelings of Trees

Tomás Feder[*] Carlos Subi[†]

February 21, 2010

Abstract

Given a graph, we define a base set to be a set of integers of size equal to the number of vertices in the graph. Given a graph and a base set, a labeling of the graph from the base set is an assignment of distinct integers from the base set to the vertices of the graph. The gap of an edge in a labeled graph is the absolute value of the difference between the labels of its endpoints. The gap of a labeled graph is the sum of the gaps of its edges.

The maximum gap graph labeling problem takes as input a graph and a base set and maximizes the gap of the graph over all possible labelings from the base set. This problem was studied in a companion paper [2], in particular in the uniform case where the base set is restricted to consecutive integers. In this note we study the related minimum uniform tree labeling problem, where the base set is uniform, the input graph is a tree, and we minimize the gap of the graph aver all possible labelings from the uniform base set. We prove that if the tree is a caterpillar, has at most one vertex of degree at least three, or has $O(\log n/\log\log n)$ leaves, then the problem can be solved exactly in polynomial time, while if the tree has $t \geq 1$ vertices of degree 3 or more, then the problem can be approximated within a factor of $3 + 2\log_2 t$.

1 Introduction

Given a graph G vertex set $V(G) = \{v_0, v_1, \ldots, v_n\}$ and edge set $E(G)$ with m edges, we define a *base set* to be a set $B = \{b_0, b_1, \ldots, b_n\}$ of $n + 1$ distinct integers, that is, with as many integers as vertices in G. The base set is said to be *uniform* if $b_i = i$, that is, if $B = \{0, 1, \ldots, n\}$.

Given a graph G and a base set B, a *labeling* L of the graph from the base set is a one-to-one assignment of integers $\ell(v)$ from the base set B to the vertices v of the graph G. The *gap of an edge* vw in a labeled graph is the absolute difference $|\ell(v) - \ell(w)|$. The *gap of a labeled graph* G is the sum $\sum_{vw\in E(G)} |\ell(v) - \ell(w)|$ of the gaps of its edges.

The *minimum graph labeling problem* takes as input a graph G and a base set B and produces a labeling of G with minimum gap. We consider the case where the graph G is a tree and the labeling is uniform, and prove that if the tree is a caterpillar, has at most

[*]268 Waverley St., Palo Alto, CA 94301. E-mail: `tomas@theory.stanford.edu`.

[†]E-mail: `carlos_subi@hotmail.com`.

one vertex of degree at least three, or has $O(\log n/\log\log n)$ leaves, then the problem can be solved exactly in polynomial time, while if the tree has $t \geq 1$ vertices of degree 3 or more, then the problem can be approximated within a factor of $3 + 2\log_2 t$.

The problem is relate to the corresponding maximum gap problem studied in the related paper [2], where connections to the graceful tree conjecture were shown [1].

2 Results

Lemma 1 *For any optimal solution of the minimum gap tree labeling problem for a tree T, the values 0 and n are assigned to leaves, and once these two leaves are identified the problem reduces to removing the edges of the path joining them and solving the resulting subtree components.*

PROOF: Let P be a path joining leaves that contains the values 0 and n. This path costs at least n, and may cost exactly n only if 0 and n are the starting and ending leaves. If we remove the edges of P, the cost of the remaining forest is at least the sum of the optimums for each subtree component, and this optimum can be achieved only by fining optima for each such subtree and juxtaposing them with increasing values as we go on the path from 0 to n at the leaves. ■

Theorem 1 *If the tree T is a path, a star, or more generally a caterpillar or a tree with at most one vertex of degree at least three, then the minimum uniform gap tree labeling problem is trivially solved exactly; if the tree T has ℓ leaves and t vertices of degree three or more (with $t \leq \ell - 1$), then there is an algorithm that solves the problem exactly in time polynomial in $\min(t^\ell, \ell^{t^2})n$, thus polynomial in n for $\ell = O(\log n/\log\log n)$, or for t constant, or for $t = O(\sqrt{\log n/\log\log n})$ and $\ell = \log^{O(1)} n$.*

PROOF: If T is a path, we assing consecutive integers along the path, so all gaps are 1. This follows from the Lemma by assigning 0 and n to the end vertices of the path and proceeding in order as suggested.

If T is a star, we assign the middle value of the range of integers for an odd number of vertices to the center of the star, or as close as possible middle value of the range of integers for an even number vertices to the center of the star. This follows from the Lemma as we assign 0 and n to leaves, remove their incident edges and proceed inductively.

More generally, if T consists of a vertex v of degree k to which are attached paths of lengths $r_1 \leq r_2 \leq \cdots \leq r_k$, then place v, path of length r_1 to the left of v, path of length r_2 to the right of v, and so on alternatingly away from v. This follows from the Lemma as we must proceed to assign 0 and n to distinct leaves, and the longer the path from 0 to n the smaller the resulting subtree after removal of this path will be, where smaller is meant in the sense of containment.

Also more generally than for paths and stars, if T is a caterpillar, then one can easily show that 0 and n must be assigned to leaves attached at the respective ends of the main path of the caterpillar, as any other choice for leaves 0 and n according to the Lemma would give subtrees that can be further separated while decreasing the cost to obtain a solution for 0 and n chosen as described above at the extremes. After removing the path joining the extremes 0 and n, we are left with disjoint stars which are solved as described above.

Finally, if there are ℓ leaves and t vertices of degree at least three (with $t \leq \ell-2$), we have at most t^2 choices for 0 and n in the Lemma, as we do not need to specify the actual leaf, just the vertex of degree three or more where a leaf path, the longest such path from a leaf will be chosen. Removing the edges of the path joining the leaf will reduce the number $t+\ell$ by two. The number of such vertices is at most $2(\ell-1)$, and is thus reduced to at most $2(\ell-2)$. Proceeding inductively on $2(\ell-1)$ gives a bound of $t^{2\ell}$ on the number of cases, thus a running time polynomial in $t^{\ell}n$. This bound is thus polynomial in n if $\ell = O(\log n/\log\log n)$.

Similarly, we may consider the first chosen path that traverses at least one path joining two of the t vertices of degree three or more, in t^2 ways and decide for each of the t vertices how many leaf path pairs for one such vertex will be removed first, in ℓ^t ways. These prior removals are done in order of decreasing total lenght of paths. There are thus $t^2\ell^t$ cases, and the number $t-1$ of path joining two vertices of degree three or more is decreased by one each time, for a total of $(t^2\ell^t)^{t-1}$ cases, and complexity polynomial in $\ell^{t^2}n$. This bound is thus polynomial in n if t is constant, or if $t = O(\sqrt{\log n/\log\log n})$, $\ell = O(\log^{O(1)} n$. ∎

Theorem 2 *In general, if the tree T has $t \geq 1$ vertices of degree 3 or more, then the problem can be approximated within a factor of* $3+2\log_2 t$.

Proof: Suppose $t \geq 1$. There exist a vertex v of degree 3 or more such that each subtree attached to v has at most $t/2$ vertices of degree 3 or more. The vertex assigned farthest form v in an optimal solution has value 0 or n, is in a subtree T' attached at T, and the farthest vertex is at distance at least $n/2$ from v. If we assign to v the value 0, and put the vertices in the optimal solution in consecutive values $n, n-1, \ldots$, we save because the values are closer, but we may pay n for the edge coming out of v instead of $n/2$ in the optimal solution, costing a factor of 2 for T'. We may now remove T' and proceed similarly for the remaining T''. In total, this will cost us a factor of 2 with respect to the optimal solution. We only save if in the end we choose the farthest T' from 0 to be the T' with the most vertices. Since each T' has by induction assumption at most $t/2$ vertices, it will cost us inductively a factor at most $1+2\log_2 t_{T'}$, plus a factor of 2, for putting T back together as shown above, for a total of $3+2\log_2 t$. ∎

More generally, if G has bounded treewidth and bounded degree, then a similar argument gives an $O(logn)$ approximation for the minimum uniform gap labeling problem.

References

[1] M. Edwards and L. Howard, A survey of graceful trees, Atlantic Electronic Journal of Mathematics, Volume 1, Number 1, Summer 2006, 5–30.

[2] T. Feder and C. Subi, Maximum gap labelings of graph, submitted.