Elegant distance constrained labelings of trees

Jiří Fiala[1]*, Petr A. Golovach[2], and Jan Kratochvíl[1]

[1] Institute for Theoretical Computer Science***
and Department of Applied Mathematics,
Charles University, Prague
{fiala,honza}@kam.mff.cuni.cz
[2] Department of Applied Mathematics,
Syktyvkar State University, Syktyvkar, Russia
golovach@syktsu.ru

Abstract. In our contribution to the study of graph labelings with three distance constraints we introduce a concept of elegant labelings: labelings where labels appearing in a neighborhood of a vertex can be completed into intervals such that these intervals are disjoint for adjacent vertices.
We justify introduction of this notion by showing that use of these labelings provides good estimates for the span of the label space, and also provide a polynomial time algorithm to find an optimal elegant labeling of a tree for distance constraints $(p, 1, 1)$. In addition several computational complexity issues are discussed.

1 Introduction

In the past decades graph theoretic models of telecommunication networks became natural and frequent subject both in theory and in practice. One of the possible applications considers an allocation of frequencies to transmitters, such that a possible interference is minimized. The notion of distance constrained labeling reflects the fact that interference decreases with increasing distance between transmitters, hence close frequencies should be used only on distant transmitters.

* Participation at the conference WG 2004 has been in part generously supported by the Deutsche Forschungsgemeinschaft (DFG) and proRWTH! (Freunde und Förderer der RWTH Aachen).
*** Supported by the Ministry of Education of the Czech Republic as project LN00A056.

For given integral parameters $p_1 \geq \cdots \geq p_k$ called *distance constraints*, an
$L(p_1, p_2, \ldots, p_k)$-*labeling* of a graph G assigns integers to vertices of G such that any pair of vertices that are at distance at most $i \leq k$ get labels that differ by at least p_i. The *span* of a labeling is the difference between the lowest and the highest labels used in the labeling. The graph invariant $\lambda_{(p_1,\ldots,p_k)}(G)$ is the minimum span among all $L(p_1, p_2, \ldots, p_k)$-labelings of G.

Clearly, $L(1)$-labelings are graph colorings, $L(1, \ldots, 1)$-labelings are colorings of the k-th distance power of the underlying graph G. A considerable attention was paid to the first "non-chromatic" collection of distance constraints, namely $(p_1, p_2) = (2, 1)$, suggested by Roberts and formally introduced by Griggs and Yeh in 1992 [1]. A variety of results appeared, among others we shall mention a nontrivial dynamic-programming algorithm for computing $\lambda_{(2,1)}(T)$ for trees by Chang and Kuo [2] and a long lasting conjecture stating that for any graph G, it holds that $\lambda_{(2,1)}(G) \leq \Delta(G)^2$, where $\Delta(G)$ stands for the maximum degree of a vertex in G.

From the computational complexity point of view it is also interesting that for an arbitrary constant c, the problem of testing whether $\lambda_{(2,1)}(G) \leq c$ is solvable in linear time when restricted to graphs of bounded treewidth, while the computational complexity of determining $\lambda_{(2,1)}(G)$ for the same class of graphs remains open.

Other collections of distance constraints were also considered by several authors. Labelings of meshes were considered in [3, 4] while $L(p_1, 1, \ldots)$-labelings of trees and interval graphs were studied in [5]. Further hardness results on $L(2, 1, \ldots, 1)$-labelings of restricted classes of graphs can be found in [6].

The computational complexity of finding $\lambda_{(p_1,p_2)}(T)$ is not fully resolved yet even for trees. For example, this problem becomes tractable when p_2 divides p_1, but the precoloring extension and the list-coloring versions of this problem are both NP-complete otherwise [7]. On the other hand, as follows from works on graph properties expressible in Monadic Second Order Logic [8, 9], if the span of a possible labeling is bounded by constant c the test whether $\lambda_{(p_1,\ldots,p_k)}(G) \leq c$ can be performed in linear time for a graph of bounded treewidth (an explicite algorithm is presented in [10]).

Distance constrained labelings can be generalized in several ways — one of the possible directions is the use of different metrics on the label space. Such labelings with constraints $(2, 1)$ were considered in [11] as special graph homomorphisms that are required to be locally injective. In our study we follow this concept and prove several of our results also for the cyclic metric on the label space.

In this paper we show that with an additional requirement on the labeling — that label space of the neighborhood of each vertex can be completed into an interval such that these intervals are disjoint for adjacent vertices — we can obtain both good estimates on the graph invariants $\lambda_{(p_1,p_2,p_3)}(T)$ for trees, but moreover an optimal so called *elegant* $L(p,1,1)$-labeling of a tree can be computed in a polynomial time.

Besides the results on computational complexity we provide also a necessary and a sufficient conditions for a tree to allow an elegant $C(2,1,1)$-labeling of the minimal possible span. The main motivation of this study is our belief that further exploration of properties of elegant and non-elegant labelings of trees might bring a new insight and new methods to finally resolve the computational complexity of the problem of determining $\lambda_{(p_1,\dots,p_k)}$ and in particular $\lambda_{(p_1,p_2)}$ on this class of graphs.

Our results on trees are finally accompanied with an NP-hardness proof of the $L(2,1,1)$-labeling problem on general graphs, which is presented in the appendix.

2 Preliminaries

All graphs considered in this paper are simple, i.e. without loops and multiple edges. For a vertex $u \in V_G$ the set of all neighbors of u in G is denoted by $N(u)$, the size of $N(u)$ is the *degree* $\deg(u)$ of the vertex u.

A connected graph without a cycle as a subgraph is called a *tree*, its vertices of degree one are called *leaves*, the other are *inner* vertices. A *star* is a graph isomorphic to the complete bipartite graph $K_{1,n}$, $n \geq 1$. The symbol $\omega(G)$ denotes the size of a maximum complete subgraph of G.

The graph distance $\operatorname{dist}(u,v)$ is the number of edges in a shortest path connecting vertices u and v. The k-th distance power G^k of a graph G is the graph on the same vertex set $V_{G^k} = V_G$ where edges of G^k connect distinct vertices that are at distance at most k in G, i.e. $E_{G^k} = \{(u,v) : 1 \leq \operatorname{dist}_G(u,v) \leq k\}$.

For integers $0 \leq a \leq b \leq t$, we define *discrete intervals* $(\operatorname{mod} t+1)$ in the following way: $[a,b] = \{a, a+1, \dots, b\}$ and $[b,a] = \{b, b+1, \dots, t, 0, 1, \dots a\}$.

The term $[t]$-*labeling of* G stands for a mapping $V_G \to [0,t]$.

For our purposes we use both linear and cyclic metric spaces in the definition of distance constrained labelings.

Definition 1. *Let $p_1 \geq p_2 \geq \cdots \geq p_k \geq 1$ be a k-tuple of integral distance constraints. A $[t]$-labeling f of G is said to be an* $L(p_1, p_2, \ldots, p_k)$-labeling of span t *if $|f(u) - f(v)| \geq p_i$ whenever $1 \leq \mathrm{dist}(u, v) \leq i \leq k$.*

A $[t]$-labeling f is called a $C(p_1, p_2, \ldots, p_k)$-labeling of span t *if for any pair of distinct vertices u, v at distance at most $i \leq k$, it holds that $p_i \leq |f(u) - f(v)| \leq t + 1 - p_i$.*

For both kinds of labelings we introduce an additional property of elegance:

Definition 2. *A $[t]$-labeling f is called* elegant *if for every vertex u, there exists an interval $I_u \pmod{k+1}$, such that $f(N(u)) \subseteq I_u$ and for every edge $(u, v) \in E_G : I_u \cap I_v = \emptyset$.*

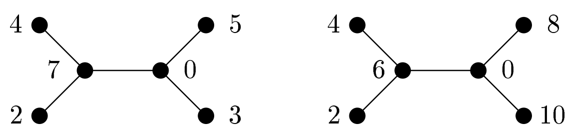

Fig. 1. An example of a tree T with $c_{(2,2,1)}(T) = 7 < 10 = c^*_{(2,2,1)}(T)$.

Observe that only triangle-free graphs may admit elegant labelings. On the other hand, it is not hard to deduce that every tree allows an elegant labeling for an arbitrary collection of distance constraints. An example of a $C(2, 2, 1)$-labeling and of an elegant $C(2, 2, 1)$-labeling of a tree T is depicted in Fig. 1.

The minimum t for which a graph G allows an $L(p_1, p_2, \ldots, p_k)$-labeling, and $C(p_1, p_2, \ldots, p_k)$-labeling resp., of span t is denoted by $\lambda_{(p_1,\ldots,p_k)}(G)$ and $c_{(p_1,\ldots,p_k)}(G)$, resp. The corresponding parameters for elegant labelings are indicated by asterisks (and are left to be $+\infty$ if no elegant labeling exists). Note that $\lambda_{(1)}(G) = c_{(1)}(G) = \chi(G) - 1$, where $\chi(G)$ denotes the chromatic number of G.

Observation 1 *For any distance constraints $(p_1, \ldots, p_k)$ and any graph G, it holds that*

$$p_k(\omega(G^k) - 1) \leq \lambda_{(p_1,\ldots,p_k)}(G) \leq p_1(\chi(G^k) - 1),$$

$$\lambda_{(p_1,\ldots,p_k)}(G) \leq c_{(p_1,\ldots,p_k)}(G) \leq c^*_{(p_1,\ldots,p_k)}(G),$$

$$\lambda_{(p_1,\ldots,p_k)}(G) \leq \lambda^*_{(p_1,\ldots,p_k)}(G) \leq c^*_{(p_1,\ldots,p_k)}(G).$$

Proof. The proof follows from the fact that every labeling with respect to the cyclic metric is also a valid labeling for linear metric, and that elegant labelings are also valid labelings. Moreover vertices of every complete subgraph of G^k should get labels pairwise at least p_k apart and a coloring of G^k can be transformed to an $L_{(p_1,\ldots,p_k)}$-labeling by using labels that form an arithmetic progression of difference p_1 as colors.

3 Tree labelings with 3 distance constraints

The concept of elegant labelings became useful in considering three distance constraints. The reason is, that in this case it is enough to maintain separation p_3 only between intervals associated to adjacent vertices instead of checking every pair of vertices at distance three.

Observe first that all $[t]$-colorings of a star $K_{1,n}$ (including labelings with at least one constraint) are elegant $(\mathrm{mod}\, t+1)$, since only two intervals play a role — the interval for the center $I_c = [f(c)+1, f(c)-1] (\mathrm{mod}\, t+1)$ and all other intervals can be chosen as $[f(c), f(c)]$.

3.1 An upper bound for elegant $C(p_1, p_2, p_3)$-labelings

We present an upper bound on distance constrained labelings of a tree with circular metric. It is well known that powers of trees are chordal graphs (see [12, 13]) and that $\chi(T^k) = \omega(T^k)$. Observe that in contrary to the general upper bound of Observation 1 for the parameter $\lambda_{(p_1,p_2,p_3)}(G)$, the coefficient by the main term $\omega(T^3)$ becomes p_2 instead of p_1 and hence it provides an essential improvement when $p_2 \ll p_1$ and $\omega(T^3)$ is sufficiently large.

Theorem 2. *For any $p_1 \geq p_2 \geq p_3 \geq 1$ and any tree T different from a star, it holds that $c^*_{(p_1,p_2,p_3)}(T) \leq p_2\omega(T^3) + p_1 + \max\{p_1 - p_2, p_3\} - 3$.*

Proof. By induction on the number s of inner vertices of T we construct an elegant labeling of T such that for each vertex u, $f(N(u))$ is an arithmetic progression of length $\deg(u)$ and difference p_2.

When $s = 2$, let u and u' be the two inner vertices of T of degrees $d, d' \geq 1$. We choose

$$t = \omega(T^3)p_2 + p_1 + \max\{p_1 - p_2, p_3\} - 2p_2 - 1$$

and define a $[t]$-labeling f of the tree T explicitly as $f(N(u)) = \{0, p_2, 2p_2, \ldots, (d-1)p_2\}$ where $f(u') = (d-1)p_2$ and the other labels are distributed on leaves of $N(u)$ arbitrarily. For $r = (d-1)p_2 + p_1$ we similarly lay out labels $\{r, r+p_2, r+2p_2, \ldots r+(d'-1)p_2\}$ on $N(u')$ such that $f(u) = r$.

To show that f is a valid $C(p_1, p_2, p_3)$-labeling we denote first by v, v' the two vertices of the minimum and the maximum label, i.e. $f(v) = 0$ and $f(v') = r + (d'-1)p_2 = \omega(T^3)p_2 + p_1 - 2p_2$.

Since $\operatorname{dist}(v, v') = 3$ we need $\omega(T^3)p_2 + p_1 - 2p_2 \leq t + 1 - p_3$, which is assured by the choice of t. For the adjacent vertices v and u we need $(d-1)p_2 + p_1 \leq t + 1 - p_1$, which holds as well, because $t \geq \omega(T^3)p_2 + p_1 - 2p_2 - 1 + p_1 - p_2 \geq \omega(T^3)p_2 + p_1 - 2p_2 - 1 + p_1 - (d'-1)p_2 = (d-1)p_2 + 2p_1 - 1$. The same inequality can be analogously derived for the labels of v' and u. Observe, that these conditions on u, u', v and v' imply, that the distance constraints are valid also for other pairs of vertices.

Now suppose that T has at least three inner vertices. Since inner vertices induce a subtree of T called the *inner tree of* T, it is possible to choose a pair (u, v) of adjacent inner vertices such that v is a leaf in the inner tree and the sum $\deg(u) + \deg(v)$ is minimized. We remove all vertices adjacent to v with exception of u and denote the resulting tree by T'. By the choice of (u, v) we have $\omega((T')^3) = \omega(T^3) \geq \deg(u) + \deg(v)$.

By the induction hypothesis the tree T' allows an elegant labeling f' of span $t = \omega(T^3)p_2 + p_1 + \max\{p_1 - p_2, p_3\} - 3$. Now assume that the arithmetic progression on $f'(N(u))$ is of form $a, a+p_2, \ldots, a + (\deg(u) - 1)p_2, \pmod{t+1}$. Then the vertices of $N(v)$ should avoid interval $I_1 = [a - p_3 + 1, a + (\deg(u) - 1)p_2 + p_3 - 1]$ due to the constraint on distance three as well as the interval $I_2 = [f'(v) - p_1 + 1, f'(v) + p_1 - 1]$.

Since $f'(v)$ is at distance at least $p_3 - 1$ from the boundary of I_1, and similarly at least $p_1 - 1$ points apart from the boundary of I_2 we get that $|I_1 \cap I_2| = p_3 + \max\{(\deg(u) - 1)p_2 + p_3, p_1\} - 1 \geq p_3 + \max\{p_2 + p_3, p_1\} - 1$.

Then $I = [0, t] \setminus (I_1 \cup I_2)$ is an interval of size

$$\begin{aligned} |I| &= t + 1 - |I_1| - |I_2| + |I_1 \cap I_2| \\ &\geq \deg(u)p_2 + \deg(v)p_2 + p_1 + \max\{p_1 - p_2, p_3\} - 3 - \\ &\quad - \deg(u)p_2 + p_2 - p_3 - 2p_1 + 2 + \max\{p_2 + p_3, p_1\} \\ &= \deg(v)p_2 + p_2 - 1 \end{aligned}$$

an hence can accommodate an arithmetic progression A of length $\deg(v)$ and difference p_2, which contains $f'(u)$ as one of its elements.

We extend the labeling f' into a labeling f of T by using elements of $A \setminus f'(u)$ as the labels of the leaf vertices adjacent to v in T. This concludes the proof.

For a particular choice of $(p_1, p_2, p_3) = (2, 1, 1)$, we have obtained an almost a tight bound:

Corollary 1. *Every tree T satisfies*

$$\omega(T^3) - 1 \leq \lambda_{(2,1,1)}(T) \leq \lambda^*_{(2,1,1)}(T) \leq \omega(T^3),$$

and for any tree T different from a star it holds

$$\omega(T^3) - 1 \leq c_{(2,1,1)}(T) \leq c^*_{(2,1,1)}(T) \leq \omega(T^3).$$

Proof. If T is a star then it can be easily seen that $\lambda_{2,1,1}(T) = \omega(T^3)$ and it was already mentioned that any of its labelings is elegant.

The bound $c^*_{2,1,1}(T) \leq \omega(T^3)$ when the tree T is different from a star follows from Theorem 2. All other inequalities and bounds were shown in Observation 1.

3.2 An algorithm to compute $c^*_{(p,1,1)}(T)$

The proof of Theorem 2 was constructive, hence it can be straightforwardly converted into a polynomial-time algorithm which finds a $C(p_1, p_2, p_3)$-labeling within the claimed upper bound.

For the special choice of distance constraints $p_2, p_3 = 1$ the computation of $\lambda^*_{(p,1,1)}(T)$ and $c^*_{(p,1,1)}(T)$ can be resolved in a polynomial time. We describe here an algorithm for deciding whether $c^*_{(p,1,1)} \leq k$. The algorithm for linear metric differs only in minor details. We use a dynamic programming approach, similarly as it was used in the algorithm for computation of $\lambda_{(2,1)}(T)$ (see [2, 7]).

Let T be a tree and k be a positive integer. Our algorithm tests the existence of an elegant $C(p, 1, 1)$-labelling of T of span k. We may assume that $k \leq n + 2p - 4$, where n is the number of vertices of T, since if $k > n + 2p - 4$, such a labeling always exists due to Theorem 2.

We first choose a leaf r as the root of T, which defines the parent-child relation between every pair of adjacent vertices. For any edge (u, v) such that u is a child of v, we denote by T_{uv} the subtree of T rooted in v and containing u and all descendants of u. For every such edge and for every pair of integers $i, j \in [0, k]$ and an interval I $(\mathrm{mod}(k+1))$ such that $j \in I$, we introduce a boolean function $\phi(u, v, i, j, I)$, which is evaluated true if and only if T_{uv} has an elegant $C(p, 1, 1)$-labelling f where $f(u) = i$, $f(v) = j$ and $I_u = I$. This function ϕ can be calculated as follows:

1. Set an initial value $\phi(u, v, i, j, I) =$ false for all edges (u, v), integers $i, j \in \{0, 1, \ldots, k\}$ and intervals I $(j \in I)$.
2. If u is a leaf adjacent to v then we set $\phi(u, v, i, j, I) =$ true for all integers $i, j \in [0, k] : p \leq |i - j| \leq k - p$ and intervals I such that $j \in I$ and $i \notin I$.
3. Let us suppose that ϕ is already calculated for all edges of T_{uv} except (u, v). Denote by $v_1, v_2, \ldots, v_m$ children of u. For all pairs of integers $i, j \in [0, k] : p \leq |i - j| \leq k - p$ and for all intervals $I : j \in I, i \notin I$ we consider the set system $\{M_1, M_2, \ldots, M_m\}$, where

   $$M_t = \{s : s \in I \backslash \{j\}, \exists \text{ interval } J : \phi(v_t, u, s, i, J) = \text{true}, i \in J, I \cap J = \emptyset\}$$

   We set $\phi(u, v, i, j, I) =$ true if the set system $\{M_1, M_2, \ldots, M_m\}$ allows a system of distinct representatives, i.e. if there exists an injective function $r : [1, m] \to [0, k]$ such that $r(t) \in M_t$ for all $t \in [1, m]$.

The correctness of calculation of the function ϕ follows by an easy inductive argument. The only nontrivial point is that in the constructed entry $f(v)$ differs from $f(x)$ for every child x of v_t, because $f(v) = j \in I$, and $f(x) \in J$, where $I \cap J = \emptyset$.

Now we evaluate the complexity of computation of this function. It is calculated for $n - 1$ edges. Since each interval I is defined by the pair of it's endpoints, the set of arguments has the cardinality $O(nk^4)$. Computation of ϕ for leafs (see step 2) demands $O(1)$ operation for each argument. The recursive step (see item 3) takes time $O(mk^3)$ for constructing the sets M_t and then $O((m + k)^2 mk)$ for the testing of the existence of the system of distinct representatives (we have m sets of cardinality of no more than k). Since $m \leq n$ and $k \leq n + 2p - 4$, this step demands $O(n^3 k)$ operations for a single collection of arguments. So the total time of computation of ϕ is equal to $O(n^4 k^5)$ and this function can be calculated for all sets of arguments polynomially.

To finish the description of the algorithm we have only to note that an elegant $C(p, 1, 1)$-labelling of span k exists if and only if there are integers $i, j \in [0, k]$ and a interval I $(j \in I)$, for which $\phi(r, w, i, j, I) =$ true where w is the only child of the root r.

It suffices to test at most $O(n)$ values of k, which provides the total $O(n^{10})$ time complexity. Observe that for linear metric the algorithm basically remains the same, with the exception that also pairs i, j such that $|i - j| > k - p$ are allowed in steps 2) and 3).

Thus we proved following theorem:

Theorem 3. *For any tree T, $\lambda^*_{(p,1,1)}(T)$ and $c^*_{(p,1,1)}(T)$ can be computed in a polynomial time.*

For the computation of $\lambda^*_{(2,1,1)}(T)$ (or $c^*_{(2,1,1)}(T)$) it is necessary to run this algorithm only once for $k = \omega(T^3) - 1$. If the algorithm returns positive answer, then $\lambda^*_{(2,1,1)}(T) = \omega(T^3) - 1$, else $\lambda^*_{(2,1,1)}(T) = \omega(T^3)$.

Finally note, that if we wanted to generalize the above algorithm to arbitrary distance constraints (p_1, p_2, p_3), it would require resolving of a system of *distant* representatives in the step 3), which is an NP-hard problem in general [7], and moreover it is exactly the same bottleneck of a possible polynomial algorithm for computing $\lambda_{(p_1,p_2)}$ on trees for a general pair of distance constraints $p_1 > p_2 > 1$ [7].

3.3 Perfect labelings

In order to illustrate the above notions, we notice that for any tree we are able to show that either $c_{(2,1,1)}(T) = \omega(T^3) - 1$ and find such a labeling, called *perfect*, or we find an elegant labeling of span $\omega(T^3)$, leaving the possibility that T may allow a perfect labeling but no such labeling can be elegant (we leave as an open question whether a tree with this property exists). It would certainly be interesting to characterize the trees that satisfy $c_{(2,1,1)}(T) = c^*_{(2,1,1)}(T) = \omega(T^3) - 1$.

We present a necessary condition that a tree must satisfy to allow a perfect elegant labeling. We first classify edges of the tree with respect to the fact whether their neighborhood induces a maximum clique in T^3 or not. Hence, an edge $(u, v) \in E_T$ will be called saturated if $\deg(u) + \deg(v) = \omega(T^3)$, and it will be called unsaturated otherwise.

Theorem 4. *If a tree allows a prefect elegant labeling, then every inner vertex is incident with at least two unsaturated edges.*

Proof. Assume for the contrary that an inner vertex v is incident with at most one unsaturated edge. For any neighbor u incident with v along a saturated edge it holds that $\deg(u) + \deg(v) = \omega(T^3)$, hence for any perfect elegant labeling follows $I_u = [0, \omega(T^3) - 1] \setminus I_v$.

Since $I_v = [a, b]$ is an interval of length $\deg(v)$, each element of I_v is used as a label of some $u \in N(v)$. As v is incident with at most one unsaturated edge, at least one of a or b is used as a label of a neighbor w connected to v via a saturated edge. But then the label of w is one unit away from I_w, a contradiction.

If we interpret this condition in the construction of Theorem 2, we get:

Corollary 2. *A tree allows a perfect elegant labeling if it can be rooted such that each inner vertex has at least two children connected to it by unsaturated edges.*

There exist trees with at least two unsaturated edges incident with each inner vertex, but which allow no labeling of span $\omega(T^3) - 1$ (neither elegant nor not elegant). An example of such a tree is depicted in Fig. 2

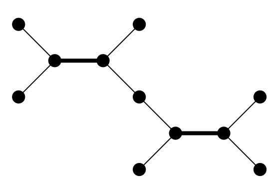

Fig. 2. A tree with $c_{(2,1,1)}(T) = \omega(T^3)$ (saturated edges indicated in bold).

4 Computational complexity of the $L(2,1,1)$-labeling problem

To complete the picture we shortly present a full computational complexity characterization of the decision problem whether $\lambda_{(2,1,1)} \leq k$ for general graphs.

Theorem 5. *The decision problem whether $\lambda_{(2,1,1)} \leq k$ is* NP*-complete for every $k \geq 5$ and it is solvable in polynomial time for all $k \leq 4$.*

Proof. We start with the second part of the theorem and prove that the labeling problem is tractable for $k \leq 4$. Only finitely many connected graphs allow a $\lambda_{(2,1,1)}$-labeling of span at most 3. So, without loss of generality we may consider only the case $k = 4$.

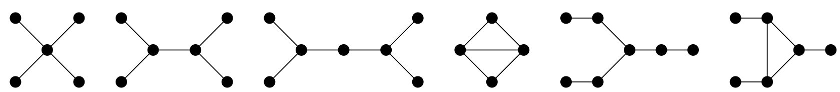

Fig. 3. Some graphs of $\lambda_{(2,1,1)}(G) > 4$.

It can be easily seen, that if G is a graph for which an $L(2,1,1)$-labelling of span 4 exists, then it can not contain as a subgraph any of the graphs depicted in Fig. 3. Clearly, the maximum degree of G is at most 3 and each connected component of G is formed by a path or by a cycle, where some vertices are equipped with an additional leaf, or two consecutive vertices may also be joined by a path of length 2. It is not difficult to observe that such graphs have treewidth bounded by 3, and hence the existence of an $L(2,1,1)$-labelling of span 4 can be tested in linear time by dynamic programming (e.g., [10]).

For $k \geq 5$, we reduce the Not-All-Equal p-Satisfiability (NAE p-SAT) problem. An instance of NAE p-SAT is a formula Φ in conjunctive normal form with p positive literals in each clause (no negations). It is well known [14] that for all $p \geq 3$, the decision problem whether such Φ allows a satisfying assignment where each clause contains also a negatively valued literal is NP-complete.

For each variable x_i we construct a gadget consisting of a chain of m_i copies of the graph depicted in Fig 4, where m_i is the number of occurrences of x_i and $p = \lceil \frac{k}{2} \rceil$, $r = \lceil \frac{p-1}{2} \rceil$. In the figure the symbol E_n stands for an independent set with n vertices, K_n for a complete graph, and M_n for a matching on n edges.

It can be explored by a case analysis that any $L(2,1,1)$ labeling of span k of the constructed variable gadget satisfies:

- All vertices u_i are labelled by the same label, either by 0 or by k.
- The vertices v_i are given labels either from the set $L = \{0, 2, 4, \ldots, k-4+(k \mod 2)\}$, when u_i's are labeled by k, or otherwise from the set $L' = \{k-l, l \in L\}$.

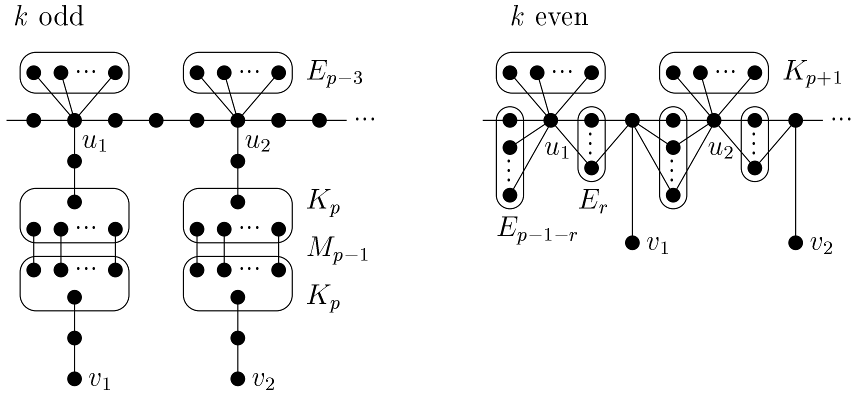

Fig. 4. Variable gadgets.

We finalize the construction of the graph G_Φ such that for each clause C of the formula Φ we insert an extra new vertex w_C and for each variable x which appears in the clause we link w_C with one of the vertices v of the vertex gadgets associated with x. (Each v-type vertex is adjacent to only one w_C).

The properties of the variable gadgets assure that G_Φ allows an $L(2,1,1)$-labeling of span k if and only if Φ has a required assignment. These labelings are related to assignments e.g. by letting $x = \mathsf{true}$ whenever the vertices u_i of the gadget for x are all labeled by k, and $x = \mathsf{false}$ if u_i get 0.

Clearly, as for any clause vertex w_C it holds $\deg(w_C) \geq |L| = |L'|$, these labelings indicate only valid assignments, i.e., at least one of the adjoining gadgets represents positively valued variable and at least one stands for a negatively valued one.

In the opposite direction, each assignment for Φ can be converted into an $L(2,1,1)$-labeling of G_Φ in a straightforward way.

References

1. Griggs, J.R., Yeh, R.K.: Labelling graphs with a condition at distance 2. SIAM Journal of Discrete Mathematics **5** (1992) 586–595
2. Chang, G.J., Kuo, D.: The $L(2,1)$-labeling problem on graphs. SIAM Journal of Discrete Mathematics **9** (1996) 309–316

3. van den Heuvel, J., Leese, R.A., Shepherd, M.A.: Graph labeling and radio channel assignment. Journal of Graph Theory **29** (1998) 263–283
4. Leese, R.A.: Radio spectrum: a raw material for the telecommunications industry. 10th Conference of the European Consortium for Mathematics in Industry, Goteborg (1998)
5. Bertossi, A.A., Pinotti, M.C., Rizzi, R.: Channel assignment on strongly-simplicial graphs. In: International Parallel and Distributed Processing Symposium, 17th IPDPS '03, Nice, IEEE Computer Society (2003) 222
6. Golovach, P.A.: Systems of pair of q-distant representatives and graph colorings. Zap. nau. sem. POMI **293** (2002) 5–25 in Russian.
7. Fiala, J., Kratochvíl, J., Proskurowski, A.: Distance constrained labelings of precolored trees. In: Theoretical Computer Science, 7th ICTCS '01, Torino. Number 2202 in Lecture Notes in Computer Science, Springer Verlag (2001) 285–292
8. Courcelle, B.: The monadic second-order logic of graphs. I: Recognizable sets of finite graphs. Inf. Comput. **85** (1990) 12–75
9. Arnborg, S., Lagergren, J., Seese, D.: Easy problems for tree-decomposable graphs. J. Algorithms **12** (1991) 308–340
10. Kohl, A., Schreyer, J., Tuza, Z., Voigt, M.: Choosability problems for (d, s)-colorings. in preparation (2003)
11. Fiala, J., Kratochvíl, J.: Partial covers of graphs. Discussiones Mathematicae Graph Theory **22** (2002) 89–99
12. Kearney, P.E., Corneil, D.G.: Tree powers. Journal of Algorithms **29** (1998) 111–131
13. Lin, Y.L., Skiena, S.S.: Algorithms for square roots of graphs. SIAM Journal on Discrete Mathematics **8** (1995) 99–118
14. Schaefer, T.J.: The complexity of the satisfability problem. In: Proceedings of the 10th Annual ACM Symposium on Theory of Computing. ACM (1978) 216–226