The Complexity of $L(p, q)$-Edge-Labelling

Gaétan Berthe[1], Barnaby Martin[2], Daniël Paulusma[2], and Siani Smith[2]

[1] ENS de Lyon, Lyon, France
[2] Department of Computer Science, Durham University, U.K.

Abstract. The $L(p, q)$-Edge-Labelling problem is the edge variant of the well-known $L(p, q)$-Labelling problem. It is equivalent to the $L(p, q)$-Labelling problem itself if we restrict the input of the latter problem to line graphs. So far, the complexity of $L(p, q)$-Edge-Labelling was only partially classified in the literature. We complete this study for all $p, q \geq 0$ by showing that whenever $(p, q) \neq (0, 0)$, the $L(p, q)$-Edge-Labelling problem is NP-complete. We do this by proving that for all $p, q \geq 0$ except $p = q = 0$, there is an integer k so that $L(p, q)$-Edge-k-Labelling is NP-complete.

1 Introduction

This paper studies a problem that falls under the distance-constrained labelling framework. Given any fixed nonnegative integer values p and q, an $L(p, q)$-k-*labelling* is an assignment of *labels* from $\{0, \ldots, k-1\}$ to the vertices of a graph such that adjacent vertices receive labels that differ by at least p, and vertices connected by a path of length 2 receive labels that differ by at least q [5]. Some authors instead define the latter condition as being vertices at distance 2 receive labels which differ by at least q (e.g. [7]). These definitions are the same so long as $p \geq q$ and much of the literature considers only this case (e.g. [11]). If $q > p$, the definitions diverge. For example, in an $L(1, 2)$-labelling, the vertices of a triangle K_3 need labels $\{0, 1, 2\}$ in the second definition but $\{0, 2, 4\}$ in the first. We use the *first* definition, in line with [5]. The decision problem of testing if for a given integer k, a given graph G admits an $L(p, q)$-k-labelling is known as $L(p, q)$-Labelling. If k is *fixed*, that is, not part of the input, we denote the problem as $L(p, q)$-k-Labelling.

The $L(p, q)$-labelling problem has been heavily studied, both from the combinatorial and computational complexity perspectives. For a starting point, we refer the reader to the comprehensive survey of Calamoneri [5].[3] The $L(1, 0)$-Labelling is the traditional Graph Colouring problem (COL), whereas $L(1, 1)$-Labelling is known as (Proper) Injective Colouring [3, 2, 9] and Distance 2 Colouring [13, 16]. The latter problem is studied explicitly in many papers (see [5]), just as is $L(2, 1)$-Labelling [8, 11, 12] (see also [5]). The $L(p, q)$-labelling problem is also studied for special graph classes, see in particular [6] for a complexity dichotomy for trees. Janczewski et al. [11] proved that if $p > q$, then $L(p, q)$-Labelling is NP-complete for planar bipartite graphs.

[3] See `http://wwwusers.di.uniroma1.it/~calamo/survey.html` for later results.

Regime	Reduction from	Place in article	k at least
$p = 0$ and $q > 0$	3-COL	Section 3	$3q$
$2 \leq q/p$	NAE-3-SAT	Appendix A	$(n-1)p + q + 1$
$1 < q/p \leq 2$	NAE-3-SAT	Section 4	$5p + 1$
$q/p = 1$	3-COL	[14]	$4p$
$2/3 < q/p \leq 1$	3-COL	Section 5	$3p + q + 1$
$q/p = 2/3$	1-in-3-SAT	Appendix B	$4p$
$1/2 < q/p < 2/3$	2-in-4-SAT	Appendix C	$p + 4q + 1$
$0 < q/p \leq 1/2$	NAE-3-SAT	Appendix D [12]	$3p + 1$
$p > 0$ and $q = 0$	3-COL	Section 2	$3p$

Table 1. Table of results. The fourth row follows from [14] (which proves the case $p = q = 1$) and applying Lemma 1. The eighth row is obtained from a straightforward generalization of the result in [12] for the case where $p = 2$ and $q = 1$. The fourth column gives the minimal k for which we prove NP-completeness. In the second row choose minimal $n \geq 4$ so that $(n-3)p \geq q$.

We consider the edge version of the problem. The *distance* between two edges e_1 and e_2 is the length of a shortest path that has e_1 as its first edge and e_2 as its last edge minus 1 (we say that e_1 and e_2 are *adjacent* if they share an end-vertex or equivalently, are of distance 1 from each other). The $L(p, q)$-Edge-Labelling problem considers an assignment of the labels to the edges instead of the vertices, and now the corresponding distance constraints are placed instead on the edges. Owing to space constraints some proofs and cases are omitted. Please see the full version of this article at [1]. In particular, references to the appendix are intended for that version.

In [12], the complexity of $L(2, 1)$-Edge-k-Labelling is classified. It is in P for $k < 6$ and is NP-complete for $k \geq 6$. In [14], the complexity of $L(1, 1)$-Edge-k-Labelling is classified. It is in P for $k < 4$ and is NP-complete for $k \geq 4$. In this paper we complete the classification of the complexity of $L(p, q)$-Edge-k-Labelling in the sense that, for all $p, q \geq 0$ except $p = q = 0$, we exhibit k so we can show $L(p, q)$-Edge-k-Labelling is NP-complete. That is, we do not exhibit the border for k where the problem transitions from P to NP-complete (indeed, we do not even prove the existence of such a border). The authors of [12] were looking for a more general result, similar to ours, but found the case $(p, q) = (2, 1)$ laborious enough to fill one paper [15]. In fact, their proof settles for us all cases where $p \geq 2q$. We now give our main result.

Theorem 1. *For all $p, q \geq 0$ except if $p = q = 0$, there exists an integer k so that $L(p, q)$-Edge-k-Labelling is NP-complete.*

The proof follows by case analysis as per Table 1, where the corresponding section for each of the subresults is specified. We are able to reduce to the case that $\gcd(p, q) = 1$, due to the forthcoming Lemma 1. We prove NP-hardness by reduction from graph 3-colouring and several satisfiability variants. These latter are known to be NP-hard from Schaefer's classification [17]. Each section begins with a theorem detailing the relevant NP-completeness. The case $p = q = 0$ is trivial (never use more than one colour) and is therefore omitted.

Our hardness proofs involve gadgets that have certain common features, for example, the vertex-variable gadgets are generally star-like. For one case, we have a computer-assisted proof (as we will explain in detail).

By Theorem 1 we obtain a complete classification of $L(p, q)$-Edge-Labelling.

Corollary 1. *For all $p, q \geq 0$ except $p = q = 0$, $L(p, q)$-Edge-Labelling is NP-complete.*

Note that $L(p, q)$-Edge-Labelling is equivalent to $L(p, q)$-Labelling for line graphs (the line graph of a graph G has vertex set $E(G)$ and two vertices e and f in it are adjacent if and only if e and f are adjacent edges in G). Hence, we obtain another dichotomy for $L(p, q)$-Labelling under input restrictions, besides the ones for trees [6] and if $p > q$, (planar) bipartite graphs [11].

Corollary 2. *For all $p, q \geq 0$ except $p = q = 0$, $L(p, q)$-Labelling is NP-complete for the class of line graphs.*

2 Preliminaries

We use the terms colouring and labelling interchangeably. A special role will be played by the *extended n-star* (especially for $n = 4$). This is a graph built from an n-star $K_{1,n}$ by subdividing each edge (so it becomes a path of length 2). Instead of referring to the problem as $L(p, q)$-Labelling (or $L(h, k)$-Labelling) we will use $L(a, b)$-Labelling to free these other letters for alternative uses.

The following lemma is folklore and applies equally to the vertex- or edge-labelling problem. Note that $\gcd(0, b) = b$.

Lemma 1. *Let $\gcd(a, b) = d > 1$. Then the identity is a polynomial time reduction from $L(a/d, b/d)$-(Edge)-k-Labelling to $L(a, b)$-(Edge)-kd-Labelling.*

This result and the known NP-completeness of Edge-3-Colouring [10] imply:

Corollary 3. *For all $a > 0$, $L(a, 0)$-Edge-$3a$-Labelling is NP-complete.*

3 Case $a = 0$ and $b > 0$

By Lemma 1 we only have to consider $a = 0$ and $b = 1$.

Theorem 2. *The problem $L(0, 1)$-Edge-3-Labelling is NP-complete.*

Let us use colours $\{0, 1, 2\}$. Our NP-hardness proof involves a reduction from 3-COL but we retain the nomenclature of variable gadget and clause gadget (instead of vertex gadget and edge gadget) in deference to the majority of our other sections. Our variable gadget consists of a triangle attached on one of its vertices to a leaf vertex of a star. Our clause gadget consists of a bull, each of whose pendant edges (vertices of degree 1) has an additional pendant edge added (that is, they are subdivided). This is equivalent to a triangle with a path of length 2 added to each of two of the three vertices. We draw our variable gadget in Figure 1 and our clause gadget in Figure 2.

Lemma 2. *In any valid $L(0,1)$-edge-3-labelling of the variable gadget, each of the pendant edges must be coloured the same.*

Proof. Each of the edges in the triangle must be coloured distinctly as there is a path of length two from each to any other (by this we mean with a single edge in between, though they are also adjacent). Suppose the triangle edge that has two nodes of degree 2 in the variable gadget is coloured i. It is this colour that must be used for all of the pendant edges. The remaining edge may be coloured by anything from $\{0,1,2\} \setminus \{i\}$. However, we will always choose the option $i-1 \bmod 3$. $\square$

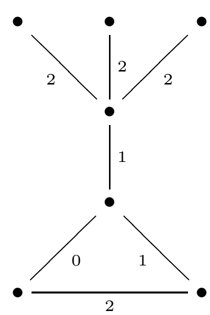

Fig. 1. The variable gadget for Theorem 2.

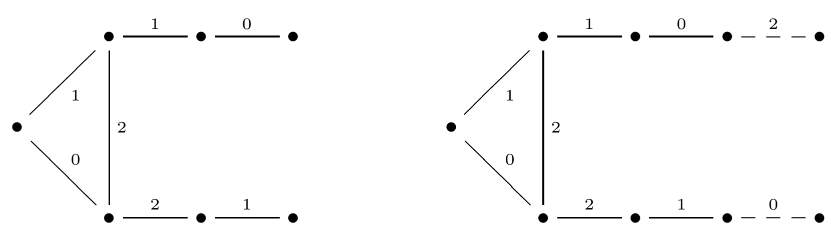

Fig. 2. The clause gadget for Theorem 2 (left) drawn also together with its interface with a variable gadget (right). The dashed line is an inner edge of the variable gadget.

Lemma 3. *In any valid $L(0,1)$-edge-3-labelling of the clause gadget, the two pendant edges must be coloured distinctly.*

Proof. Each of the edges in the triangle must be coloured distinctly as there is a path of length two from each to any other. Suppose the triangle edge that has two nodes of degree 3 in the clause gadget is coloured (w.l.o.g.) 2. The remaining edges in the triangle must be given 0 and 1, in some order. This then determines the colours of the remaining edges and enforces that the two pendant edges must be coloured distinctly. However, suppose we had started first by colouring distinctly the pendant edges. We could then choose a colouring of the remaining edges of the clause gadget so as to enforce the property that, if a pendant edge is coloured i, then its neighbour (in the clause gadget) is coloured $i+1 \bmod 3$. This is the colouring we will always choose. □

We are now ready to prove Theorem 2.

Proof (Proof of Theorem 2.). We reduce from 3-COL. Let G be an instance of 3-COL involving n vertices and m edges. Let us explain how to build an instance G' for $L(0,1)$-Edge-3-Labelling. Each particular vertex may only appear in at most m edges (its degree), so for each vertex we take a copy of the variable gadget which has m pendant edges. For each edge of G we use a clause gadget to unite an instance of these pendant edges from the corresponding two variable gadgets. We use each pendant edge from a variable gadget in at most one clause gadget. We identify the pendant edge of a variable gadget with a pendant edge from a clause gadget so as to form a path from one to the other. We claim that G is a yes-instance of 3-COL iff G' is a yes-instance of $L(0,1)$-Edge-3-Labelling.

(Forwards.) Take a proper 3-colouring of G and induce these colours on the pendant edges of the corresponding variable gadgets. Distinct colours on pendant edges can be consistently united in a clause gadget since we choose, for a pendant edge coloured i: $i-1 \bmod 3$ for its neighbour in the variable gadget, and $i+1 \bmod 3$ for its neighbour in the clause gadget.

(Backwards.) From a valid $L(0,1)$-edge-3-labelling of G', we infer a 3-colouring of G by reading the pendant edge labels from the variable gadget of the corresponding vertex. The consistent labelling of each vertex follows from Lemma 2 and the fact that it is proper follows from Lemma 3. □

4 Case $1 < \frac{b}{a} \leq 2$

In this section we prove the following result.

Theorem 3. *If $1 < \frac{b}{a} \leq 2$, the problem $L(a,b)$-Edge-$(5a+1)$-Labelling is* NP*-complete.*

We proceed by a reduction from (monotone) NAE-3-SAT. This case is relatively simple as the variable gadget is built from a series of extended 4-stars chained together, where each has a pendant 5-star to enforce some benign property. We will use colours from the set $\{0,\ldots,5a\}$.

Lemma 4. *Let $1 < \frac{b}{a} \leq 2$. In any valid $L(a,b)$-edge-$(5a+1)$-labelling of the extended 4-star, if one pendant edge is coloured 0 then all pendant edges are coloured in the interval $\{0,\ldots,a\}$; and if one pendant edge is coloured $5a$ then all pendant edge are coloured in the interval $\{4a,\ldots,5a\}$.*

Proof. Suppose some pendant edge is coloured by 0 and another pendant is coloured by $l' \notin \{0, \ldots, a\}$. There are four inner edges of the star that are at distance 1 or 2 from these, and one another. If $l' < 2a$, then at least $2a$ labels are ruled out, which does not leave enough possibilities for the inner edges to be labelled in (at best) $\{2a+1, \ldots, 5a\}$. If $l' \geq 2a$, then it is not possible to use labels for the inner edges that are all strictly above l'. It is also not possible to use labels for the inner edges that are all strictly below l. In both cases, at least $2a$ labels are ruled out. Thus the labels, read in ascending order, must start no lower than a and have a jump of $2a$ at some point. It follows they are one of: $a, 3a, 4a, 5a$; or $a, 2a, 4a, 5a$; or $a, 2a, 3a, 5a$. This implies that l' is itself a multiple of a (whichever one was omitted in the given sequence). But now, since $b > a$, there must be a violation of a distance 2 constraint from l'. □

We would like to chain extended 4-stars together to build our variable gadgets, where the pendant edges represent variables (and enter into clause gadgets) and we interpret one of the regimes $\{0, \ldots, a\}$ and $\{4a, \ldots, 5a\}$ as true, and the other as false. However, the extended 4-star can be validly $L(a, b)$-edge-$(5a+1)$-labelled in other ways that we did not yet consider. We can only use Lemma 4 if we can force one pendant edge in each extended 4-star to be either 0 or $5a$. Fortunately, this is straightforward: take a 5-star and add a new edge to one of the edges of the 5-star creating a path of length 2 from the centre of the star to the furthest leaf. This new edge can only be coloured 0 or $5a$. In Figure 3 we show how to chain together copies of the extended 4-star, together with pendant 5-star gadgets at the bottom, to produce many copies of exactly one of the regimes $\{0, \ldots, a\}$ and $\{4a, \ldots, 5a\}$. Note that the manner in which we attach the pendant 5-star only produces a valid $L(a, b)$-edge-$(5a+1)$-labelling because $2a \geq b$ (otherwise some distance 2 constraints would fail). So long as precisely one pendant edge per extended 4-star is used to encode a variable, then each encoding can realise all labels within each of these regimes, and again this can be seen by considering the pendant edges drawn top-most in Figure 3, which can all be coloured anywhere in $\{4a, \ldots, 5a\}$. Let us recap, a *variable gadget* (to be used for a variable that appears in an instance of NAE-3-SAT m times) is built from chaining together m extended 4-stars, each with a pendant 5-star, exactly as is depicted in Figure 3 for $m = 3$. The following is clear from our construction. The designation *top* is with reference to the drawing in Figure 3. In Figure 3, the case drawn corresponds to $\{4a, \ldots, 5a\}$, where the case $\{0, \ldots, a\}$ is symmetric.

Lemma 5. *Any valid $L(a, b)$-edge-$(5a+1)$-labelling of a variable gadget is such that the top pendant edges are all coloured from precisely one of the sets $\{0, \ldots, a\}$ and $\{4a, \ldots, 5a\}$. Moreover, any colouring of the top pendant edges from one of these sets is valid.*

The clause gadget will be nothing more than a 3-star (a claw) which is formed from a new vertex uniting three (top) pendant edges from their respective variable gadgets. The following is clear.

Lemma 6. *A clause gadget is in a valid $L(a, b)$-edge-$(5a+1)$-labelling in the case where two of its edges are coloured $0, a$ and the third $5a$; or two of its edges*

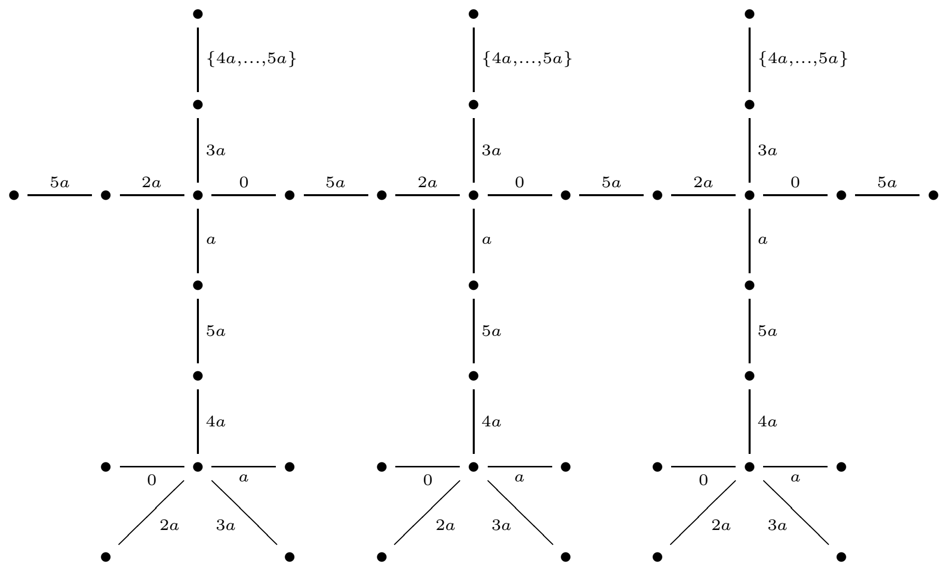

Fig. 3. Three extended 4-stars chained together, each with a pendant 5-star below, to form a variable gadget for Theorem 3. The pendant edges drawn on the top will be involved in clauses gadget and each of these three edges can be coloured with anything from $\{4a,\ldots,5a\}$. If the top pendant edge is coloured $5a$ it may be necessary that the inner star edge below it is coloured not $3a$ but $2a$ (cf. Figure 4). This is fine, the chaining construction works when swapping $2a$ and $3a$.

are coloured $4a, 5a$ *and the third* 0. *If all three edges come from only one of the regimes* $\{0,\ldots,a\}$ *and* $\{4a,\ldots,5a\}$, *it can not be in a valid* $L(a,b)$*-edge-*$(5a+1)$*-labelling.*

We are now ready to prove Theorem 3.

Proof (Proof of Theorem 3.). We reduce from (monotone) NAE-3-SAT. Let Φ be an instance of NAE-3-SAT involving n occurrences of (not necessarily distinct) variables and m clauses. Let us explain how to build an instance G for $L(a,b)$-Edge-$(5a+1)$-Labelling. Each particular variable may only appear at most n times, so for each variable we take a copy of the variable gadget which is n extended 4-stars, each with a pendant 5-star, chained together. Each particular instance of the variable belongs to one of the free (top) pendant edges of the variable gadget. For each clause of Φ we use a 3-star to unite an instance of these free (top) pendant edges from the corresponding variable gadgets. Thus, we add a single vertex for each clause, but no new edges (they already existed in the variable gadgets). We claim that Φ is a yes-instance of NAE-3-SAT if and only if G is a yes-instance of $L(a,b)$-Edge-$(5a+1)$-Labelling.

(Forwards.) Take a satisfying assignment for Φ. Let the range $\{0,\ldots,a\}$ represent true and the range $\{4a,\ldots,5a\}$ represent false. This gives a valid labelling of the inner edges in the extended 4-stars, as exemplified in Figure 3. In each clause, either there are two instances of true and one of false; or the converse.

Let us explain the case where the first two variable instances are true and the third is false (the general case can easily be garnered from this). Colour the (top) pendant edge associated with the first variable as 0, the second variable a and the third variable $5a$. Plainly these can be consistently united in a claw by the new vertex that appeared in the clause gadget. We draw the situation in Figure 4 to demonstrate that this will not introduce problems at distance 2. Thus, we can see this is a valid $L(a, b)$-edge-$(5a + 1)$-labelling of G.

(Backwards.) From a valid $L(a, b)$-edge-$(5a + 1)$-labelling of G, we infer an assignment Φ by reading, in the variable gadget, the range $\{0, \ldots, a\}$ as true and the range $\{4a, \ldots, 5a\}$ as false. The consistent valuation of each variable follows from Lemma 5 and the fact that it is in fact not-all-equal follows from Lemma 6. □

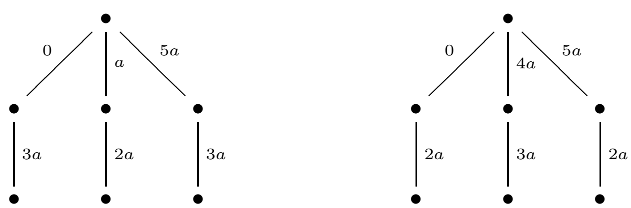

Fig. 4. The clause gadget and its interface with the variable gadgets (where we must consider distance 2 constraints). Both possible evaluations for not-all-equal are depicted.

5 Case $\frac{2}{3} < \frac{b}{a} < 1$

In this section we prove the following result.

Theorem 4. *If $\frac{2}{3} < \frac{b}{a} < 1$, then the problem $L(a, b)$-*Edge*-$(3a + b + 1)$-*Labelling *is* NP*-complete.*

The regimes of the following lemma are drawn in Figure 5.

Lemma 7. *Let $1 < \frac{a}{b} < \frac{3}{2}$. In an $L(a, b)$-edge-$(3a + b + 1)$-labelling c of the extended 4-star, there are three regimes for the pendant edges. The first is $\{b, \ldots, a\}$, the second is $\{2a + b, \ldots, 3a\}$, and the third is $\{a + b, \ldots, 2a\}$.*

Proof. In a valid $L(a, b)$-edge-$(3a + b + 1)$-labelling, we note $c_1 < c_2 < c_3 < c_4$ the colours of the 4 edges in the middle of the extended 4-star, and l_1, l_2, l_3, l_4 the colours of the pendant edges such that l_i is the colour of the pendant edge connected to the edge of colour c_i.

Claim 1. For all i, $c_1 < l_i < c_4$.

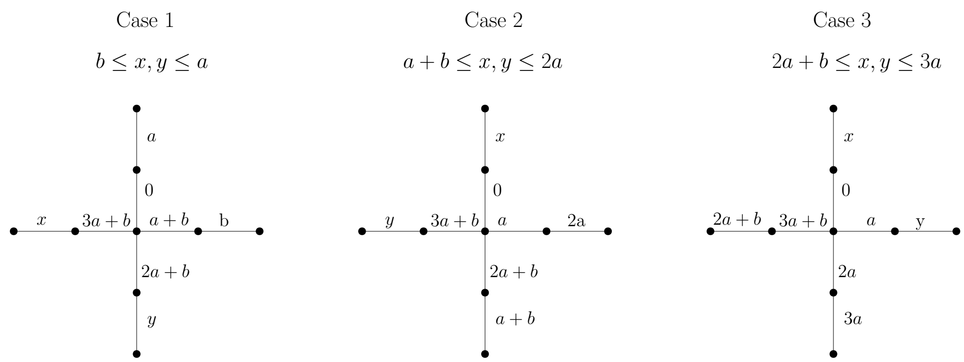

Fig. 5. The regimes of Theorem 4.

We only have to prove one inequality, as the other one is obtained by symmetry. If $l_i \leq c_1$ (bearing in mind also $b < a$), we have:

$$3a + b \geq c_4 - l_i = (c_1 - l_i) + (c_2 - c_1) + (c_3 - c_2) + (c_4 - c_3) \geq 3a + b.$$

So $(c_1, c_2, c_3, c_4) = (b, a+b, 2a+b, 3a+b)$, but $a > b$ so there is no possible value for l_1, which is not possible. So $c_1 < l_i$, and by symmetry $l_i < c_4$.

Claim 2. There exists $i \in \{1, 2, 3\}$ such that $c_{i+1} - c_i \geq a + b$.

We suppose the contrary. We have proved $c_1 < l_2, l_3 < c_4$. If $l_2 < c_2$, then $c_2 - c_1 = c_2 - l_2 + l_2 - c_1 \geq a + b$, impossible. If $c_2 < l_2 < c_3$, then $c_3 - c_2 = c_3 - l_2 + l_2 - c_2 \geq a + b$, impossible. So $c_3 < l_2 < c_4$. Symmetrically, we obtain $c_1 < l_3 < c_2$. So $c_1 < l_3 < c_2 < c_3 < l_2 < c_4$, and we get: $c_4 - c_1 \geq (l_3 - c_1) + (c_2 - l_3) + (c_3 - c_2) + (l_2 - c_3) + (c_4 - l_2) \geq 4b + a > 3a + b$, which is not possible.

Now we are in a position to derive the lemma, with the three regimes coming from the three possibilities of Claim 2. If $i = 1$, then the inner edges of the star are $0, a+b, 2a+b, 3a+b$ and the pendant edges come from $\{b, \ldots, a\}$. If $i = 2$, then the inner edges of the star are $0, a, 2a+b, 3a+b$ and the pendant edges come from $\{a+b, \ldots, 2a\}$. If $i = 3$, then the inner edges of the star are $0, a, a+b, 3a+b$ and the pendant edges come from $\{2a+b, \ldots, 3a\}$. $\square$

The *variable gadget* may be taken as a series of extended 4-stars chained together. In the following, the "top" pendant edges refer to one of the two free pendant edges in each extended 4-star (not involved in the chaining together). The following is a simple consequence of Lemma 7 and is depicted in Figure 6.

Lemma 8. *Any valid $L(a, b)$-edge-$(3a + b + 1)$-labelling of a variable gadget is such that the top pendant edges are all coloured from precisely one of the sets $\{b, \ldots, a\}$, $\{a + b, \ldots, 2a\}$ or $\{2a + b, \ldots, 3a\}$. Moreover, any colouring of the top pendant edges from one of these sets is valid.*

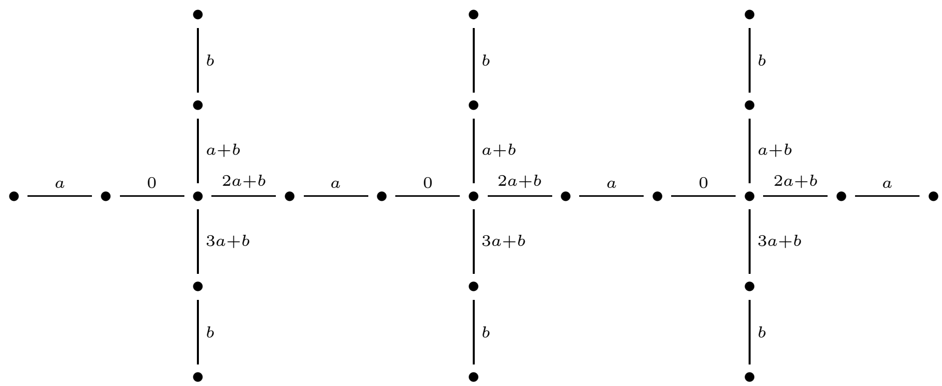

Fig. 6. Three extended 4-stars chained together, to form a variable gadget for Theorem 4. The pendant edges drawn on the top will be involved in clauses gadget. Suppose the top pendant edges are coloured b (as is drawn). In order to fulfill distance 2 constraints in the clause gadget, we may need the inner star vertices adjacent to them to be coloured not always $a+b$ (for example, if that pendant edge b is adjacent in a clause gadget to another edge coloured $a+b$). This is fine, the chaining construction works when swapping inner edges $a+b$ and $3a+b$ wherever necessary.

The clause gadget will be nothing more than a 2-star (a path) which is formed from a new vertex uniting two (top) pendant edges from their respective variable gadgets. The following is clear.

Lemma 9. *A clause gadget is in a valid $L(a,b)$-edge-$(3a+b+1)$-labelling in the case where its edges are coloured distinctly. If they are coloured the same, then it can not be in a valid $L(a,b)$-edge-$(3a+b+1)$-labelling.*

We are now ready to prove Theorem 4.

Proof (Proof of Theorem 4.). We reduce from 3-COL. Let G be an instance of 3-COL involving n vertices and m edges. Let us explain how to build an instance G' for $L(a,b)$-EDGE-$(3a+b+1)$-LABELLING. Each particular vertex may only appear in at most m edges (m is an upper ground on its degree), so for each vertex we take a copy of the variable gadget which is m extended 4-stars chained together. Each particular instance of the vertex belongs to one of the free (top) pendant edges of the variable gadget. For each edge of G we use a 2-star to unite an instance of these free (top) pendant edges from the corresponding two variable gadgets. Thus, we add a single vertex for each edge of G, but no new edges in G' (they already existed in the variable gadgets). We claim that G is a yes-instance of 3-COL if and only if G' is a yes-instance of $L(a,b)$-EDGE-$(3a+b+1)$-LABELLING.

(Forwards.) Take a proper 3-colouring of G and induce these pendant edge labels on the corresponding variable gadgets according to the three regimes of

Lemma 7. For example, map colours 1, 2, 3 to $b, a+b, 2a+b$. Plainly distinct pendant edge labels can be consistently united in a 2-claw by the new vertex that appeared in the clause gadget. Thus, we can see this is a valid $L(a,b)$-edge-$(3a+b+1)$-labelling of G'.

(Backwards.) From a valid $L(a,b)$-edge-$(3a+b+1)$-labelling of G', we infer a 3-colouring of G by reading the pendant edge labels from the variable gadget of the corresponding vertex and mapping these to their corresponding regime. The consistent valuation of each variable follows from Lemma 8 and the fact that it is proper (not-all-equal) follows from Lemma 9. □

6 Final Remarks

We give several directions for future work. First, determining the boundary for k between P and NP-complete, in $L(p,q)$-Edge-k-Labelling, for all p,q is still open except if $(p,q)=(1,1)$ and $(p,q)=(2,1)$. For $(p,q)=(1,1)$ it is known to be 4 (it is in P for $k<4$ and is NP-complete for $k \geq 4$) [14]; and for $(p,q)=(2,1)$ it is known to be 6 (it is in P for $k<6$ and is NP-complete for $k \geq 6$) [12].

A second open line of research concerns $L(p,q)$-Labelling for classes of graphs that omit a single graph H as an induced subgraph (such graphs are called *H-free*). A rich line of work in this vein includes [3], where it is noted, for $k \geq 4$, that $L(1,1)$-k-Labelling is in P over H-free graphs, when H is a linear forest; for all other H the problem remains NP-complete. If k is part of the input and $p=q=1$, the only remaining case is $H = P_1 + P_4$ [2]. Corollary 2 covers, for every $(p,q) \neq (0,0)$, the case where H contains an induced claw (as every line graph is claw-free). For bipartite graphs, and thus for H-free graphs for all H with an odd cycle, the result for $L(p,q)$-k-Labelling is known from [11], at least in the case $p>q$.

As our final open problem, for $d \geq 1$, the complexity of $L(p,q)$-Labelling on graphs of diameter at most d has, so far, only been determined for $a,b \in \{1,2\}$ [4].

References

1. Berthe, G., Martin, B., Paulusma, D., Smith, S.: The complexity of l(p, q)-edge-labelling. CoRR **abs/2008.12226** (2020), https://arxiv.org/abs/2008.12226
2. Bok, J., Jedlicková, N., Martin, B., Paulusma, D., Smith, S.: Injective colouring for H-free graphs. Proc. CSR 2021, LNCS **12730**, 18–30 (2021)
3. Bok, J., Jedličková, N., Martin, B., Paulusma, D., Smith, S.: Acyclic, star and injective colouring: A complexity picture for H-free graphs. Proc. ESA 2020, LIPIcs **173**, 22:1–22:22 (2020)
4. Brause, C., Golovach, P.A., Martin, B., Paulusma, D., Smith, S.: Acyclic, star, and injective colouring: Bounding the diameter. Proc. WG 2021, LNCS **12911**, 336–348 (2021)
5. Calamoneri, T.: The $L(h,k)$-labelling problem: An updated survey and annotated bibliography. Computer Journal **54**, 1344–1371 (2011)
6. Fiala, J., Golovach, P.A., Kratochvíl, J.: Computational complexity of the distance constrained labeling problem for trees (extended abstract). Proc. ICALP 2008, LNCS **5125**, 294–305 (2008)

7. Fiala, J., Kloks, T., Kratochvíl, J.: Fixed-parameter complexity of lambda-labelings. Discrete Applied Mathematics **113**, 59–72 (2001)
8. Griggs, J.R., Yeh, R.K.: Labelling graphs with a condition at distance 2. SIAM Journal on Discrete Mathematics **5**, 586–595 (1992)
9. Hahn, G., Kratochvíl, J., Širáň, J., Sotteau, D.: On the injective chromatic number of graphs. Discrete Mathematics **256**, 179–192 (2002)
10. Holyer, I.: The NP-completeness of edge-coloring. SIAM Journal on Computing **10**, 718–720 (1981)
11. Janczewski, R., Kosowski, A., Małafiejski, M.: The complexity of the $L(p, q)$-labeling problem for bipartite planar graphs of small degree. Discrete Mathematics **309**, 3270–3279 (2009)
12. Knop, D., Masarík, T.: Computational complexity of distance edge labeling. Discret. Appl. Math. **246**, 80–98 (2018)
13. Lloyd, E.L., Ramanathan, S.: On the complexity of distance-2 coloring. Proc. ICCI 1992 pp. 71–74 (1992)
14. Mahdian, M.: On the computational complexity of strong edge coloring. Discrete Applied Mathematics **118**, 239–248 (2002)
15. Masarík, T.: Private communication (2020)
16. McCormick, S.: Optimal approximation of sparse hessians and its equivalence to a graph coloring problem. Mathematical Programming **26**, 153–171 (1983)
17. Schaefer, T.J.: The complexity of satisfiability problems. STOC 1978 pp. 216–226 (1978)