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

Gaétan Berthe[1], Barnaby Martin[2*], Daniël Paulusma[2] and Siani Smith[3]

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

*Corresponding author(s). E-mail(s): barnabymartin@gmail.com;

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.

Keywords: $L(p,q)$-labeling, colouring, dichotomy, computational complexity, NP-hard

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

Fig. 1 Colouring a triangle under the two definitions of $L(1,2)$-labelling: first (left) and second (right). Note that distinct vertices in a triangle are at distance one from one another (right), yet there is a path of length two between them as well (left).

$L(1,2)$-labelling, the vertices of a triangle K_3 can take labels $\{0,1,2\}$ in the second definition but need $\{0,2,4\}$ in the first. We illustrate this difference in Figure 1. 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].[1] The $L(1,0)$-Labelling is the traditional Graph Colouring problem (COL), whereas $L(1,1)$-Labelling is known as (Proper) Injective Colouring [2, 3, 9] and Distance 2 Colouring [13, 17]. 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.

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.

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 [15], 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 [16]. In fact, their proof settles for us all cases where $p \geq 2q$. We now give our main result.

[1] 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	Section 4	$(n-1)p+q+1$
$1<q/p\leq 2$	NAE-3-SAT	Section 5	$5p+1$
$q/p=1$	3-COL	[15]	$4p$
$2/3<q/p<1$	3-COL	Section 6	$3p+q+1$
$q/p=2/3$	1-in-3-SAT	Section 7	$4p$
$1/2<q/p<2/3$	2-in-4-SAT	Section 8	$p+4q+1$
$0<q/p\leq 1/2$	NAE-3-SAT	Section 9 [12]	$3p+1$
$p>0$ and $q=0$	3-COL	Section 2	$3p$

Table 1 Table of results. The case $2=q/p$ is covered by two regimes. The fourth row follows from [15] (which proves the case $p=q=1$) and applying Lemma 4. 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 $(n-3)p\geq q$.

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 4. We prove NP-hardness by reduction from graph 3-colouring and several satisfiability variants. 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 2 *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 3 *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 4 *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 5 *For all* $a > 0$, $L(a,0)$-EDGE-$3a$-LABELLING *is* NP*-complete.*

Let us discuss the NP-complete problems from which we reduce in this article. 3-COL takes as input a graph and asks whether there is a proper 3-colouring of the vertices (that is, one in which no two adjacent vertices take the same colour). NAE-3-SAT takes as input a collection of clauses each of which contains 3 literals. It asks whether there is a truth assignment to variables so that in each clause there is both a true and a false literal. The instance is *monotone* if all literals are positive. For $a < b$, a-in-b-SAT takes as input a collection of clauses each of which contains b literals. It asks whether there is a truth assignment to variables so that in each clause there are precisely a true literals. The instance is *monotone* if all literals are positive. The fact that our satisfiability problems are NP-hard follows from [18].

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

By Lemma 4 we only have to consider $a = 0$ and $b = 1$.

Theorem 6 *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 is a triangle with a path of length 2 added to each of two of the three vertices. We draw our variable gadget in Figure 2 and our clause gadget in Figure 3.

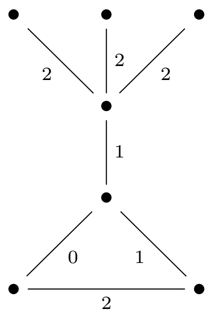

Fig. 2 The variable gadget for Theorem 6.

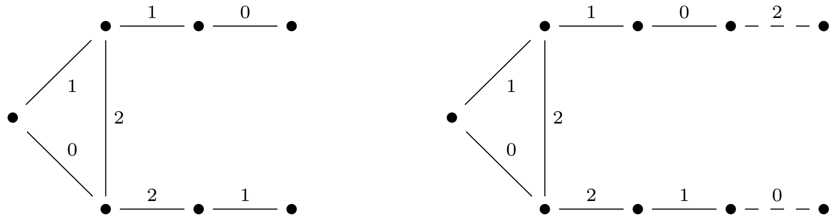

Fig. 3 The clause gadget for Theorem 6 (left) drawn also together with its interface with a variable gadget (right). The dashed line is an inner edge of the variable gadget.

Lemma 7 *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 $i-1 \bmod 3$. $\square$

Lemma 8 *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 6.

Proof of Theorem 6. 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 7 and the fact that it is proper follows from Lemma 8. □

4 Case $2 \le \frac{b}{a}$

In the case $2 \le \frac{b}{a}$, we can no longer get away with just an extended 4-star on which to base our variable gadget (as we did in Section 5). We need to move to higher degree. On the other hand, we will be able to dispense with the pendant 5-stars.

Theorem 9 *If $2 \le \frac{b}{a}$, let $n \ge 4$ be such that $(n-3)a \ge b$ then problem $L(a,b)$-Edge-$((n-1)a+b+1)$-Labelling is* NP*-complete.*

We will need the following lemma.

Lemma 10 *Let $2 \le \frac{b}{a}$ and let $n \ge 4$ be such that $(n-3)a \ge b$. In any valid $L(a,b)$-edge-$((n-1)a+b+1)$-labelling of the extended n-star, either all pendant edges are coloured in the interval $\{(n-2)a+b, \dots, (n-1)a+b\}$ or all pendant edges are coloured in the interval $\{0, \dots, a\}$.*

Proof Suppose some pendant edge is coloured by l' in $\{a+1, \dots, (n-2)a+b-1\}$. Consider the $n-1$ inner edges at distance 2 from it. Reading their labels in ascending order there must be a jump of at least $2b \ge a+b+1$ at some point unless the lowest

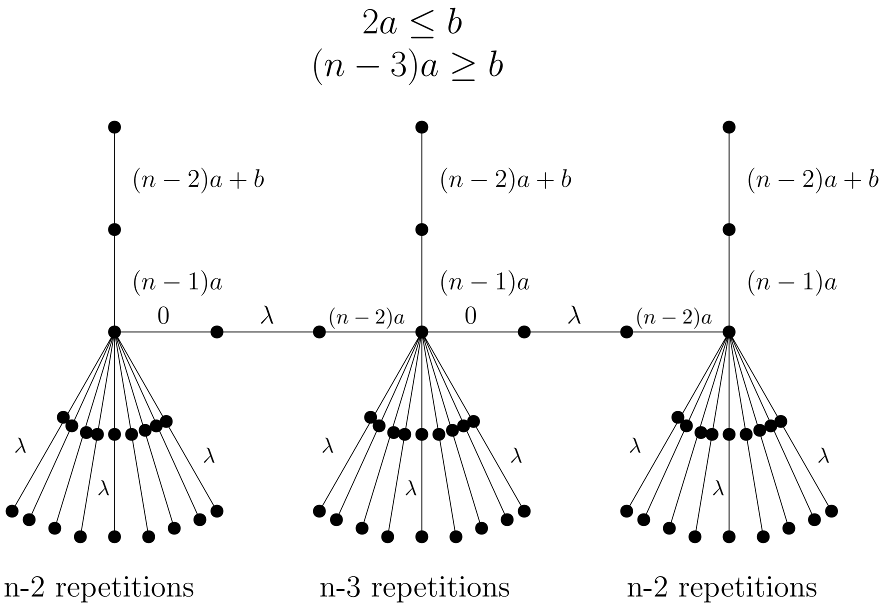

Fig. 4 The variable gadget for Theorem 9. The pendant edges drawn on the top will be involved in clauses gadget and each of these three edges can be coloured with anything from $\{(n-2)a+b,\ldots,(n-1)a+b\}$.

label is itself $a+b+1$. But now we have run out of labels, because $(n-2)a+(a+b+1) > (n-1)a+b$ which is the last label.

Suppose now that some pendant edge is coloured by l'_1 in $\{0,\ldots,a\}$ and another pendant edge is coloured by l'_2 in $\{(n-2)a+b,\ldots,(n-1)a+b\}$. It is now not possible to choose $n-2$ labels to complete the opposing inner edges, because l_1 and l_2 (inner edges adjacent to outer edges with labels l'_1 and l'_2, respectively) together must remove more than $b \geq 2a$ possibilities for labels at both the top and the bottom of the order. Using $2b > b+2a$, this leaves no more than $(n-3)a$ which is not enough space for $n-2$ labels spaced by a in the $n-2$ inner edges.

Finally, we note a valid colouring of the form $0,\ldots,(n-1)a$ for the inner edges of the extended n-star, with $\{(n-2)a+b,\ldots,(n-1)a+b\}$ enforced on the pendant edges (and the whole range from $\{(n-2)a+b,\ldots,(n-1)a+b\}$ is possible adjacent to the label $(n-1)a$). The other regime comes from order-inverting the colours. $\square$

The stipulation $(n-3)a \geq b$ plays no role in the previous lemma. It is needed in order to chain together extended n-stars to form the *variable gadget* whose construction we now explain. The variable gadget is made from a series of extended n-stars joined in a chain. They can join to one another in a path running from one's inner star edge labelled 0 to another's inner star edge labelled $(n-2)a$. In this fashion, the inner star edge labelled $(n-1)a$ is free for the (top) pendant edge that acts as the point of contact for clauses. This inner star edge may sometimes need to be labelled $(n-2)a$ (cf. Figure 5) in which case the other inner star edge labelled $(n-3)a$ will be needed to perform the chaining. In the following lemma, the designation *top* is with reference to the drawing in Figure 4.

Lemma 11 *Let $n \geq 4$ be such that $(n-3)a \geq b$. Any valid $L(a,b)$-edge-$(n-1)a+b+1$-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 $\{(n-2)a+b,\ldots,(n-1)a+b\}$. 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 12 *Let $n \geq 4$ be such that $(n-3)a \geq b$. A clause gadget is in a valid $L(a,b)$-edge-$(n-1)a+b+1$-labelling in the case where two of its edges are coloured $0, a$ and the third $(n-1)a+b$; or two of its edges are coloured $(n-2)a+b, (n-1)a+b$ and the third 0. If all three edges come from only one of the regimes $\{0,\ldots,a\}$ and $\{(n-2)a+b,\ldots,(n-1)a+b\}$, it cannot be in a valid $L(a,b)$-edge-$(n-1)a+b+1$-labelling.*

We are now ready to prove Theorem 9.

Proof of Theorem 9. We reduce from (monotone) NAE-3-SAT. Choose n such that $(n-3)a \geq b$. 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-$(n-1)a+b+1$-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 n-stars 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-$(n-1)a+b+1$-Labelling.

(Forwards.) Take a satisfying assignment for Φ. Let the range $\{0,\ldots,a\}$ represent true and the range $\{(n-2)a+b,\ldots,(n-1)a+b\}$ represent false. This gives a valid labelling of the inner vertices in the extended n-stars, as exemplified in Figure 4. 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 $(n-1)a+b$. 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 5 to demonstrate that this will not introduce problems at distance 2. Thus, we can see this is a valid $L(a,b)$-edge-$(n-1)a+b+1$-labelling of G.

(Backwards.) From a valid $L(a,b)$-edge-$(n-1)a+b+1$-labelling of G, we infer an assignment Φ by reading, in the variable gadget, the range $\{0,\ldots,a\}$ as true and the range $\{(n-2)a+b,\ldots,(n-1)a+b\}$ as false. The consistent valuation of each variable follows from Lemma 11 and the fact that it is in fact not-all-equal follows from Lemma 12. ∎

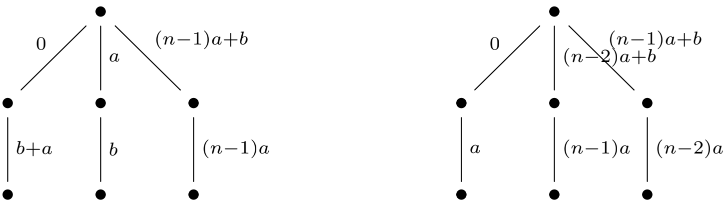

Fig. 5 The clause gadget and its interface with the variable gadgets (where we must consider distance 2 constraints). Both possible evaluations for not-all-equal are depicted. Note the difference $(n-2)a + b - (n-1)a = b - a > a$.

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

In this section we prove the following result.

Theorem 13 *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 14 *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 (indeed, they are at distance 1 from 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'. □

Let us remark that the colourings as restricted in Lemma 14 are achievable, and we will use them in the sequel.

We would like to chain extended 4-stars together to build our variable gadgets, where the pendant edges represent variables (and enter into clause

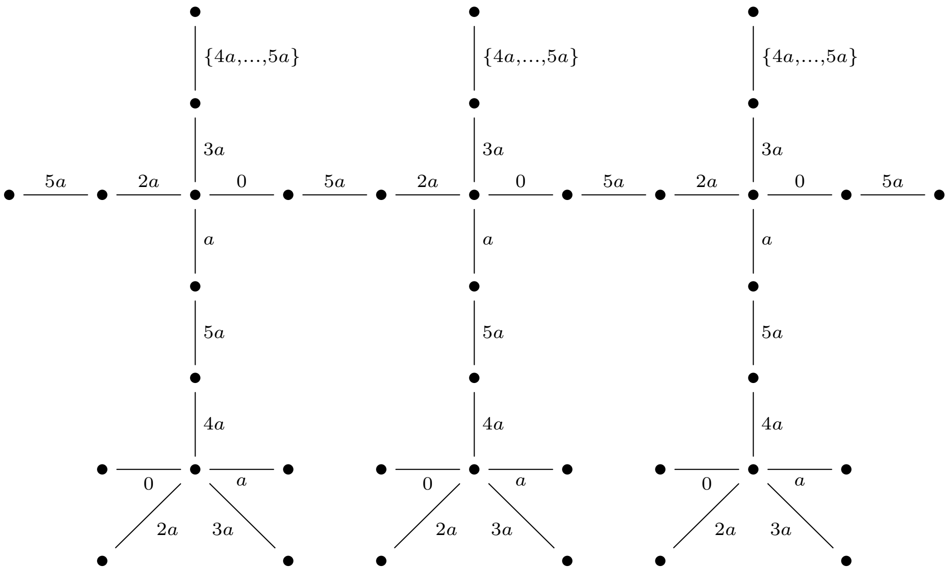

Fig. 6 Three extended 4-stars chained together, each with a pendant 5-star below, to form a variable gadget for Theorem 13. The pendant edges drawn on the top will be involved in clauses gadget and each of these three edges can be coloured with anything from $\{4a,\ldots,5a\}$. If the top pendant edge is coloured $5a$ it may be necessary that the inner star edge below is coloured not $3a$ but $2a$ (cf. Figure 7). This is fine, the chaining construction works when swapping $2a$ and $3a$.

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 14 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 6 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 6, 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 6 for $m=3$. The following is clear from our construction. The designation *top* is with reference to the drawing in Figure 6. In Figure 6, the case drawn corresponds to $\{4a,\ldots,5a\}$, where the case $\{0,\ldots,a\}$ is symmetric.

Lemma 15 *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 $\{0,\dots,a\}$ and $\{4a,\dots,5a\}$. 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 16 *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 are coloured $4a, 5a$ and the third 0. If all three edges come from only one of the regimes $\{0,\dots,a\}$ and $\{4a,\dots,5a\}$, it can not be in a valid $L(a,b)$-edge-$(5a+1)$-labelling.*

We are now ready to prove Theorem 13.

Proof of Theorem 13. 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 $\{0,\dots,a\}$ represent true and the range $\{4a,\dots,5a\}$ represent false. This gives a valid labelling of the inner edges in the extended 4-stars, as exemplified in Figure 6. 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 7 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 $\{0,\dots,a\}$ as true and the range $\{4a,\dots,5a\}$ as false. The consistent valuation of each variable follows from Lemma 15 and the fact that it is in fact not-all-equal follows from Lemma 16. □

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

In this section we prove the following result.

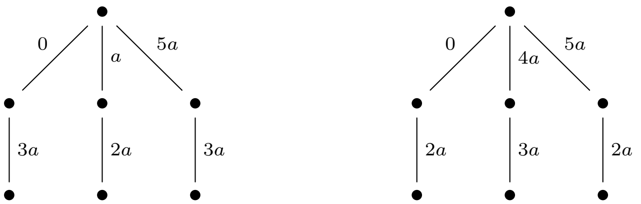

Fig. 7 The clause gadget and its interface with the variable gadgets (where we must consider distance 2 constraints). Both possible evaluations for not-all-equal are depicted.

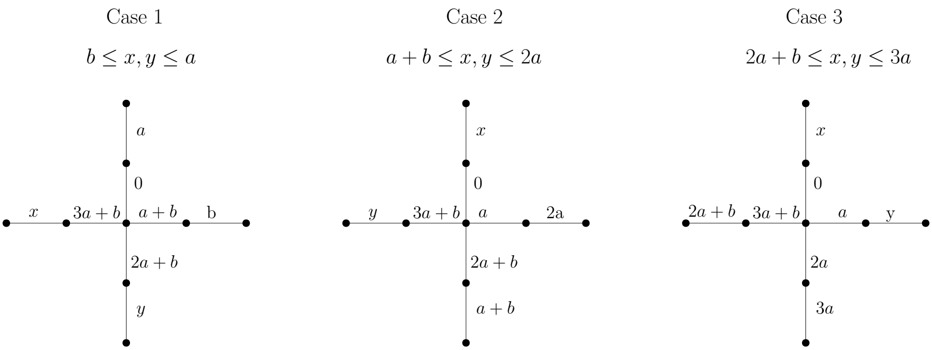

Fig. 8 The regimes of Theorem 17.

Theorem 17 *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 8.

Lemma 18 *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 $\{b,\dots,a\}$, the second is $\{2a+b,\dots,3a\}$, and the third is $\{a+b,\dots,2a\}$.*

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, $c_1 < l_i < c_4$.

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 $c_{i+1} - c_i \geq a+b$.

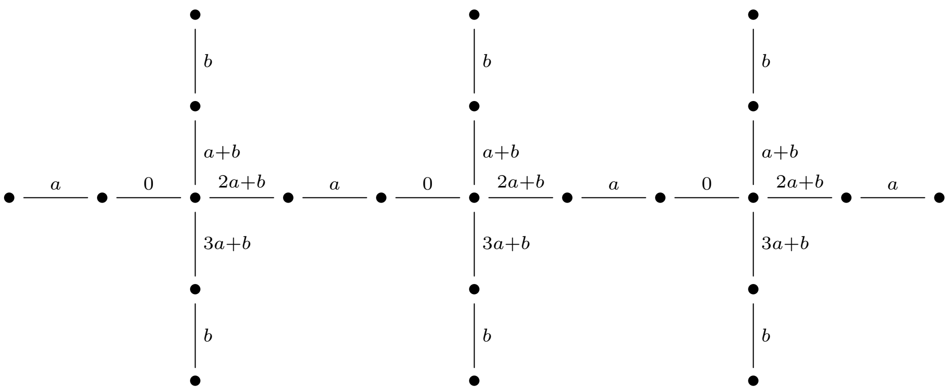

Fig. 9 Three extended 4-stars chained together, to form a variable gadget for Theorem 17. The pendant edges drawn on the top will be involved in clauses gadget. Suppose the top pendant edges are coloured b (as is drawn). In order to fulfill distance 2 constraints in the clause gadget, we may need the 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.

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 $\{2a + b, \ldots, 3a\}$. □

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 18 and is depicted in Figure 9.

Lemma 19 *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.*

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 20 *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 17.

Proof of Theorem 17. 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 18. 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 19 and the fact that it is proper (not-all-equal) follows from Lemma 20. □

7 Case $\frac{b}{a} = \frac{2}{3}$

In light of Lemma 4, it suffices to find k so that $L(3,2)$-EDGE-k-LABELLING is NP-hard.

Theorem 21 *The problem $L(3,2)$-EDGE-12-LABELLING problem is NP-complete.*

We use the colours $\{0,\ldots,11\}$. The following can be verified by hand; we used a computer.[2]

Lemma 22 *In a valid $L(3,2)$-edge-12-labelling of the extended 4-star, the possible labels of the three other pendant edges after one label is fixed are given in the following dictionary:*

$$2 : \{2,3,9\}$$

[2]The Python program for checking this can be found in the following github repository: `https://github.com/G-Berthe/Lpq-edge-labelling/`. Use, e.g.: `extended_four_star=[(0,1),(0,2),(0,3),(0,4),(1,5),(2,6),(3,7),(4,8)]` with `plotPoss(extended_four_star, 3, 2, 12, {(1,5):[2]})` to find $2:\{2,3,9\}$.

$$\begin{aligned}
3 &: \{2, 3\} \\
5 &: \{5, 6\} \\
6 &: \{5, 6\} \\
8 &: \{8, 9\} \\
9 &: \{2, 8, 9\}
\end{aligned}$$

For example, if one of the pendant edges is labelled 6*, then the others must be labelled by entries from* $\{5, 6\}$*. The possible multiplicities of these labellings is not specified in this dictionary.*

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).

Lemma 23 *Any valid $L(3, 2)$-edge-12-labelling of a variable gadget is such that the top pendant edges are all coloured from precisely one of the sets $\{5, 6\}$ or $\{2, 3, 8, 9\}$. 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, which is formed from a new vertex uniting three (top) pendant edges from their respective variable gadgets. This is drawn in Figure 11. The following is clear.

Lemma 24 *A clause gadget is in a valid $L(3, 2)$-edge-12-labelling precisely in the case where its edges are coloured one from $\{5, 6\}$ and two from $\{2, 3, 8, 9\}$.*

Proof of Theorem 21. We reduce from (monotone) 1-in-3-SAT. Let Φ be an instance of 1-in-3-SAT involving n occurrences of (not necessarily distinct) variables and m clauses. Let us explain how to build an instance G for $L(3, 2)$-Edge-12-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 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 1-in-3-SAT if and only if G is a yes-instance of $L(3, 2)$-Edge-12-Labelling.

(Forwards.) Take a satisfying assignment for Φ. Let the range $\{5, 6\}$ represent true and the range $\{2, 3, 8, 9\}$ represent false. In particular, every clause has two false and one should be chosen as (e.g.) 2 and the other 9. Thus, where a variable is false, some of top pendant edges are labelled 2 and others 9 (and this is shown in Figure 10). In each clause, we will have (say) $2, 9, 5$. 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 10 to demonstrate that this will not introduce problems at distance 2. Thus, we can see this is a valid $L(3, 2)$-edge-12-labelling of G.

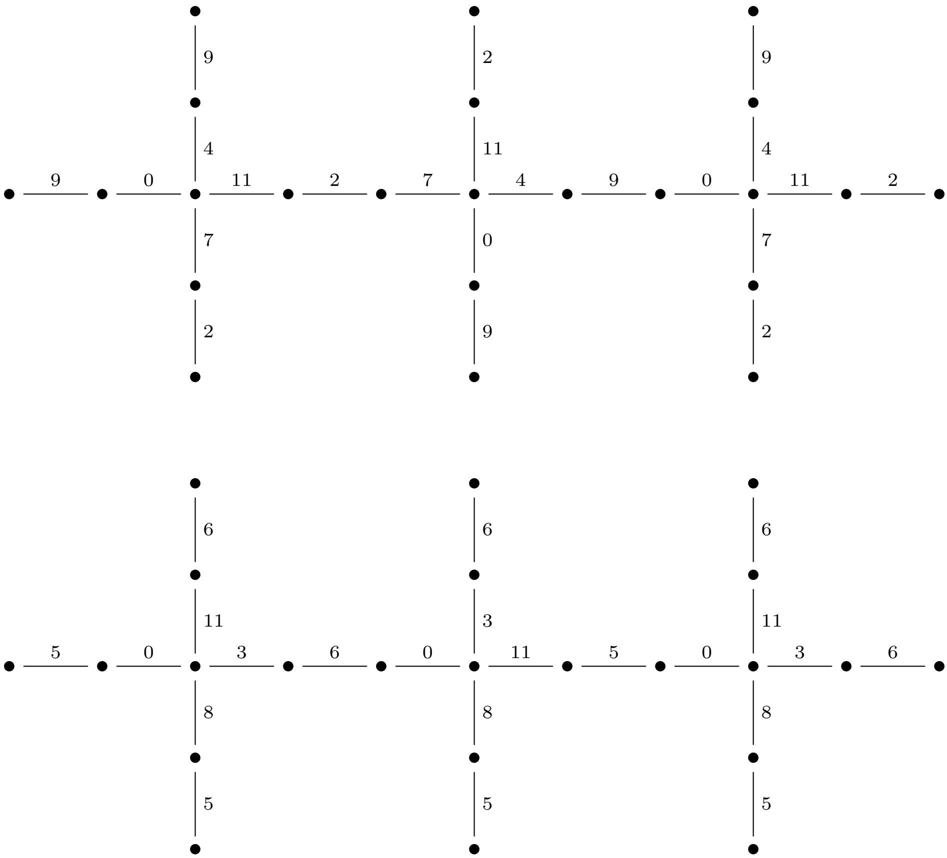

Fig. 10 Three extended 4-stars chained together, to form a variable gadget for Theorem 21. The pendant edges drawn on the top will be involved in clauses gadget. We show in the upper drawing how both sides of the regime representing false can be achieved (2 and 9). We show in the lower drawing how it works with $\{5, 6\}$.

(Backwards.) From a valid $L(3, 2)$-edge-12-labelling of G, we infer an assignment Φ by reading, in the variable gadget, range $\{5, 6\}$ as true and the range $\{2, 3, 8, 9\}$ as false. The consistent valuation of each variable follows from Lemma 23 and the fact that it is in fact not-all-equal follows from Lemma 24. $\square$

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

Theorem 25 *If $\frac{1}{2} < \frac{b}{a} < \frac{2}{3}$, then the problem $L(a, b)$-*Edge-*$(4b + a + 1)$*-Labelling *is* NP*-complete.*

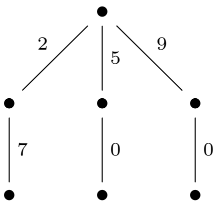

Fig. 11 The clause gadget and its interface with the variable gadgets (where we must consider distance 2 constraints).

This is probably the most involved case in terms of the sophistication of the proof. We need some lemmas before we can specify our gadgets.

Lemma 26 *If* $0 < b < a$ *and* $\lambda < 3a + b$*, with* $k = \lambda + 1$*, any edge* k*-labelling of the extended* 4*-star must involve inner edge labels of* $(0 \leq)p < q < r < s(< k)$ *so that both* $q - p \geq 2b$ *and* $s - r \geq 2b$*.*

Proof The assumption $\lambda < 3a + b$ forces: $\lambda - s < b$, $p < b$, $q - p, r - q, s - r < a + b$. Consider colouring the edge beside that edge which is coloured by r. This can't be coloured by anything other than something between p and q, forcing $q - p \geq 2b$. Similarly, consider colouring the edge beside that edge which is coloured by q. This can't be coloured by anything other than something between r and s, forcing $s - r \geq 2b$. □

□

Corollary 27 *Let* $a \leq 2b$*. The minimal* k *so that the extended* 4*-star gadget can be edge* k*-labelled is* $4b + a + 1$*.*

Proof We know it is at least $4b+a+1$ from the previous lemma. Further, the colouring alluded to in the previous proof extends to a valid colouring. Set labels (p, q, r, s) to $(0, 2b, 2b + a, 4b + a)$. Then, the edges next to p and q can be coloured $3b + a$, and the edges next to r and s can be coloured b. □

Lemma 28 *Let* $\frac{1}{2} < \frac{b}{a} < \frac{2}{3}$ *and* $k = 4b + a + 1$*. The extended* 4*-star gadget can be edge-*k*-labelled only such that two pendant edges are* b *and the other two are* $3b + a$*.*

Proof The inequality $\frac{1}{2} < \frac{b}{a}$ proves it is a correct labelling.

We have $\lambda = 4b + a < 3a + b$ so from the previous lemma we deduce the inner edge labels are $0, 2b, 2b + a, 4b + a$. Adjacent to $2b + a$ must be b and the same is true for $4b + a$. Adjacent to $2b$ must be $3b + a$ and the same is true of 0. □

Note that colours are in the set $\{0, \ldots, 4b + a\}$. Below, in Figure 12, we give two gadgets for the variables, the *end gadget* and the (basic part of the) *variable gadget*. The variable gadget admits a number of edge-$(4b + a + 1)$-labellings, but we want the only possibilities to be that drawn and one that swaps $3b+a$ and b. This we enforce by attaching an end gadget at the end (e.g. the left-hand end). For example, one may join it by adding new edges (in the present colouring of the end gadget, that would force the other colouring of the variable gadget). That is, we join the end gadget using the two edges drawn at the bottom below to the (basic part of the) variable gadget using the two edges drawn (say) to the left below. The join is accomplished by adding two new edges, one for each position. That is, one edge joins left and bottom, while the other edge joins right and top. In the variable gadget, the variables will extend from the 10-cycles, but this is possible only on one side. We now

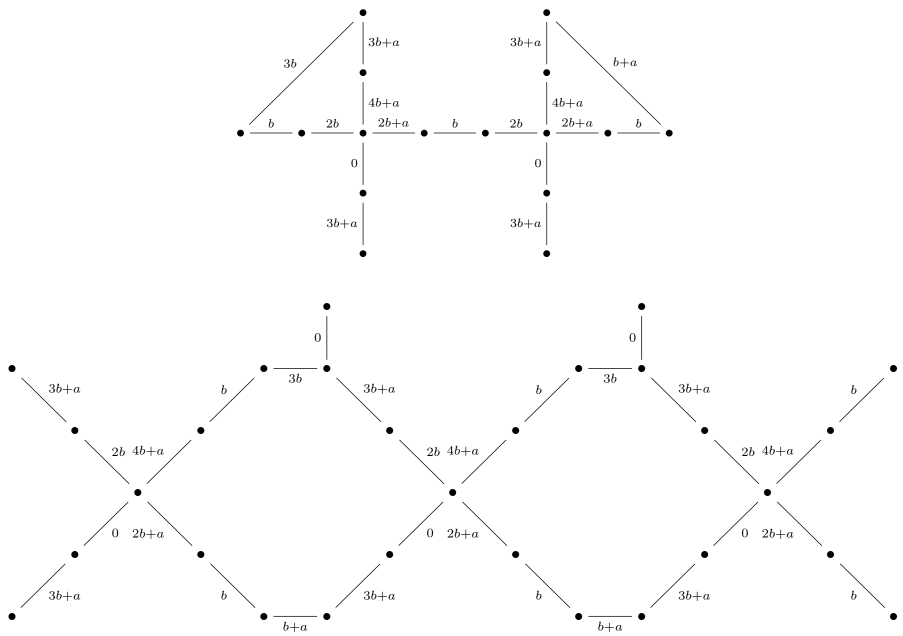

Fig. 12 End gadget (above) and basic part of variable gadget (below).

meet, in Figure 13, a full variable gadget drawn with a variable protrusion, in this case built from two 0 edges (the symmetric form gives two $4b+a$ edges).

Summing up, we derive the following lemma.

Lemma 29 *In a full variable gadget complete with an end gadget, any valid edge-$(4b+a+1)$-labelling has the property that the pendant edges from the basic part of the variable gadgets, which form the vertical edges in the full variable protrusion, are either all* 0 *or are all* $4b+a$*.*

The clause gadget is derived from an extended 4-star, whose properties we gave already in Lemma 28. Specifically, we extend the paths in the extended 4-star from length two to length four where they join the top node from a variable protrusion. Let us call this a triply extended 4-star. This is drawn in

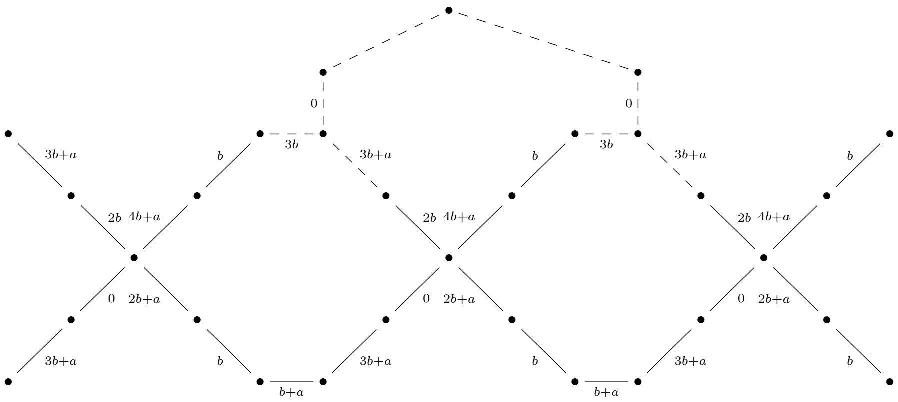

Fig. 13 A full variable gadget drawn with a variable protrusion. Note that each variable protrusion, as the gadget repeats, must be of the same kind. This is demonstrated in Figure 14 where it is shown that the alternative colouring is impossible. The dashed lines in the present drawing also appear in our depiction of the clause gadget in Figures 15 and 16.

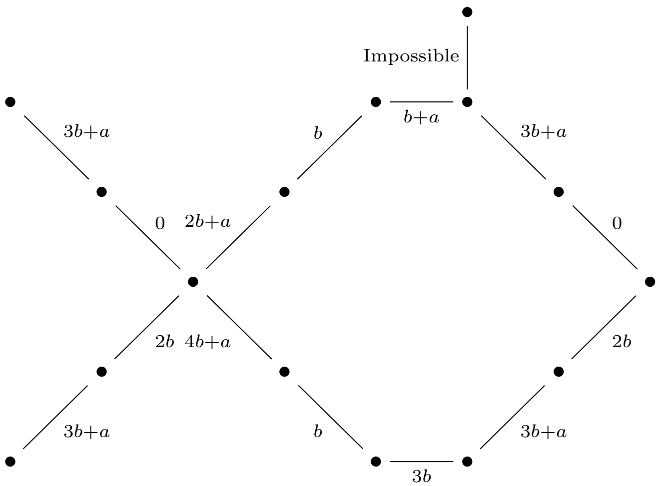

Fig. 14 Demonstration that the variable protrusions are determined once the left-hand leaves of the first extended 4-star are chosen (remember they are ultimately made equal by the end gadget).

Figure 15, where we also show the interface with the variable gadgets, together with a valid colouring.

Proof of Theorem 25. We reduce from (monotone) 2-in-4-SAT. Let Φ be an instance of 2-in-4-SAT involving n occurrences of (not necessarily distinct) variables and m

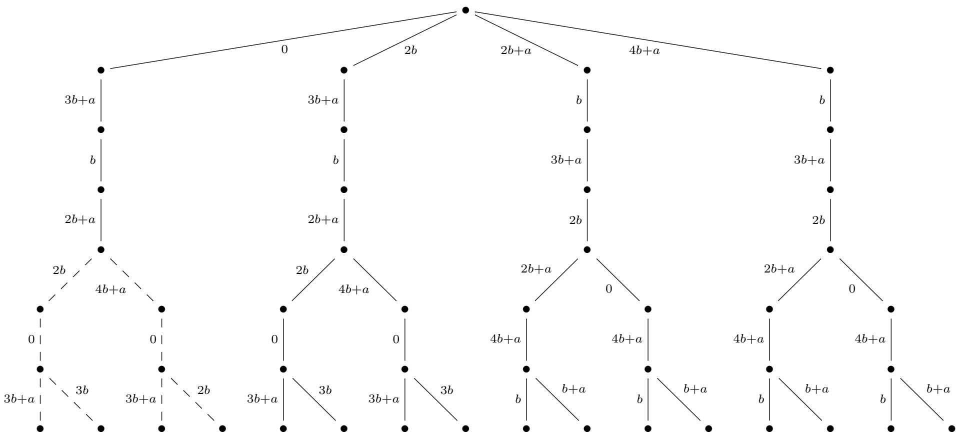

Fig. 15 The clause gadget and its interface with the variable gadgets (where we must consider distance 2 constraints). In the first of the four variables, on the left-hand branch, we show in dashed lines the corresponding variable protrusion.

clauses. Let us explain how to build an instance G for $L(a, b)$-EDGE-$(4b + a + 1)$-LABELLING. Each particular variable only appears at most n times, so for each variable we take a full variable gadget with n variable protrusions. Each particular instance of the variable belongs to the top vertex of a variable protrusion (one of these is drawn in Figure 13, but none appears in Figure 12). For each clause of Φ we use a triply extended 4-star to unite some instance of these top vertices of the variable protrusions from the corresponding full variable gadgets. We claim that Φ is a yes-instance of 2-in-4-SAT if and only if G is a yes-instance of $L(a, b)$-EDGE-$(4b + a + 1)$-LABELLING.

(Forwards.) Take a satisfying assignment for Φ. Let 0 represent true and $4b + a$ represent false. Then, every clause has two true and two false variables and these can be consistently united in an triply extended 4-star as in Figure 15. This is a valid $L(a, b)$-edge-$(4b + a + 1)$-labelling of G.

(Backwards.) From a valid $L(a, b)$-edge-$(4b + a + 1)$-labelling of G, we infer an assignment Φ by reading, in the full variable gadget, 0 as true and $4b + a$ as false. The consistent valuation of each variable follows from Lemma 29 and the fact that it is 2-in-4 follows from Lemma 28, bearing in mind the impossibility of colouring a path in the clause gadget as in Lemma 30 and Figure 16. □

Lemma 30 *The colouring depicted in Figure 16 cannot be completed from the initial colouring of the second to top edge as b and the lower six edges as 0 (above) and $3b + a$ and $3b$ (below).*

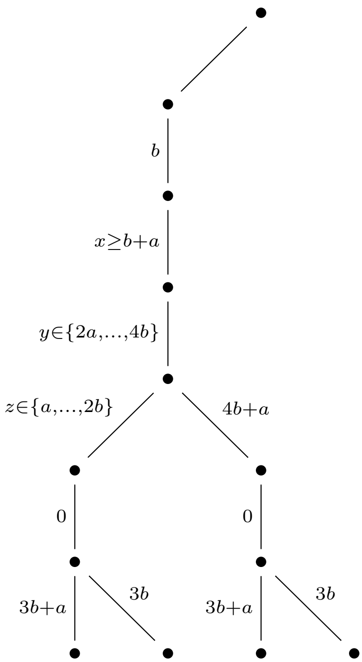

Fig. 16 An impossible colouring on a path in a clause gadget that shows (together with the valid colouring of Figure 15) that the clause gadget enforces 2-in-4-SAT.

Proof Case 1 : $x \leq y$.
As x and y are neighbours, we have $x \leq y - a \leq 4b - a$. So the distance between x and z is $\leq 4b - a - a = 4b - 2a < b$.

This is not possible as we need the distance between x and z to be $\geq b$.

Case 2 : $y \leq x$.
As x and y are neighbours, we have $x \geq y + a \geq 3a$. So $4b + a - x \leq 4b - 2a = 4b - 2a < b$.

This is not possible as we need $4b + a - x \geq b$. □

9 Case $0<\frac{b}{a} \leq \frac{1}{2}$

We follow the exposition of [12], which addresses the $L(2,1)$-edge-7-labelling problem. With permission we have used (an adaptation) of their diagrams. Note that in [12], they would call the problem we address the $L(a,b)$-edge-**3a**-labelling problem as, in their exposition $3a$ refers to the set $\{0,\ldots,3a\}$.

Theorem 31 *The $L(a,b)$-edge-$(3a+1)$-labelling problem, for $a \geq 2b$, is* NP*-hard.*

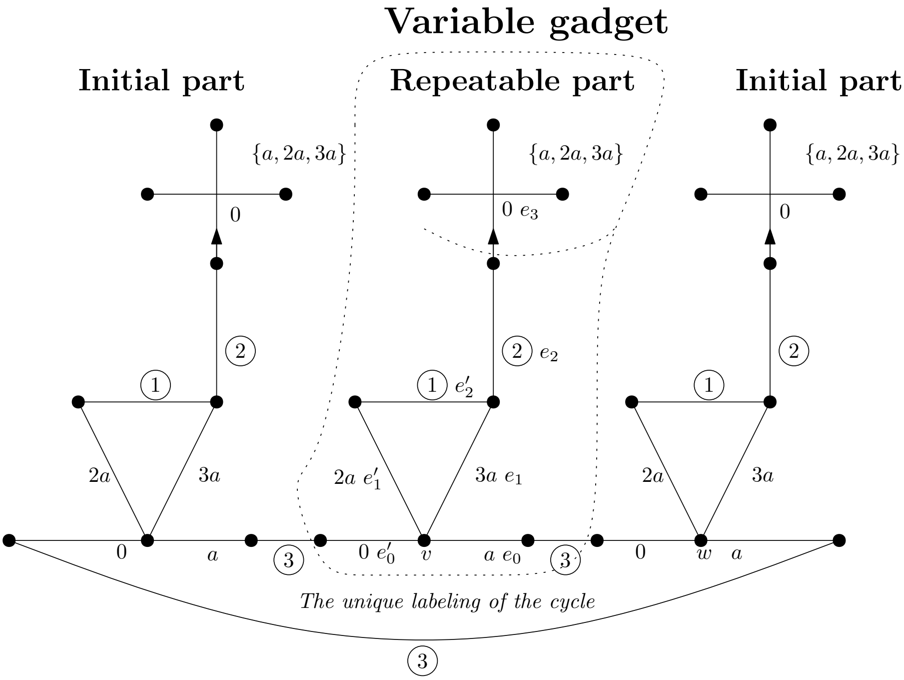

Fig. 17 Variable gadget (adapted from [12]).

Proof By reduction from (monotone) NAE-3-SAT using the gadgets and properties detailed in Lemmas 32 and 33, below. □

For $1 \le k \le 3$, we define the sets $\textcircled{k} = [\![(k-1)a+b, ka-b]\!]$. The edges of a 4-star have to be coloured $0, a, 2a, 3a$, in any valid $L(a,b)$-edge-$(3a+1)$-labelling. Then any neighbouring edge of the star has to be in one these sets, of the form $\textcircled{k}$. These properties we will now use without further comment.

A variable is represented by the *variable gadget* of Figure 17.

Lemma 32 *In any valid $L(a,b)$-edge-$(3a+1)$-labelling of the variable gadget, the three edges free in the top of a 4-star at the top of a repeatable section must be coloured (in all repeatable and initial parts) by either $\{a,2a,3a\}$ or $\{0,a,2a\}$.*

Proof Let us consider various possibilities for the colouring of $\{e_0', e_0\}$ and $\{e_1', e_1\}$ (up to the order inverting map that takes $(0,\ldots,3a)$ to $(3a,\ldots,0)$). These are drawn in Table 2 (essentially reproduced from [12]) together (in some cases) with why they lead to contradiction. The Cases III-VI are straightforward. For Cases I and II we need to argue why $\{e_0', e_0\}$ cannot be $\{a, 2a\}$. In this case, the cycle must continue (bearing in mind that every edge with a vertex of degree 4 must be from $\{0, a, 2a, 3a\}$) in a certain way. To the right it must continue: $(a, 2a, \textcircled{1}, 3a, 2a, \textcircled{1}, 3a, 2a, \ldots)$.

	e'_0 e_0	e'_1 e_1	e'_2 e_2	e_3
I.	a $2a$	0 $3a$	in 2 in 1	$a, 2a$ impossible in the cycle
II.	a $2a$	$3a$ 0	in 2 in 3	$a, 2a$ impossible in the cycle
III.	0 $3a$	a $2a$	impossible —	—
IV.	0 $2a$	a $3a$	impossible —	—
V.	0 a	$3a$ $2a$	in 1 impossible	—
VI.	0 a	$2a$ $3a$	in 1 in 2	0

Table 2 Variable gadget table (adapted from [12]).

However, to the left it must continue: $(2a, a, 3, 0, a, 3, 0, a, \ldots)$. These paths can now never join together in a cycle. This rules out Cases I and II.

The remaining labellings, Case VI and its various symmetries, are possible and result in the claimed behaviour. □

The clause is represented by the *clause gadget* of Figure 18.

Lemma 33 *Consider any valid $L(a,b)$-edge-$(3a+1)$-labelling of the clause gadget, such that the input parts of the variable gadgets satisfy the previous lemma. Two of the input variable gadget parts must come from one of the regimes $\{a, 2a, 3a\}$ or $\{0, a, 2a\}$, and the other input part from the other regime. In particular, if all three input variable gadget parts come from only one of the regimes, then this can not be extended to a valid $L(a,b)$-edge-$(3a+1)$-labelling.*

Proof Let us consider various possibilities for the colouring of e_1 and $\{e'_1, e''_1\}$ (up to the order inverting map that takes $(0, \ldots, 3a)$ to $(3a, \ldots, 0)$). These are drawn in Table 3 (essentially reproduced from [12]) together (in some cases) with why they lead to contradiction.

Cases III and IV show the valid possibilities. The three edges where the 4-star unites the variable gadget repeatable parts has only two possibilities for each of the variable regimes $\{a, 2a, 3a\}$ or $\{0, a, 2a\}$ (namely, $\{0, 2a\}$ and $\{a, 3a\}$, respectively). The claimed behaviour is clear. □

10 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 $k \geq 4$) [15]; and for

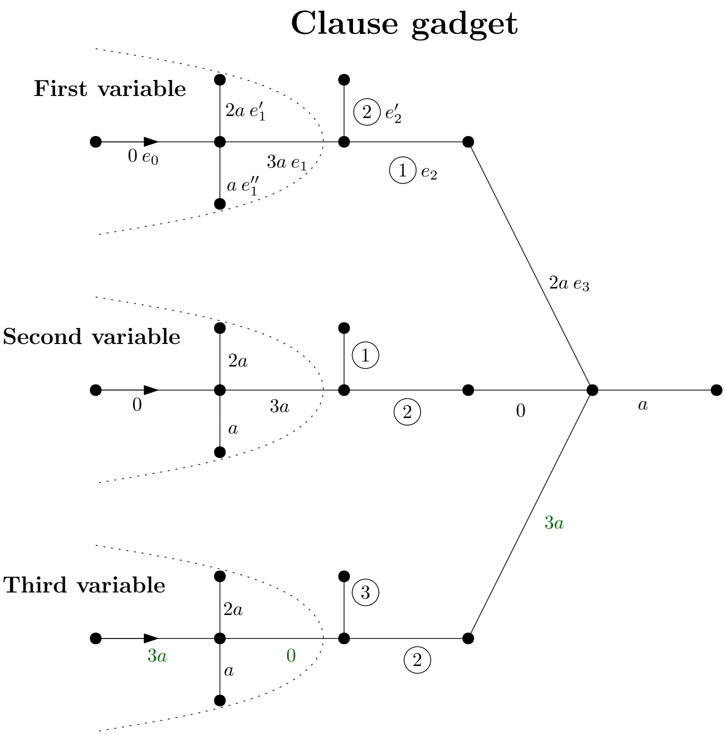

Fig. 18 Clause gadget (adapted from [12]).

$(p,q) = (2,1)$ 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 3 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].

Acknowledgments. An extended abstract of this paper, omitting numerous proofs, appeared at The 16th International Conference and Workshops on Algorithms and Computation (WALCOM) 2022 [1].

	e_0	e_1'' e_1' e_1	e_2' e_2	e_3
I.	0	$2a$ or $3a$ $3a$ or $2a$ a	Both in 3. Impossible Both in 3. Impossible	—
II.	0	a or $3a$ $3a$ or $2a$ $2a$	Both in 1. Impossible Both in 1. Impossible	—
III.	0	$2a$ or a a or $2a$ $3a$	In 1 In 2	0 or 3
IV.	0	$2a$ or a a or $2a$ $3a$	In 2 In 1	$2a$ or 3

Table 3 Clause gadget table (adapted from [12]).

References

[1] Berthe, G., Martin, B., Paulusma, D., Smith, S.: The complexity of L(p, q)-edge-labelling. Proc. WALCOM 2022, LNCS **13174** 175–186 (2022)

[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. Electron. J. Comb. 29(2) P2.43 (2022)

[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] Lovász, L.: Coverings and coloring of hypergraphs. Proc. 4th Southeastern Conference on Combinatorics, Graph Theory, and Computing (Florida Atlantic Univ., Boca Raton, 1973), Congressus Numerantium, **VIII**, 3–12 (1973)

[15] Mahdian, M.: On the computational complexity of strong edge coloring. Discrete Applied Mathematics **118**, 239–248 (2002)

[16] Masarík, T.: Private communication (2020)

[17] McCormick, S.: Optimal approximation of sparse hessians and its equivalence to a graph coloring problem. Mathematical Programming **26**, 153–171 (1983)

[18] Schaefer, T.J.: The complexity of satisfiability problems. STOC 1978 pp. 216–226 (1978)