A tight quasi-polynomial bound for Global Label Min-Cut[*]

Lars Jaffke[1,2], Paloma T. Lima[3], Tomáš Masařík[1], Marcin Pilipczuk[1], and Ueverton S. Souza[1,4]

[1]University of Warsaw, Poland
[2]University of Bergen, Norway
[3]IT University of Copenhagen, Denmark
[4]Fluminense Federal University, Brazil

July 18, 2022

Abstract

We study a generalization of the classic Global Min-Cut problem, called Global Label Min-Cut (or sometimes Global Hedge Min-Cut): the edges of the input (multi)graph are labeled (or partitioned into color classes or hedges), and removing all edges of the same label (color or from the same hedge) costs one. The problem asks to disconnect the graph at minimum cost.

While the st-cut version of the problem is known to be NP-hard, the above global cut version is known to admit a quasi-polynomial randomized $n^{\mathcal{O}(\log \mathrm{OPT})}$-time algorithm due to Ghaffari, Karger, and Panigrahi [SODA 2017]. They consider this as "strong evidence that this problem is in P". We show that this is actually not the case. We complete the study of the complexity of the Global Label Min-Cut problem by showing that the quasi-polynomial running time is probably optimal: We show that the existence of an algorithm with running time $(np)^{o(\log n/(\log\log n)^2)}$ would contradict the Exponential Time Hypothesis, where n is the number of vertices, and p is the number of labels in the input. The key step for the lower bound is a proof that Global Label Min-Cut is W[1]-hard when parameterized by the *number of uncut labels*. In other words, the problem is difficult in the regime where almost all labels need to be cut to disconnect the graph. To turn this lower bound into a quasi-polynomial-time lower bound, we also needed to revisit the framework due to Marx [Theory Comput. 2010] of proving lower bounds assuming Exponential Time Hypothesis through the Subgraph Isomorphism problem parameterized by the number of edges of the pattern. Here, we provide an alternative simplified proof of the hardness of this problem that is more versatile with respect to the choice of the regimes of the parameters.

1 Introduction

Given a weighted graph $G = (V, E)$ with n vertices and m edges, in the (Global) MIN-CUT problem we are asked to find a cut $[S, T]$ of G minimizing the total weight of edges crossing from S to T, where a cut of G is a partition of V into two non-empty subsets S and $T = V \setminus S$. This problem is one of the most fundamental problems in computer science, and dates back to the earliest days of the field [8].

An st-cut of a graph G is a cut $[S, T]$ having $s \in S$ and $t \in T$, and a minimum cut of G is a cut having an edge-cut set of minimum total weight. We denote by MIN-st-CUT the problem of computing the minimum st-cut of a given graph. A minimum st-cut of a graph G can be found by using any algorithm for computing a maximum st-flow. While the history of minimum st-cut and maximum flow algorithms dates back to the 1950s [8, 9], revolutionary results are obtained regularly up to today: we mention [10, 18, 20], and, most notably, the recent breakthrough near-linear algorithm for polynomially bounded capacities [3].

Due the duality of maximum st-flows and minimum st-cuts, a typical strategy to solve the MIN-CUT problem consists of finding minimum st-cuts for a fixed vertex s and all $n - 1$ possible choices of t and then selecting the smallest one. However, a few number of non-flow-based methods for solving MIN-CUT have been developed [11, 16, 17, 22, 23, 27]. In particular, in 1993, Karger [16] developed a simple randomized algorithm to compute a minimum cut of a connected graph, which later inspired many subsequent works.

A natural generalization of MIN-CUT on graphs is the MIN-CUT problem on hypergraphs, which aims to determine the smallest number of hyperedges to be removed to disconnect a given hypergraph. In 1997, Stoer and Wagner [27] presented a non-flow-based algorithm for MIN-CUT on hypergraphs with $O(n \cdot m + n^2 \log n)$ running time. This approach was generalized by Queyranne [24, 25] for minimizing symmetric submodular functions. In [24, 25], it was shown that if V is a finite set and $f : 2^V \to \mathbb{R}$ is a polynomial-time computable function that is symmetric and submodular then there is a polynomial-time algorithm that minimizes f over all proper subsets of V. Rizzi [26] observed that the hypergraph cut function is symmetric and submodular, which implies that Min-Cut on hypergraphs can be solved by using polynomial-time algorithms for submodular function minimization [24, 25]. Note that MIN-CUT on hypergraphs is a particular case of a more general cut problem on edge-colored multigraphs: if the vertex set of each hyperedge i is viewed as a connected subgraph whose edges are colored with color c_i then the goal is to find a cut $[S, T]$ minimizing the number of colors in $\partial(S)$, the set of edges with exactly one endpoint in S.

In this paper, we consider the more general problem where each color class is allowed to induce more than one connected component. Let $G = (V, E)$ be an edge-colored multigraph with n vertices, m edges, and an edge coloring $c : E \to \{1, 2, \ldots, p\}$, not necessarily proper. We let $c(\partial(S))$ be the set of colors that appear in $\partial(S)$. GLOBAL LABEL MIN-CUT is the problem of finding a subset $S \subseteq V$, $S \neq \emptyset, V$ that minimizes the size of $c(\partial(S))$.

GLOBAL LABEL MIN-CUT
Input: A multigraph $G = (V, E)$ with an edge coloring $c : E \to \{1, 2, \ldots, p\}$ and an integer $k > 0$.
Output: YES if G has a proper subset $\emptyset \neq S \subsetneq V$ such that $|c(\partial(S))| \leqslant k$, No otherwise.

GLOBAL LABEL MIN-CUT naturally captures a problem in network survivability in which the failure of a single connection implies failure of a whole set of connections, see for instance [5, 4, 12]. In the literature, the colors are sometimes called *labels* or *hedges*; in this work, we stick to *colors*.

Similarly, in LABEL MIN-st-CUT, we are asked to find an edge-cut set that separates a given pair s, t of vertices using at most k colors (or, as few colors as possible). Note that the analogously defined MIN-CUT on "hyperedge-colored" hypergraphs is in fact the same problem as GLOBAL LABEL MIN-CUT.

The computational complexity of LABEL MIN-st-CUT and GLOBAL LABEL MIN-CUT has been widely investigated in recent years [1, 2, 4, 5, 7, 12, 28, 29, 30, 31, 32, 33]. As first observed in [29], if we define a function f on 2^V that counts the number of colors in an edge-cut set resulting from a vertex subset S, that is, $f(S) = c(\partial(S))$, then we can easily verify that f is not submodular. Therefore we cannot use

the same approach as for MIN-CUT on hypergraphs via submodular function minimization to solve GLOBAL LABEL MIN-CUT.

In the special case when every color contains at most r edges, Coudert et al. [4] showed that GLOBAL LABEL MIN-CUT can be solved in polynomial time, but LABEL MIN-st-CUT is NP-hard even when each color contains at most two edges. Furthermore, Blin et al. [1] presented a randomized polynomial-time algorithm for GLOBAL LABEL MIN-CUT that returns an optimal colored cut of G with probability at least $|V|^{-2r}$.

Several approximation and hardness results for LABEL MIN-st-CUT are presented in [28, 29, 30, 32, 33]. Zhang and Fu [31] showed that LABEL MIN-st-CUT is NP-hard even if the maximum length of any path is equal to two. Regarding the parameterized complexity of LABEL MIN-st-CUT, Fellows et al. [7] showed that the problem is W[2]-hard when parameterized by the number of colors of the solution, and W[1]-hard when parameterized by the number of edges of the solution. Coudert et al. [5] showed that GLOBAL LABEL MIN-CUT can be solved in time $2^s \cdot n^{O(1)}$, where s is the number of colors that induce more than one nontrivial connected component.

Most importantly, Ghaffari, Karger, and Panigrahi [12] showed that GLOBAL LABEL MIN-CUT can be solved exactly in time $n^{\mathcal{O}(\log \text{OPT})}$, where n is the number of vertices of the graph and OPT denotes the cost of the optimum solution, and a $(1 + \varepsilon)$-approximate solution can be found in time $n^{\mathcal{O}(\log(1/\varepsilon))}$. This results shows that GLOBAL LABEL MIN-CUT is probably not NP-hard, contrary to its st-cut variant, LABEL MIN-st-CUT.

Despite the above progress, the exact computational complexity of GLOBAL LABEL MIN-CUT was left open by all these works. To the best of our knowledge, the first time that the computational complexity of GLOBAL LABEL MIN-CUT was mentioned as an open question it was in [4]. Note that the result of [12] places GLOBAL LABEL MIN-CUT between the classes of problems solvable in polynomial and quasi-polynomial time. After 15 years of research, the community is still asking for an answer to this question. More recently, in [33], Zhang and Tangi explicitly stated that "a challenging problem is to determine the exact complexity of GLOBAL LABEL CUT." In addition, prior to [12] it was pointed out [4, 29] that unlike the situation of MIN-CUT and MIN st-CUT where both are polynomial-time solvable, GLOBAL LABEL MIN-CUT seems to behave differently than its local counterpart (LABEL MIN-st-CUT) and should be easier than it; the result of [12] confirms this suspicion.

Our results. Ghaffari, Karger, and Panigrahi write that their randomized $n^{\mathcal{O}(\log \text{OPT})}$-time algorithm provides "strong evidence that this problem is in P" [12, abstract]. We show that this is probably not the case, and settle the question of the complexity of GLOBAL LABEL MIN-CUT: one of our main results states that the quasi-polynomial complexity is tight under the Exponential Time Hypothesis [15].[1]

Theorem 1. *Unless the Exponential Time Hypothesis fails, there is no algorithm solving GLOBAL LABEL MIN-CUT in time $(np)^{o(\log n/(\log\log n)^2)}$.*

Observe that the lower bound of Theorem 1 is tight up to a $\mathcal{O}((\log \log n)^2)$ factor in the exponent with the algorithm of [12], as $\text{OPT} \leqslant p$ and $n^{\mathcal{O}(\log \text{OPT})} = \text{OPT}^{\mathcal{O}(\log n)} \leqslant p^{\mathcal{O}(\log n)}$. The key ingredient in the proof of Theorem 1 is the proof that GLOBAL LABEL MIN-CUT is W[1]-hard when parameterized by *the number of uncut colors*, that is, the value $p - k$.

Theorem 2. *The GLOBAL LABEL MIN-CUT problem, parameterized by $a := p - k$, the number of uncut colors, is W[1]-hard. Furthermore, unless the Exponential Time Hypothesis fails, there is no computable function f and an algorithm for the problem with running time bound $f(a) \cdot (np)^{o(a/\log a)}$.*

Theorem 2 is proven in Section 2. A natural step to obtain Theorem 1 from Theorem 2 is to pipeline the reduction of Theorem 2 (if it started in the classic MULTICOLORED CLIQUE problem) with a standard reduction from 3-CNF SAT to MULTICOLORED CLIQUE that makes a correct choice of the parameters (here, from an input n-variable m-clause formula we would want a MULTICOLORED CLIQUE instance with

---

[1]The Exponential Time Hypothesis, a now-standard assumption in fine-grained complexity, together with the Sparsification Lemma asserts that one cannot resolve satisfiability of an n-variable m-clause 3-CNF formula in time $2^{o(n+m)}$ [15].

roughly $2^{\mathcal{O}(\sqrt{n+m})}$ vertices and looking for a clique of size $\mathcal{O}(\sqrt{n+m})$). Unfortunately, the reduction of Theorem 2 makes heavy use of the so-called edge-choice gadgets and therefore starts instead in the SUBGRAPH ISOMORPHISM problem, parameterized by the number of edges of the pattern. Hardness of this source problem is provided by Marx [21]; his reduction is involved and not directly amenable to such choose-convenient-range-of-parameters tricks.

To cope with this issue, we revisit the framework of Marx [21] and provide a simplified and streamlined proof of the hardness of SUBGRAPH ISOMORPHISM that is now versatile to a choice of parameter range. The exact statement (and necessary definitions) can be found in Section 3. (We remark that the main goal of the framework of Marx [21] was to provide hardness for the *treewidth parameterization* and the "number of edges" parameter was only a side corollary there; our simplified proof is tailored for the latter parameter and has no implications for the treewidth parameterization.)

2 Hardness of DUAL GLOBAL LABEL MIN-CUT

In this section we give the main reduction that leads to the quasi-polynomial time lower bound under the ETH for GLOBAL LABEL MIN-CUT, and a proof of Theorem 2. To give the reduction, we slightly change the perspective on the problem. A computationally expensive case in the algorithm of [12] is when a large number of colors is incident with a large number of vertices. In particular, this forces a high depth of recursion in the execution of the algorithm. In this situation, each color is incident with many vertices, so only very few (say at most a) colors do not appear in an optimum solution. The algorithm of [12] in fact in this case reverts to brute-forcing all $\binom{p}{a}$ sets of colors that are not in the solution (recall that p is the total number of colors).

This scenario reveals that the number of colors *not* present in a minimum colored cut is a relevant parameter for understanding the computational complexity of GLOBAL LABEL MIN-CUT. We therefore consider its dual problem, parameterized by solution size. (Note that finding the maximum-size set of colors that do not belong to some cut of a graph G is equivalent to finding the maximum number of colors such that the union of their edges forms a disconnected subgraph of G.)

DUAL GLOBAL LABEL MIN-CUT
Input: Vertex set W, graphs $G_1, \ldots, G_p$ on vertex set W, integer a.
Output: YES if there exists a set $I \subseteq [p]$ with $|I| = a$ such that $(W, \cup_{i \in I} E(G_i))$ is disconnected, NO otherwise.

We will confirm our intuition that indeed, the case described above makes the problem computationally hard, by giving a reduction that proves W[1]-hardness of DUAL GLOBAL LABEL MIN-CUT parameterized by a, and, together with a reduction given in Section 4, an $(np)^{o(\log n/(\log \log n)^2)}$ time lower bound for GLOBAL LABEL MIN-CUT under the ETH. The reduction is from the PARTITIONED SUBGRAPH ISOMORPHISM problem.

PARTITIONED SUBGRAPH ISOMORPHISM (PSI)
Input: Graph H, called *pattern*, graph K, called *host*, with vertex partition $\uplus\{V_x \mid x \in V(H)\}$ such that $|V_x| = n$ for all $x \in V(H)$.
Output: YES if there is a set $\{v_x \in V_x \mid x \in V(H)\} \subseteq V(K)$ such that for all $x, y \in V(H)$, $xy \in E(H)$ implies $v_x v_y \in E(K)$, NO otherwise.

To give an overview of the main idea behind the reduction of the following theorem, we consider for one moment a *partitioned* version of DUAL GLOBAL LABEL MIN-CUT, where in addition we are given a partition $(\mathcal{G}_1, \ldots, \mathcal{G}_a)$ of $\{G_1, \ldots, G_p\}$ and we require a solution to select one graph from each $\mathcal{G}_i$.

The following simple reduction from PSI to PARTITIONED DUAL COLORED MIN-CUT (PDCMC) would suffice if we could restrict our attention to this problem instead of the non-partitioned version. Let (H, K) be an instance of PSI where H is the pattern graph and K the host graph, and let $E(H) = \{e_1, \ldots, e_a\}$. We may assume that H is connected. We construct an instance $(W, \mathcal{G}_1, \ldots, \mathcal{G}_a)$ of PDCMC as follows.

We let $W = \{t\} \cup V(K)$. For all $i \in [a]$, let $e_i = xy$, and consider the set of edges $E(V_x, V_y) \subseteq E(K)$ with one endpoint in V_x and the other in V_y. For all $v_x v_y \in E(V_x, V_y)$, we add a graph $G(i, v_x, v_y)$ to $\mathcal{G}_i$ that consists of the edge $v_x v_y$, and all edges tz, where $z \in V_x \cup V_y \setminus \{v_x, v_y\}$ (cf. Figure 1).

For one direction, suppose that $X = \{v_x \in V_x \mid x \in V(H)\}$ is a solution to (H, K). Then, taking the union over all $e_i = xy$ of the graphs $G(i, v_x, v_y)$ (where $v_x, v_y \in X$) is a solution to $(W, \mathcal{G}_1, \ldots, \mathcal{G}_a)$, since no vertex in X is adjacent to t in the resulting graph. For the other direction, let $(G_1 \in \mathcal{G}_1, \ldots, G_a \in \mathcal{G}_a)$ be a solution to the PDCMC instance. Consider some $x \in V(H)$ that is incident with two edges $e_{i_y} = xy$ and $e_{i_z} = xz$ in H. Let $G_{i_y} = G(i_y, v_x, v_y)$ and $G_{i_z} = G(i_z, w_x, w_z)$. In $G_{i_y} \cup G_{i_z}$, if (i) $v_x = w_x$, then there is a component C not containing t, but containing $v_x = w_x$, v_y, and w_z, while if (ii) $v_x \neq w_x$, then t is connected to all vertices of V_x, and hence also all vertices of V_y and V_z. Now suppose for a contradiction that the edges in K corresponding to the graphs $\mathcal{G}_i$ do not give a solution to (H, K). Then, for at least one vertex $x \in V(H)$, case (ii) has to apply, which allows us to conclude that $\bigcup_{i \in [a]} G_i$ was in fact connected as long as K is connected (which can be easily ensured), a contradiction.

The main technical contribution of the following theorem is a fairly non-trivial construction that emulates the requirement that we have to choose one graph from each part of the partition $(\mathcal{G}_1, \ldots, \mathcal{G}_a)$. To do so, we equip each graph $G_i \in \mathcal{G}_i$ with a set of *padding edges* with the following properties.

1. For each $i \in [a]$ and distinct $G_{i_1}, G_{i_2} \in \mathcal{G}_i$, the padding edges in $G_{i_1} \cup G_{i_2}$ connect the entire vertex set W.

2. For all $G_1 \in \mathcal{G}_1, \ldots, G_a \in \mathcal{G}_a$, the padding edges in $\bigcup_{i \in [a]} G_i$ leave W (highly) disconnected.

To make this construction work, we have to add many more vertices to W, and an additional challenge is to keep the size of W sufficiently small to obtain the desired lower bound. In particular, to obtain a lower bound of the form $(np)^{o(\log n / \text{polyloglog}(n))}$ for GLOBAL LABEL MIN-CUT, we require the dependence of the size of W on a, the number of edges in the pattern graph, to be no more than $2^{\mathcal{O}(a \text{polylog}(a))}$.

Theorem 3. *There is a reduction that given a* PARTITIONED SUBGRAPH ISOMORPHISM *instance* (H, K), *where H is connected, and with $|V(H)| = h$, $|E(H)| = a$, and n being the number of vertices in each part of the partition of $V(K)$, constructs an equivalent* DUAL GLOBAL LABEL MIN-CUT *instance* $(W, G_1, \ldots, G_p, a)$ *with* $|W| = \mathcal{O}(h(4a)^a n)$, *and* $p = |E(K)|$ *in time* $|W|^{\mathcal{O}(1)}$.

PROOF. Let (H, K), n, h, and a be as in the statement of the theorem. We show how to construct an instance $(W, G_1, \ldots, G_p, a)$ of DUAL GLOBAL LABEL MIN-CUT with the claimed properties. Throughout the proof, for $xy \in E(H)$, we let $E(V_x, V_y) = \{v_x v_y \in E(K) \mid v_x \in V_x, v_y \in V_y\}$.

We choose a prime ρ with $\lceil n^{1/a} \rceil < \rho \leq 2\lceil n^{1/a} \rceil$ (whose existence can be shown by elementary number theory) and let $b = 2a$. Then we have that

$$(\rho - 1)^a \geq n \text{ and } \rho^b \geq n^2. \tag{1}$$

We let $\mathbb{F}_\rho$ be the field of integers modulo ρ and let $\mathbb{F}_\rho^\star$ be its multiplicative subgroup, i.e., $\mathbb{F}_\rho^\star = \{1, \ldots, \rho - 1\}$. For each $x \in V(H)$, we fix an arbitrary injective map $f_x \colon V_x \to (\mathbb{F}_\rho^\star)^a$. Note that the existence of such maps is guaranteed by (1). For each $xy \in E(H)$, let $g_{xy} \colon V_x \times V_y \to (\mathbb{F}_\rho^\star)^b$ be the map defined as $g_{xy}(v_x v_y) = f_x(v_x) \circ f_y(v_y)$.

For each $x \in V(H)$, we let $W_x = (\mathbb{F}_\rho \times \{0, \ldots, b\})^a$ and $\widehat{W}_x = (\mathbb{F}_\rho \times \{0\})^a \subseteq W_x$, and we embed f_x into $\widehat{W}_x$ with the following map $\widehat{f}_x \colon V_x \to \widehat{W}_x$. For each $v_x \in V_x$, we let

$$\widehat{f}_x(v_x) = ((f_x(v_x)[1], 0), \ldots, (f_x(v_x)[a], 0)).$$

The vertex set of the DUAL GLOBAL LABEL MIN-CUT instance is $W = \{t\} \cup \bigcup_{x \in V(H)} W_x$, where t is a new vertex.
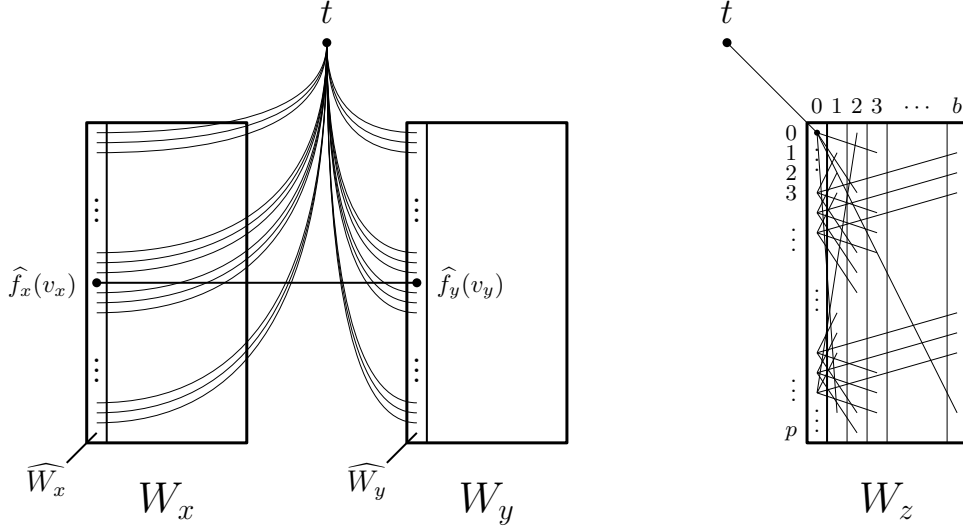
Figure 1: On the left an illustration of a set $A(\alpha, v_x, v_y)$ and on the right an illustration of a padding set $\mathsf{pad}(\alpha, v_x, v_y, z)$. Note that the vector $g_{xy}(v_x, v_y)$ determines the shape of the stars in $\mathsf{pad}(\alpha, v_x, v_y, z)$. We would like to remark though that the picture presented here is drastically simplified and should not be interpreted "literally", as most dimensions of the elements of the sets considered here do not show.

The graphs of the Dual Global Label Min-Cut instance are constructed as follows, see Figure 1 for an illustration. Throughout the following, let $E(H) = \{e_1, \ldots, e_a\}$. Let $\alpha \in [a]$ and $e_\alpha = xy$; for each $v_x v_y \in E(V_x, V_y)$, we let

$$A(\alpha, v_x, v_y) = \{\widehat{f}_x(v_x), \widehat{f}_y(v_y)\} \cup \left\{ tz \mid z \in (\widehat{W}_x \cup \widehat{W}_y) \setminus \{\widehat{f}_x(v_x), \widehat{f}_y(v_y)\}. \right\} \qquad (2)$$

We obtain the graph $G(\alpha, v_x, v_y)$ by taking $A(\alpha, v_x, v_y)$ and adding the following set of "padding" edges $\mathsf{pad}(\alpha, v_x, v_y, z)$ for each $z \in V(H)$, containing the following edges between vertices of $W_z \cup \{t\}$:

$$\forall q_1, \ldots, q_{\alpha-1}, q_{\alpha+1}, \ldots, q_a \in \mathbb{F}_\rho \times \{0, \ldots, b\} \colon \{t, (q_1, \ldots, q_{\alpha-1}, (0,0), q_{\alpha+1}, \ldots, q_a)\}, \text{ and}$$
$$\forall q_1, \ldots, q_{\alpha-1}, q_{\alpha+1}, \ldots, q_a \in \mathbb{F}_\rho \times \{0, \ldots, b\} \quad \forall r \in \mathbb{F}_\rho \quad \forall i \in [b] \colon$$
$$\{(q_1, \ldots, q_{\alpha-1}, (r, 0), q_{\alpha+1}, \ldots, q_a), (q_1, \ldots, q_{\alpha-1}, (r + g_{xy}(v_x, v_y)[i], i), q_{\alpha+1}, \ldots, q_a)\} \qquad (3)$$

Note that the edges described in (3) form a star whose center is in $\widehat{W}_z$. In particular, for each $r \in \mathbb{F}_\rho$, we construct one star whose center is $(r, 0)$; the shape of each such star is determined by the vector $g_{xy}(v_x, v_y)$. The (edge set of the) graph $G(\alpha, v_x, v_y)$ is then obtained as:

$$G(\alpha, v_x, v_y) = A(\alpha, v_x, v_y) \cup \bigcup_{z \in V(H)} \mathsf{pad}(\alpha, v_x, v_y, z)$$

We make an easy but crucial observation.

Observation 4. *Let $\alpha \in [a]$, $e_\alpha = xy$, and $v_x v_y \in E(V_x, V_y)$. Then, $G(\alpha, v_x, v_y)$ spans W.*

The graphs $G(\alpha, v_x, v_y)$ make up the graphs of the Dual Global Label Min-Cut instance; note we have $|E(K)|$ of them and therefore $m = |E(K)|$ as required by the statement of the theorem. Moreover, since $b = 2a$ and $\rho \leq 2\lceil n^{1/a} \rceil$, we have that

$$|W| = 1 + h(\rho(b+1))^a = 1 + h\rho^a(2a+1)^a = \mathcal{O}(h(2n^{1/a})^a(2a)^a) = \mathcal{O}(h2^a(2a)^a n) = \mathcal{O}(h(4a)^a n),$$

as claimed. It remains to prove the correctness of the reduction.

Claim 1. *Let $\alpha \in [a]$, $e_\alpha = xy$, and $v_x v_y, v'_x v'_y \in E(V_x, V_y)$ be distinct. Then, $G(\alpha, v_x, v_y) \cup G(\alpha, v'_x, v'_y)$ is connected.*

5

PROOF OF CLAIM. We show that for each vertex $w \in W \setminus \{t\}$, there is a path from w to t using only the edges from $G(\alpha, v_x, v_y) \cup G(\alpha, v'_x, v'_y)$. To do that, let $z \in V(H)$ be such that $w \in W_z$, and let $q_1, \ldots, q_a \in \mathbb{F}_\rho \times \{0, \ldots, b\}$ be such that $w = (q_1, \ldots, q_a)$. We will be able to "move within" the α-th coordinate of $(q_1, \ldots, q_a)$, and for better readability we introduce the following shorthand: for $(r, \beta) \in \mathbb{F}_\rho \times \{0, \ldots, b\}$, we let $(r, \beta)_\alpha$ stand for $(q_1, \ldots, q_{\alpha-1}, (r, \beta), q_{\alpha+1}, \ldots, q_a)$. Let $q_\alpha = (r, \beta)$. We will show that we can reach $(0, 0)_\alpha$, which is adjacent to t, from $w = (r, \beta)_\alpha$.

Let $g_{xy}(v_x v_y) = (r_1, \ldots, r_b)$, $r_0 = 0$, $g_{xy}(v'_x v'_y) = (r'_1, \ldots, r'_b)$, and $r'_0 = 0$. Since $v_x v_y$ and $v'_x v'_y$ are distinct, there is some $\gamma \in [b]$ such that $r_\gamma \neq r'_\gamma$. Let $r^* = r - r_\beta$ and note that in the case $\beta \neq 0$, there is an edge between $(r, \beta)_\alpha$ and $(r^*, 0)_\alpha$ in $G(a, v_x, v_y)$. Moreover, the following is a path in $G(\alpha, v_x, v_y) \cup G(\alpha, v'_x, v'_y)$:

$$(r^*, 0)_\alpha, (r^* + r_\gamma, \gamma)_\alpha, (r^* + r_\gamma - r'_\gamma, 0)_\alpha, (r^* + 2r_\gamma - r'_\gamma, \gamma)_\alpha, (r^* + 2(r_\gamma - r'_\gamma), 0)_\alpha, \ldots \quad (4)$$

Since ρ is prime and $r_\gamma \neq r'_\gamma$, the path shown in (4) eventually leads to $(0, 0)_\alpha$, and therefore to t. ⌟

Claim 2. *For each $\alpha \in [a]$ with $e_\alpha = x_\alpha y_\alpha$, let $v_{x_\alpha} v_{y_\alpha} \in E(V_{x_\alpha}, V_{y_\alpha})$. If $\mathcal{G} = \bigcup_{\alpha \in [a]} G(\alpha, v_{x_\alpha}, v_{y_\alpha})$ is disconnected, then (H, K) is a YES-instance.*

PROOF OF CLAIM. To prove the claim, we need to show that $U = \{v_{x_\alpha} \mid \alpha \in [a]\}$ contains exactly one vertex from each V_x where $x \in V(H)$. The existence of the necessary edges to show the H-subgraph in K is then guaranteed by the choice made in the statement of the claim. To do so, we show that all vertices in $\widehat{U} = \{\widehat{f_{x_\alpha}}(v_{x_\alpha}) \mid \alpha \in [a]\}$ are in the same connected component of $\mathcal{G}$, and in particular that this component does not contain t. This yields the desired property, for if there was some $x \in V(H)$ such that $|U \cap V_x| \geq 2$, then for $\widehat{W_x}$, $\left|\widehat{U} \cap \widehat{W_x}\right| \geq 2$, which implies that for some distinct $\alpha, \alpha' \in [a]$, and distinct $v_x, v'_x \in V_x$, $\mathcal{G}$ contains both $G(\alpha, v_x, v_y)$ and $G(\alpha', v'_x, v_{y'})$ for some $y \neq y'$ and $v_y \in V_y$ and $v_{y'} \in V_{y'}$. As $\widehat{f_x}(v_x)$ is adjacent to t in $G(\alpha', v'_x, v_{y'})$ since $v_x \neq v'_x$, we obtain a contradiction.

Since $\mathcal{G}$ is disconnected, there is some $z \in V(H)$ such that W_z contains a vertex that is in a different connected component than t; call this vertex $\mathbf{w}_0$. We show that $\mathbf{w}_0$ has a path to a vertex in $\widehat{W_z}$ in $\mathcal{G}$. For all $\alpha \in [a]$, we let $g_{x_\alpha y_\alpha}(v_{x_\alpha}, v_{y_\alpha}) = (r_1^\alpha, \ldots, r_b^\alpha)$ and $r_0^\alpha = 0$. Let $\mathbf{w}_0 = ((s_1, \gamma_1), \ldots, (s_a, \gamma_a)) \in \mathbb{F}_\rho \times \{0, \ldots, b\}$. For all $\alpha \in [a]$, let

$$\mathbf{w}_\alpha = ((s_1 - r_{\gamma_1}^1, 0), \ldots, (s_\alpha - r_{\gamma_\alpha}^\alpha, 0), (s_{\alpha+1}, \gamma_{\alpha+1}), \ldots, (s_a, \gamma_a)).$$

Then, $\mathbf{w}_0, \mathbf{w}_1, \ldots, \mathbf{w}_a$ is a path in $\mathcal{G}$, since for all $\alpha \in [a]$, the edge $\mathbf{w}_{\alpha-1} \mathbf{w}_\alpha$ is in $G(\alpha, v_{x_\alpha}, v_{y_\alpha})$. Moreover, $\mathbf{w}_a \in \widehat{W_z}$. For each α such that z is an endpoint of $e_\alpha = x_\alpha y_\alpha$, assume $z = x_\alpha$, only one vertex in $\widehat{W_z}$ is not adjacent to t in $G(\alpha, x_\alpha, y_\alpha)$; namely $\widehat{f_{x_\alpha}}(v_{x_\alpha})$, see (2). Therefore, $\mathbf{w}_a = \widehat{f_{x_\alpha}}(v_{x_\alpha})$ for *every* such α. This allows us to furthermore conclude that $\widehat{f_{y_\alpha}}(v_{y_\alpha})$, which is connected to $\widehat{f_{x_\alpha}}(v_{x_\alpha})$ by an edge in $G(\alpha, v_{x_\alpha}, v_{y_\alpha})$, see (2), is in the same connected component as $\mathbf{w}_0$, and therefore disconnected from t. Since H is connected, we can iteratively apply this argument to conclude that all vertices in $\widehat{U} = \{\widehat{f_{x_\alpha}}(v_{x_\alpha}) \mid \alpha \in [a]\}$ are in the same connected component of $\mathcal{G}$ that does not contain t, which concludes the proof by the above discussion. ⌟

Claim 3. *If $\{v_x \mid x \in V(H)\}$ is an H-subgraph in K, then $\widehat{U} = \{\widehat{f_x}(v_x) \mid x \in V(H)\}$ are disconnected from t in $\mathcal{G} = \bigcup_{\alpha \in [a], e_\alpha = x_\alpha y_\alpha} G(\alpha, v_{x_\alpha}, v_{y_\alpha})$.*

PROOF OF CLAIM. Suppose for a contradiction that there is some vertex in $\widehat{U}$ that is connected by a path to t. In particular, let $x \in V(H)$ be such that there is such a vertex $\widehat{f_x}(v_x) = \widehat{u}$ that has a path P to t in $\mathcal{G}$ that is fully contained in $W_x \cup \{t\}$. Let $\mathbf{w}$ be the other endpoint of the edge in P that is incident with t. By construction we have that $\mathbf{w} \neq \widehat{u}$. Let $\mathbf{w} = ((r_1, 0), \ldots, (r_a, 0))$ and $\widehat{u} = ((r_1^*, 0), \ldots, (r_a^*, 0))$. Since $\mathbf{w} \neq \widehat{u}$, there is some $\alpha \in [a]$ such that $r_\alpha \neq r_\alpha^*$. Since there is no edge $v'_{x_\alpha} v'_{y_\alpha} \in E(V_x, V_y)$ with $v_{x_\alpha} v_{y_\alpha} \neq v'_{x_\alpha} v'_{y_\alpha}$ such that $G(\alpha, v'_{x_\alpha}, v'_{y_\alpha}) \subseteq \mathcal{G}$, we conclude that all vertices reachable from $\mathbf{w}$ in $\mathcal{G}$ have the value $(r_\alpha + g_{x_\alpha y_\alpha}(v_{x_\alpha} v_{y_\alpha})[\beta], \beta)$ for some $\beta \in \{0, \ldots, b\}$ in the α-th coordinate, which means in particular for $\beta = 0$ that their α-th coordinate is $(r_\alpha, 0) \neq (r_\alpha^*, 0)$, and therefore such a path P cannot exist, a contradiction. ⌟

To conclude the correctness proof, suppose that (H, K) is a Yes-instance. Then Claim 3 implies that $(W, G_1, \ldots, G_p, a)$ is a Yes-instance as well. Conversely, suppose that $(W, G_1, \ldots, G_p, a)$ is a Yes-instance. By Claim 1, we know that a solution to the instance must be of the form as required by Claim 2 which yields that (H, K) is a Yes-instance. $\square$

The following theorem due to Marx [21] has an immediate consequence for the parameterized complexity of DUAL GLOBAL LABEL MIN-CUT.

Theorem 5 (Marx [21]). *PARTITIONED SUBGRAPH ISOMORPHISM where the host graph has n vertices and the pattern graph h edges parameterized by h is W[1]-hard, and cannot be solved in time $f(h)n^{o(h/\log h)}$, for any computable function f, unless the ETH fails.*

Note that we can always assume that H is connected in the previous theorem. If not, then we add a new color class to H containing a single universal vertex x_H, as well as a universal vertex x_K to K with $V_{x_H} = \{x_K\}$. Therefore, Theorems 3 and 5 imply:

Theorem 2. *The GLOBAL LABEL MIN-CUT problem, parameterized by $a := p - k$, the number of uncut colors, is W[1]-hard. Furthermore, unless the Exponential Time Hypothesis fails, there is no computable function f and an algorithm for the problem with running time bound $f(a) \cdot (np)^{o(a/\log a)}$.*

To obtain the quasi-polynomial time lower bound for GLOBAL LABEL MIN-CUT under the ETH, we need some more ingredients proved in Section 3; and we wrap it up in Section 4.

3 Fine-tuned reduction of Marx

Two connected subgraphs H_1 and H_2 of a graph H *touch* if they share a vertex or if there is an edge of H with one endpoint in $V(H_1)$ and another endpoint in $V(H_2)$. An *embedding* of a graph G in a graph H is an assignment ϕ that assigns to every $v \in V(G)$ a nonempty connected subgraph $\phi(v)$ in H (called the *branch set of v*) such that for every edge $uv \in E(G)$, the subgraphs $\phi(u)$ and $\phi(v)$ touch. The *depth* of an embedding ϕ is $\max_{x \in V(H)} |\{v \in V(G) \mid x \in \phi(v)\}|$, that is, the maximum number of subgraphs $\phi(v)$ that meet in a single vertex of H. In literature, such embeddings are sometimes also called *congested minor models* and depth is called *congestion* or *ply*.

In this section, we prove the following version of Theorem 3.1 of [21]. This version is more fine-tuned for our application: a lower bound for binary CSPs parameterized by the number of constraints.

Theorem 6. *There exists a constant C and a polynomial-time algorithm that, given an integer $k \geqslant 2$ and a graph G with n vertices and m edges, outputs a graph H with $|V(H)| + |E(H)| \leqslant k$ and an embedding of G into H. Furthermore, with probability at least 0.5, the depth of the output embedding is at most*

$$C(1 + k^{-1}(n + m)) \cdot \log k.$$

We start with a few simplification steps in the proof of Theorem 6.

1. For $k = \mathcal{O}(1)$, the algorithm of Theorem 6 can output just a single-vertex graph H and embed the whole G into that vertex. Thus, we can assume k is larger than any fixed constant, in particular, $k \geqslant 8$.

2. Our constructed graph H will have maximum degree at most 3 and will satisfy $|V(H)| \geqslant \lfloor k/4 \rfloor \geqslant k/8$, so any isolated vertices of G can be equidistributed across the vertices of H adding at most $1 + k^{-1}n$ to the congestion. Thus, we can assume G has no isolated vertices.

3. We can assume G has maximum degree at most 3: for every vertex v of degree $\deg_G(v) > 3$, we replace v with a cycle on $\deg_G(v)$ vertices and equidistribute the edges incident to v among the vertices of the cycle. This increases $n + m$ by at most a factor of 3. Since the original graph G is a minor of the modified one, it suffices to find an embedding of the modified graph.

4. Finally, if at this point $n + m \leqslant k$, then we can output $H = G$ (and, if $|V(G)| < \lfloor k/4 \rfloor$, a number of isolated vertices to have $\lfloor k/4 \rfloor$ vertices in G as it is required in Point 2 above) and a trivial embedding, so assume $n + m > k$.

In what follows we will need the following variant of Chernoff bound (this is the same as used in [21]).

Theorem 7 (Chernoff bound). *Let $X_1, \ldots, X_n$ be independent random variables with values in $\{0, 1\}$. Let $X = \sum_{i=1}^{n} X_i$, let $\mu = \mathbb{E}X$, and let $r \geqslant \mu$ be a real. Then,*

$$\Pr(X > \mu + r) \leqslant \left(\frac{e\mu}{r}\right)^r.$$

PROOF. The case $\mu = 0$ is trivial, so assume $\mu > 0$. Let $\delta = r/\mu \geqslant 1$. We use the standard Chernoff bound as in Wikipedia:

$$\Pr(X > (1 + \delta)\mu) \leqslant \left(\frac{e^\delta}{(1 + \delta)^{1+\delta}}\right)^\mu.$$

Hence

$$\Pr(X > \mu + r) = \Pr(X > (1 + \delta)\mu) \leqslant \left(\frac{e^\delta}{(1 + \delta)^{1+\delta}}\right)^\mu$$

$$\leqslant \left(\frac{e}{1 + \delta}\right)^{\mu(1+\delta)} = \left(\frac{e\mu}{\mu + r}\right)^{\mu+r} \leqslant \left(\frac{e\mu}{r}\right)^r. \qquad \square$$

3.1 Concurrent flows and expanders

For a constant $\alpha > 0$, we say that a (multi)graph H is an α-*expander* if for every nonempty set $S \subseteq V(G)$ of size at most $|V(G)|/2$, the number of edges with exactly one endpoint in S is at least $\alpha|S|$. For the choice of the graph H in Theorem 6, we can rely on any polynomial-time construction of a constant-degree expander. We encapsulate it in the following standard claim.

Theorem 8. *There exists a constant $\delta > 0$ and polynomial-time algorithm that, given an integer $\ell \geqslant 1$, outputs a (simple) graph H_ℓ on ℓ vertices that is a δ-expander and has maximum degree at most 3.*

Obtaining a multicommodity flow. We now recall some tools from [13, 21]. Fix a graph G and a set $W \subseteq V(G)$. A pair (A, B) is a *separation* in G if $A \cup B = V(G)$ and there is no edge between $A \setminus B$ and $B \setminus A$. The *sparsity* of (A, B) (w.r.t. W) is defined as

$$\alpha^W(A, B) = \frac{|A \cap B|}{|A \cap W| \cdot |B \cap W|}.$$

Let $\alpha^W(G)$ be the minimum sparsity of a separation of G. We have the following observation.

Lemma 9. *If G is a δ-expander of maximum degree 3, then*

$$\alpha^{V(G)}(G) \geqslant \frac{\delta}{3 + \delta}|V(G)|^{-1}.$$

PROOF. Let (A, B) be a separation of G. Without loss of generality assume that $|A| \leqslant |B|$, so $|A \setminus B| \leqslant |V(G)|/2$ Then, by the expander property for the set $A \setminus B$ we have

$$\delta|A \setminus B| \leqslant 3|A \cap B|.$$

Hence

$$|A| \leqslant \frac{3 + \delta}{\delta}|A \cap B|.$$

Therefore

$$\alpha^{V(G)}(A, B) = \frac{|A \cap B|}{|A| \cdot |B|} \geqslant \frac{\delta}{3 + \delta} \cdot |V(G)|^{-1}. \qquad \square$$

For the definition of a concurrent flow and sampling from it, we follow the notation of [14].

A *flow* $f_{u,v}$ between two vertices u and v is a weighted collection of pairwise distinct paths between u and v (if $u = v$ then the only possible path in $f_{u,v}$ has length 0). The *units of flow sent through a vertex* w by $f_{u,v}$ is the sum of the weights of the paths in $f_{u,v}$ that contain w. The *value* of $f_{u,v}$ is defined as the units of flow sent through u (or equivalently through v). A *concurrent flow of value ν and congestion γ* is a collection of $|V(G)|^2$ flows $(f_{u,v})_{(u,v) \in V(G) \times V(G)}$ such that

- $f_{u,v}$ sends exactly ν units of flow from u to v; and

- for each vertex w the total flow over all flows $f_{u,v}$ sent through w is at most γ.

We need the following relation between the minimum possible congestion and sparsity of separations.

Theorem 10 ([6, 19]). *Let γ be the minimum possible congestion of concurrent flow of value 1 in a graph G. Then there exists a separation with sparsity (w.r.t. $|V(G)|$) $\mathcal{O}(\log |V(G)|/\gamma)$.*

From the above toolbox we obtain the following corollary.

Lemma 11. *There exists a constaint c and an algorithm that, given an integer $\ell \geqslant 2$, in time polynomial in ℓ computes a graph H with ℓ vertices and maximum degree at most 3 and a concurrent flow $(f_{u,v})_{u,v \in V(H)}$ of congestion at most $c\ell \log \ell$.*

PROOF. Apply Theorem 8 to ℓ, obtaining a δ-expander $H := H_\ell$. As $\alpha^{V(H)}(H) \geqslant \frac{\delta}{3+\delta} \ell^{-1}$ by Lemma 9, there exists in H a concurrent flow of value 1 and congestion at most $c\dot\ell \log \ell$ for some universal constant c (that depends on δ and the constants hidden in Theorem 10). Finally, we observe that the problem of finding a concurrent flow of value 1 and minimum possible congestion can be formulated as a linear program, and therefore solved in polynomial time. $\square$

3.2 Constructing the embedding

We set $\ell := \lfloor k/4 \rfloor$; note that $\ell \geqslant 2$ due to the assumption $k \geqslant 8$. We apply Lemma 11 to ℓ obtaining a graph H on ℓ vertices and maximum degree at most 3, and a concurrent flow $(f_{u,v})_{u,v \in V(H)}$ of value 1 and congestion at most $c\ell \log \ell \leqslant ck \log k$. Without loss of generality, we assume $c \geqslant 1$. Note that $|V(H)| + |E(H)| \leqslant \ell + 3\ell \leqslant k$ as desired. We assume $V(H) = [\ell]$ for ease of the notation.

Similarly as in [13, 14, 21], we treat $f_{u,v}$ as a probability distribution over paths from u to v: the probability of choosing a path equals the amount of flow passed along this path (its weight in $f_{u,v}$).

The crucial tool for the construction is the following analysis.

Claim 4. *Fix integer $p \geqslant 1$ and consider the following random process. For every $x \in V(H)$ and every $i \in [p]$, randomly pick (with uniform distribution) a vertex $Y(x,i) \in V(H)$ and then a path $P(x,i)$ from x to $Y(x,i)$ according to the distribution $f_{x,Y(x,i)}$. Then, with probability more than 0.9, the family of paths $\mathcal{P} = \{P(x,i) \mid x \in V(H), i \in [p]\}$ has congestion at most $10cp \log \ell$ (where c is the constant from Lemma 11).*

PROOF. Fix a vertex $w \in V(H)$. For $x, y \in V(H)$, let $f_{x,y}(w)$ be the amount of flow $f_{x,y}$ that passes through w. For $x \in V(H)$ and $i \in [p]$, let $X(x,i)$ be a $\{0,1\}$-valued random variable indicating if w lies on the path $P(x,i)$. Then,

$$\mathbb{E}X(x,i) = \Pr(X(x,i) = 1) = \ell^{-1} \sum_{y \in V(H)} \Pr(w \in P(x,i) \mid Y(x,i) = y) = \ell^{-1} \sum_{y \in V(H)} f_{x,y}(w).$$

Hence, if we define $X = \sum_{x \in V(H)} \sum_{i=1}^{p} X(x,i)$, we have

$$\mathbb{E}X = \sum_{x \in V(H)} \sum_{i=1}^{p} \ell^{-1} \sum_{y \in V(H)} f_{x,y}(w) = p\ell^{-1} \sum_{x,y \in V(H)} f_{x,y}(w) \leqslant p\ell^{-1} c\ell \log \ell = cp \log \ell.$$

By the aforementioned version of the Chernoff bound (Theorem 7), we obtain:

$$\Pr(X > 10cp\log\ell) \leqslant \left(\frac{e\mathbb{E}X}{9cp\log\ell}\right)^{9cp\log\ell} \leqslant e^{-9cp\log\ell} \leqslant \ell^{-9} < \ell^{-1}/10.$$

In the last two inequalities we used $c \geqslant 1$ and $\ell \geqslant 2$. Thus, by the union bound, the probability that for every $w \in V(H)$ at most $10cp\log\ell$ of $\mathcal{P}$ pass through w is at least 0.9. $\qquad\square$

We are now ready to construct the final embedding ϕ of G into H. We start with splitting $V(G)$ into ℓ buckets of as equal as possible sizes: that is, we take arbitrary $\zeta : V(H) \to [\ell]$ such that for every $a \in [\ell]$, we have $|\zeta^{-1}(i)| \in \{\lfloor |V(G)|/\ell\rfloor, \lceil |V(G)|/\ell\rceil\}$. We start by setting $\phi(v) = \{\zeta(v)\}$ for every $v \in V(G)$. Then, for every $xy \in E(G)$ with $\zeta(x) \neq \zeta(y)$ we proceed as follows:

1. Uniformly at random pick $Z(xy) \in V(H)$.

2. Sample a path $P(xy, x)$ from x to $Z(xy)$ according to the distribution $f_{x,Z(xy)}$ and add $P(xy, x)$ to $\phi(x)$.

3. Sample a path $P(xy, y)$ from y to $Z(xy)$ according to the distribution $f_{y,Z(xy)}$ and add $P(xy, y)$ to $\phi(y)$.

Clearly, the constructed ϕ is indeed an embedding of G to H. It remains to bound its congestion.

To this end, consider a vertex $w \in V(H)$ and a vertex $v \in V(G)$ such that $w \in \phi(v)$. We say that w is a *type-0* member of $\phi(v)$ if $\zeta(v) = w$. We say that w is a *type-1* member of $\phi(v)$ if for some $xy \in E(G)$, with $\zeta(x) < \zeta(y)$, w lies on the sampled path $P(xy, x)$. We say that w is a *type-2* member of $\phi(v)$ if for some $xy \in E(G)$, with $\zeta(x) < \zeta(y)$, w lies on the sampled path $P(xy, y)$. Note that w is a member of at least one type (but possibly of multiple types).

A fixed vertex w is a type-0 member of exactly $|\zeta^{-1}(w)| \leqslant 1 + \ell^{-1}n$ sets $\phi(v)$. The crucial observation is that the number sets $\phi(v)$ which w is a type-1 member can be bounded using Claim 4 with $p := 3(1 + \ell^{-1}n)$. Indeed, for a fixed vertex $a \in V(H)$, let $E_{a,1}$ be the set of edges $xy \in E(G)$ with $a = \zeta(x) < \zeta(y)$. Since G is of maximum degree at most 3, we have $|E_{a,1}| \leqslant 3|\zeta^{-1}(a)| \leqslant 3(1 + \ell^{-1}n) = p$. Crucially, the sets $(E_{a,1})_{a \in V(H)}$ are pairwise disjoint. Thus, the choices of $Z(xy)$ for all $a \in V(H)$ and $xy \in E_{a,1}$ are independent. Hence, by Claim 4, with probability at least 0.9 every $w \in V(H)$ is a type-1 member of at most $10cp\log\ell$ sets $\phi(v)$.

A symmetrical argument shows that with probability at least 0.9 every $w \in V(H)$ is a type-2 member of at most $10cp\log\ell$ sets $\phi(v)$. Consequently, by the union bound, with probability at least 0.8 the congestion of the constructed embedding is at most

$$(1 + \ell^{-1}n) + 2 \cdot 10c(1 + \ell^{-1}n)\log\ell \leqslant (20c + 1)(1 + \ell^{-1}n)\log\ell \leqslant (120c + 6)(1 + k^{-1}n)\log k.$$

In the last inequality we used $\ell = \lfloor k/4 \rfloor$ and $k \geqslant 8$. This finishes the proof of Theorem 6.

4 Wrapping up the proof of quasi-polynomial time lower bound

In this section we finish the proof of the quasi-polynomial time lower bound for Global Label Min-Cut under the ETH. The lower bound follows from an application of Theorem 3 to an instance of Partitioned Subgraph Isomorphism constructed from a 3-Cnf-Sat instance with the use of Theorem 6 with appropriate choices of parameters along the way.

Theorem 1. *Unless the Exponential Time Hypothesis fails, there is no algorithm solving* Global Label Min-Cut *in time* $(np)^{o(\log n/(\log\log n)^2)}$.

Proof. We give a randomized reduction from 3-Cnf-Sat to Dual Global Label Min-Cut that succeeds with probability at least 0.5.

Let ψ be the 3-CNF-SAT instance on N variables and M clauses, and let G be the incidence graph of ψ. We call the vertices in G corresponding to variables in ψ the *variable vertices*, and the vertices corresponding to clauses *clause vertices*. Let G be the incidence graph of ψ and let $k = \sqrt{N + M}$. We apply Theorem 6 on (G, k). Let C be its constant, let H be the graph, and ϕ the embedding of G in H returned by its algorithm. If the congestion of this embedding is more than $C(1 + k^{-1}(N + M)) \cdot \log k$, then we report fail. Note that this happens with probability less than 0.5. We may therefore assume that the congestion $\mathsf{con}(\phi)$ of the embedding ϕ is at most

$$\mathsf{con}(\phi) \leq C(1 + k^{-1}(N + M)) \cdot \log k = \mathcal{O}(\sqrt{N + M} \log \sqrt{N + M}), \tag{5}$$

and that

$$|V(H)| + |E(H)| \leq k = \sqrt{N + M}. \tag{6}$$

We now give a CSP instance ψ_H that will be transformed to a PARTITIONED SUBGRAPH ISOMORPHISM instance to which we can apply Theorem 3. To obtain ψ_H, first think of ψ as a CSP instance ψ_G on the graph G, in the following way: The variables of ψ_G are the vertices of G. The domain of each variable corresponding to a variable vertex is $\{\mathsf{true}, \mathsf{false}\}$, and the domain of each variable corresponding to a clause vertex it is $[\ell]$, where $\ell \leq 3$ is the number of literals in that clause. Each truth assignment to the variables of ψ naturally corresponds to a valuation of the variable vertex variables in ψ_G. If such an assingment is a satifying assignment, then each clause vertex variable can receive a value that indicates which literal satisfies it. To ensure that this only works when a truth assignment is indeed satisfying, we add the following constraints to ψ_G: Let vz be an edge in G, where v is a variable vertex and z is a clause vertex, corresponding to a clause of size ℓ. Suppose v is the i-th literal in z. Let $R = \{\mathsf{true}, \mathsf{false}\} \times [\ell] \setminus \{i\}$. If v occurs positively in z, then we add a constraint $((v, z), R \cup \{(\mathsf{true}, i)\})$ to ψ_G, and if v occurs negated, we add $((v, z), R \cup \{(\mathsf{false}, i)\})$ to ψ_G. In either case, we denote the corresponding constraint by $((v, z), R_{vz})$. Such constraints ensure that if z receives value i, then in any satisfying valuation, the truth assignment of v satisfies z.

We now obtain ψ_H, a CSP instance on H, from ψ_G by "routing ϕ through ψ_G". Concretely, this means the following. For each $v \in V(G)$, let D_v denote the domain of v in ψ_G. The domain of each $w \in V(H)$ in ψ_H is $\mathcal{D}_w = \bigtimes_{v \in V(G), w \in V(\phi(v))} D_v$. This way, assigning a value to w in ψ_H corresponds to assigning values to all $v \in V(G)$ that are mapped to w via ϕ. We add two types of constraints. The first one simply checks that if a vertex $v \in V(G)$ is mapped to several vertices in H, then the valuation stays consistent. The second one checks the edge constraints from ψ_G at a point where the subgraphs of their endpoints in H touch. We use the following notation. For $w \in V(H)$, $(v_1, \ldots, v_r) \in V(G)^r$ with $w \in \bigcap_{j \in [r]} V(\phi(v_j))$, and $(i_1, \ldots, i_r) \in D_{v_1} \times \ldots \times D_{v_r}$, we let $\mathcal{D}_w[v_1 \leftarrow i_1, \ldots, v_r \leftarrow i_r]$ denote the subset of $\mathcal{D}_w$, where for all $j \in [r]$, value of v_j is fixed to i_j. In the following, domains of variables in ψ_H may get shrunk. For ease of notation, however, we always denote the current domain of $w \in V(H)$ by $\mathcal{D}_w$. We do the following:

1. (*Consistency.*) For each $v \in V(G)$ and each edge $w_1 w_2 \in E(H)$ such that $\{w_1, w_2\} \subseteq V(\phi(v))$, we add a constraint $((w_1, w_2), \bigcup_{i \in D_v} \mathcal{D}_{w_1}[v \leftarrow i] \times \mathcal{D}_{w_2}[v \leftarrow i])$. (Note that since $\phi(v)$ is connected, this indeed ensures consistency.)

2. (*Touching.*) For each $uv \in E(G)$, by definition, $\phi(u)$ and $\phi(v)$ touch. In this part, we ensure that ψ_H respects the constraints in ψ_G imposed by the edges in G. For each $uv \in E(G)$, we do the following.

   (a) (*Vertex Touching.*) Let $w \in V(\phi(u)) \cap V(\phi(v))$; we restrict $\mathcal{D}_w$ to
   $$\bigcup_{(i,j) \in R_{uv}} \mathcal{D}_w[u \leftarrow i, v \leftarrow j].$$

   (b) (*Edge Touching.*) Let $wz \in E(H)$ such that $w \in V(\phi(u))$ and $z \in V(\phi(v))$; we add a constraint $((w, z), \mathcal{R}_{wz}^{uv})$ to ψ_H, where
   $$\mathcal{R}_{wz}^{uv} = \bigcup_{(i,j) \in R_{uv}} \mathcal{D}_w[u \leftarrow i] \times \mathcal{D}_z[v \leftarrow j].$$

11

It follows directly from this description that ψ is satisfiable if and only if ψ_H is. Moreover, the domain of each variable in ψ_H (vertex in H) is at most $3^{\mathsf{con}(\phi)}$.

We now transform ψ_H into a PARTITIONED SUBGRAPH ISOMORPHISM instance (H, K); H serves as the pattern graph and K is obtained as follows. For all $x \in V(H)$, we add a set of vertices $V_x = \mathcal{D}_x$ to K, i.e., V_x is the domain of the variable x in ψ_H. This way, a choice of one vertex per V_x naturally corresponds to a valuation of the variables in ψ_H. To ensure that such a choice gives an H-subgraph in K precisely when the corresponding valuation satisfies ψ_H, we add the following edges to H: For each $wz \in E(H)$, and each pair $u \in V_w$, $v \in V_z$, we add the edge uv to K if and only if for each constraint of the form $((w, z), \mathcal{R})$ in ψ_H, $(u, v) \in \mathcal{R}$, in other words, it is satisfied by assigning u to w and v to z. Again it is clear from the description that the instance (H, K) is equivalent to ψ_H, and therefore to ψ. If H is not connected, then we add a new color class to H containing a single universal vertex x_H, as well as a universal vertex x_K to K with $V_{x_H} = \{x_K\}$.

We now apply the reduction of Theorem 3 to (H, K) and obtain an instance $\mathcal{I}$ of DUAL GLOBAL LABEL MIN-CUT. Let $L = N + M$. To analyze the size of $\mathcal{I}$, recall that by (6), $|V(H)| + |E(H)| \le \sqrt{L}$, and since each color class of K has at most $3^{\mathsf{con}(\phi)} = 2^{\mathcal{O}(\sqrt{L} \log \sqrt{L})}$ vertices, we have that $|V(K)| = 2^{\mathcal{O}(\sqrt{L} \log \sqrt{L})}$ and $|E(K)| = 2^{\mathcal{O}(\sqrt{L} \log \sqrt{L})}$, by (5). Let n be the number of vertices in $\mathcal{I}$ and let p be the number of labels. By Theorem 3,

$$ n = \mathcal{O}\Big(|V(H)|(4|E(H)|)^{|E(H)|}|V(K)| \Big) = 2^{\mathcal{O}(\sqrt{L} \log \sqrt{L})}, \text{ and } p = |E(K)| = 2^{\mathcal{O}(\sqrt{L} \log \sqrt{L})}, $$

and $\mathcal{I}$ can be constructed in $2^{\mathcal{O}(\sqrt{L} \log \sqrt{L})}$ time. To conclude, an algorithm for DUAL GLOBAL LABEL MIN-CUT running in $(np)^{o(\log n/(\log \log n)^2)}$ would yield a 3-CNF-SAT algorithm running in

$$ (np)^{o(\log n/(\log \log n)^2)} = \left(2^{\sqrt{L} \log \sqrt{L}}\right)^{o\left(\frac{\sqrt{L} \log \sqrt{L}}{(\log \sqrt{L} + \log \log \sqrt{L})^2}\right)} = 2^{o\left(L \frac{(\log \sqrt{L})^2}{(\log \sqrt{L} + \log \log \sqrt{L})^2}\right)} = 2^{o(N+M)} $$

time, contradicting the ETH. $\qquad\square$

References

[1] Guillaume Blin, Paola Bonizzoni, Riccardo Dondi, Romeo Rizzi, and Florian Sikora. Complexity insights of the minimum duplication problem. *Theor. Comput. Sci.*, 530:66–79, 2014. `doi:10.1016/j.tcs.2014.02.025`.

[2] Augusto Bordini, Fábio Protti, Thiago Gouveia da Silva, and Gilberto Farias de Sousa Filho. New algorithms for the minimum coloring cut problem. *International Transactions in Operational Research*, 26(5):1868–1883, 2019. `doi:https://doi.org/10.1111/itor.12494`.

[3] Li Chen, Rasmus Kyng, Yang P. Liu, Richard Peng, Maximilian Probst Gutenberg, and Sushant Sachdeva. Maximum flow and minimum-cost flow in almost-linear time. *CoRR*, abs/2203.00671, 2022. `arXiv:2203.00671`, `doi:10.48550/arXiv.2203.00671`.

[4] David Coudert, Pallab Datta, Stephane Perennes, Hervé Rivano, and Marie-Emilie Voge. Shared risk resource group complexity and approximability issues. *Parallel Process. Lett.*, 17(2):169–184, 2007. `doi:10.1142/S0129626407002958`.

[5] David Coudert, Stéphane Pérennes, Hervé Rivano, and Marie-Emilie Voge. Combinatorial optimization in networks with shared risk link groups. *Discrete Mathematics and Theoretical Computer Science*, 18(3), 2016. `doi:10.46298/dmtcs.1297`.

[6] Uriel Feige, MohammadTaghi Hajiaghayi, and James R. Lee. Improved approximation algorithms for minimum weight vertex separators. *SIAM J. Comput.*, 38(2):629–657, 2008. `doi:10.1137/05064299X`.

[7] Michael R Fellows, Jiong Guo, and Iyad Kanj. The parameterized complexity of some minimum label problems. *Journal of Computer and System Sciences*, 76(8):727–740, 2010. `doi:10.1016/j.jcss.2010.02.012`.

[8] L. R. Ford and D. R. Fulkerson. Maximal flow through a network. *Canadian Journal of Mathematics*, 8:399–404, 1956. `doi:10.4153/CJM-1956-045-5`.

[9] L. R. Ford and D. R. Fulkerson. Flows in networks. *Princeton University Pres, Princeton, NJ*, 1962.

[10] Yu Gao, Yang P. Liu, and Richard Peng. Fully dynamic electrical flows: Sparse maxflow faster than goldberg-rao. In *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 516–527, 2022. `doi:10.1109/FOCS52979.2021.00058`.

[11] Paweł Gawrychowski, Shay Mozes, and Oren Weimann. Minimum Cut in $O(mlog^2n)$ Time. In Artur Czumaj, Anuj Dawar, and Emanuela Merelli, editors, *47th International Colloquium on Automata, Languages, and Programming (ICALP 2020)*, volume 168 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 57:1–57:15, Dagstuhl, Germany, 2020. Schloss Dagstuhl–Leibniz-Zentrum für Informatik. `doi:10.4230/LIPIcs.ICALP.2020.57`.

[12] Mohsen Ghaffari, David R. Karger, and Debmalya Panigrahi. Random contractions and sampling for hypergraph and hedge connectivity. In Philip N. Klein, editor, *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2017, Barcelona, Spain, Hotel Porta Fira, January 16-19*, pages 1101–1114. SIAM, 2017. `doi:10.1137/1.9781611974782.71`.

[13] Martin Grohe and Dániel Marx. On tree width, bramble size, and expansion. *J. Comb. Theory, Ser. B*, 99(1):218–228, 2009. `doi:10.1016/j.jctb.2008.06.004`.

[14] Meike Hatzel, Pawel Komosa, Marcin Pilipczuk, and Manuel Sorge. Constant congestion brambles. *CoRR*, abs/2008.02133, 2020. URL: `https://arxiv.org/abs/2008.02133`, `arXiv:2008.02133`.

[15] Russell Impagliazzo, Ramamohan Paturi, and Francis Zane. Which problems have strongly exponential complexity? *J. Comput. Syst. Sci.*, 63(4):512–530, 2001. `doi:10.1006/jcss.2001.1774`.

[16] David R. Karger. Global min-cuts in rnc, and other ramifications of a simple min-cut algorithm. In *Proceedings of the Fourth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '93, page 21–30, USA, 1993. Society for Industrial and Applied Mathematics. URL: `https://dl.acm.org/doi/10.5555/313559.313605`.

[17] David R. Karger. Minimum cuts in near-linear time. *J. ACM*, 47(1):46–76, jan 2000. `doi:10.1145/331605.331608`.

[18] Tarun Kathuria, Yang P. Liu, and Aaron Sidford. Unit capacity maxflow in almost $m^{4/3}$ time. *SIAM Journal on Computing*, 0(0):FOCS20–175–FOCS20–204, 2022. `doi:10.1137/20M1383525`.

[19] Frank Thomson Leighton and Satish Rao. Multicommodity max-flow min-cut theorems and their use in designing approximation algorithms. *J. ACM*, 46(6):787–832, 1999. `doi:10.1145/331524.331526`.

[20] Aleksander Madry. Computing maximum flow with augmenting electrical flows. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 593–602, 2016. `doi:10.1109/FOCS.2016.70`.

[21] Dániel Marx. Can you beat treewidth? *Theory Comput.*, 6(1):85–112, 2010. `doi:10.4086/toc.2010.v006a005`.

[22] David W Matula. A linear time 2+ ε approximation algorithm for edge connectivity. In *Proceedings of the fourth annual ACM-SIAM Symposium on Discrete algorithms*, pages 500–504, 1993. URL: https://dl.acm.org/doi/10.5555/313559.313872.

[23] Hiroshi Nagamochi and Toshihide Ibaraki. Computing edge-connectivity in multigraphs and capacitated graphs. *SIAM Journal on Discrete Mathematics*, 5(1):54–66, 1992. doi:10.1137/0405004.

[24] Maurice Queyranne. A combinatorial algorithm for minimizing symmetric submodular functions. In *Proceedings of the Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '95, pages 98–101, USA, 1995. Society for Industrial and Applied Mathematics. URL: https://dl.acm.org/doi/10.5555/313651.313669.

[25] Maurice Queyranne. Minimizing symmetric submodular functions. *Mathematical Programming*, 82(1-2):3–12, 1998. doi:10.1007/BF01585863.

[26] Romeo Rizzi. NOTE - on minimizing symmetric set functions. *Combinatorica*, 20(3):445–450, 2000. doi:10.1007/s004930070017.

[27] Mechthild Stoer and Frank Wagner. A simple min-cut algorithm. *J. ACM*, 44(4):585–591, jul 1997. doi:10.1145/263867.263872.

[28] Linqing Tang and Peng Zhang. Approximating minimum label s-t cut via linear programming. In *Proc. of the 10th Latin American Symposium on Theoretical Informatics (LATIN)*, volume 7256 of *LNCS*, pages 655–666, 2012. doi:10.1007/978-3-642-29344-3_55.

[29] Peng Zhang. Efficient algorithms for the label cut problems. In *Proc. of the 11th Annual Conference on Theory and Applications of Models of Computation (TAMC)*, volume 8402 of *LNCS*, pages 259–270, 2014. doi:10.1007/978-3-319-06089-7_18.

[30] Peng Zhang, Jin-Yi Cai, Lin-Qing Tang, and Wen-Bo Zhao. Approximation and hardness results for label cut and related problems. *Journal of Combinatorial Optimization*, 21(2):192–208, 2011. doi:10.1007/s10878-009-9222-0.

[31] Peng Zhang and Bin Fu. The label cut problem with respect to path length and label frequency. *Theor. Comput. Sci.*, 648:72–83, 2016. doi:10.1016/j.tcs.2016.08.006.

[32] Peng Zhang, Bin Fu, and Linqing Tang. Simpler and better approximation algorithms for the unweighted minimum label s-t cut problem. *Algorithmica*, 80(1):398–409, Jan 2018. doi:10.1007/s00453-016-0265-1.

[33] Peng Zhang and Linqing Tang. Minimum label s-t cut has large integrality gaps. *Information and Computation*, 275:104543, 2020. doi:https://doi.org/10.1016/j.ic.2020.104543.