Inapproximability of H-Coloring Problems

Kamyar Khodamoradi * Akbar Rafiey † Arash Rafiey‡

Abstract

This paper studies the inapproximability of Hypergraph Coloring, HC. We are given two hypergraphs $\mathcal{G}$ and $\mathcal{H}$ (assume the hyperedges are ordered) together with a cost function c, specifying the cost of coloring a given vertex of $\mathcal{G}$ with a given vertex of $\mathcal{H}$. The goal is to find a homomorphism, a.k.a. coloring, from $V(\mathcal{G})$ to $V(\mathcal{H})$ so that it preserves adjacency (the image of every hyperedge in $\mathcal{G}$ is a hyperedge in $\mathcal{H}$) and its cost (sum over individual cost) is minimized. When $\mathcal{H}$ is a fixed target hypergraph, we denote this problem by MHC($\mathcal{H}$). Some prominent problems that this framework captures are (Hypergraph) Vertex Cover, Min Sum k-Coloring, Multiway Cut, Min Ones, and others.

We present the first general hardness of the approximation for MHC. More precisely, we prove that every instance of MHC(H), H being a digraph, is either polynomial-time solvable or **APX**-complete. Moreover, we show the existence of a universal constant $0 < \delta < 1$ such that it is **NP**-hard to approximate MHC(H) within a factor of $(2 - \delta)$ for all digraphs H where MHC(H) is **NP**-complete. We use structural properties of digraph H where MHC(H) is **NP**-complete and develop an array of gap-preserving approximation reductions. The underlying structural properties used in our results can be extended to hypergraphs by considering the obstruction of MHC(H) for digraphs to hypergraphs, yielding hardness results for hypergraphs.

*Computer Science Department, University of Regina, SK, Canada. Email : kamyar.khodamoradi@uregina.ca

†Halıcıoğlu Data Science Institute, University of California San Diego, USA. Email : arafiey@sfu.ca

‡Department of Math and Computer Science, Indiana State University, Indiana, USA. Email: arash.rafiey@indstate.edu.
Research supported in part by NSF grant 1751765.

1 Introduction

Let $\mathcal{G}$ be a hypergraph. We denote the vertex set and the edge set of $\mathcal{G}$ by $V(\mathcal{G})$ and $E(\mathcal{G})$, respectively. However, we assume the hyperedges of $\mathcal{G}$ are ordered, and hence, we refer to them as arcs of $\mathcal{G}$. Thus, (x, y, z) and (z, x, y) are considered two different arcs of $\mathcal{G}$. $\mathcal{G}$ is called k-uniform hypergraph if all the arcs have size k. A digraph is a 2-uniform hypergraph.

A *homomorphism* of hypergraph $\mathcal{D}$ to a hypergraph $\mathcal{H}$, also known as an $\mathcal{H}$-Coloring of $\mathcal{D}$, is a mapping $f : V(\mathcal{D}) \to V(\mathcal{H})$, such that for each arc $(x_1, x_2, \dots, x_k) \in \mathcal{D}$, $(f(x_1), f(x_2), \dots, f(x_k))$ is an arc of $\mathcal{H}$. We say a mapping f does not satisfy arc $(x_1, x_2, \dots, x_k)$, if $(f(x_1), f(x_2), \dots, f(x_k))$ is not an arc of $\mathcal{H}$. The homomorphism problem parameterized by target hypergraphs, denoted HC($\mathcal{H}$), takes a hypergraph $\mathcal{D}$ as input and asks whether there is a homomorphism from $\mathcal{D}$ to $\mathcal{H}$. Therefore, by fixing the hypergraph $\mathcal{H}$ we obtain a class of problems, one for each hypergraph $\mathcal{H}$. For instance, HC(H), when H is an edge, is exactly the problem of determining whether the input graph G is bipartite, known as the 2-Coloring problem. Similarly, if H is a clique on k vertices, then HC(H) is the classical k-Coloring problem.

There are several optimization versions of HC($\mathcal{H}$) problem, two of which have attracted a lot of attention. One is to find a mapping $f : V(\mathcal{D}) \to V(\mathcal{H})$ that maximizes (minimizes) the number of satisfied (unsatisfied) arcs in $\mathcal{D}$. This problem is known under the name of Max CSP (Min CSP); an example is the Max Cut problem where the target graph H is an edge. This line of research has received a lot of attention in the literature and there are very strong results concerning various aspects of approximability of Max CSP and Min CSP [2, 12, 16, 28, 34, 36]. A beautiful result of [36] established optimal (in)approximability for the Max CSP problem where the goal is to find an assignment which maximizes a weighted fraction of satisfied arcs. The author of [36] showed how to use the basic SDP relaxation to obtain a constant factor approximation. Moreover, he proved that the approximation ratio can not be improved under the Unique Games Conjecture (UGC) [27]. It was noted in the same paper [36] that the techniques do no apply to CSPs where *all* the arcs must be satisfied such as Vertex Cover and 3-Coloring problems. This type of problems are known as strict CSPs [33].

The focus of this paper is on an optimization version of the HC problem where we can express problems such as Vertex Cover and 3-Coloring. In this optimization version of HC($\mathcal{H}$) problem, we are not only interested in the existence of a homomorphism (i.e., satisfying *all* the arcs), but want to find the "best homomorphism". The Minimum Hypergraph Coloring problem to $\mathcal{H}$, denoted by MHC($\mathcal{H}$), for a given input hypergraph $\mathcal{D}$, and a cost function $c(x, i), x \in V(\mathcal{D}), i \in V(\mathcal{H})$, seeks a homomorphism f of $\mathcal{D}$ to H that minimizes the total cost $\sum_{x \in V(\mathcal{D})} c(x, f(x))$. The cost function c can take non-negative rational values.

MHC($\mathcal{H}$):

Input:	Hypergraph $\mathcal{D}$, and a cost function $c : V(\mathcal{D}) \times V(\mathcal{H}) \to \mathbb{Q}_{\geq 0}$.
Objective:	Find a *homomorphism* $f : V(\mathcal{D}) \to V(\mathcal{H})$ that minimizes $\sum_{x \in V(\mathcal{D})} c(x, f(x))$.

The MHC problem offers a natural way to model and generalizes many optimization problems.

Example 1.1 (Vertex Cover)**.** This problem can be seen as MHC(H) where $V(H) = \{0, 1\}, E(H) = \{11, 01\}$, and $c(u, 0) = 0$, $c(u, 1) = 1$ for every $u \in V(G)$ where G is the input graph.

For k-Hypergraph Vertex Cover, when the input is a hypergraph $\mathcal{G}$, the target hypergraph $\mathcal{H}$ consists of all the arcs $\{\{0, 1\}^t - (0, \dots, 0), t \leq k\}$, and $c(u, 0) = 0$, $c(u, 1) = 1$ for every $u \in V(\mathcal{G})$ where $\mathcal{G}$ is the input hypergraph.

Example 1.2 (Chromatic Sum)**.** In this problem, we are given a graph G, and the objective is to find a proper coloring of G with colors $\{1, \dots, k\}$ with *minimum color sum.* This can be seen as MHC(H) where H is a clique of size k with $V(H) = \{1, \dots, k\}$ and the cost function is defined as $c(u, i) = i$. The problem Chromatic Sum appears in many applications, such as resource allocation problems [4].

Example 1.3 (Multiway Cut)**.** Let G be a graph where each edge e has a non-negative weight $w(e)$. There are also k specific (terminal) vertices, $s_1, s_2, \dots, s_k$ of G. The goal is to partition the vertices of G into k parts so that each part $i \in \{1, 2, \dots, k\}$, contains s_i and the sum of the weights of the edges between different parts is minimized. Let H be a graph with vertex set $\{a_1, a_2, \dots, a_k\} \cup \{b_{i,j} \mid 1 \leq i < j \leq k\}$. The edge set of H is $\{a_i a_i, a_i b_{i,j}, b_{i,j} a_j, a_j a_j \mid 1 \leq i < j \leq k\}$. Now obtain the graph G' from G by replacing every edge $e = uv$ of G with the edges $ux_e, x_e v$ where x_e is a new vertex. The cost function c is as follows. $c(s_i, a_i) = 0$, else $c(s_i, d) = |G|$ for $d \neq a_i$. For every $u \in G \setminus \{s_1, s_2, \ldots, s_k\}$, set $c(u, s_i) = 0$. Set $c(x_e, b_{i,j}) = w(e)$. Now, finding a minimum multiway cut in G is equivalent to solving $\mathsf{MHC}(H)$ for G' and c.

Example 1.4 (Min-Ones for 3LIN)**.** We are given a set of equations of type $x_{i_1} \oplus x_{i_2} \oplus x_{i_3} = 0/1$. The goal is to solve this system of equations so that the number of variables assigned to 1 is minimized. This is an instance of $\mathsf{MHC}(\mathcal{H})$ where $\mathcal{H} = \{(0,0,1), (0,1,0), (1,0,0), (1,1,1)\}$ and with the cost function $c(x_i, 0) = 0$, and $c(x_i, 1) = 1$.

Example 1.5 (List Hypergraph Coloring (LHC))**.** LHC($\mathcal{H}$), seeks, for a given input hypergraph $\mathcal{D}$ and lists $L(x) \subseteq V(\mathcal{H}), x \in \mathcal{D}$, a homomorphism f from $\mathcal{D}$ to $\mathcal{H}$ such that $f(x) \in L(x)$ for all $x \in \mathcal{D}$. This is equivalent to $\mathsf{MHC}(H)$ (with total cost zero) with $c(u, i) = 0$ if $i \in L(u)$, otherwise, $c(u, i) = 1$. This problem is also known as List H-Coloring.

The MHC problem generalizes many other problems such as (Weighted) Min Ones [1, 6, 26], Min Sol [25, 37], a large class of linear programs of bounded integers, retraction problems [10], Minimum Sum Coloring [4, 11, 32], and various optimum cost chromatic partition problems [15, 23, 24, 31].

We start off with inapproximability of digraphs i.e., hypergraphs with arcs of size two which we believe are the most important instances. Later, we will discuss how to extend our hardness results to hypergraphs. In terms of graphs and digraphs, the complexity of *exact minimization* of $\mathsf{MHC}(H)$ is well-understood. A complete complexity classifications were given in [13] for undirected graphs and in [21] for digraphs. More precisely, the result in [21] states that if H admits a so-called *k-min-max* ordering, then $\mathsf{MHC}(H)$ is polynomial time solvable and otherwise it is **NP**-complete. We will use this characterization in our paper.

There are only a few results concerning (in)approximability of MHC parameterized by a target graph or digraph. The authors of [17] initiated the study of (constant factor) approximation algorithms of $\mathsf{MHC}(H)$. They proved a dichotomy in the case of bipartite graphs. That is, for any fixed bipartite graph H, $\mathsf{MHC}(H)$ is approximable within (constant) factor $|V(H)|$ if H is a *co-circular arc* graph, otherwise $\mathsf{MHC}(H)$ is not approximable unless $\mathbf{P} \neq \mathbf{NP}$. Interestingly, they showed such bipartite graphs can be characterized by the existence of a special type of vertex orderings. A bipartite graph is co-circular arc if and only if it admits a vertex ordering called *min ordering* [17]. This dichotomy result was extended to graphs in [35]. It was shown that for any fixed graph H, $\mathsf{MHC}(H)$ is approximable within (constant) factor $|V(H)|$ if H is a *bi-arc* graph, otherwise $\mathsf{MHC}(H)$ is not approximable unless $\mathbf{P} \neq \mathbf{NP}$. In this paper we provide a lower bound for MHC for graphs and show that for any graph H, $\mathsf{MHC}(H)$ is 1.128-approx hard under the assumption $\mathbf{P} \neq \mathbf{NP}$.

Theorem 1.6. *[Inapproximability for graphs] For every* graph H *where MHC(H) is* ***NP****-complete, it is* ***NP****-hard to approximate MHC(H) within factor* 1.128 *of its optimal cost. Moreover, under UGC it is hard to approximate MHC(H) within factor* 1.242.

Observe that the **NP**-completeness of the LHC problem leads to inapproximability results for $\mathsf{MHC}(\mathcal{H})$:

Observation 1.7. If LHC($\mathcal{H}$) is **NP**-complete then $\mathsf{MHC}(\mathcal{H})$ is not approximable within any factor, unless $\mathbf{P} = \mathbf{NP}$.

Hence, the dichotomy for the LHC problem [20] implies that $\mathsf{MHC}(H)$ is not approximable for digraphs that contain a *digraph asteroidal triple* (DAT), also known as *bounded width* digraphs. Moreover, the dichotomy for the MHC problem states that $\mathsf{MHC}(H)$ is **NP**-complete if H does not admit a k-min-max ordering [21]. Thus, the primary focus of this paper is to provide inapproximability results for ***all*** digraphs that do not admit a k-min-max ordering i.e., all digraphs H for which $\mathsf{MHC}(H)$ is **NP**-complete. We prove the following:

Theorem 1.8. *[Inapproximability for digraphs] For every digraph H where MHC(H) is* ***NP****-complete, i.e., H does not admit a k-min-max ordering for any $k \geq 1$, it is* ***NP****-hard to approximate MHC(H) within a factor* 1.021 *of its optimal cost.*

We provide an overview of our proofs in Section 3. The proof of Theorem 1.8 is presented in Section 4.2, and the proof of Theorem 1.6 is included in the Appendix.

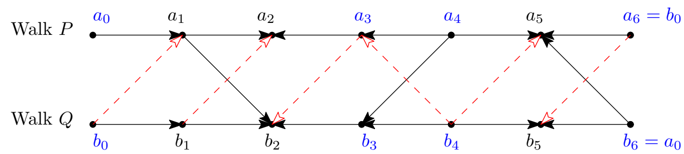

Figure 1: A symmetrically invertible pair a_0, b_0 with 3 switches. Red dashed-arcs are the missing faithful arcs. a_1b_2 is a faithful arc from P to Q, b_3a_4 is a faithful arc from Q to P, a_5a_0 is a faithful arc from P to Q.

2 Definitions and preliminaries

A digraph $H = (V(H), A(H))$ consists of vertex set $V(H)$ and a set of arcs $A(H) \subseteq V(H) \times V(H)$. When no confusion arises, we use the shorthand notations $uv \in A(D)$ in place of $(u, v) \in A(D)$, and $u \in D$ in place of $u \in V(D)$. For arc uv in D, v is an out-neighbor of u and u is an in-neighbor of v. We say uv is a *symmetric arc* of D if both uv and vu are arcs of D. A *graph* is a digraph whose every arc is symmetric. We denote the edge set of a graph G by $E(G)$, and when two vertices x and y are adjacent we say xy (yx) is an edge of G.

Vertex ordering. An ordering $v_1 < v_2 < \cdots < v_n$ of $V(H)$ is a

- *min ordering* if and only if $uv \in A(H), u'v' \in A(H)$ and $u < u', v' < v$ imply that $uv' \in A(H)$;
- *min-max ordering* if and only if $uv \in A(H), u'v' \in A(H)$ and $u < u', v' < v$ imply that $uv', u'v \in A(H)$.
- *k-min-max ordering* if there is a partition of the vertices of H into $V_0, V_1, \ldots, V_{k-1}$ so that each arc of H is from some V_i to V_{i+1}, and $<$ is a min-max ordering on the sub-digraph induced by $V_i \cup V_{i+1}$ for every i (here $V_k = V_0$).

Oriented path, cycle and avoidance definition. We say that $uv \in A(H)$ is an arc from u to v. Sometimes, we emphasize this by saying that uv is a *forward* arc of H, and also say vu is a *backward* arc of H. In what follows when we mention walk, path, and cycle we mean oriented walk, oriented path, and oriented cycle, respectively, unless specified otherwise. For a walk $P = x_0, x_1, \ldots, x_n$ and any $i \leq j$, we denote by $P[x_i, x_j]$ the walk $x_i, x_{i+1}, \ldots, x_j$. We call $P[x_i, x_j]$ a *prefix* of P if $i = 0$. For two walks, P and Q where the end vertex of P is the same as the beginning of Q, let PQ be the walk obtained from concatenation of P and Q. We define two walks $P = x_0, x_1, \ldots, x_n$ and $Q = y_0, y_1, \ldots, y_n$ in H to be *congruent* if they follow the same pattern of forward and backward arcs, i.e., x_ix_{i+1} is a forward arc if and only if y_iy_{i+1} is a forward arc, and x_ix_{i+1} is a backward arc if and only if y_iy_{i+1} is a backward arc. Suppose that the walks P, Q are congruent. We say an arc x_iy_{i+1} is *a faithful arc from P to Q* if it is a forward (backward) arc when x_ix_{i+1} is a forward (backward) arc (respectively). A *faithful arc from Q to P* is defined similarly. We say P *avoids* Q if there is no faithful arc from P to Q. We say P and Q *avoid each other* if P avoids Q, and Q avoids P. We say that P and Q *weakly avoid each other* if for every $0 \leq i \leq n-1$, either x_iy_{i+1} is not a faithful arc (from P to Q) or y_ix_{i+1} is not a faithful arc (from Q to P). Note that this definition is slightly less strict than saying that P and Q avoid each other.

The obstructions to k-min-max ordering are mainly due to the notion of *symmetrically invertible pair*. If H does not admit a k-min-max ordering, then there are oriented walk P (from a to b) and oriented walk Q (from b to a) so that P and Q weakly avoid each other. Furthermore, we assume there is at least one faithful arc between P and Q. We can partition P and Q into segments so that in each segment the faithful arcs are in one direction (from P to Q only or from Q to P only). When the direction of the faithful arcs changes, we say a *switch* occurs (see Figure 1).

Let $\mathcal{H}$ be a hypergraph. We say $\mathcal{H}$ admits a min-max ordering if there is an ordering $a_1, a_2, \ldots, a_n$ of vertices of $\mathcal{H}$ so if $\overline{\mathbf{e_1}} = (a_{i_1}, \ldots, a_{i_r})$ and $\overline{\mathbf{e_2}} = (a_{j_1}, \ldots, a_{j_r})$ are arcs of $\mathcal{H}$, then $(a_{\min\{i_1,j_1\}}, a_{\min\{i_2,j_2\}}, \ldots, a_{\min\{i_r,j_r\}})$ and $(a_{\max\{i_1,j_1\}}, a_{\max\{i_2,j_2\}}, \ldots, a_{\max\{i_r,j_r\}})$ are arcs of $\mathcal{H}$.

Digraph class	Approximation hardness assuming $\mathbf{P} \neq \mathbf{NP}$	Approximation hardness under UGC
(A)	$\sqrt{2} - \epsilon$	$2 - \epsilon$
(B)	1.128	1.242
(C)	1.076	1.137
(D)	1.021	—
(Z)	$\sqrt{2} - \epsilon$ or no approximation	$2 - \epsilon$ or no approximation

Table 1: Summery of our hardness results.

3 Overview of the proofs and future directions

Consider a symmetrically invertible pair a, b with a walk P from a to b and a walk Q from b to a, where P and Q weakly avoid each other. Additionally, suppose there are some faithful arcs from P to Q. Based on the existence of two walks P and Q and the number of switches, we categorize the class of digraphs that do not admit a k-min-max ordering into the following categories:

(A) digraphs with symmetrically invertible pairs with one switch,

(B) digraphs with symmetrically invertible pairs with three switches,

(C) digraphs with symmetrically invertible pairs with five switches,

(D) digraphs with symmetrically invertible pairs with at least seven switches,

(Z) digraphs with zero switches (and hence, no symmetrically invertible pairs).

The graph corresponding to Vertex Cover problem $\{aa, ab\}$ belongs to class (A) and the inapproximability of $\mathsf{MHC}(H)$ for any (graph) digraph H in class (A) is $\sqrt{2} - \epsilon$ under the $\mathbf{P} \neq \mathbf{NP}$ assumption and $2 - \epsilon$ under the Unique Game Conjecture (UGC). Our inapproximability bound for $\mathsf{MHC}(H)$ when H is in class (B) is 1.128 under $\mathbf{P} \neq \mathbf{NP}$ and 1.242 under the UGC. Interestingly, any **graph** H for which $\mathsf{MHC}(H)$ is $\mathbf{NP}$-complete belongs to class (A) or (B). For any digraph H in class (C), we show that it is $\mathbf{NP}$-hard to approximate $\mathsf{MHC}(H)$ within a factor better than 1.076, and it is UG-hard to approximate $\mathsf{MHC}(H)$ with a ratio better than 1.137. For digraph H in class (D), we show that it is $\mathbf{NP}$-hard to approximate $\mathsf{MHC}(H)$ within a factor 1.0134 of its optimal cost. Finally, we note that for class (Z), we may deal with a digraph H for which LHC is $\mathbf{NP}$-complete, and therefore, MHC does not admit any approximation (one concrete example of such digraphs is the class of oriented cycles). For all the other cases, we show an inapproximability of $\sqrt{2} - \epsilon$ assuming $\mathbf{P} \neq \mathbf{NP}$ and $2 - \epsilon$ under the UGC. Table 1 summarizes our hardness results.

For digraphs in classes (B) and (C), we give a hardness reduction starting from Vertex Cover, and for class (D), we provide a gap-preserving reduction starting from Max-3-SAT.

To find a hardness reduction in class (B), we start with an arbitrary graph G and construct a 3-partite graph G_3 with partite sets V_0, V_1, V_2 where each V_i is a copy of $V(G)$. For every $u \in V_0$ and $v \in V_1$ we put an edge uv (in G_3) if their corresponding vertices are adjacent in G. Moreover, connect each vertex in V_2 to its corresponding vertex in V_0 by an edge and to its corresponding vertex in V_1 by an edge. Let $P = a_0, \dots, a_1, \dots, a_2, \dots, b_0$ and $Q = b_0, \dots, b_1, \dots, b_2, \dots, a_0$ be two walks corresponding to symmetrically invertible pair a_0, b_0. Suppose there are faithful arcs from $P[a_0, a_1]$ to $Q[b_0, b_1]$ and faithful arcs from $Q[b_1, b_2]$ to $P[a_1, a_2]$, and finally there are faithful arcs from $P[a_2, b_0]$ to $Q[b_2, a_0]$. We construct digraph D from G_3 as follows. Each edge of G_3 between V_i and V_{i+1} (sum modulo 3) is replaced by an oriented path homomorphic to the i-th segment of P. We define the cost function $c : V(D) \times V(H) \to \mathbb{Q}_{\geq 0} \cup \{+\infty\}$ where $\forall v_0 \in V_0$, $c(v_0, a_0) = 1$, $c(v_0, b_0) = 0$ and $c(v_0, d) = |G|$, $d \notin \{a_0, b_0\}$. $\forall v_1 \in V_1$, $c(v_1, b_1) = 1$, $c(v_1, a_1) = 0$ and $c(v_1, d) = |G|$, $d \notin \{a_1, b_1\}$. $\forall v_2 \in V_2$, $c(v_2, a_2) = 1$, $c(v_1, b_2) = 0$ and $c(v_0, d) = |G|$, $d \notin \{a_2, b_2\}$. For a vertex v of D between V_i and V_{i+1}, $i = 0, 1, 2$, the cost of mapping v to its corresponding vertex in the i-th segment of P and Q is zero and to anything else the cost is $|G|$. We argue that if the cost of a homomorphism from D

to H is less than 1.128 (1.242 under UGC) of its optimal, then the Vertex Cover can be approximated better than $\sqrt{2} - \epsilon$ ($2 - \epsilon$ under UGC).

A similar treatment is applied to obtain the hardness of approximation for class (C) digraphs. In this case, we need a 5-partite graph G_5 constructed from an input graph G, where the edges of G_5 are between consecutive partite sets (also from the last partite set to the first one). However, if we continue with this type of construction for digraphs with a symmetrically invertible pair with k switches, then the hardness of approximation bound would be of the form $(1 + \frac{\alpha}{k})$ for some constant $\alpha < 1$. Therefore, we need to develop a totally different strategy and use the hardness of Max 3-SAT. We construct a graph G from an instance of 3-SAT and then partition the vertices of G into k parts corresponding to k switches in P and Q. Next, we deploy a delicate engineering for replacing each edge of G by an oriented path homomorphic to appropriate sub-paths of P and Q and obtain digraph D and define the cost function. Next, we show that if the minimum cost homomorphism from D to H can be approximated better than 1.0134 then Max 3-SAT admits an approximation factor better than 7/8.

Remark 3.1. For the hardness results we obtain under the complexity assumption $\mathbf{P} \neq \mathbf{NP}$, similar to all other works in the literature, we rely on the PCP characterization of the class **NP**, which implies, one way or another, we are using Gap-SAT as the starting point of our reduction. We have used various known hardness results that suit our case analysis better in order to get our **APX**-hardness results.

Remark 3.2. In Section F, we extend the definitions of symmetrically invertible pairs to hypergraphs by introducing an auxiliary digraph H and a pair digraph H^+ to capture the structural properties of hypergraph $\mathcal{H}$ when it does not admit a min-max ordering. We apply our reduction technique developed for digraphs to the case of hypergraphs.

3.1 Future directions

We remark that for various special cases of the MHC problem, inapproximability results are known. The most notable example is the Vertex Cover problem where the target graph H is $\{aa, ab\}$. It is a classical result that Vertex Cover has a 2-approximation algorithm, and inapproximability results are also known. It is **NP**-hard to approximate Vertex Cover within factor 1.3606 [7]. Later, the factor was improved to $(\sqrt{2} - \epsilon)$ for any $\epsilon > 0$ [29]. Moreover, assuming UGC, Vertex Cover cannot be approximated within any constant factor better than 2. There are other classes of target digraphs H, where MHC(H) does not admit $(2 - \epsilon)$-approximation, for any $\epsilon > 0$. One particular example is the class of *oriented cycles*. When MHC(H) is **NP**-complete for oriented cycle H, then it is **NP**-hard to approximate MHC(H) within factor $\sqrt{2} - \epsilon$, and UGC hard to approximate MHC(H) within factor $2 - \epsilon$ for every $\epsilon > 0$. See Theorem E.3 in Section E.

However, there are classes of digraph H where MHC(H) is **NP**-complete and MHC(H) admits a 2-approximation algorithm, particularly the class of oriented trees admitting a min ordering. We do not have a strong intuition that there is no better than 2-approximation algorithm for MHC(H). On the other hand, this class of digraphs is a sub-class of class (B) discussed earlier in this section, and hence, MHC(H) can not be approximated better than 1.242 under UGC and does not admit a 1.128-approximation, assuming $\mathbf{P} \neq \mathbf{NP}$. Perhaps a more intriguing question to ask is the following.

Open Problem 3.3. *Is there a digraph H for which MHC(H) is **NP**-complete, yet a δ, $\delta < 0.9866$, exists such that MHC(H) admits an $(2 - \delta)$-approximation?*

For future work, a direction is sought. First, MHC($\mathcal{H}$) might admit an approximation algorithm only if LHC($\mathcal{H}$) is polynomial time solvable. LHC($\mathcal{H}$) can be solved in polynomial time if $\mathcal{H}$ can be partitioned into a "bounded width" part and an "affine" part. Otherwise, this LHC is **NP**-complete [5]. For the affine case, it was observed in [26] that, based on hardness result for Nearest Codeword problem [1], Min-Ones for 3LIN is not possible to approximate within a factor of $\Omega(2^{\log^{1-\varepsilon} n})$, unless $\mathbf{NP} \subseteq \mathbf{QP}$. Moreover, in bounded width case, Lemma 8.14 of [26] shows that Min Ones for Horn SAT cannot be approximated to within a factor of $\Omega(2^{\log^{1-\varepsilon} n})$, unless $\mathbf{NP} \subseteq \mathbf{QP}$. Using their example, we give an example of hypergraph $\mathcal{H}$ consisting of three uniform hypergraphs (different sizes) for which MHC($\mathcal{H}$) does not admit a constant approximation (see

Lemma F.5). Therefore, the constant factor approximable cases of $\mathsf{MHC}(\mathcal{H})$ must be bounded width and have at most two *uniform* sub-hypergraphs.

4 Inapproximability of MHC

We say that an optimization problem $\mathcal{P}$ is *α-approx-hard*, $\alpha > 0$, if it is **NP**-hard to find an α-approximation for $\mathcal{P}$. Note that if $\mathcal{P}$ is a maximization problem, then $\alpha \leq 1$; if it is a minimization problem, then $\alpha \geq 1$. We also use another notion of inapproximability under the Unique Game Conjecture (UGC) [27]. We say an optimization problem $\mathcal{P}$ is *α-UG-hard* if it is UG-hard to approximate $\mathcal{P}$ within factor α. See [3] for further details.

A nice property of the MHC problem is that the hardness results for approximation are "carried over" by induced sub-digraphs. This means if $\mathsf{MHC}(H)$ is α-approx-hard or it is α-UG-hard, then the same holds for any digraph which has H as its induced sub-digraph. Informally speaking, such a property holds since the cost functions in the MHC problem are part of inputs, hence, modifying cost functions gives rise to hardness results for every digraph H' which has H as its induced sub-digraph. This is proved formally as follows.

Lemma 4.1 (Sub-digraph hardness)**.** *Let H be an induced sub-digraph of a digraph H'. If $\mathsf{MHC}(H)$ is α-approx-hard [α-UG-hard], then $\mathsf{MHC}(H')$ is α-approx-hard [α-UG-hard].*

Proof. Let G, H together with the cost function $c : G \times H \to \mathbb{Q}_{\geq 0} \cup \{+\infty\}$ be an instance of $\mathsf{MHC}(H)$. Construct an instance of MHC (H') with digraphs G, H' and cost function $c' : G \times H' \to \mathbb{Q}_{\geq 0} \cup \{+\infty\}$ where $c'(u, i) = c(u, i)$ for every $u \in G$ and $i \in H$, otherwise, for every $u \in G$ and $i \in H' \setminus H$, $c'(u, i) = +\infty$. Notice that the cost of any homomorphism from G to H is strictly less than $+\infty$.

Notice that $f'^{*} : V(G) \to V(H')$, the minimum cost homomorphism from G to H', does not map any of the vertices of G to any vertex in $H' \setminus H$ due to the way we have defined c'. Therefore, f'^{*} is also the minimum cost homomorphism for H. Now it is straightforward to see that if an algorithm approximates $f^{*} : V(G) \to V(H)$, the minimum cost homomorphism from G to H within a factor α, it is, in fact, computing an α-approximation of f'^{*}. $\square$

The above lemma provides us the flexibility to focus only on obstructions that render $\mathsf{MHC}(H)$ **NP**-complete. The **NP**-complete cases for MHC has been characterized in terms of k-min-max ordering.

Theorem 4.2 ([21])**.** *Let H be a digraph. Then $\mathsf{MHC}(H)$ is polynomial time solvable if H admits a k-min-max ordering, for some $k \geq 1$. Otherwise, $\mathsf{MHC}(H)$ is **NP**-complete.*

The obstructions for min-max ordering and k-min-max ordering have been characterized in [22]. In the rest of this section, we focus on inapproximability of such instances of MHC. Next, we explain the necessary definitions and terminology to state these obstructions.

4.1 Symmetrically invertible pairs

We observe that if $<$ is a min-max ordering of a digraph H and $P = x_0, x_1, \ldots, x_n$ and $Q = y_0, y_1, \ldots, y_n$ are two congruent walks in H that weakly avoid each other, then $x_0 < y_0$ if and only if $x_n < y_n$. Indeed, if $x_i < y_i$ and $y_{i+1} < x_{i+1}$, then the min-max property is not satisfied for $x_i, y_i, x_{i+1}, y_{i+1}$; a similar contradiction arises if $y_i < x_i$ and $x_{i+1} < y_{i+1}$.

Definition 4.3 (Symmetrically invertible pair)**.** A *symmetrically invertible pair* in H is a pair of distinct vertices a and b with two congruent walks P and Q in H, where P is from a to b and Q is from b to a such that P and Q weakly avoid each other. Moreover, there is at least one faithful arc between P and Q. We say P, Q are the associated walks with the pair a, b.

Now it is clear that, if H has a symmetrically invertible pair, then it cannot have a min-max ordering.

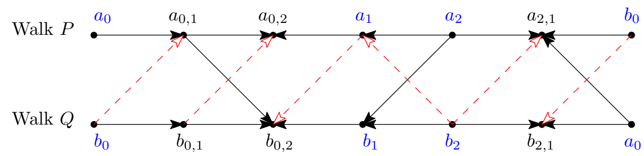

Figure 2: A symmetrically invertible pair a_0, b_0 with 3 switches. Dashed-arcs are the missing faithful arcs. $a_{0,1}b_{0,2}$ is a faithful arc from P to Q, b_1a_2 is a faithful arc from Q to P, $a_{2,1}a_0$ is a faithful arc from P to Q.

Let a_0, b_0 be a symmetrically invertible pair with associated walks P and Q. Without loss of generality, let P and Q be partitioned into k pieces as follows (see Figure 2) :

$$P = a_0, a_{0,1}, a_{0,2}, \dots, a_{0,l_0}, a_1, a_{1,1}, a_{1,2}, \dots a_{1,l_1}, a_2, \dots, a_{k-1}, a_{k-1,1}, \dots, a_{k-1,l_{k-1}}, b_0,$$

$$Q = b_0, b_{0,1}, b_{0,2}, \dots, b_{0,l_0}, b_1, b_{1,1}, b_{1,2}, \dots, b_{1,l_1}, b_2, \dots, b_{k-1}, b_{k-1,1}, \dots, b_{k-1,l_{k-1}}, a_0.$$

For $0 \leq i < j \leq k-1$, let $P_{i,j}$ (resp. $Q_{i,j}$) denote $P[a_i, a_j]$ of P (resp. $Q[b_i, b_j]$ of Q). Furthermore, assume that $P_{i,i+1}$ avoids $Q_{i,i+1}$ when i is odd and there is at least one faithful arc from $Q_{i,i+1}$ to $P_{i,i+1}$. One can assume such arc always exists due to the definition of the symmetrically invertible pair. Likewise, assume $Q_{i,i+1}$ avoids $P_{i,i+1}$ when i is even and there is at least one faithful arc from $P_{i,i+1}$ to $Q_{i,i+1}$ (here $0 \leq i \leq k-1$). We refer to k as *the number of switches* for walks P and Q.

With the walks P and Q as above, notice that a_1, b_1 is also a symmetrically invertible pair. Indeed, setting $P' = P[a_1, a_{k-1}]P[a_{k-1}, b_0]Q[b_0, b_1]$ and $Q' = Q[b_1, b_{k-1}]Q[b_{k-1}, a_0]P[a_0, a_1]$ we will have the pair (P', Q') of walks associated to (a_1, b_1). Moreover, when k is even, there is some faithful arc from $Q[b_{k-1}, a_0]P[a_0, a_1]$ to $P[a_{k-1}, b_0]Q[b_0, b_1]$ whereas there is no faithful arc from $P[a_{k-1}, b_0]Q[b_0, b_1]$ to $Q[b_{k-1}, a_0]P[a_0, a_1]$. Therefore, when k is even then (Q', P') has $k-1$ switches, thereby, (a_1, b_1) has at most $k-1$ switches, by definition. This observation in particular implies the following proposition which we will use in our hardness reductions.

Proposition 4.4 (Odd number of switches)**.** *Let H be a digraph with a pair of symmetrically invertible vertices. Then, H contains a symmetrically invertible pair with an odd number of switches.*

4.2 Hardness of approximation for digraphs

The techniques used here are based on the elegant characterization and structural properties of digraphs admitting a k-min-max ordering. The following corollary characterizes digraphs that do not admit a k-min-max ordering for any $k \geq 1$. It is obtained from Theorems 4.2, A.1, A.2 that we borrow from [21, 22].

Corollary 4.5. *Let H be a digraph. Then MHC(H) is **NP**-complete if one the following occurs:*

1. *H is balanced (see appendix A), and it does not admit a min-max ordering.*

2. *H contains walks P and Q, from a, b to b, a (respectively) where P, Q weakly avoids each other and P have some faithful arcs to Q or Q have some faithful arcs to P (at least one switch).*

3. *H contains three congruent walks P, Q, R from a, b, a to b, a, a, respectively such that :*

   - *Q has no faithful arc to P,*

   - *if there is a faithful arc from i-th vertex of Q to $(i+1)$-th vertex to R, then there is no faithful arc from j-vertex of R to $(j+1)$-vertex of P, $i+1 \leq j$, from R to P.*

4. *There is a homomorphism $f : V(H) \to \overrightarrow{C}_k$, $k > 1$, but H contains a symmetrically invertible pair a, b with $f(a) = f(b)$.*

Next we prove a sequence of hardness results for various cases, namely Lemmas 4.6, 4.7, 4.8, 4.9, and at the end argue these lemmas are sufficient to cover the cases in Corollary 4.5.

Lemma 4.6 (General case and one switch)**.** *Let H be a digraph that contains three congruent walks P, Q, R from a, b, a to b, a, a, respectively. Suppose Q has no faithful arc to P, and if there is a faithful arc from the i-th vertex of Q to $(i+1)$-th vertex of R, then there is no faithful arc from j-th vertex of R to $(j+1)$-th vertex of P, $i+1 \leq j$. Then,*

1. *MHC(H) is $(\sqrt{2}-\epsilon)$-approx-hard for every $\epsilon > 0$.*
2. *MHC(H) is $(2-\epsilon)$-UG-hard for every $\epsilon > 0$.*

Proof. Let G be the graph described in [7]. We orient each edge of G to obtain the digraph G. Vertex cover in digraphs is defined in the same way as for graphs. We construct an instance of $\mathsf{MHC}(H)$ as follows. Let p_i, r_i, q_i, $0 \leq i \leq t$ be the vertices of P, Q, R, respectively. Construct digraph D by replacing each arc $e = uv$ of G by an oriented path $S_e : u = u_0, u_1, \dots, u_{t-1}, u_t = v$ which is congruent to P, Q, R. Define the cost function $c : V(D) \times V(H) \to \mathbb{Q}_{\geq 0}$ by the following rules. For every arc $e = uv$ of G

- $c(u, b) = c(v, b) = 0$ and $c(u, a) = c(v, a) = 1$,
- for every $u_i \in S_e$, $1 \leq i \leq t-1$, $c(u_i, p_i) = c(u_i, r_i) = c(u_i, q_i) = 0$,
- for every $u_i \in S_e$, $1 \leq i \leq t-1$, and $d \notin \{p_i, q_i, r_i\}$ set $c(u_i, d) = |G|$.

Next we show that G has a vertex cover of size m if and only if there exists a homomorphism from D to H with total cost m.

Let VC be a vertex cover in G. Define a mapping $f : V(D) \to V(H)$ by setting $f(u) = a$ if $u \in VC$ and $f(u) = b$ if $u \in G \setminus VC$. For every vertex u_i of S_e, $1 \leq i \leq t-1$, corresponding to arc $e = uv$ of G, set $f(u_i) = p_i$ when $f(u = u_0) = a$, $f(u_t = v) = b$; set $f(u_i) = r_i$ when $f(u) = f(v) = a$, and finally set $f(u_i) = q_i$ when $f(u) = b$, $f(v) = a$. Since S_e can be mapped to one of the P, Q, R depending on $f(u), f(v)$, it is easy to see that f is a homomorphism from D to H with $c(f) = m$.

Conversely, let $f : V(D) \to V(H)$ be a homomorphism with cost $c(f) = m < |G|$. Let $VC = \{u \in D \mid f(u) = a\}$. We show that VC is a vertex cover in G of size m. Let $u, v \in G \setminus VC$, and for contradiction suppose $e = uv$ is an arc of G. This means $f(u) = b$, and $f(v) = b$, and hence, S_e is mapped to a walk T in H from b to b and congruent to Q. However, because $c(f) < |G|$ we conclude that $f(u_i) \in \{p_i, q_i, r_i\}$, and since there is no faithful arc from Q to P, and there is no faithful arc from R to P after a faithful arc from Q to R, there is no such T; this is a contradiction to f being a homomorphism. Therefore, VC is a vertex cover of size $c(f) = m$.

Let f^* be an optimal minimum cost homomorphism from D to H. For contradiction, suppose for some $\lambda > 0$, there exists a $(1+\lambda)$-approximation algorithm for $\mathsf{MHC}(H)$ that finds a homomorphism $f : V(D) \to V(H)$ with $c(f) < (1+\lambda)c(f^*)$. Now obtain a vertex cover VC in G from f, and an optimal vertex cover VC^* in G from f^*. Thus, we have $|VC| < (1+\lambda)(|VC^*|)$. By setting $\lambda = \sqrt{2} - 1 - \epsilon$, we conclude that vertex cover in G can be approximated within factor $\sqrt{2} - \epsilon$, a contradiction to [7].

Using the UGC assumption and appealing to hardness of approximation for vertex cover from [30], we conclude that $\mathsf{MHC}(H)$ is $(2-\epsilon)$-UG-hard for any $\epsilon > 0$. ∎

Lemma 4.7 (3 switches)**.** *Let H be a digraph where MHC(H) is **NP**-complete. Suppose H contains a symmetrically invertible pair with three switches. Then MHC(H) is 1.128-approx-hard, and it is 1.242-UG-hard.*

Proof. Let a_0, b_0 be a symmetrically invertible pair with three switches. Let $P = a_0, a_{0,1}, a_{0,2}, \dots, a_{0,l_0}, a_1, a_{1,1}, \dots, a_{1,l_1}, a_2, a_{2,1}, \dots, a_{2,l_2}, b_0$ and let $Q = b_0, b_{0,1}, \dots, b_{0,l_0}, b_1, b_{1,1}, \dots, b_{1,l_1}, b_2, b_{2,1}, \dots, b_{2,l_2}, a_0$. Let $P_i = P[a_i, a_{i+1}]$ and $Q_i = Q[b_i, b_{i+1}]$, $i = 0, 1$ and $P_2 = P[a_2, b_0]$, $Q_2 = P[b_2, a_0]$. Moreover, P_i has faithful arcs to Q_i, but Q_i has no faithful arc to P_i, $i = 0, 2$ and Q_1 has faithful arcs to P_1, but P_1 has no faithful arc to Q_1. Let G be a graph with vertex set $\{x_1, \dots, x_n\}$ considered by [7, 29].
Construction of the 3-partite graph G' from G: Let V_0, V_1, V_2 be 3 disjoint copies of the vertices of G. Let $V_0 = \{u_1, u_2, \dots, u_n\}$, $V_1 = \{v_1, v_2, \dots, v_n\}$, and $V_2 = \{w_1, \dots, w_n\}$ where u_r, v_r, w_r are the copy of $x_r \in G$.

For each edge $e = x_r x_s$ of G, add the edges $u_r v_s$ and $u_s v_r$ into G' where $u_r, u_s \in V_0$ (corresponding to vertices $x_r, x_s \in G$, respectively) and $v_r, v_s \in V_1$ (corresponding to $x_r, x_s \in G$, respectively). For every $v_r \in V_1$, $w_r \in V_2$ corresponding to $x_r \in G$, add the edge $v_r w_r$ into G'. For every $u_r \in V_0$, $w_r \in V_2$ corresponding to $x_r \in G$, add the edge $u_r w_r$ into G'.
Construction of digraph D from G', and defining the cost function: For every edge $e = uv$ of G' with $u \in V_i$, $v \in V_{i+1}$, $i = 0, 1, 2$ (sum module 3), replace e by a new path $Y_i = y_i, y_{i,1}, \ldots, y_{i,l_i}, y_{i+1}$ which is congruent to P_i and Q_i, by identifying y_i with u and y_{i+1} by v. Let D be the resulting digraph. Define the cost function $c : D \times H \to \mathbb{Q}_{\geq 0}$, for Y_i as follows. $c(y_{i,j}, a_{i,j}) = c(y_{i,j}, b_{i,j}) = 0$ and $c(y_{i,j}, t) = 2|G|$ for every other vertex $t \in P \cup Q$. For every $u \in V_i$, set $c(u, a_i) = 1$, $c(u, b_i) = 0$ when $i = 0, 2$ and set $c(u, a_i) = 0$, $c(u, b_i) = 1$ when $i = 1$. In any other case, the cost of mapping u to a vertex in $P \cup Q$ is $2|G|$.
From a vertex cover in G to a homomorphism from D to H: Let VC be a vertex cover in G. Then define the mapping $f : V(D) \to V(H)$ as follows. For $u_r \in V_0$, set $f(u_r) = a_0$ if $x_r \in VC$ (u_r is the copy of x_r) else set $f(u_r) = b_0$. For every $v_r \in V_1$, set $f(v_r) = b_1$ if $x_r \in VC$ (v_r is the copy of $x_r \in G$) else $f(v_r) = a_1$. For every $w_r \in V_2$, set $f(w_r) = a_2$ if $x_r \notin VC$ (w_r is the copy of $x_r \in G$) else $f(w_r) = b_2$. Let $Y_0 = y_0, y_{0,1}, \ldots, y_{0,l_0}, y_1$ be a path in D from $u_r \in V_0$ to $v_s \in V_1$ ($y_0 = u_r$ and $y_1 = v_s$). We extend f to the vertices of Y_0 as follows. Since VC is a vertex cover in G, by definition there is no path in D between a vertex $u_r \in V_0$, with $f(u_r) = b_0$, and vertex $v_s \in V_0$, with $f(v_s) = a_1$. Thus, f maps Y_0 to $P_0 \cup Q_0$ according the following rules.

- if $f(u_r) = a_0$ and $f(v_s) = a_1$. Then, f maps Y_0 to P_0.
- if $f(u_r) = b_0$ and $f(v_s) = b_1$. Then f maps Y_0 to Q_0.
- if $f(u_r) = a_0$ and $f(v_s) = b_1$. Let $a_{0,l} b_{0,l+1}$ be a first faithful arc from P_0 to Q_0. Now, set $f(y_{0,j}) = a_{0,j}$ if $j \leq l$ else set $f(y_0, j) = b_{0,j}$. Notice that f is a homomorphism that maps Y_0 to $P_0 \cup Q_0$ (using the faithful arc $a_{0,l} b_{0,l+1}$).

Similarly one can extend f to make it a homomorphism from Y_i to $P_i \cup Q_i$ where $Y_i = y_i, y_{i,1}, \ldots, y_{i,l_i}, y_{i+1}$, $i = 1, 2$ (sum module 3) $y_i \in V_i$ to $y_{i+1} \in V_{i+1}$.

Now it is easy to see that f is a homomorphism from D to H with total cost $2|VC| + |G| - |VC| = |VC| + |G|$.
From a homomorphism from D to H to a vertex cover in G: Let $f : V(D) \to V(H)$ be a homomorphism with the total cost less than $2|G|$. We modify f so that for every $u_r \in V_0$, $v_r \in V_1$, and $w_r \in V_2$ (where u_r, v_r, w_r are copies of the same vertex $x_r \in G$) $f(u_r) = a_0$ if and only if $f(v_r) = b_1$ if and only if $f(w_r) = b_2$. Suppose for some $u_r \in V_0$, $f(u_r) = a_0$ and $f(v_r) = a_1$. Note that there is edge $v_r w_r$ in G', and hence, there is oriented walk Y_1 from v_r to w_r homomorphic to $P[a_1, a_2]$. Since $P[a_1, a_2]$ does not have a faithful arc to $Q[b_1, b_2]$, and f is a homomorphism, we must have $f(w_r) = a_2$. Now, we modify f, and obtain f^1, by setting $f^1(v_r) = b_1$, and $f(w_r) = b_2$. Furthermore, Y_2, an oriented path from w_r to u_r in D, is assigned under f^1 to $Q[b_2, a_0]$, and the path Y_1 between v_r and w_r is mapped to $Q[b_1, b_2]$. Finally, any path Y_0 from some $u_i \in V_0$ to v_r with $f(u_i) = a_0$, under f^1 is mapped to $P[a_0, a_{0,l}]Q[b_{0,l+1}, b_1]$ where $a_{0,l} b_{0,l+1}$ is a faithful arc from $P[a_0, a_1]$ to $Q[b_0, b_1]$. It is clear that f^1 is also a homomorphism from D to H with the same cost as f. We continue this process until we obtain a homomorphism f^t so that $f^t(u_r) = a_0$ if and only if $f^t(v_r) = b_1$, and if and only if $f^t(w_r) = b_2$ for every $1 \leq r \leq n$. Therefore, for simplicity, we may assume $f^t = f$. Let $VC = \{x_r \in G \mid f(u_r) = a_0 \text{ where } u_r \in V_0 \text{ is the copy of } x_r \in G\}$. Now, it is not difficult to show that VC is a vertex cover in G of size $|G| - c(f)$.
*Showing the 1.128-approximation is **NP**-hard:* We show that it is **NP**-hard to find a homomorphism $f : V(D) \to V(H)$ with $c(f) < (1 + \lambda)c(f^*)$ (here $\lambda = 0.155$, and f^* is the optimal minimum cost homomorphism from D to H). For contradiction, suppose there is a polynomial-time algorithm that produces such a homomorphism f. Then, $c(f) = |VC| + |G|$ and $c(f^*) = |VC^*| + |G|$ (here VC^* is the optimal vertex cover in G). We have $|VC| + |G| < (1 + \lambda)(|VC^*| + |G|)$.

Thus, $|VC| < (1+\lambda)|VC^*| + \lambda|G|$, and hence, $|VC| - \lambda|G| < (1+\lambda)|VC^*|$. We may assume $|VC| \geq 0.639|G|$, thanks to the construction in [7]. Therefore, we have $|VC|(1 - \frac{\lambda}{0.639}) \leq |VC| - \lambda|G| < (1 + \lambda)|VC^*|$, and consequently, we have $|VC| < \frac{1+\lambda}{1-\frac{\lambda}{0.639}}|VC^*|$.

Setting $\frac{(1+\lambda)0.639}{0.639-\lambda} = \sqrt{2}$, we get a contradiction since, as shown in [29], the vertex cover cannot be approximated within any factor better than $\sqrt{2} - \epsilon$. Thus, $1 + \lambda = 1.128$ and it is NP-hard to approximate

MHC(H) within the factor 1.128. Moreover, setting $\frac{(1+\lambda)0.639}{0.639-\lambda} = 2$, ($\lambda = 0.242$) we get a contradiction with the $(2-\epsilon)$-UG-hardness for the Vertex Cover [30]. $\square$

Lemma 4.8 (5 switches)**.** *For digraph H containing symmetrically invertible pair with five switches, MHC(H) is 1.076-approx-hard, and it is 1.137-UG-hard.*

Lemma 4.9 (7 and more switches)**.** *Let H be a digraph containing symmetrically invertible pair with $k \geq 7$ switches. Then MHC(H) is 1.021-approx-hard.*

Proof. Let F be a 3-SAT formula with variables $\alpha_0, \alpha_1, \dots, \alpha_n, \neg\alpha_0, \neg\alpha_1, \dots, \neg\alpha_n$. Without loss of generality, we assume that α_i and $\neg\alpha_i$ both appear in F as otherwise, if only α_i appears in F then we can set α_i to be true and eliminate the clauses containing α_i (the same treatment for $\neg\alpha_i$). It is easy to see that a random assignment on average satisfied at least $\frac{7}{8}$ fraction of clauses. On the other hand, it was shown in [16] that for every $\delta > 0$, it is **NP**-hard to satisfy more that $(\frac{7}{8} + \delta)$ fraction of clauses. In other words, Max 3-SAT is $(\frac{7}{8} + \delta)$-approx-hard for every $\delta > 0$.
Construction of graph G from F. For each clause $C_i = (\alpha_i \vee \beta_i \vee \gamma_i)$ we consider three vertices $x_i, y_i, z_i \in G$ corresponding to $\alpha_i, \beta_i, \gamma_i$ (respectively), and the edges x_iy_i, x_iz_i, y_iz_i. We add an edge x_ix_j (y_iy_j, z_iz_j) if a_j (b_j, c_j) in clause C_j is the negation of $a_i(b_i, c_i)$ in clause C_i.

Let $V_0, V_1, \dots, V_{n-1}, V_n$ be a partition of $V(G)$ so that V_0 contains vertices of G corresponding to α_0 and $\neg\alpha_n$, V_1 consists of vertices of G corresponding to $\neg\alpha_0$ and α_1, V_2 consists of vertices of G corresponding to $\neg\alpha_1$ and α_2, and so on. Finally, V_n consists of vertices of G corresponding to $\neg\alpha_{n-1}$ and α_n. Notice that there is an edge from V_i to V_{i+1} (sum modulo $n+1$). Let P and Q be two congruent walks in H starting at a_0, b_0 and ending at b_0, a_0 (respectively) that weakly avoid each other. As mentioned earlier, we partition P, Q into k pieces as follows.

$$P = \overbrace{a_0, a_{0,1}, a_{0,2}, \dots, a_{0,l_0}, a_1, a_{1,1}, a_{1,2}, \dots a_{1,l_1}, a_2}^{P_{0,2}}, \dots, a_{k-1}, a_{k-1,1}, a_{k-1,2}, \dots, a_{k-1,l_{k-1}}, a_k = b_0$$
$$Q = \underbrace{b_0, b_{0,1}, b_{0,2}, \dots, b_{0,l_0}, b_1, b_{1,1}, b_{1,2}, \dots, b_{1,l_1}, b_2}_{Q_{0,2}}, \dots, b_{k-1}, b_{k-1,1}, \dots, b_{k-1,l_{k-1}}, b_k = a_0$$

Recall that for $i < j$, $P_{i,j}$ is $P[a_i, a_j]$ (the potion of P from a_i to a_j) and $Q_{i,j}$ is $Q[b_i, b_j]$ (the portion of Q from b_i to b_j). Then, $P_{i,i+1}$ avoids $Q_{i,i+1}$ when i is odd and there is a faithful arc from $Q_{i,i+1}$ to $P_{i,i+1}$. Also, $Q_{i,i+1}$ avoids $P_{i,i+1}$ when i is even and there is at least one faithful arc from $P_{i,i+1}$ to $Q_{i,i+1}$ $(0 \leq i \leq k-1)$.

For $0 < i < j \leq k-1$, let $P_{j,i}$ be the walk $P_{j,k}Q_{0,i}$; concatenation of $P_{j,k}$ and $Q_{0,i}$. In other words, $P_{j,i}$ starts from a_j and it follows P to $a_k = b_0$ (to the end of P), and then it continues on Q from b_0 to b_i. Similarly, $Q_{j,i} = Q_{j,k}P_{0,i}$.

When $i + 4 \leq j$, let $X_{i,j}$ be a path that is congruent to $P_{i,j}$. When $j - i = 1$, then $X_{i,i+1}$ is a path that is congruent to $P_{i,i+1}$. When $2 \leq j - i \leq 3$, then let $X_{j,i}$ is a path congruent to $P_{j,i}$. Notice that since $X_{i,j}$, $P_{i,j}$, and $Q_{i,j}$ are all congruent, every vertex in $X_{i,j}$ has its corresponding vertices in $P_{i,j}$ and $Q_{i,j}$. For all $i + 4 \leq j$, let g_a be a homomorphism from $X_{i,j}$ to $P_{i,j}$ so that for every $u \in X_{i,j}$, $g_a(u)$ is its corresponding vertex in $P_{i,j}$. For all $0 \leq i \leq k-1$, extend g_a so that it is a homomorphism from $X_{i,i+1}$ to $P_{i,i+1}$ where for every $u \in X_{i,i+1}$, $g_a(u)$ is its corresponding vertex in $P_{i,i+1}$. Finally, for all $i + 2 \leq j \leq i + 3$, extend g_a to a homomorphism from $X_{j,i}$ to $P_{j,i}$ so that for every $u \in X_{j,i}$, $g_a(u)$ is its corresponding vertex in $P_{j,i}$.

Let g_b be the corresponding homomorphism to g_a, i.e., a homomorphism from $X_{i,j}$ to $Q_{i,j}$ and from $X_{j,i}$ to $Q_{j,i}$ (respectively for $i + 4 \leq j$, $j < i + 4$) so that for every $u \in X_{i,j}$ and $X_{j,i}$, $g_b(u)$ is its corresponding vertex in $Q_{i,j}$ and $Q_{j,i}$, respectively. Notice that we often use a copy of $X_{i,j}$, say $Y_{i,j}$ and assume $g_a(y) = g_a(x)$ for $y \in Y_{i,j}$, $x \in X_{i,j}$, where y is the copy of x. The same is considered for g_b.

In what follows, we construct an instance of MHC(H) with input digraph D and target digraph H where H is a digraph containing $P \cup Q$ as an induced sub-digraph.
Construction of the digraph D and defining the cost function: Let D be a digraph constructed from G as follows. The vertices of D consists of $U_0, U_1, \dots, U_n$ where each U_t is a copy of the vertices in V_t $(t = 0, 1, \dots, n)$.

Let $e = uv$ be an arbitrary edge of G with $u \in V_{i'}$ and $v \in V_{j'}$. Let $i = i' \mod k$, and $j = j' \mod k$. When $j = i + 1$ or $i + 4 \leq j$ we add a copy of $X_{i,j}$ between $u \in U_{i'}$ and $v \in U_{j'}$ (u is corresponding to $u \in V_{i'}$

and v is corresponding to $v \in V_{j'}$) identifying the first vertex of $X_{i,j}$ by u and the last vertex of $X_{i,j}$ by v. When $i+2 \leq j \leq i+3$, then place a copy of $X_{j,i}$ identifying v with the beginning of $X_{j,i}$, and u with the end of $X_{j,i}$.

Now define the cost function $c : V(D) \times V(H) \to \mathbb{Q}_{\geq 0}$ as follows. For every $u \in U_{i'}$ if i is even then set $c(u, a_i) = 1$, $c(u, b_i) = 0$, and in any other case the cost is $2|G|$, that is, $c(u, d) = 2|G|$ when $d \notin \{a_i, b_i\}$. For every $u \in U_{i'}$ if i is odd then set $c(u, b_i) = 1$, $c(u, a_i) = 0$, and in any other case the cost is $2|G|$.

Let $Y_{i,j}$ ($Y_{j,i}$ when $j-i = 2,3$) be a copy of $X_{i,j}$ ($X_{j,i}$) connecting vertices u and v in D. Define the cost function for $w \in Y_{i,j}$ ($w \in Y_{j,i}$) as follows. Set $c(w, d) = 2|G|$ when $d \notin \{g_a(w), g_b(w)\}$. Initially, set $c(w, g_a(w)) = c(w, g_b(w)) = 0$. We change it to 1 according to the following cases.

1. i is even, j is even, and $j - i \geq 4$. If $g_b(w) \in \{b_{i+1}, b_{j-1}\}$ then set $c(w, g_b(w)) = 1$.
2. i is odd, j is odd, and $j - i \geq 4$. If $g_a(w) \in \{a_{i+1}, a_{j-1}\}$ then set $c(w, g_a(w)) = 1$.
3. i is even, j is odd, and $j - i \geq 4$. If $g_b(w) = b_{i+1}$ then $c(w, b_{i+1}) = 1$. If $g_a(w) = a_{j-1}$ then set $c(w, a_{j-1}) = 1$.
4. i is odd, j is even, and $j - i \geq 4$. If $g_a(w) = a_{i+1}$ then set $c(w, a_{i+1}) = 1$. If $g_b(w) = b_{j-1}$ then set $c(w, b_{j-1}) = 1$.
5. i is even, j is even and $j - i = 2$. If $g_b(w) \in \{b_{j+1}, b_{i-1}\}$ then set $c(w, g_b(w)) = 1$.
6. i is odd, j is odd and $j - i = 2$. If $g_a(w) \in \{a_{j+1}, a_{i-1}\}$ then set $c(w, g_a(w)) = 1$.
7. i is even, j is odd, and $j - i = 3$. If $g_a(w) = a_{j+1}$ then set $c(w, a_{j+1}) = 1$. If $g_b(w) = b_{i-1}$ then set $c(w, b_{i-1}) = 1$.
8. i is odd, j is even, and $j - i = 3$. If $g_b(w) = b_{j+1}$ then set $c(w, b_{j+1}) = 1$. If $g_a(w) = a_{i-1}$ then set $c(w, a_{i-1}) = 1$.

From an independent set in G to a homomorphism from D to H: Let I be an independent set in G. Define the mapping $f : V(D) \to V(H)$ as follows. For every $u \in U_{2i'}$ set $f(u) = a_{2i}$ if $u \notin I$ else set $f(u) = b_{2i}$. For every $u \in U_{2i'+1}$ set $f(u) = b_{2i+1}$ if $u \notin I$ else set $f(u) = a_{2i+1}$.

Let $e = uv$ be an edge of G and let $Y_{i,j}$ ($j = i+1$ or $j - i \geq 4$) be a copy of $X_{i,j}$ in D where the first vertex of $Y_{i,j}$ is u and the last vertex of $Y_{i,j}$ is v. Let $Y_{j,i}$ ($2 \leq j - i \leq 3$) be a copy of $X_{j,i}$ in D where the first vertex of $Y_{j,i}$ is v and the last vertex of $Y_{j,i}$ is u. We show that f can be defined on $Y_{i,j}$ ($Y_{j,i}$, $j-i = 2,3$) so that it becomes a homomorphism from $Y_{i,j}$ ($Y_{j,i}$) to $P_{i,j} \cup Q_{i,j}$ ($P_{j,i} \cup Q_{j,i}$).

Claim 4.10. *f can be defined on $Y_{i,j}$ ($Y_{j,i}$, $j - i = 2,3$) so that it becomes a homomorphism from D to H. Moreover, the cost of f is $|G| - |I| + \sum_{i+1<j} |E_{i,j}|$ where $|E_{i,j}|$ is the number of edges $e = uv$ between V_i, V_j with one endpoint of e in I and another one outside I.*

From a homomorphism from D to H to an independent set in G: Let f be a homomorphism from D to H with total cost less than $2|G|$. Let $u \in U_{i'}$ and $v \in U_{j'}$, with $j - i > 1$, and uv is an edge of G. Suppose $f(u) = b_i$, $f(v) = b_j$, and i, j both are even. Notice that $c(f(Y_{i,j})$ ($c(f(Y_{j,i}))$ is 2. In this case we modify f and assign u to a_i, and then the image of $Y_{i,j}$ ($Y_{j,i}$, when $2 \leq j - i \leq 3$) changes so that f is still a homomorphism from $Y_{i,j}$ to $P_{i,j} \cup Q_{i,j}$. This was explained in Cases 1,2,3,4. Notice that by doing so, the value of f does not change and f still is a homomorphism from D to H.

Similarly, when $f(u) = a_i$, $f(v) = a_j$, and both i, j are odd we modify f so that $f(u) = b_i$, and consequently $f(Y_{i,j})$ is changed accordingly. Again in this case the value of f stays the same and f is a homomorphism from D to H. Now define set $I = \{u_i \in U_{i'} \mid f(u_i) = b_i \text{ and } i \text{ is even}\} \cup \{v_i \in U_{i'} \mid f(v_i) = a_i \text{ and } i \text{ is odd}\}$. We show that I is an independent set in G. Let $u \in U_{i'} \cap I$, and $v \in V_{j'} \cap I$. We show that $uv \notin E(G)$. For contradiction, suppose uv is an edge of G.

First suppose $j - i = 1$. Now by definition $f(u) = b_i$ and i is even or $f(u_i) = a_i$ and i is odd. Suppose i is even. Notice that by definition of the cost function, the image of $Y_{i,i+1}$ under f lies in $P_{i,i+1} \cup Q_{i,i+1}$. However, since there is no faithful arc from $Q_{i,i+1}$ to $P_{i,i+1}$, the image of $Y_{i,i+1}$ under f is $Q_{i,i+1}$, and

$f(v_i) \neq a_{i+1}$, a contradiction. When i (still in case $j - i = 1$) is odd we have $f(u) = a_i$, and again since there no faithful arc from $P_{i,i+1}$ to $Q_{i,i+1}$, the image of $Y_{i,i+1}$ under f is $P_{i,i+1}$, and $f(v_i) \neq b_{i+1}$, a contradiction.

Now consider the case where $2 < j - i$. Then, according to the modification of f, when $f(u) = b_i$ and i, j both are even, then $f(v_j) \neq b_j$, and hence, $v_j \notin I$. When $f(u) = b_i$ and i is even and j is odd then $f(v_j) \neq a_j$, and hence, $v_j \notin I$. Similarly, when $f(u) = a_i$ and i is even then $f(v_j) \neq a_j$.

Therefore, I is an independent set in G. Notice that $c(f) = |G| - |I| + \sum_{i+1<j} |E_{i,j}|$ where $|E_{i,j}|$ is the number of edges between V_i, V_j where one end point is in I and the other point is outside I.

Getting contradiction to Max 3-SAT inapproximability: Let f^* be an optimal minimum cost homomorphism from D to H. Suppose that for some $\lambda > 0$, there exists a $(1+\lambda)$-approximation algorithm for $\mathsf{MHC}(H)$ that finds a homomorphism $f : V(D) \to V(H)$ with $c(f) < (1+\lambda)c(f^*)$. Let $|E_I| = \sum_{i<j} |E_{i,j}|$ (corresponding to homomorphism f), and let $|E^*| = \sum_{i<j} |E^*_{i,j}|$ (corresponding to optimal homomorphism f^*).

Thus, we have $|G| - |I| + |E_I| < (1+\lambda)(|G| - |I^*| + |E^*|)$. Note that the number of edges in G corresponding to each clause of F is $|G|$ (each vertex lies on a triangle). Therefore, $|E_I|, |E^*| \leq |G|$. First assume $|E^*| \leq |E_I|$ (in the other case we replace $|E^*|$ by $|E_I|$ on the right side of the inequality). Thus, we have $(1+\lambda)|I^*| < |I| + \lambda(|G| + |E^*|)$, and consequently $(1+\lambda)|I^*| < 2\lambda|G| + |I|$. Since there is an assignment that satisfies at least $\frac{7}{8}$ of the clauses, we conclude that $I^* \geq \frac{7|G|}{24}$. Thus, $(1+\lambda)|I^*| < \frac{48}{7}\lambda|I^*| + |I|$. Now, $(1 - \frac{41}{7}\lambda)|I^*| < |I|$. Therefore, $(1 - \frac{1}{8})|I^*| < |I|$ which is a contradiction to Max 3-SAT admits a better than $\frac{7}{8}$ approximation algorithm. □

Theorem 4.11 (Restatement of Theorem 1.8)**.** *Let H be a digraph that does not admit a k-min-max ordering for any $k \geq 1$. Then $\mathsf{MHC}(H)$ is (1.021)-approx-hard.*

Proof. We consider four scenarios and argue that it does cover all the digraphs that do not admit k-min-max ordering for any $k \geq 1$.

First scenario: H contains a symmetrically invertible pair with at least one switch. The proof follows from Lemma 4.6, Lemma 4.7, Lemma 4.8, and Lemma 4.9.

Second scenario: H is balanced. According to Lemma A.4, H may have symmetrically invertible pair a, b with associated walks P, Q that weakly avoid each (with at least one switch), and hence, we are done according to the First scenario. Another possibility by Lemma A.4 (2), is the existence of three walks P, Q, R (in H) from a, b, a to b, a, a such that Q avoids P, and hence, by Lemma 4.6 we get the desired conclusion.

Third scenario: H is homomorphic to some directed cycle $\overrightarrow{C_k}$ and has a symmetrically invertible pair belonging to the same set. This case is handled by the First scenario.

Fourth scenario: H is not homomorphic to any directed cycle $\overrightarrow{C}_k$. Suppose H contains two induced oriented cycles of net lengths l, r, where l, r are co-prime. Now according to Lemma A.3 H may have symmetrically invertible pair a, b with associated walks P, Q (with at least one switch). Another possibility is that there exist, three walks P, Q, R from a, b, a to b, a, a such that Q avoids P, and by Lemma 4.6 we get the conclusion.

Now the cases for digraphs not admitting a k-min-max ordering, $k \geq 1$, are as follows. First, assume H does not contain an induced oriented cycle of net length greater than one. In this case, either H is balanced, or H has an induced oriented cycle of net length one. In the latter, H contains a symmetrically invertible pair a, b, and congruent walks P, Q from a, b to b, a respectively such that P, Q weakly avoid each other. This case is considered in the First scenario. When H is balanced and does not admit a min-max ordering, then in the Second scenario, we can apply Lemma 4.6.

Next, we assume that H contains an induced oriented cycle of net length $k > 1$. When homomorphism $f : V(H) \to \overrightarrow{C_k}$ exists, then according to Theorem A.2, H has symmetrically invertible pair a, b with $f(a) = f(b)$, and we appeal to the similar argument as in First scenario. When H is not homomorphic to $\overrightarrow{C_k}$ then we argue in the Fourth scenario Lemma 4.6 can be applied.

We conclude that if H does not admit not admit a k-min-max ordering for every $k \geq 1$, then $\mathsf{MHC}(H)$ is 1.021-approx-hard. □

References

[1] Sanjeev Arora, László Babai, Jacques Stern, and Z Sweedyk. The hardness of approximate optima in lattices, codes, and systems of linear equations. *Journal of Computer and System Sciences*, 54(2):317–331, 1997.

[2] Per Austrin. Towards sharp inapproximability for any 2-csp. *SIAM Journal on Computing*, 39(6):2430–2463, 2010.

[3] Per Austrin, Subhash Khot, and Muli Safra. Inapproximability of vertex cover and independent set in bounded degree graphs. In *Proceedings of the 24th Annual IEEE Conference on Computational Complexity, CCC 2009, Paris, France, 15-18 July 2009*, pages 74–80. IEEE Computer Society, 2009.

[4] Amotz Bar-Noy, Mihir Bellare, Magnús M Halldórsson, Hadas Shachnai, and Tami Tamir. On chromatic sums and distributed resource allocation. *Information and Computation*, 140(2):183–202, 1998.

[5] Andrei A Bulatov. Tractable conservative constraint satisfaction problems. In *Logic in Computer Science (LICS) , 2003. Proceedings. 18th Annual IEEE Symposium on*, pages 321–330. IEEE, 2003.

[6] Nadia Creignou, Sanjeev Khanna, and Madhu Sudan. *Complexity classifications of Boolean constraint satisfaction problems*, volume 7 of *SIAM monographs on discrete mathematics and applications*. SIAM, 2001.

[7] Irit Dinur and Samuel Safra. On the hardness of approximating minimum vertex cover. *Annals of Mathematics*, 162(1):439–485, 2005.

[8] Tomás Feder and Pavol Hell. List homomorphisms to reflexive graphs. *Journal of Combinatorial Theory, Series B*, 72(2):236–250, 1998.

[9] Tomas Feder, Pavol Hell, and Jing Huang. List homomorphisms and circular arc graphs. *Combinatorica*, 19(4):487–505, 1999.

[10] Tomás Feder, Pavol Hell, Peter Jonsson, Andrei Krokhin, and Gustav Nordh. Retractions to pseudoforests. *SIAM Journal on Discrete Mathematics*, 24(1):101–112, 2010.

[11] Krzysztof Giaro, Robert Janczewski, Marek Kubale, and Michał Małafiejski. A 27/26-approximation algorithm for the chromatic sum coloring of bipartite graphs. In *International Workshop on Approximation Algorithms for Combinatorial Optimization (APPROX)*, pages 135–145. Springer, 2002.

[12] Michel X Goemans and David P Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM (JACM)*, 42(6):1115–1145, 1995.

[13] Gregory Gutin, Pavol Hell, Arash Rafiey, and Anders Yeo. A dichotomy for minimum cost graph homomorphisms. *European Journal of Combinatorics*, 29(4):900–911, 2008.

[14] Gregory Gutin, Arash Rafiey, and Anders Yeo. Minimum cost homomorphism dichotomy for oriented cycles. *Graphs and Combinatorics*, 25(4):521, 2009.

[15] Magnús M Halldórsson, Guy Kortsarz, and Hadas Shachnai. Minimizing average completion of dedicated tasks and interval graphs. In *Approximation, Randomization, and Combinatorial Optimization: Algorithms and Techniques*, pages 114–126. Springer, 2001.

[16] Johan Håstad. Some optimal inapproximability results. *Journal of the ACM (JACM)*, 48(4):798–859, 2001.

[17] Pavol Hell, Monaldo Mastrolilli, Mayssam Mohammadi Nevisi, and Arash Rafiey. Approximation of minimum cost homomorphisms. In *European Symposium on Algorithms (ESA)*, pages 587–598. Springer, 2012.

[18] Pavol Hell and Jaroslav Nešetřil. On the complexity of h-coloring. *Journal of Combinatorial Theory, Series B*, 48(1):92–110, 1990.

[19] Pavol Hell and Jaroslav Nesetril. *Graphs and homomorphisms*. Oxford University Press, 2004.

[20] Pavol Hell and Arash Rafiey. The dichotomy of list homomorphisms for digraphs. In *Proceedings of the twenty-second annual ACM-SIAM symposium on Discrete Algorithms (SODA)*, pages 1703–1713. Society for Industrial and Applied Mathematics, 2011.

[21] Pavol Hell and Arash Rafiey. The dichotomy of minimum cost homomorphism problems for digraphs. *SIAM Journal on Discrete Mathematics*, 26(4):1597–1608, 2012.

[22] Pavol Hell and Arash Rafiey. Monotone proper interval digraphs and min-max orderings. *SIAM Journal on Discrete Mathematics*, 26(4):1576–1596, 2012.

[23] Klaus Jansen. Approximation results for the optimum cost chromatic partition problem. *Journal of Algorithms*, 34(1):54–89, 2000.

[24] Tao Jiang and Douglas B West. Coloring of trees with minimum sum of colors. *Journal of Graph Theory*, 32(4):354–358, 1999.

[25] Peter Jonsson and Gustav Nordh. Introduction to the maximum solution problem. In *Complexity of Constraints*, pages 255–282. Springer, 2008.

[26] Sanjeev Khanna, Madhu Sudan, Luca Trevisan, and David P Williamson. The approximability of constraint satisfaction problems. *SIAM Journal on Computing*, 30(6):1863–1920, 2001.

[27] Subhash Khot. On the power of unique 2-prover 1-round games. In *Proceedings on 34th Annual ACM Symposium on Theory of Computing, May 19-21, 2002, Montréal, Québec, Canada*, pages 767–775. ACM, 2002.

[28] Subhash Khot, Guy Kindler, Elchanan Mossel, and Ryan O'Donnell. Optimal inapproximability results for max-cut and other 2-variable csps? *SIAM Journal on Computing*, 37(1):319–357, 2007.

[29] Subhash Khot, Dor Minzer, and Muli Safra. On independent sets, 2-to-2 games, and grassmann graphs. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*, pages 576–589. ACM, 2017.

[30] Subhash Khot and Oded Regev. Vertex cover might be hard to approximate to within 2-epsilon. *J. Comput. Syst. Sci.*, 74(3):335–349, 2008.

[31] Leo G Kroon, Arunabha Sen, Haiyong Deng, and Asim Roy. The optimal cost chromatic partition problem for trees and interval graphs. In *International Workshop on Graph-Theoretic Concepts in Computer Science (WG)*, pages 279–292. Springer, 1996.

[32] Ewa Kubicka and Allen J Schwenk. An introduction to chromatic sums. In *Proceedings of the 17th conference on ACM Annual Computer Science Conference*, pages 39–45. ACM, 1989.

[33] Amit Kumar, Rajsekar Manokaran, Madhur Tulsiani, and Nisheeth K. Vishnoi. On lp-based approximability for strict csps. In *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2011, San Francisco, California, USA, January 23-25, 2011*, pages 1560–1573. SIAM, 2011.

[34] Michael Lewin, Dror Livnat, and Uri Zwick. Improved rounding techniques for the max 2-sat and max di-cut problems. In *International Conference on Integer Programming and Combinatorial Optimization (IPCO)*, pages 67–82. Springer, 2002.

[35] Akbar Rafiey, Arash Rafiey, and Thiago Santos. Toward a dichotomy for approximation of h-coloring. In *46th International Colloquium on Automata, Languages, and Programming, ICALP 2019, July 9-12, 2019, Patras, Greece*, volume 132 of *LIPIcs*, pages 91:1–91:16, 2019.

[36] Prasad Raghavendra. Optimal algorithms and inapproximability results for every csp? In *Proceedings of the fortieth annual ACM symposium on Theory of computing (STOC)*, pages 245–254. ACM, 2008.

[37] Hannes Uppman. The complexity of three-element min-sol and conservative min-cost-hom. In *International Colloquium on Automata, Languages, and Programming (ICALP)*, pages 804–815. Springer, 2013.

A Obstruction and structural characterization

The *net length* of a walk is the number of forward arcs minus the number of backward arcs. A closed walk is *balanced* if it has net length zero, otherwise, it is *unbalanced*. Note that in an unbalanced closed walk we may always choose a direction in which the net length is positive (or negative). A digraph is *unbalanced* if it contains an unbalanced closed walk (or equivalently an unbalanced cycle); otherwise it is *balanced*. It is easy to see that a digraph is balanced if and only if it admits a *labeling* of vertices by non-negative integers so that each arc goes from a vertex with a label i to a vertex with a label $i+1$. In other words, a balanced digraph H admits a homomorphism to some induced directed path $\overrightarrow{P}_k$, $k > 1$. We observe that an unbalanced digraph H has only a limited range of possible values of k for which it could have a homomorphism to induced directed cycle on k vertices ,$\overrightarrow{C}_k$, and hence, a limited range of possible values of k for which it could have a k-min-max ordering. It is easy to see that an oriented cycle C admits a homomorphism to $\overrightarrow{C}_k$ only if the net length of C is divisible by k [19]. Thus, any oriented cycle of net length $q > 0$ in H limits the possible values of k to the divisors of q. If H is balanced, one can see that H has a k-min-max ordering for some k if and only if it has a min-max ordering [22].

Theorem A.1 ([22])**.** *A digraph H admits a min-max ordering if and only if H has no induced cycle of net length greater than one and no symmetrically invertible pair.*

The deep structural Theorem A.2 characterizes the digraph admitting k-min-max ordering and it provides a forbidden obstruction characterizations for k-min-max ordering. According to this theorem, it is polynomial-time to decide whether a given digraph H admits a min-max ordering or a k-min-max ordering, $k > 1$. We use this theorem in our reduction.

Theorem A.2 ([22])**.** *Let H be a weakly connected digraph i.e., the underlying graph of H is connected. Suppose H is homomorphic to $\vec{C}_k$ under homomorphism f.*

Then H admits a k-min-max ordering if and only if it contains no induced oriented cycle of positive net length other than k, and no symmetrically invertible pair u, v with $f(u) = f(v)$.

In the following two lemmas, we study the structural properties of digraphs H, where $\mathsf{MHC}(H)$ is **NP**-complete, in two particular cases. The cases considered here are important because they are the building blocks of our hardness reductions. We first discuss the effect of two induced oriented cycles C_1, C_2 with net length $l > k > 0$, respectively, and provide some sub-structure useful in our reduction. The following lemma can be obtained from [21] (Theorem 7.3), and [22].

Lemma A.3. *Let H be a digraph and suppose H contains two induced oriented cycles C_1, C_2 of net length $l > k > 0$, respectively. Then the following hold.*

1. *H contains a symmetrically invertible pair a, b, and congruent walks P, Q from a, b to b, a respectively such that P, Q weakly avoid each other, and P and Q have at least three switches.*

2. *H contains three congruent walks P, Q, R from a, b, a to b, a, a, respectively such that :*

   - *Q has no faithful arc to P,*
   - *if there is an i-th faithful arc from Q to R, then there is no j-th faithful arc, $i + 1 \leq j$, from R to P.*

In the following lemma, we layout a structural property of balanced digraph H that does not admit a min-max ordering. This property is used in one of our hardness reductions.

Lemma A.4. *Let H be a balanced digraph which does not admit a min-max ordering. Then the following hold.*

1. *H contains a symmetrically invertible pair a, b, and congruent walks P, Q from a, b to b, a respectively such that P, Q weakly avoid each other, and P and Q have at least three switches.*

2. *H contains three congruent walks P, Q, R from a, b, a to b, a, a, respectively such that :*

   - *Q has no faithful arc to P,*
   - *if there is an i-th faithful arc from Q to R, then there is no j-th faithful arc, $i + 1 \leq j$, from R to P.*

Remark A.5. Some of the balanced digraphs satisfying item (2) belong to class (Z) described in the introduction.

Observation A.6. Suppose a, b is a symmetrically invertible pair with associated walks P and Q with exactly one switch. Then P and Q satisfy the condition (2) in Lemmas A.3 and A.4. In other words, if P has some faithful arc to Q, but Q has no faithful arc to P, then one can assume R is a walk then that starts from P and it takes the first faithful arc from P to Q and then it follows Q to the end.

B Proof of Lemma 4.8: 5 switches

Proof of Lemma 4.8

Proof. Let a_0, b_0 be an invertible pair with five switches. Let $P = a_0, a_{0,1}, a_{0,2}, \dots, a_{0,l_0}, a_1, a_{1,1}, \dots, a_{1,l_1}, a_2, \dots, a_4, a_{4,1}, \dots, a_{4,l_4}, a_5 = b_0$ and let $Q = b_0, b_{0,1}, \dots, b_{0,l_0}, b_1, b_{1,1}, \dots, b_{1,l_1}, b_2, \dots, b_4, b_{4,1}, \dots, b_{4,l_4}, b_5 = a_0$. Let $P_i = P[a_i, a_{i+1}]$ and $Q_i = Q[b_i, b_{i+1}]$. Moreover, P_{2i} has faithful arcs to Q_{2i}, but Q_{2i} has no faithful arc to P_{2i}, $0 \leq i \leq 2$ and Q_{2i+1} has faithful arcs to P_{2i+1}, but P_{2i+1} has no faithful arc to Q_{2i+1}, $0 \leq i \leq 1$. Let G be a graph with vertex set $\{x_1, \dots, x_n\}$ considered in [7, 29].

Construction of the 5-partite graph G' from G: Let $V_0, V_1, \dots, V_4$ be 5 disjoint copies of the vertices of G. For each edge $e = x_r x_s$ of G, add the edges $u_r v_s$ and $u_s v_r$ into G' where $u_r, u_s \in V_0$ (corresponding to vertices $x_r, x_s \in G$ respectively) and $v_r, v_s \in V_1$ (corresponding to $x_r, x_s \in G$ respectively). For $u_r \in V_i$, $v_r \in V_{i+1}$ ($i = 1, 2, 3, 4$, sum module 5) corresponding to $x_r \in G$, add the edge $u_r v_r$ into G'.

Construction of digraph D from G', and defining the cost function: For every edge $e = uv$ of G' with $u \in V_i$, $v \in V_{i+1}$ (sum module 5), replace e by a new path $Y_i = y_i, y_{i,1}, \dots, y_{i,l_i}, y_{i+1}$ which is congruent to P_i and Q_i, by identifying y_i with u and y_{i+1} by v. Let D be the resulting digraph. Define the cost function $c : D \times H \to \mathbb{Q}_{\geq 0}$, for Y_i as follows. $c(y_{i,j}, a_{i,j}) = c(y_{i,j}, b_{i,j}) = 0$ and $c(y_{i,j}, t) = 3|G|$ for every other vertex $t \in P \cup Q$. For every $u \in V_i$, set $c(u, a_i) = 1$, $c(u, b_i) = 0$ when $i = 0, 2, 4$ and set $c(u, a_i) = 0$, $c(u, b_i) = 1$

when $i = 1, 3$. In any other case, the cost of mapping u to a vertex in $P \cup Q$ is $3|G|$.

From a vertex cover in G to a homomorphism from D to H: Let VC be a vertex cover in G. Define the mapping $f : V(D) \to V(H)$ as follows. For every $u \in V_0$ which is a copy of vertex $x \in G$ set $f(u) = a_0$ if $x \in VC$, else set $f(u) = b_0$. For every vertex $v \in V_1$ which is a copy of vertex $x \in G$, set $f(v) = b_1$ if $x \in VC$, else $f(v) = a_1$. For every vertex $u \in V_i$, with $i = 2, 4$, if $x \notin VC$ (here u is the copy of vertex x) then set $f(u) = a_i$ else set $f(u) = b_i$. For every vertex $u \in V_3$ if $x \in VC$ then set $f(u) = b_3$ else $f(u) = a_3$ (here u is the copy of vertex x). Now consider an edge $e = uv$ in G' with $u \in V_i$, and $v \in V_{i+1}$. In D the edge e has been replaced by oriented path $Y_i = y_i, y_{i,1}, \ldots, y_{i,l_i}, y_{i+1}$ between u and v ($y_i = u$, $y_{i+1} = v$). Define f for Y_i as follows.

- If $f(y_i) = a_i$ and $f(y_{i+1}) = a_{i+1}$ then set $f(y_{i,j}) = a_{i,j}$.
- If $f(y_i) = b_i$ and $f(y_{i+1}) = b_{i+1}$ then set $f(y_{i,j}) = b_{i,j}$.
- If i is even, $f(y_i) = a_i$, and $f(y_{i+1}) = b_{i+1}$. Then, let $a_{i,l}b_{i,l+1}$ be the first faithful arc from P_i to Q_i. Set $f(y_{i,j}) = a_{i,j}$ if $j \leq l$ else $f(y_{i,j}) = b_{i,j}$.
- If i is odd, $f(y_i) = b_i$, and $f(y_{i+1}) = a_{i+1}$. Then, let $b_{i,l}a_{i,l+1}$ be the first faithful arc from Q_i to P_i. Set $f(y_{i,j}) = b_{i,j}$ if $j \leq l$ else $f(y_{i,j}) = a_{i,j}$.

Using a similar argument, as the one in the proof of Lemma 4.7, it is not difficult to see that f is a homomorphism from D to H with cost $c(f) = |VC| + 2|G|$.

From a homomorphism from D to H to a vertex cover in G: Let $f : V(D) \to V(H)$ be a homomorphism of cost less than $3|G|$. We modify f so that for every $v_i \in V_i$, $0 \leq i \leq 4$, where all are the copy of the same vertex $x \in G$, $f(v_0) = a_0$ if and only if $f(v_j) = b_j$, $1 \leq j \leq 4$.

Suppose for some $u_0 \in V_0$, $f(u_0) = a_0$ and $f(u_1) = a_1$ where $u_0 \in V_0$ and $u_1 \in V_1$, and are the copy of the same vertex $y \in G$. Let u_2, u_3, u_4 be the copy of y in V_2, V_3, V_4, respectively. Let $a_{0,l}b_{0,l+1}$ be the first faithful arc from P_0 to Q_0, and let $R_0 = P[a_0, a_{0,l}]Q[b_{0,l+1}, b_1]$.

According to the construction, let Y_i, $i = 1, 2, 3, 4$, be the oriented path in D from u_i to u_{i+1} (sum module 5) and congruent to P_i. Notice that since there is no faithful arc from P_i to Q_i, for odd i and f is a homomorphism from D to H, if $f(u_i) = a_i$ then $f(u_{i+1}) = a_{i+1}$. Likewise when $i > 0$ is even and $f(u_i) = b_i$ then $f(u_{i+1}) = b_{i+1}$. By this observation, $f(u_2) = a_2$ (f maps Y_1 to P_1). We modify f as follows. Set $f(u_1) = b_1$, and $f(u_2) = b_2$, and the new f maps Y_1 to Q_1. Set $f(u_3) = b_3$, and the new f maps Y_2 to Q_2 ($f(u_3) = b_3$) and f maps Y_3 to Q_3, and finally maps Y_4 to Q_4. A similar treatment is performed when $f(v_1) = b_1$ but $f(v_0) \neq a_0$ for $v_0 \in V_0$, and $v_1 \in V_1$ (where v_0, v_1 are copies of the same vertex in G). Moreover, for the path Y_0 between $v_0 \in V_0$, and u_1, if $f(v_0) = a_0$ then the new f maps Y_0 to R_0. Notice that by doing so f is still a homomorphism from D to H with the same cost.

Now, suppose $f(u_0) = a_0$ and $f(u_1) = b_1$ where u_0, u_1 are the copies of the same vertex $y \in G$. Then, we change the image of f so that f maps Y_i to Q_i, $i = 1, 2, 3, 4$ without increasing its values.

Therefore, we conclude that $c(f) = 2|G| + |\{u \in V_0 \mid f(u) = a_0\}|$. Now let $VC = \{x \in V(G) \mid f(u) = a_0$ where u is the copy of $x\}$. It is easy to see that VC is a vertex cover in G.

Showing the hardness 1.076 *and* 1.137. We show that it is **NP**-hard to find a homomorphism $f : V(G') \to V(H)$ with $c(f) < (1+\lambda)c(f^*)$ (here $\lambda = 0.076$, and f^* is the optimal minimum cost homomorphism from G' to H). For contradiction, suppose that there is a polynomial-time algorithm that produces such a homomorphism f. Thus, $c(f) = |VC| + 2|G|$ and $c(f^*) = |VC^*| + 2|G|$ (here VC^* is the optimal vertex cover in G). We have $|VC| + 2|G| < (1+\lambda)(|VC^*| + 2|G|)$.

Thus, $|VC| < (1+\lambda)|VC^*| + 2\lambda|G|$, and consequently, $|VC| - 2\lambda|G| < (1+\lambda)|VC^*|$. We may assume $|VC| \geq 0.639|G|$. This follows from the construction in [7]. Therefore, we have $|VC|(1 - \frac{2\lambda}{0.639}) < (1+\lambda)|VC^*|$. By setting $\frac{1+\lambda}{1-\frac{2\lambda}{0.639}} = \sqrt{2}$, and hence, $\lambda = 0.076$ we get a contradiction that the vertex cover cannot be approximated within the factor than $\sqrt{2}$ according to [7]. Therefore, we obtain $(1 + \lambda) = 1.076$ hardness result assuming $\mathbf{P} \neq \mathbf{NP}$.

Moreover, setting $\lambda = \frac{1}{7.299} \approx 0.137$, we get a contradiction to $(2 - \epsilon)$-UG-hardness for the vertex cover according to [30]. Therefore, we obtain $(1 + \lambda) = 1.13$ hardness result assuming UGC. □

C Proof of Claim 4.10: 7 switches

We extend f to $Y_{i,j}$ and $Y_{j,i}$ in order to obtain a homomorphism from D to H. To do so we consider the following cases.

Case 1. $j - i \geq 4$.

1. Suppose $u, v \notin I$. Then, for every $w \in Y_{i,j}$ set $f(w) = g_a(w)$ when i, j are even and when both are odd set $f(w) = g_b(w)$. In any of these two cases the cost of mapping $Y_{i,j}$ to $P_{i,j}$ (or $Q_{i,j}$) under f is 2. If i is even and j is odd then f maps the first portion of $Y_{i,j}$ to $P_{i,i+1}$ and then using the first faithful arc from $P_{i,i+1}$ to $Q_{i,i+1}$, the rest of the $Y_{i,j}$ is mapped to $Q_{i,j}$. Note that in this case again according to definition of the costs in (3), the cost of mapping $Y_{i,j}$ to $P_{i,j} \cup Q_{i,j}$ under f is 2. Analogously, when i is odd and j is even, f maps $Y_{i,j}$ to $Q_{i,i+1} \cup P_{i,j}$, and the cost of mapping $Y_{i,j}$ to $P_{i,j} \cup Q_{i,j}$) under f is 2.

2. Suppose $u \in I$, and $v \notin I$, and i, j both are even. Note that we have $f(u) = b_i$ and $f(v) = a_j$. Let $b_{i+1,l}a_{i+1,l+1}$ be a first faithful arc from $Q_{i+1,i+2}$ to $P_{i+1,i+2}$. Now for $w \in Y_{i,j}$ if $g_b(w)$ is before $b_{i+1,l+1}$ then set $f(w) = g_b(w)$; otherwise, set $f(w) = g_a(w)$. Observe that f is a homomorphism from $Y_{i,j}$ to $P_{i,j} \cup Q_{i,j}$, because it maps the first part of $Y_{i,j}$ to $Q_{i,j}$ according to g_b, and then using the faithful arc $b_{i+1,l}a_{i+1,l+1}$, the rest of $Y_{i,j}$ is mapped according to g_a. The cost of mapping $Y_{i,j}$ to $P_{i,j} \cup Q_{i,j}$ under f is 2.

3. Suppose $u \in I$, $v \notin I$, i is even and j is odd. Note that we have $f(u) = b_i$ and $f(v) = b_j$. In this case for every $w \in Y_{i,j}$, set $f(w) = g_b(w)$. Notice that by definition if $g_b(w) \in \{b_{i+1}, b_{j-1}\}$, then $c(w, g_b(w)) = 1$. Thus, the cost of mapping $Y_{i,j}$ to $Q_{i,j}$ under f is 2.

4. Suppose $u \in I$, $v \notin I$, and i is odd and j is even. Note that we have $f(u) = a_i$ and $f(v) = a_j$. In this case for every $w \in Y_{i,j}$, $f(w) = g_a(w)$. Notice that by definition if $g_a(w) \in \{a_{i+1}, a_{j-1}\}$, then $c(w, g_a(w)) = 1$. Thus, the cost of mapping $Y_{i,j}$ to $P_{i,j}$ under f is 2.

5. Suppose $u \in I$, $v \notin I$, and both i and j are odd. Note that we have $f(u) = a_i$ and $f(v) = b_j$. Let $a_{i+1,l}b_{i+1,l+1}$ be the first faithful arc from $P_{i+1,i+2}$ to $Q_{i+1,i+2}$. Now for $w \in Y_{i,j}$ if $g_a(w)$ is before $a_{i+1,l+1}$ then set $f(w) = g_a(w)$; otherwise, set $f(w) = g_b(w)$. Observe that f is a homomorphism from $Y_{i,j}$ to $P_{i,j} \cup Q_{i,j}$, because it maps the first part of $Y_{i,j}$ to $P_{i,j}$ according to g_a, and then using the faithful arc $a_{i+1,l}b_{i+1,l+1}$, the rest of $Y_{i,j}$ is mapped according to g_b. The cost of mapping $Y_{i,j}$ to $P_{i,j} \cup Q_{i,j}$ under f is 2.

6. Suppose $u \notin I$, $v \in I$, and i, j both even. Analogous to (2) we define f.

7. Suppose $u \notin I$, $v \in I$, and i, j both odd. Analogous to (5) we define f.

8. Suppose $u \notin I$, $v \in I$, and i is even and j is odd. Analogous to (4) we define f.

9. Suppose $u \notin I$, $v \in I$, and i is odd and j is even. Analogous to (3) we define f.

Case 2. $j - i = 1$.

1. Suppose $u \in I$ and $v \notin I$. Then, we have $f(u) = b_i$ and $f(v) = b_{i+1}$. Now set $f(w) = g_b(w)$ for every $w \in Y_{i,i+1}$.

2. Suppose $v \in I$ and $u \notin I$. Then we have $f(u) = a_i$ and $f(v) = a_{i+1}$. Now set $f(w) = g_a(w)$ for every $w \in Y_{i,i+1}$.

3. Suppose $v, u \notin I$, and i is even. Then, we have $f(u) = a_i$ and $f(v) = b_{i+1}$. Let $a_{i,l}b_{i,l+1}$ be the first faithful arc from $P_{i,i+1}$ to $Q_{i,i+1}$. Now for every $w \in Y_{i,i+1}$, if $g_a(w)$ is before $a_{i,l+1}$ then set $f(w) = g_a(w)$; otherwise, set $f(w) = g_b(w)$.

4. Suppose $v, u \notin I$, and i is odd. Then, we have $f(u) = b_i$ and $f(v) = a_{i+1}$. Let $b_{i,l}a_{i,l+1}$ be the first faithful arc from $Q_{i,i+1}$ to $P_{i,i+1}$. Now for every $w \in Y_{i,i+1}$, if $g_b(w)$ is before $b_{i,l+1}$ then set $f(w) = g_b(w)$; otherwise, set $f(w) = g_a(w)$.

Case 3. $j - i = 2$.

1. Suppose $u, v \notin I$, i is even. Then, we have $f(u) = a_i$ and $f(v) = a_{i+2}$. Let $a_{j+2,l}b_{j+2,l+1}$ be the first faithful arc from $P_{j+2,j+3}$ to $Q_{j+2,j+3}$. Now for $w \in Y_{j,i}$, set $f(w) = g_a(w)$ if $g_a(w)$ is before $a_{j+2,l+1}$; otherwise, set $f(w) = g_b(w)$. Note that f is a homomorphism from $Y_{j,i}$ to $P_{j,i} \cup Q_{j,i}$, and the cost of mapping $Y_{j,i}$ to H under f is 2.

2. Suppose $u, v \notin I$, and i is odd. Analogous to (1) define f.

3. Suppose $u \in I$, $v \notin I$, and i is even. Then, we have $f(u) = b_i$ and $f(v) = a_{i+2}$. In this case for $w \in Y_{j,i}$ set $f(w) = g_a(w)$.

4. Suppose $u \in I$, $v \notin I$, and i is odd. In this case for $w \in Y_{j,i}$ set $f(w) = g_b(w)$.

5. Suppose $u \notin I$, $v \in I$, and i is even. Now for $w \in Y_{j,i}$, set $f(w) = g_b(w)$.

6. Suppose $u \notin I$, $v \in I$, and i is even. Now for $w \in Y_{j,i}$, set $f(w) = g_a(w)$.

Case 4. $j - i = 3$.

1. Suppose $u, v \notin I$ and i is even. For every $w \in Y_{j,i}$ set $f(w) = g_b(w)$.

2. Suppose $u, v \notin I$ and i is odd. For every $w \in Y_{j,i}$ set $f(w) = g_a(w)$,

3. Suppose $u \in I$, $v \notin I$, and i is even. Notice that we have $f(u) = b_i$ and $f(v) = b_j$. Let $b_{j+1,l}a_{j+1,l+1}$ be the first faithful arc from $Q_{j+1,j+2}$ to $P_{j+1,j+2}$. Now for $w \in Y_{j,i}$, set $f(w) = g_b(w)$ if $g_b(w)$ is before $b_{j+1,l+1}$; otherwise, set $f(w) = g_a(w)$. The cost of mapping $Y_{j,i}$ to $Q_{j,i}$ under f is 2. This is because $k \geq 7$, and therefore f maps a vertex of $Y_{j,i}$ to b_{i-1} and the cost of this mapping is 1.

4. Suppose $u \in I$, $v \notin I$, and i is odd. Let $a_{j+1,l}b_{1+2,l+1}$ be the first faithful arc from $P_{j+1,j+2}$ to $Q_{j+1,j+2}$. Now for $w \in Y_{j,i}$, set $f(w) = g_a(w)$ if $g_a(w)$ is before $a_{j+1,l+1}$; otherwise, set $f(w) = g_b(w)$. The cost of mapping $Y_{j,i}$ to $P_{j,i}$ under f is 2.

5. Suppose $u \notin I$, $v \in I$, and i is even. Let $b_{i-1,l}a_{i-1,l+1}$ be a last faithful arc from $Q_{i-1,i}$ to $P_{i-1,i}$. Now for $w \in Y_{j,i}$, if $g_a(w)$ is not after $b_{i-1,l}$ then set $f(w) = g_a(w)$; otherwise, $f(w) = g_b(w)$. Similar to argument in (3,4), f is a homomorphism of total cost 2.

6. Suppose $u \notin I$, $v \in I$, and i is odd. Let $a_{i-1,l}b_{i-1,l+1}$ be a last faithful arc from $P_{i-1,i}$ to $Q_{i-1,i}$. Now for $w \in Y_{j,i}$, if $g_b(w)$ is not after $a_{i-1,l}$ then set $f(w) = g_b(w)$; otherwise, $f(w) = g_a(w)$. Similar to argument in (3,4), f is a homomorphism of total cost 2.

D Inapproximability results for graphs

We use Lemma 4.7, to prove Theorem 1.6. First, we need two structural theorems on the polynomial cases of MHC(H) when H is a graph.

Theorem D.1. *Let H be a bipartite graph. Then MHC(H) is polynomial-time solvable if and only if H admits a min-max ordering (i.e., does not contain an induced cycle of length at least six, or a bipartite claw, or a bipartite net, or a bipartite tent, see Figure 3).*

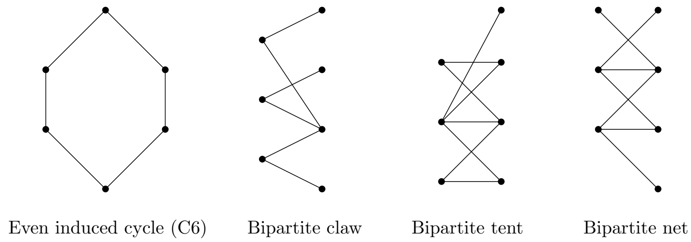

Figure 3: Obstruction to min-max ordering in bipartite graphs, and making $\mathsf{MHC}(H)$ **NP**-complete.

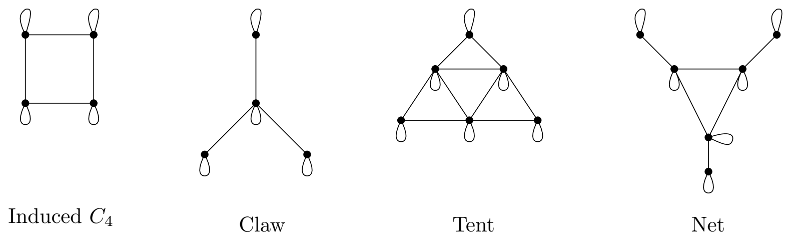

Figure 4: Obstruction to min-max ordering in reflexive graphs, and making $\mathsf{MHC}(H)$ **NP**-complete.

Theorem D.2. *Let H be graph with at least one self-loop vertex. Then MHC(H) is polynomial-time solvable if and only if H is reflexive (every vertex has a self-loop) and admits a min-max ordering (i.e., does not contain an induced cycle of length at least four, or a claw, or a net, or a tent, see Figure 4).*

Theorem D.3 (Restatement of Theorem 1.6)**.** *Let H be a graph where MHC(H) is **NP**-complete. Then MHC(H) is at least* 1.128*-approx-hard and* 1.242*-UG-hard.*

Proof. We consider two cases where H is irreflexive (no vertex has a self-loop) and the case where H has a vertex with a self-loop.

H **is irreflexive:** Without loss of generality, we can assume H is bipartite, as otherwise, CSP(H) is **NP**-complete (due to [18]). Hence, $\mathsf{LHC}(H)$ is **NP**-complete, and by Observation 1.7, $\mathsf{MHC}(H)$ does not have any approximation. By this argument and by Lemma 4.1 (hardness of approximation for sub-graph), if a sub-graph of H is not bipartite, again $\mathsf{MHC}(H)$ does not admit any approximation. Therefore, we continue by assuming that H is bipartite. Moreover, when the bipartite graph H contains an induced even cycle of length at least 6, $\mathsf{LHC}(H)$ is **NP**-complete due to [9], and hence, by Observation 1.7 $\mathsf{MHC}(H)$ admits no approximation. By Theorem D.1 and Lemma 4.1, it remains to consider the cases where H is either bipartite claw, bipartite tent, or bipartite net. It is also easy to transform the input instance G, H, c of $\mathsf{MHC}(H)$ into D, H', c of MHC (H') by orienting all the edges of G from left to right and obtain digraph D, as well as orienting all the edges of H from left to right and obtain digraph H'. The cost function stays the same. Thus, we will have bipartite di-claw, bipartite di-tent, and bipartite di-tent (see Figure 5). Each bipartite di-claw, bipartite di-tent, and bipartite di-tent has a symmetrically invertible pair with three switches (see Figure 5). Thus, by Lemma 4.7, MHC (H') when H' is one of the bipartite di-claw, di-tent, di-net, $\mathsf{MHC}(H')$ is 1.128-approx-hard, and it is 1.242-UG-hard. Therefore, when H is one of the bipartite claw, tent, net then $\mathsf{MHC}(H)$ is 1.128-approx-hard, and it is 1.128-UG-hard.

H **has vertices with self-loops:** We show that H must be reflexive; meaning every vertex has a loop. Otherwise, H must contain an induced sub-graph $H_1 = \{aa, ab\}$ where b does not have a self-loop (recall that we assume H is connected). As we mention in the introduction, Vertex Cover problem is an instance of MHC

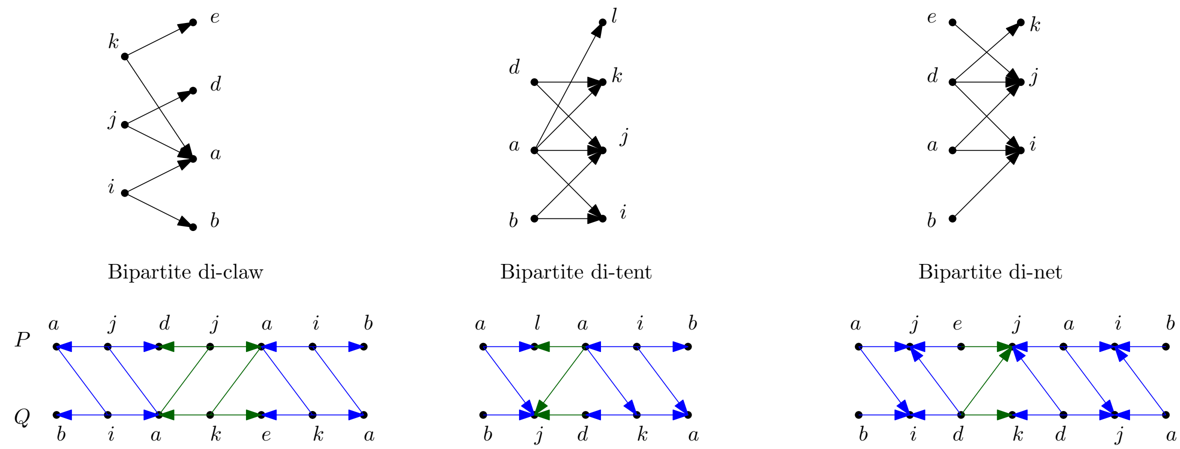

Figure 5: Invertible pair for bipartite di-claw, di-tent, and di-net with three switches. The blue and green colors show the switches.

(H_1). Vertex Cover is $(\sqrt{2}-\epsilon)$-approx-hard and $(2-\epsilon)$-UG-hard for every $\epsilon > 0$. Therefore, MinHOM(H_1) is $(\sqrt{2}-\epsilon)$-approx-hard and $(2-\epsilon)$-UG-hard for every $\epsilon > 0$. By the hardness of approximation for sub-graphs (Lemma 4.1), we obtain better hardness bounds than the claim of the theorem. Therefore, we may assume that H is reflexive henceforth.

If H contains an induced cycle of length at least 4 (when removing the self-loops), LHC(H) is **NP**-complete due to [8], and hence, by Observation 1.7, MHC(H) does not admit any approximation. Thus, by Theorem D.2 and Lemma 4.1, we need to consider the case where H is a claw, tent or net. When H is any of these three graphs, H contains an invertible pair (see Figure 4). By a similar treatment, if H is reflexive and MHC(H) is **NP**-complete then MHC(H) is 1.128-approx-hard and 1.242-UG-hard. This completes the proof of the theorem.

□

E Special cases: oriented cycles and oriented trees

We start this section by some necessary technical definition. Before proceeding, we need a technical definition. Let $P = x_0, x_1, \ldots, x_n$ be a walk in H of net length $k \geq 0$. We say that P is *constricted from below* if the net length of any prefix $P[x_0, x_j]$ is non-negative, and is *constricted from above* if the net length of any prefix is at most k. We also say that P is *constricted* if it is constricted both from below and from above. For a walk P in digraph H let P^{-1} denote the reverse of P.

We use the following well-known lemma (for a proof, see Lemma 2.36 in [19]).

Lemma E.1. *Let P_1 and P_2 be two constricted walks of net length r. Then there is a constricted path P of net length r that admits a homomorphism f_1 to P_1 and a homomorphism f_2 to P_2, such that each f_i takes the starting vertex of P to the starting vertex of P_i and the ending vertex of P to the ending vertex of P_i.*

We shall call P a *common pre-image* of P_1 and P_2. We often use the image of P under f_1, f_2 to obtain P_1', P_2' which we call the *embedded pre-images* of P_1, P_2.

E.1 Oriented cycles

We first need the following technical lemma about induced oriented cycles.

Lemma E.2. *Let C be an induced oriented cycle, where MHC (C) is **NP**-complete. Then C contains three congruent walks P, Q, R from a, b, b to b, a, b, respectively such that:*

- *P has no faithful arc to Q,*
- *if there is an i-th faithful arc from P to R, then there is no j-th faithful arc, $i+1 \leq j$, from R to Q.*

Proof. The result in [14], shows that if an oriented cycle C, is not balanced, then MHC (C) is polynomial-time solvable. Thus, we may assume that C is balanced and does not admit a min-max ordering. Let a, b be a symmetrically invertible pair with the walks P, Q from a, b to b, a respectively that P, Q avoid each other. Notice that an induced oriented cycle is chordless by definition. Since C is induced, there is no faithful arc from P to Q and there is no faithful arc from Q to P, as otherwise, one can find a chord in C; a contradiction to C being induced. Let $\overrightarrow{P}_k$ be a directed path on $1, 2, \ldots, k$.

Since C is balanced, by definition, there is homomorphism $f : C \to \overrightarrow{P}_k$, where k is the number of levels in C. We may assume that $f(a) = f(b) = 1$ (lowest level of C) Let c be a vertex of C with $f(c) = k$ (c is on the highest level of C). Let d be the vertex on Q corresponding to c. Let $P = P_1P_2$, and $Q = Q_1Q_2$, where P_1 is part of P from a to c, P_2 is part of P from c to b, and Q_1 is part of Q from b to d, Q_2 is part of Q from d to a. Observe that $P_2^{-1}, Q_2^{-1}, Q_1^{-1}$ are constricted walks and have net length $k-1$.

First notice that if two congruent walks avoid each other, then their reverses also avoid each other. So P^{-1}, Q^{-1} avoid each other. Moreover, if P, Q avoid each other then their embedded pre-images also avoid each other. For two walks R, S let RS be the walks obtained by identifying the end of R with the beginning of S.

Notice that by Lemma E.1, $P_2^{-1}, Q_2^{-1}, Q_1^{-1}$ are constricted and have embedded pre-images P_2', Q_2', R. Now, $P_3 = P_1P_2'$, $Q_3 = Q_1Q_2'$ and $R_1 = Q_1R$ are the three desired walks from a, b, b to b, b, a. This is because P_2', Q_2' avoid each other, and there is no faithful arc from P_2' to R (unless at the end) because C is chordless. □

Theorem E.3. *Let C be an induced oriented cycle that does not admit a min-max ordering. Then*

1. *MHC(H) is $(\sqrt{2} - \epsilon)$-approx hard for every $\epsilon > 0$.*
2. *MHC(H) is $(2 - \epsilon)$-UG-hard for any $\epsilon > 0$.*

Proof. By Lemma E.2 there exists three congruent walks P, Q, R satisfying the conditions of Lemma 4.6, and hence, the theorem is established. □

Proof of Lemma A.4 We may assume that we have a symmetrically invertible pair a, b and corresponding walks P, Q with no faithful arcs between P and Q. It is not difficult to see that we may assume that a, b are on the lowest level of P and Q. Now by similar argument, as in the proof of Lemma E.2, one can obtained the three desired walks P, Q, R. □

E.2 Oriented trees

It was shown in [17] that when H is any of the bipartite claw, bipartite net, bipartite tent (see Figure 6 when we ignore the direction of the arcs) then MHC(H) admits a constant approximation algorithm. Notice that by adding one extra arc to any of the digraphs depicted in Figure 6 (i.e. 37 to Di-net,Di-tent, and 36 to Di-claw) the resulting digraph admits a min-max ordering. Therefore, is not difficult to see that this constant factor is 2, by analysing the approximation ratio of the approximation algorithm in paper [17]. In fact from the analysis of the approximation algorithm in [17, 35] one can conclude that if by adding one extra arc to H, the resulting (di)graph admits a min-max then MHC(H) admits a 2-approximation algorithms. Using this argument one can obtain the following :

Proposition E.4. *There exists an infinite family of oriented trees T with the following properties:*

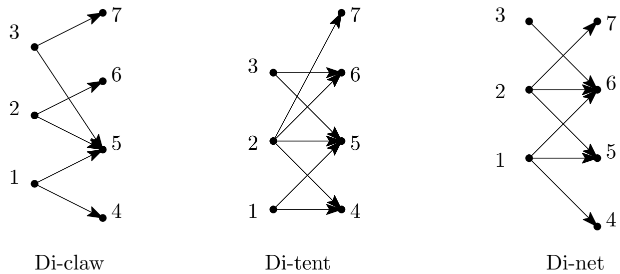

Figure 6: Obstruction to min-max ordering in bipartite digraphs; making $\mathsf{Strict\text{-}CSP}$ (H) **NP**-complete.

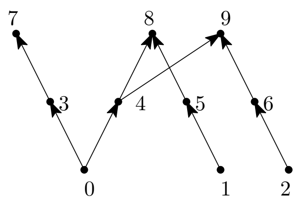

Figure 7: Example of oriented tree

- *T does not admit a min-max ordering, and hence,* MHC *(T) is* ***NP****-complete,*
- MHC *(T) admits a 2-approximation algorithm.*

Proof. We start with the smallest such oriented tree, establishing the proposition, and then generalize it to bigger trees. Let T be the oriented tree depicted in Figure 6. The ordering $0, 1, 2, 3, 4, 5, 6, 7, 8, 9$ is a min ordering of T. By adding arc 59, the same ordering becomes a min-max ordering. Now the idea in [35], is to formulate MHC (T) as an LP, with the cost function of minimizing the MHC (H). For each vertex $v \in G$ and $i \in T$, there is a variable $0 \leq X_{v,i} \leq 1$. In the LP there are some constraints that avoid mapping an arc of the input digraph G to arc 59. Once the LP returns the fractional values, a uniform random variable X from real interval $[0, 1]$, is used to round the fractional variables $X_{v,i}$ to 1 if $X \leq X_{v,i}$, and otherwise, to round $X_{v,i}$ to zero. If $X_{v,i}$ is set to 1 then it means in the final homomorphism, f we set $f(v) = i$. If for an arc uv of G, according to this rounding, we set $f(u) = 5$ and $f(v) = 9$, then $f(u)$ is set to 4 and the image of any in-neighbor of u, is set to 0. From the analysis of the algorithm in [35], one can conclude that this procedure, yields a homomorphism whose total cost is at most twice the optimal cost.

Now consider the oriented tree T_1, obtaining from T, by replacing arcs $03, 04, 15, 26$ by arbitrary constricted oriented paths P_1, P_2, P_3, P_4 (respectively) each of net length r (see Figure 8). By using the same argument as above one can show that MHC (T_1) admits a 2-approximation algorithm.

□

F Generalization to r-uniform $(r > 2)$ hypergraphs

In this section, we introduce an auxiliary directed graph and paired digraph for a given hypergraph $\mathcal{H}$, aimed at capturing its structural properties when it lacks a min-max ordering.

Definition F.1 (Projection)**.**

– For an arc $\overline{\mathbf{e}}$ in $\mathcal{H}$, let $\overline{\mathbf{e}}_t$ denote the elements of $\overline{\mathbf{e}}$ appearing in coordinate t of $\overline{\mathbf{e}}$. Let $\overline{\mathbf{e}}_{t,r}$ represent the pair (a, b) where a and b appear in coordinates t and r of $\overline{\mathbf{e}}$, respectively. In other words, the projection of $\overline{\mathbf{e}}$ over t and r is (a, b).

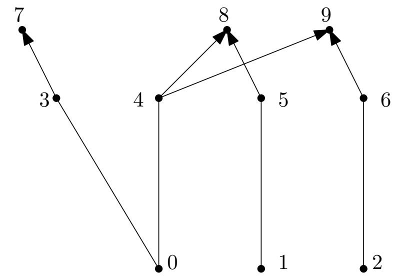

Figure 8: Each solid line is a constricted oriented path, with net length r.

– For two given arcs $\overline{\mathbf{e}}^1$ and $\overline{\mathbf{e}}^2$ of $\mathcal{H}$ with the same size r, let $P_t(\overline{\mathbf{e}}^1, \overline{\mathbf{e}}^2)$ be the set of arcs $\overline{\mathbf{e}}$ of size r in $\mathcal{H}$ such that $\overline{\mathbf{e}}_t \in \{\overline{\mathbf{e}}_t^1, \overline{\mathbf{e}}_t^2\}$.

Definition F.2 (H and H^+)**.** Let $\mathcal{H}$ be a hypegraph. Define H to be the digraph obtained from $\mathcal{H}$ as follows. The vertex set of H is the same as vertex set of $\mathcal{H}$. The arc set of $A(H) = \{ab \mid (a, b) = \overline{\mathbf{e}}_{r,s}$ for some arc $\overline{\mathbf{e}}$ in $\mathcal{H}\}$

Define H^+ to be the digraph with the vertex set $\{(a_1, a_2) \mid a_1, a_2 \in V(\mathcal{H})\}$ and consisting of arcs $(a_1, a_2)(b_1, b_2)$ so that there exist coordinates r and s where the following hold.

- there exist $\overline{\mathbf{e}}^1, \overline{\mathbf{e}}^2$ of $\mathcal{H}$ both of the same size so that $\overline{\mathbf{e}}^l_{r,s} = (a_l, b_l)$, $1 \leq l \leq 2$
- $\forall\ \overline{\boldsymbol{\omega}} \in P_s(\overline{\mathbf{e}}^1, \overline{\mathbf{e}}^2)$ at least one of the $\overline{\boldsymbol{\omega}}_{r,s} \neq (a_1, b_2)$ and $\overline{\boldsymbol{\omega}}_{r,s} \neq (a_2, b_1)$ holds.

Every directed path $W \in H^+$ gives rise to two walks P and Q in H such that P and Q weakly avoid each other. We denote this pair of walks as $W = (P, Q)$.

Lemma F.3. *Let $\mathcal{H}$ be a hypergraph. Suppose there exists a directed path $W = (P, Q)$ in H^+ from (a, b) to (b, a) (with P from a to b and Q from b to a) such that Q avoids P, but P has a faithful arc to Q. Then MHC($\mathcal{H}$) is $(\sqrt{2} - \epsilon)$-approx-hard.*

Proof. Let G be an arbitrary graph. We may assume that G is the graph used in [7]. Orient each edge of G arbitrary and obtain digraph D. Now replace every arc uv in D by a path S_{uv} which is congruent with P and has new vertices except u and v. Now we construct an instance of $\mathsf{MHC}(\mathcal{H})$ as follows. Consider S_{uv} in D, and walks P and Q as in the statement of the lemma. Let aa', bb' be the i-th arcs of P and Q. According to the construction of H^+, there are arcs $\overline{\mathbf{e}}^1, \overline{\mathbf{e}}^2 \in \mathcal{H}$ so that $\overline{\mathbf{e}}^1_{r,s} = (a, a')$, and $\overline{\mathbf{e}}^2_{r,s} = (b, b')$. We replace the i-th arc of S_{uv} with an arc $\overline{\mathbf{x}} = (x_1, x_2, \dots, x_r)$ of the same size as $\overline{\mathbf{e}}^1$. Let $\mathcal{G}$ be the resulting hypergraph . The cost function c for vertices of $\overline{\mathbf{x}}$ is $c(x_j, a_j) = 0$ if $a_j \in \{\overline{\mathbf{e}}^1_j, \overline{\mathbf{e}}^2_j\}$ and for every other case the cost is $|G|$. Moreover, $c(u, a) = c(v, a) = 1$, $c(u, b) = c(v, b) = 0$ and for $d \neq \{a, b\}$, $c(u, d) = c(v, d) = |G|$.

Let VC be a vertex cover in G, then we define a homomorphism $f : V(\mathcal{G}) \to V(\mathcal{H})$ as follows. For every vertex $u \in \mathcal{G} \cap D$, set $f(u) = a$ if $u \in VC$, otherwise set $f(v) = b$. For every internal vertex x_j which is in arc $\overline{\mathbf{x}} = (x_1, x_2, \dots, x_r)$ corresponding to the i-th arc of S_{uv}, if $f(u) = a$ and $f(v) = b$, we set $f(x_j) = a_j$ where $a_j = \overline{\mathbf{e}}^1_j$, where $\mathbf{e}^1$ is the arc in $\mathcal{H}$ corresponding to the i-th arc of P. On the other hand, if $f(u) = b$ and $f(v) = a$, we set $f(x_j) = b_j$ with $b_j = \overline{\mathbf{e}}^2_j$, where $\overline{\mathbf{e}}^2$ is the i-th arc corresponding to the i-th arc of Q. By this mapping, the arc of $\mathcal{G}$ corresponding to the i-th arc in S_{uv} is mapped to arc in $\mathcal{H}$ corresponding to the i-th arc in P (or Q).

If $f(u) = a$ and $f(v) = a$, then let the arc $a_l b_{l+1}$ be a faithful arc from P to Q. Now, the first $l - 1$ arcs in $\mathcal{G}$ corresponding to the S_{uv} are mapped to the first $l - 1$ arcs corresponding of $\mathcal{H}$ corresponding to P, and then the arc in $\mathcal{G}$ corresponding to the l-th arc of S_{uv} is mapped to the arc of $\mathcal{H}$ corresponding to the faithful arc $a_l b_{i+1}$, and the rest rest of the arc in $\mathcal{G}$ corresponding to the the rest of the arcs in S_{uv} are mapped to the arcs of $\mathcal{H}$ corresponding to the arcs in Q after the l-arcs of Q.

It is now straightforward to see that f is a homomorphism from $\mathcal{G}$ to $\mathcal{H}$ with a total cost $|VC|$. Now similar to the argument in the proof of Theorem 4.6 we can show that $\mathsf{MHC}(\mathcal{H})$ is $(\sqrt{2}-\epsilon)$-approx-hard for any $\epsilon > 0$.

□

The definitions of H and H^+ derived from $\mathcal{H}$, along with the argument presented in the proof of Theorem F.3, enable us to extend the theorems from digraphs to hypergraphs.

Theorem F.4. *Let $\mathcal{H}$ be a hypergraph. Suppose H^+ contains a directed path $W = (P, Q)$ from (a, b) to (b, a) such that P and Q have k-switches $(k \geq 1)$. Then MHC($\mathcal{H}$) is 1.021-approximation hard.*

We conclude this section by providing a hardness of approximation result for the case where the target structure consists of more than one hypergraph.

Lemma F.5. *There is hypergraph $\mathcal{H}$ with more than one k-uniform hypergraph where MHC($\mathcal{H}$) does not admits any constant approximation algorithm.*

Proof. It was noted in [26] that Min Horn Deletion problem does not admit a constant approximation algorithm. Min Horn Deletion problem consists of clauses in which at most one literal appear to be positive. The goal is to find an assignment to minimize the number of variables assigned to true. This is equivalent to consider clauses of form $(x \vee \neg y \vee \neg z) \wedge (\neg u \vee \neg v) \wedge (w \vee \neg p) \wedge (\neg p)$. If we translate this a homomrphism then we will have relation $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_3$ of arity $3, 2, 2$ respectively from set $\{x_1, x_2, \ldots, x_n\}$ and target relations $\mathbb{H}_1 = \{((0,0,0), (0,0,1), (0,1,0), (0,1,1), (1,0,1), (1,1,0), (1,1,1)\}$ and $\mathbb{H}_2 = \{(0,1), (1,0), (1,1)\}$, and $\mathbb{H}_3 = \{(0,0), (0,1), (1,1)\}$. The goal is to find an assignment so that every tuple in $\mathbb{G}_i$ is mapped to its corresponding $\mathbb{H}_i$, $i = 1, 2, 3$.

Now define hypergraph $\mathcal{H}$ consists of hypergraphs $\mathcal{H}_1 = \{(0,0,0,a), (0,0,1,a), (0,1,0,a), (0,1,1,a), (1,0,1,a), (1,1,0,a), (1,1,1,a)\}$ and $\mathcal{H}_2 = \{(0,1), (1,0), (1,1)\}$, and $\mathcal{H}_3 = \{(0,0,a), (0,1,a), (1,1,a)\}$. Moreover, define input hypergraph $\mathcal{G}$, where for each tuple $(x, y, z) \in \mathbb{G}_1$ add (x, y, z, ω) into $\mathcal{G}$, for every $(x, y) \in \mathbb{G}_2$ add (x, y, ω) to $\mathcal{G}$, and for every $(x, y) \in \mathbb{G}_3$ add (x, y, ω) to $\mathcal{G}_3$. Define the cost function $c(\omega, a) = 0$, and $c(x, 0) = 0$ and $c(x, 1) = 1$ for every $x \neq \omega$. Then $\mathsf{MHC}(\mathcal{H})$ is equivalent to Min Horn Deletion problem, and hence, it does not admit a constant approximation algorithm. □