On the Copy Complexity of Width 3 Horn Constraint Systems

K. Subramani[1(✉)], P. Wojciechowski[1], and Alvaro Velasquez[2]

[1] LDCSEE, West Virginia University, Morgantown, WV, USA
k.subramani@mail.wvu.edu, pwojciec@mix.wvu.edu
[2] Air Force Research Laboratory, Rome, NY, USA
alvaro.velasquez.1@us.af.mil

Abstract. In this paper, we analyze the copy complexity of unsatisfiable width 3 Horn constraint systems, under the ADD refutation system. Recall that a linear constraint of the form $\sum_{i=1}^{n} a_i \cdot x_i \geq b$, is said to be a Horn constraint if all the $a_i \in \{0, 1, -1\}$ and at most one of the a_is is positive. A conjunction of such constraints is called a Horn constraint system (HCS). An HCS is said to have width 3, if there are at most 3 variables with non-zero coefficients per constraint. Horn constraints arise in a number of domains including but not limited to program verification, power systems, econometrics, and operations research. The ADD refutation system is both **sound** and **complete**. Additionally, it is the simplest and most natural refutation system for refuting the feasibility of a system of linear constraints. The copy complexity of an infeasible linear constraint system (not necessarily Horn) in a refutation system is the minimum number of times each constraint needs to be replicated, in order to obtain a read-once refutation. In this paper, we analyze width 3 HCSs from the perspective of copy complexity.

1 Introduction

This paper is concerned with the problem of determining bounds on the **copy complexity** of Horn constraint systems (HCSs) under the ADD refutation system [10]. A linear constraint of the form $\sum_{i=1}^{n} a_i \cdot x_i \geq b$, $b \in \mathbb{Z}$, is said to be Horn, if $\forall i, a_i \in \{0, 1, -1\}$ and at most one of the $a_i = 1$. A conjunction of such constraints is called a Horn constraint system (HCS). Horn constraints arise in a number of application domains such as program verification [2,3], lattice programming [9] and econometrics. The ADD refutation system is a refutation system with a single inference rule, viz., if two constraints l_1 and l_2 are part of the HCS or can be inferred from the HCS, then so can their sum. It is well-known that the ADD refutation system is both sound and complete from the perspective of establishing infeasibility in polyhedral constraint systems [10]. Furthermore,

K. Subramani—This research was supported in part by the Air-Force Office of Scientific Research through Grant FA9550-19-1-0177 and in part by the Air-Force Research Laboratory, Rome through Contract FA8750-17-S-7007.

B. Konev and G. Reger (Eds.): FroCoS 2021, LNAI 12941, pp. 63–78, 2021.
https://doi.org/10.1007/978-3-030-86205-3_4

this system enables the extraction of the actual refutation. When it comes to establishing infeasibility, the goal is clearly to find "short" certificates, since such certificates can be effectively verified. However, not all constraint systems have compact certificates in the ADD refutation system. In our quest for compactness, we attempt to minimize the number of times a constraint can be used by the refutation system, in order to infer a contradiction. This leads to the notion of copy complexity of a constraint system under the ADD refutation system. For the rest of the paper, we will assume that the constraint system under consideration is Horn and that the refutation system is the ADD refutation system (see Sect. 2). Accordingly, we will use the phrase "copy complexity" without reference to the accompanying refutation system.

The problem of determining the copy complexity of an HCS is known to be **NP-hard** [8].

In this paper, we investigate width 3 HCSs. In most program verification applications, the width of Horn clauses or constraints is bounded by a small constant. Accordingly, this investigation is well-motivated.

2 Statement of Problems

In this section, we define the problems studied in this paper.

Definition 1. *A system of constraints* $\mathbf{A} \cdot \mathbf{x} \geq \mathbf{b}$ *is said to be a Horn Constraint system (HCS) if:*

1. *The entries in* $\mathbf{A}$ *belong to the set* $\{0, 1, -1\}$.
2. *Each row of* $\mathbf{A}$ *contains at most one positive entry.*
3. $\mathbf{x}$ *is a real valued vector.*
4. $\mathbf{b}$ *is an integral vector.*

In the constraint $\mathbf{a} \cdot \mathbf{x} \geq b_j$, b_j is called the **defining constant**. If a Horn constraint has at most w non-zero coefficients, then it is called a width w Horn constraint. A system of width w Horn constraints is known as a width w HCS. In a Horn constraint we refer to the terms x_i and $-x_i$ as literals.

If a Horn constraint has only one non-zero coefficient, then it is called an absolute constraint. If that coefficient is 1, then it is called a positive absolute constraint.

We are interested in certificates of infeasibility. In this paper, we utilize an inference rule known as the **ADD rule** [10]. This inference rule derives a new constraint by summing a pair of constraints (either from the original system or derived by previous inferences) and is defined as follows:

$$\text{ADD} : \frac{\sum_{i=1}^{n} a_i \cdot x_i \geq b_1 \qquad \sum_{i=1}^{n} a'_i \cdot x_i \geq b_2}{\sum_{i=1}^{n} (a_i + a'_i) \cdot x_i \geq b_1 + b_2} \tag{1}$$

This inference rule plays a similar role to the role played by resolution in clausal formulas.

Using Rule (1), we can now define a linear refutation.

Definition 2. *A* linear refutation *is a sequence of applications of the ADD rule that results in a contradiction of the form* $0 \geq b$, $b > 0$.

The form of refutation defined in Definition 2 is both sound and complete when used as a proof system for linear feasibility. It is sound since any assignment that satisfies the constraints used by an application of the ADD rule also satisfies the constraint derived by that application. Additionally, ADD rule based linear refutation is **complete**. This means that repeated application of the ADD rule will eventually result in a contradiction of the form: $0 \geq b$, $b > 0$ for any linearly infeasible system. The completeness of ADD rule based linear refutations was established by Farkas [4], in a lemma that is famously known as Farkas' Lemma for systems of linear inequalities [12].

Of particular interest is a restricted form of refutation known as read-once refutation.

Definition 3. *A* **read-once** *refutation is a refutation in which each constraint, l, can be used in only one inference. This applies to constraints present in the original systems and those derived as a result of previous inferences.*

Example 1. Consider the HCS **H** defined by System (2).

$$\begin{array}{cc} l_1 : x_1 - x_2 - x_3 \geq 0 & l_2 : x_2 - x_3 \geq -1 \\ l_3 : x_3 - x_1 \geq 1 & l_4 : x_3 \geq 1 \end{array} \qquad (2)$$

System (2) has the following read-once refutation:

1. Apply the ADD rule to l_1 and l_2 to get $l_5 : x_1 - 2 \cdot x_3 \geq -1$.
2. Apply the ADD rule to l_5 and l_3 to get $l_6 : -x_3 \geq 0$.
3. Apply the ADD rule to l_6 and l_4 to get the contradiction $0 \geq 1$.

We can now define copy complexity in terms of read-once refutations.

Definition 4. *A HCS* **H** *has* **copy complexity** *k if k is the smallest integer for which there exists a multi-set of Horn constraints,* $\mathbf{H}'$ *such that:*

1. *Every constraint in* **H** *appears at most k times in* $\mathbf{H}'$.
2. *Every constraint in* $\mathbf{H}'$ *appears in* **H**.
3. $\mathbf{H}'$ *has a read-once refutation using the ADD rule.*

In this paper, we examine the following problems related to copy complexity:

Definition 5. *The copy complexity (CC_D) problem: given an HCS* **H** *and an integer k, is the copy complexity of* **H** *at most k?*

Definition 6. *The optimal copy complexity (CC_{Opt}) problem: given an HCS* **H**, *what is the smallest k such that the copy complexity of* **H** *is at most k?*

In this paper, we focus on these problems in width 3 HCSs, for the most part. The principal contributions of this paper are as follows:

1. Establishing a lower bound on the copy complexity of bounded width HCSs (Theorem 2).
2. Establishing that the CC_D problem for width 3 HCSs is **NP-complete** (Theorem 4).
3. Establishing that no algorithm for the CC_D problem for width 3 HCSs can run in time $2^{o(n)}$ unless the Exponential Time Hypothesis (**ETH**) fails (Theorem 6).
4. Establishing that the CC_{Opt} problem for width 3 HCSs is **NPO-complete** [7] (Theorem 7).

3 Observations on Copy Complexity

In this section, we observe several properties of the copy complexity of bounded width HCSs.

First, we show that the copy complexity of a width w HCS with $((w-1)\cdot n'+1)$ variables can be as large as $2^{(w-2)\cdot n'}$.

Theorem 1. *For each integer $n' \geq 0$, there exists a width w HCS $\mathbf{H}$ with $((w-1)\cdot n'+1)$ variables such that $\mathbf{H}$ has copy complexity $2^{(w-2)\cdot n'}$.*

Proof. Let $\mathbf{H}$ be the HCS constructed as follows:

1. The constraint l_1 is $-x_1 \geq 1$.
2. For $r = 2, \ldots, (w-1)\cdot n'+1$, the constraint l_r is

$$x_{r-1} - \sum_{j=r}^{\lceil \frac{r-1}{w-1} \rceil \cdot (w-1)+1} x_j \geq 0.$$

3. The constraint $l_{(w-1)\cdot n'+2}$ is $x_{(w-1)\cdot n'+1} \geq 0$.

We will show that for each $i = 0 \ldots n'$, the constraint $l_{(w-1)\cdot i+2}$ must be used at least $2^{(w-2)\cdot i}$ times by any linear refutation of $\mathbf{H}$.

Note that constraint l_1 is the only constraint in $\mathbf{H}$ that has a positive defining constant. Thus, l_1 must be in any linear refutation of $\mathbf{H}$. Additionally, the only constraint in $\mathbf{H}$ with the literal x_1 is l_2. Thus, the constraint l_2 must be used at least 2^0 times by any linear refutation of $\mathbf{H}$.

Now assume that the constraint $l_{(w-1)\cdot i+2}$ must be used at least $2^{(w-2)\cdot i}$ times by any linear refutation of $\mathbf{H}$. Note that this constraint contains the literal $-x_{(w-1)\cdot i+2}$ and that this is the only constraint in $\mathbf{H}$ containing that literal. The only constraint in $\mathbf{H}$ with the literal $x_{(w-1)\cdot i+2}$ is $l_{(w-1)\cdot i+3}$. Thus, this constraint must also be used at least $2^{(w-2)\cdot i}$ times by any linear refutation of $\mathbf{H}$.

Both constraints $l_{(w-1)\cdot i+2}$ and $l_{(w-1)\cdot i+3}$ contain the literal $-x_{(w-1)\cdot i+3}$ and these are the only constraints in $\mathbf{H}$ containing that literal. Thus, the literal $-x_{(w-1)\cdot i+3}$ is used by at least $2^{(w-2)\cdot i+1}$ constraints in any linear refutation of $\mathbf{H}$.

For each $j = 1 \ldots w-1$, the constraints $l_{(w-1)\cdot i+2}$ through $l_{(w-1)\cdot i+j+1}$ contain the literal $-x_{(w-1)\cdot i+j+1}$ and these are the only constraints in $\mathbf{H}$ containing that literal. Thus, the literal $-x_{(w-1)\cdot i+j+1}$ is used by at least $2^{(w-2)\cdot i+j-1}$ constraints in any linear refutation of $\mathbf{H}$.

Note that the only constraint in $\mathbf{H}$ with the literal $x_{(w-1)\cdot i+w} = x_{(w-1)\cdot(i+1)+1}$ is $l_{(w-1)\cdot(i+1)+2}$. Thus, any linear refutation of $\mathbf{H}$ must use the constraint $l_{(w-1)\cdot(i+1)+2}$ at least $2^{(w-2)\cdot(i+1)}$ times as desired.

Thus, for each $i = 0 \ldots n'$, the constraint $l_{(w-1)\cdot i+2}$ must be used at least $2^{(w-2)\cdot i}$ times by any linear refutation of $\mathbf{H}$. In particular, the constraint $l_{(w-1)\cdot n'+2}$ must be used at least $2^{(w-2)\cdot n'}$ times by any linear refutation of $\mathbf{H}$.

We can construct a linear refutation of $\mathbf{H}$ by using each constraint l_r, $2^{r-1-\lfloor \frac{r-1}{w-1} \rfloor}$ times. □

From, Theorem 1, there is a width w HCS $\mathbf{H}$ with $n \equiv 1 \mod (w-1)$ variables such that $\mathbf{H}$ has copy complexity at least $2^{(w-2)\cdot \frac{n-1}{w-1}}$. Utilizing a different construction, we can obtain a tighter bound based on a generalization of the Fibonacci Sequence.

Recall that the Fibonacci Sequence F_n is the sequence in which each element is the sum of the two previous elements. More formally, the Fibonacci Sequence is defined as follows: $F_0 = 0$, $F_1 = 1$, and $F_n = F_{n-1} + F_{n-2}$ for $n \geq 2$. We can generalize this definition by having each element depend on more than just the previous two elements. Our result on width w HCSs utilizes the following generalization of the Fibonacci Sequence.

For each $w \geq 1$, the width w Fibonacci Sequence $F_{w,n}$ is defined as follows: $F_{w,0} = 0$, $F_{w,1} = 1$, $F_{w,n} = \sum_{i=\max\{0,n-w\}}^{n-1} F_{w,i}$ for $n \geq 2$. Thus, in the width w Fibonacci Sequence each element depends on the sum of the previous w elements, not just the previous 2 elements. Note that the width 2 Fibonacci Sequence is simply the regular Fibonacci Sequence.

We now make a structural observation about $F_{w,n}$.

Lemma 1. *For $w \geq 1$ and $2 \leq n \leq w+1$, $F_{w,n} = 2^{n-2}$.*

Proof. Let w be a positive integer. By definition, $F_{w,0} = 0$ and $F_{w,1} = 1$. Additionally, for $n = 2 \ldots w$, $F_{w,n} = \sum_{i=0}^{n-1} F_{w,i}$. Since $F_{w,0} = 0$, we have that for $n = 2 \ldots w+1$, $F_{w,n} = \sum_{i=1}^{n-1} F_{w,i}$.

When $n = 2$, we have

$$F_{w,n} = F_{w,2} = F_{w,0} + F_{w,1} = 1 = 2^{n-2}.$$

Let n be an integer such that $2 \leq n \leq w+1$. Assume that $F_{w,i} = 2^{i-2}$ for $2 \leq i < n$. Recall that,

$$F_{w,n} = \sum_{i=1}^{n-1} F_{w,i} = 1 + \sum_{i=2}^{n-1} 2^{i-2} = 1 + (2^{n-2} - 1) = 2^{n-2}.$$

Thus, for $w \geq 1$ and $2 \leq n \leq w+1$, $F_{w,n} = 2^{n-2}$. □

We will now utilize width w Fibonacci Sequences to establish a stronger lower bound on the copy complexity of HCSs with bounded constraint width.

Theorem 2. *For each $w \geq 2$ and $n \geq 2$, there exists a width w HCS $\mathbf{H}_{w,n}$ with n variables such that the copy complexity of $\mathbf{H}_{w,n}$ is $2 \cdot F_{(w-1),n}$.*

Proof. For each $w \geq 2$ and $n \geq 2$, Let $\mathbf{H}_{w,n}$ be the HCS constructed as follows:

1. The constraint l_0 is $-x_1 - x_2 - \ldots - x_w \geq 1$.
2. For $r = 1, \ldots, n-1$, the constraint l_r is $x_r - \sum_{j=r+1}^{\min\{r+w-1,n\}} x_j \geq 0$.
3. The constraint l_n is $x_n \geq 0$.

Let R be a linear refutation of $\mathbf{H}_{w,n}$ and let $C(l_i)$ be the number of times the constraint l_i is used by R. We will show that for any refutation R of $\mathbf{H}_{w,n}$, $C(l_i) \geq 2 \cdot F_{(w-1),i}$ for $i = 2 \ldots n$.

We make the following observations about R and the structure of $\mathbf{H}_{w,n}$:

1. l_0 is the only constraint with positive defining constant. Thus, l_0 must be used by R and $C(l_0) \geq 1$
2. For each $i = 1 \ldots n$, the constraint l_i is the only constraint to use the literal x_i. Additionally, the constraints l_{i-w+1} through l_{i-1} are the only constraints in $\mathbf{H}_{w,n}$ to use the literal $-x_i$. The only exception to this is the literal $-x_w$ which also appears in the constraint l_0.
   Since R is a refutation of $\mathbf{H}_{w,n}$, the number of constraints in R that use the literal x_i and the number of constraints in R that use the literal $-x_i$ are equal. Thus, for $i = 1 \ldots n$, $i \neq w$ we have that $C(l_i) = \sum_{j=\max\{0,i-w+1\}}^{i-1} C(l_j)$ and $C(l_w) = \sum_{j=0}^{w-1} C(l_j)$.
3. For $i = 1 \ldots w$, $C(l_i) = \sum_{j=0}^{i-1} C(l_j)$. Thus, $C(l_i) \geq 2^{i-1}$. From Lemma 1, we have that for $i = 2 \ldots w$, $F_{(w-1),i} = 2^{i-2}$. Thus, for $i = 2 \ldots w$, $C(l_i) \geq 2 \cdot F_{(w-1),i}$.
4. Let i be an integer such that $w < i \leq n$. Assume that for each $j = 2 \ldots i-1$, $C(l_j) \geq 2 \cdot F_{(w-1),j}$. We have that $C(l_i) = \sum_{j=i-w+1}^{i-1} C(l_j)$. Since $(i-w+1) \geq 2$, $\sum_{j=i-w+1}^{i-1} C(l_j) \geq \sum_{j=i-w+1}^{i-1} 2 \cdot F_{(w-1),j} = 2 \cdot F_{(w-1),i}$. Thus, $C(l_i) \geq 2 \cdot F_{(w-1),i}$ as desired.

From the above observations, for each $i = 2 \ldots n$, $C(l_i) \geq 2 \cdot F_{(w-1),i}$. In particular, $C(l_n) \geq 2 \cdot F_{(w-1),n}$. Thus, the copy complexity of $\mathbf{H}_{w,n}$ is at least $2 \cdot F_{(w-1),n}$.

Let R be such that $C(l_0) = 1$, $C(l_1) = 1$, and for each $i = 2 \ldots n$, $C(l_i) = 2 \cdot F_{(w-1),i}$. It can be algebraically verified that R is a linear refutation of $\mathbf{H}_{w,n}$. Thus, copy complexity of $\mathbf{H}_{w,n}$ is $2 \cdot F_{(w-1),n}$ as desired. □

Let S_l be the following set of constraints:

$$l_1 : x_{2 \cdot l+1} - x_{2 \cdot l} - x_{2 \cdot l-1} \geq 0 \qquad l_2 : x_{2 \cdot l} - x_{2 \cdot l-1} \geq 0$$
$$l_3 : x_{2 \cdot l-1} - x_{2 \cdot l-2} - x_{2 \cdot l-3} \geq 0 \qquad l_4 : x_{2 \cdot l-2} - x_{2 \cdot l-3} \geq 0 \qquad \ldots$$
$$l_{2 \cdot l-1} : x_3 - x_2 - x_1 \geq 0 \qquad l_{2 \cdot l} : x_2 - x_1 \geq 0 \qquad l_{2 \cdot l+1} : x_1 \geq 0$$

Theorem 1, when applied to width 3 HCSs, can be extended to the following result which will be utilized later in the paper.

Theorem 3. *Let* **H** *be an HCS and let* $\{x_1, \ldots, x_{2\cdot l+1}\}$ *be a subset of the variables in* **H** *such that for each* $i = 1 \ldots (2 \cdot l + 1)$, *the only constraint in* **H** *that uses the literal* x_i *belongs to the set* S_l. *If a linear refutation* R *of* **H** *uses a constraint* $x - \sum_{i \in S} x_i \geq b$ *for some set* $S \subseteq \{1, 3, \ldots, 2 \cdot l + 1\}$, *then* R *must use the constraint* $x_1 \geq 0$ *at least* $\sum_{2\cdot i+1 \in S} 2^i$ *times.*

Proof. Let **H** be an appropriately constructed HCS. For some subset $S \subseteq \{1, 3, \ldots, 2 \cdot l + 1\}$ let $x - \sum_{i \in S} x_i \geq b$ be a constraint in **H** used by a linear refutation R of **H**.

Let x_i be a variable such that $2 \cdot i + 1 \in S$. By the definition of **H**, the only constraint in **H** with the literal $x_{2\cdot i+1}$ is $x_{2\cdot i+1} - x_{2\cdot i} - x_{2\cdot i-1} \geq 0$. Thus, this constraint must be used by R. Observe the following:

1. The constraint $x_{2\cdot i} - x_{2\cdot i-1} \geq 0$ is the only constraint with the literal $x_{2\cdot i}$. Thus, it needs to be in R. Consequently, R has at least 2 constraints with the literal $-x_{2\cdot i-1}$.
2. The constraint $x_{2\cdot i-1} - x_{2\cdot i-2} - x_{2\cdot i-3} \geq 0$ is the only constraint with the literal $x_{2\cdot i-1}$. Thus, R needs to use this constraint at least 2 times. Consequently, R has at least 2 constraints with the literal $-x_{2\cdot i-2}$.
3. The constraint $x_{2\cdot i-2} - x_{2\cdot i-3} \geq 0$ is the only constraint with the literal $x_{2\cdot i-2}$. Thus, R needs to use this constraint at least 2 times. Consequently, R has at least 4 constraints with the literal $-x_{2\cdot i-3}$.
4. For each r, R uses at least 2^r constraints with the literal $-x_{2\cdot(i-r)+1}$. Thus, the constraint $x_{2\cdot(i-r)+1} - x_{2\cdot(i-r)} - x_{2\cdot(i-r-1)+1} \geq 0$ needs to be in the refutation at least 2^r times. Consequently, R uses at least 2^r constraints with the literal $-x_{2\cdot(i-r)}$.
5. For each r, R uses at least 2^r constraints with the literal $-x_{2\cdot(i-r)}$. Thus, the constraint $x_{2\cdot(i-r)} - x_{2\cdot(i-r-1)+1} \geq 0$ needs to be in the refutation at least 2^r times. Consequently, R uses at least 2^{r+1} constraints with the literal $-x_{2\cdot(i-r-1)+1}$.
6. The constraint $x_1 \geq 0$ needs to be used at least 2^i times by R.

Thus, for each $2 \cdot i + 1 \in S$, R must use the constraint $x_1 \geq 0$ at least 2^i times. Consequently, R must use the constraint $x_1 \geq 0$ at least $\sum_{2\cdot i+1 \in S} 2^i$ times. □

4 Computational Complexity of the $\mathbf{CC}_D$ Problem

In this section, we explore the computational and approximation complexities of the copy complexity problem for width 3 HCSs.

First, we show that the problem of determining the copy complexity of an HCS is **NP-complete** even when each constraint in the HCS has at most 3 non-zero coefficients.

Let Φ be a 3-CNF formula with m' clauses over n' variables and let **H** be the HCS constructed as follows:

1. For each variable x_i of Φ, create the variables x_i and y_i. Create the constraints $-x_1 - y_1 \geq 0$, $y_1 - x_2 - y_2 \geq 0, \ldots, y_{n'-2} - x_{n'-1} - y_{n'-1} \geq 0$, $y_{n'-1} - x'_n \geq 1 - m'$. These constraints are equivalent to the constraint $-\sum_{i=1}^{n'} x_i \geq 1 - m'$.

2. For each clause $\phi_j \in \Phi$, create the variable c_j. Additionally, create the constraints $c_j \geq 1$, $c_j \geq 0$, and $c_j \geq 0$.
3. For each clause $\phi_j \in \Phi$ and each variable x_i in clause ϕ_j, create the variable $z_{i,j}$. Since each clause has at most 3 literals, there are at most $3 \cdot m'$ such variables.
4. For each variable x_i, let $Pos(i) = \{\phi_{j_1}, \ldots, \phi_{j_{|Pos(i)|}}\}$ be the set of clauses containing the literal x_i. Create the constraints $x_i - z_{i,j_1} \geq 0$, $z_{i,j_1} - c_{j_1} - z_{i,j_2} \geq 0, \ldots, z_{i,j_{|Pos(i)|}} - c_{j_{|Pos(i)|}} \geq 0$. This is equivalent to the constraint $x_i - \sum_{\phi_j \in Pos(i)} c_j \geq 0$.
5. For each variable x_i, let $Neg(i) = \{\phi_{j'_1}, \ldots, \phi_{j'_{|Neg(i)|}}\}$ be the set of clauses containing the literal $\neg x_i$. Create the constraints $x_i - z_{i,j'_1} \geq 0$, $z_{i,j'_1} - c_{j'_1} - z_{i,j'_2} \geq 0, \ldots, z_{i,j'_{|Neg(i)|}} - c_{j'_{|Neg(i)|}} \geq 0$. This is equivalent to the constraint $x_i - \sum_{\phi_j \in Neg(i)} c_j \geq 0$.

Note that **H** has $n \leq (4 \cdot m' + 2 \cdot n')$ variables.

We now show that a 3-CNF formula Φ has a solution if and only if the HCS **H** has a copy complexity of 1.

Lemma 2. *Let Φ be a 3-CNF formula and let* **H** *be the HCS constructed from Φ. Φ has a solution if and only if* **H** *has a copy complexity of* 1.

Proof. First, assume that Φ has a solution $\mathbf{x}$. We will show that **H** has copy complexity 1 by showing that **H** has a read-once linear refutation R.

We construct R as follows:

1. Add the constraints $-x_1 - y_1 \geq 0, \ldots, y_{n'-1} - x'_n \geq 1 - m'$ to R. Note that summing these constraints results in the constraint $-\sum_{i=1}^{n'} x_i \geq 1 - m'$.
2. For each variable x_i, if x_i is assigned **true** by $\mathbf{x}$, then add the constraints $x_i - z_{i,j_1} \geq 0, \ldots, z_{i,j_{|Pos(i)|}} - c_{j_{|Pos(i)|}} \geq 0$ to R. If x_i is assigned **false** by $\mathbf{x}$, then add the constraints $x_i - z_{i,j'_1} \geq 0, \ldots, z_{i,j'_{|Neg(i)|}} - c_{j'_{|Neg(i)|}} \geq 0$ to R.
3. For each clause $\phi_r \in \Phi$, let $C(r)$ be the number of times the literal $-c_r$ is used by a constraint in R so far. Since ϕ_r has at most 3 literals, $C(r) \leq 3$. Additionally, since $\mathbf{x}$ satisfies Φ, the clause ϕ_r must contain a literal $T(r)$ set to **true** by $\mathbf{x}$.
4. If $T(r)$ is the literal x_i, then the variable x_i is assigned **true** by $\mathbf{x}$. Thus, by construction, R contains the equivalent of the constraint $x_i - \sum_{\phi_r \in Pos(i)} c_r \geq 0$. Since $\Phi_r \in Pos(i)$, the literal $-c_r$ is used by a constraint in R. Thus, $C(r) \geq 1$.
5. If $T(r)$ is the literal $\neg x_i$, then the variable x_i is assigned **false** by $\mathbf{x}$. Thus, by construction, R contains the equivalent of the constraint $x_i - \sum_{\phi_r \in Neg(i)} c_r \geq 0$. Since $\Phi_r \in Neg(i)$, the literal $-c_r$ is used by a constraint in R. Thus, $C(r) \geq 1$. Consequently, for each clause $\phi_r \in \Phi$, $1 \leq C(r) \leq 3$.
6. For each clause $\phi_r \in \Phi$, add the constraint $c_r \geq 1$ and $(C(r) - 1)$ copies of the constraint $c_r \geq 0$ to R. Note that **H** has 2 copies of the constraint $c_r \geq 0$ and that $0 \leq C(r) - 1 \leq 2$.

It is easy to see that summing all of the constraints in R results in a contradiction of the form $0 \geq 1$. Thus, R is a read-once refutation of $\mathbf{H}$.

Now assume that $\mathbf{H}$ has a refutation R that uses no constraint more than once. We construct an assignment $\mathbf{x}$ to Φ as follows: for each variable x_i, if R contains a constraint of the form $x_i - z_{i,j_1} \geq 0$ such that $\phi_{j_1} \in Pos(x_i)$, then set x_i to **true**. Otherwise set x_i to **false**.

We make the following observations about R:

1. If the constraint $-x_1 - y_1 \geq 0$ is removed from $\mathbf{H}$, then $\mathbf{H}$ is feasible. Thus, this constraint must be used by R. To cancel each y_i variable, all of the constraints $-x_1 - y_1 \geq 0, \ldots, y_{n'-1} - x'_n \geq 1 - m'$ must be in R. These constraints are equivalent to the constraint $-\sum_{i=1}^{n'} x_i \geq 1 - m'$.
2. To get a contradiction, the defining constant of the derived constraint must be positive. Note that the only constraints with positive defining constant in $\mathbf{H}$ are of the form $c_r \geq 1$. There are m' such constraints, thus they must all be used by R.
3. Consider the constraint $c_r \geq 1$. The only constraints with $-c_r$ in $\mathbf{H}$ are of the form $z_{i,r} - c_r - z_{i,r'} \geq 0$ where $\phi_r \in Pos(i)$ or $\phi_r \in Neg(i)$. Thus, R must contain a constraint of this form.
4. If R contains a constraint of the form $z_{i,r} - c_r - z_{i,r'} \geq 0$ where $\phi_r \in Pos(i)$, then to cancel all the $z_{i,r}$ variables R must contain the constraints $x_i - z_{i,j_1} \geq 0, \ldots, z_{i,j_{|Pos(i)|}} - c_{j_{|Pos(i)|}} \geq 0$. Thus, x_i is set to **true** by $\mathbf{x}$. Note that $\phi_r \in Pos(i)$ if and only if ϕ_r contains the literal x_i. Thus, ϕ_r is satisfied by $\mathbf{x}$.
5. If R contains a constraint of the form $z_{i,r} - c_r - z_{i,r'} \geq 0$ where $\phi_j \in Neg(i)$, then to cancel all the $z_{i,r}$ variables R must contain the constraints $x_i - z_{i,j'_1} \geq 0, \ldots, z_{i,j'_{|Neg(i)|}} - c_{j'_{|Neg(i)|}} \geq 0$. By construction, $\mathbf{H}$ has only one constraint with the literal $-x_i$. Thus, R cannot contain both the constraint $x_i - z_{i,j'_1} \geq 0$ and a constraint of the form $x_i - z_{i,j_1} \geq 0$ such that $\phi_{j_1} \in Pos(x_i)$. Consequently, x_i is set to **false** by $\mathbf{x}$. Note that $\phi_r \in Neg(i)$ if and only if ϕ_r contains the literal $\neg x_i$. Thus, ϕ_r is satisfied by $\mathbf{x}$.

Note that $\mathbf{x}$ satisfies every clause in Φ. Thus, Φ is satisfiable. □

Theorem 4. *The CC_D problem for width* 3 *HCSs is* **NP-complete**.

Proof. For each integer k we can establish that the copy complexity of a width 3 HCS is at most k by providing a Farkas vector $\mathbf{y}$ such that $||\mathbf{y}||_\infty \leq k$. Note that $||\mathbf{y}||_\infty$ is the largest element of $\mathbf{y}$ and is called the L_∞ norm of $\mathbf{y}$. Thus, the CC_D problem for width 3 HCSs is in **NP**.

From Lemma 2, we have that given a 3-CNF formula Φ, we can construct a width 3 HCS $\mathbf{H}$ such that $\mathbf{H}$ has copy complexity 1 if and only if Φ is feasible. Thus, the CC_D problem for width 3 HCSs is **NP-complete**. □

The result in Theorem 4, relies on the fact that the problem of determining if a width 3 HCS has copy complexity 1 is **NP-complete**. However, this result can be extended to any fixed positive integer C.

Theorem 5. *Let C be a positive integer. The problem of determining if a width 3 HCS has copy complexity at most C is* **NP-complete**.

Proof. Let Φ be a 3-CNF formula and let $\mathbf{H}$ be the HCS constructed from Φ. From Lemma 2, we know that $\mathbf{H}$ has copy complexity 1 if and only if Φ is satisfiable.

We can construct an HCS $\mathbf{H}'$ from $\mathbf{H}$ as follows:

1. Initially, $\mathbf{H}' = \mathbf{H}$.
2. Let $E \subseteq \mathbb{N}$ be such that $\sum_{i \in E} 2^i = C$. For each $k = 1 \ldots |E|$, let $E(k)$ be the k^{th} element of E.
3. For each constraint l_j of $\mathbf{H}$:
   (a) Create the variables $g_{j,1}$ through $g_{j,2 \cdot \lfloor \log C \rfloor + 1}$ and the constraint $g_{j,1} \geq 0$. Additionally, create the constraints
   $g_{j,2 \cdot l+1} - g_{j,2 \cdot l} - g_{j,2 \cdot l-1} \geq 0$ and $g_{j,2 \cdot l} - g_{j,2 \cdot l-1} \geq 0$ for $l = 1 \ldots \lfloor \log C \rfloor$. Let S_j denote this set of constraints.
   (b) Create the variables $e_{j,k}$ for $k = 0 \ldots |E|$. Additionally, create the constraints $e_{j,0} - e_{j,1} \geq 0$, $e_{j,1} - g_{j,2 \cdot E(1)+1} - e_{j,2} \geq 0, \ldots, e_{j,|E|-1} - g_{j,2 \cdot E(|E|-1)+1} - e_{j,|E|} \geq 0$, and $e_{j,|E|} - g_{j,2 \cdot E(|E|)+1} \geq 0$.
   (c) Add the literal $-e_{j,0}$ to the constraint l_j.

We will now show that $\mathbf{H}'$ has copy complexity at most C if and only if $\mathbf{H}$ has copy complexity 1.

First assume that $\mathbf{H}$ has copy complexity 1. Let R be a read-once refutation of $\mathbf{H}$. We construct a refutation R' of $\mathbf{H}'$ as follows:

1. Add each constraint used by R to R'.
2. For each constraint l_j used by R, add the constraints $e_{j,0} - e_{j,1} \geq 0, \ldots, e_{j,|E|} - g_{j,2 \cdot E(|E|)+1} \geq 0$ to R'. Additionally, add enough copies of the constraints in S_j to cancel all of the $g_{j,l}$ variables. From Theorem 3, this requires $\sum_{i \in E} 2^i = C$ copies of the constraint $g_{j,1} \geq 0$.

Note that R' is a refutation of $\mathbf{H}'$ that uses each constraint at most C times. Thus, $\mathbf{H}'$ has copy complexity at most C.

Now assume that $\mathbf{H}'$ has a refutation that uses no constraint more than C times. We construct a read-once refutation R of $\mathbf{H}$ as follows: for each constraint l_j in $\mathbf{H}'$ used by R' add the corresponding constraint l_j from $\mathbf{H}$ to R. By construction, the remaining constraints in R' are used to eliminate the variable $e_{j,0}$ from each constraint l_j. Since these variables do not exist in $\mathbf{H}$, R is a refutation of $\mathbf{H}$. All that remains is to show that no constraint l_j is used more than once by R'.

Assume that for some j, the constraint l_j is used $r > 1$ times by R'. Thus, by construction, the constraint $e_{j,0} - e_{j,1} \geq 0$ must also be used r times by R'. From Theorem 3, this means that the constraint $g_{j,1} \geq 0$ needs to be used at least $r \cdot C > C$ times by R'. This contradicts the fact that R used no constraint more than C times. Thus, each constraint l_j is used at most once. Consequently, R is a read-once refutation of H. □

Since the CC_D problem is in **NP**, there exists a $2^{p(m,n)}$ algorithm for this problem, where $p(m, n)$ is some polynomial in m and n. We now show that there cannot be a $2^{o(n)}$ algorithm for the CC_D problem for width 3 HCSs unless the Exponential Time Hypothesis (**ETH**) fails [5,6].

The ETH states that for each $k \geq 3$, there exists a constant $s_k > 0$ such that k-SAT cannot be solved in time less than $O(2^{s_k \cdot n})$. In particular, this precludes a $2^{o(n)}$ time algorithm for 3-SAT. We now utilize the reduction used by Lemma 2 to establish a likely lower bound on the running time on any algorithm for solving the copy complexity problem for width 3 HCSs.

Theorem 6. *There cannot be a $2^{o(n)}$ algorithm for the CC_D problem for width 3 HCSs unless the ETH fails.*

Proof. From Lemma 2, if there is a $2^{o(n)}$ time algorithm for the copy complexity problem for HCSs, then there is a $2^{o(n'+m')}$ algorithm for 3-CNF feasibility. This violates the ETH [5,6]. Thus, it is unlikely that a $2^{o(n)}$ time algorithm exists for the CC_D problem for HCSs. □

We now show that the problem of finding the copy complexity of a width 3 HCS is **NPO complete** [1]. We do this by a reduction from the Weighted Min-Ones problem.

The Weighted Min-Ones problem is defined as follows: Given a 3CNF formula Φ and positive integer valued variable weight function w, what is the satisfying assignment to Φ with least weight of variables set to **true**. This problem is known to be **NPO-complete** [11].

Let Φ be a CNF formula with m clauses over n variables where each variable x_i has weight $w(i)$. Additionally, let W be the target weight. We construct the corresponding HCS **H** as follows:

1. Let w_{max} be the largest weight of any variable x_i of Φ. Additionally let $f = \lfloor \log w_{max} \rfloor$.
2. For each variable x_i of Φ, create the variables x_i, t_i and y_i^+.
3. Create the constraints $-x_1 - t_1 \geq 0$, $t_1 - x_2 - t_2 \geq 0, \ldots, t_{n-2} - x_{n-1} - t_{n-1} \geq 0$, and $t_{n-1} - x_n \geq 1 - m$. Let S be the set containing these constraints. Note that these constraints are the only constraints to use the variables t_i for $i = 1 \ldots (n-1)$. If any constraint in S is used more times by a refutation R of **H** than any other constraint in S, then there would be a variable t_i left over in the resultant constraint. Thus, any refutation of **H** must use all of these constraints an equal number of times. Note that together these constraints are equivalent to the constraint $-\sum_{i=1}^{n} x_i \geq m - 1$.
4. For each variable x_i, let $P(i)$ be the number of clauses in Φ containing the literal x_i, and let $N(i)$ be the number of clauses in Φ containing the literal $\neg x_i$. Create the variables $z_{i,l}^+$ and $t_{i,l}^+$ for $l = 1 \ldots P(i)$ and the variables $z_{i,l}^-$ and $t_{i,l}^-$ for $l = 1 \ldots N(i)$.

5. For each variable x_i of Φ, create the constraints $x_i \geq 0$, $x_i - t^-_{i,1} \geq 0$, $t^-_{i,1} - z^-_{i,1} \geq 0$, $t^-_{i,1} - z^-_{i,1} - t^-_{i,2} \geq 0, \ldots, t^-_{i,N(i)-1} - z^-_{i,N(i)-1} \geq 0$, $t^-_{i,N(i)-1} - z^-_{i,N(i)-1} - t^-_{i,N(i)} \geq 0$, and $t^-_{i,N(i)} - z^-_{i,N(i)} \geq 0$. For each $l = 0 \ldots N(i)$, let $S^-_{i,l}$ be the set:
$$\{x_i - t^-_{i,1} \geq 0, t^-_{i,1} - z^-_{i,1} - t^-_{i,2} \geq 0, \ldots, t^-_{i,l} - z^-_{i,l} \geq 0\}.$$
Note that the constraints in $S^-_{i,l}$ are equivalent to the constraint $x_i - \sum_{j=1}^{l} z^-_{i,j} \geq 0$.
6. For each variable x_i of Φ, create the constraints $x_i - y^+_i \geq 0$, $x_i - y^+_i - t^+_{i,1} \geq 0$, $t^+_{i,1} - z^+_{i,1} \geq 0$, $t^+_{i,1} - z^+_{i,1} - t^+_{i,2} \geq 0, \ldots, t^+_{i,P(i)-1} - z^+_{i,P(i)-1} \geq 0$, $t^+_{i,P(i)-1} - z^+_{i,P(i)-1} - t^+_{i,P(i)} \geq 0$, and $t^+_{i,P(i)} - z^+_{i,P(i)} \geq 0$. For each $l = 0 \ldots P(i)$, let $S^+_{i,l}$ be the set:
$$\{x_i - y^+_i - t^+_{i,1} \geq 0, t^+_{i,1} - z^+_{i,1} - t^+_{i,2} \geq 0, \ldots, t^+_{i,l} - z^+_{i,l} \geq 0\}.$$
Note that the constraints in $S^+_{i,l}$ are equivalent to the constraint $x_i - y^+ - \sum_{j=1}^{l} z^+_{i,j} \geq 0$.
7. For each clause $\phi_j \in \Phi$, create the variables c_j and d_j. Additionally, create the constraint $c_j - d_j \geq 1$.
8. For each clause $\phi_j \in \Phi$, create the variables $d_{j,1}$ through $d_{j,2\cdot\lfloor \log W \rfloor+1}$ and the constraint $d_{j,1} \geq 0$. Additionally, create the constraints $d_{j,2\cdot l+1} - d_{j,2\cdot l} - d_{j,2\cdot l-1} \geq 0$ and $d_{j,2\cdot l} - d_{j,2\cdot l-1} \geq 0$ for $l = 1 \ldots \lfloor \log W \rfloor$. Let S'_j denote this set of constraints.
9. Let $E_W \subseteq \mathbb{N}$ be such that $\sum_{j\in E_W} 2^j = W$. For each $k = 1 \ldots |E_W|$, let $E(W, k)$ be the k^{th} element of E_W. For each clause ϕ_j, create the variables $h_{j,k}$ for $k = 1 \ldots |E_W|$. Additionally, create the constraints $d_j - h_{j,1} \geq 0$, $h_{j,1} - d_{j,2\cdot E(W,1)+1} - h_{i,2} \geq 0, \ldots, h_{j,|E_W|-1} - d_{j,2\cdot E(W,|E_W|-1)+1} - h_{j,|E_W|} \geq 0$, and $h_{j,|E_W|} - d_{j,2\cdot E(W,|E_W|)+1} \geq 0$.
10. For each clause $\phi_j \in \Phi$ and each variable x_i, if the literal x_i appears in the clause ϕ_j, add the constraints $z^+_{i,l} - c_j \geq 0$ for $l = 1 \ldots P(i)$ to **H**. If the literal $\neg x_i$ appears in the clause ϕ_j, add the constraints $z^-_{i,l} - c_j \geq 0$ for $l = 1 \ldots N(i)$ to **H**.
11. Create the variables g_1 through $g_{2\cdot f+1}$ and the constraint $g_1 \geq 0$. Additionally, create the constraints $g_{2\cdot l+1} - g_{2\cdot l} - g_{2\cdot l-1} \geq 0$ and $g_{2\cdot l} - g_{2\cdot l-1} \geq 0$ for $l = 1 \ldots f$. Let S_f denote this set of constraints.
12. For each variable x_i of Φ, let $E_i \subseteq \mathbb{N}$ be such that $\sum_{j\in E_i} 2^j = w(i)$. For each $k = 1 \ldots |E_i|$, let $E(i, k)$ be the k^{th} element of E_i. Create the variables $e_{i,k}$ for $k = 1 \ldots |E_i|$. Additionally, create the constraints $y^+_i - e_{i,1} \geq 0$, $e_{i,1} - g_{2\cdot E(i,1)+1} - e_{i,2} \geq 0, \ldots, e_{i,|E_i|-1} - g_{2\cdot E(i,|E_i|-1)+1} - e_{i,|E_i|} \geq 0$, and $e_{i,|E_i|} - g_{2\cdot E(i,|E_i|)+1} \geq 0$.

We now show that a CNF formula Φ has a solution in which the total weight of **true** variables is at most W if and only if the HCS **H** has a copy complexity of at most W.

Lemma 3. *Let Φ be a CNF formula with weighted variables and let* **H** *be the HCS constructed from Φ. Φ has a solution in which the total weight of* **true** *variables is at most W if and only if* **H** *has a copy complexity of at most W.*

Proof. First, assume that Φ has a solution $\mathbf{x}$ such that $W^* = \sum_{x_i : x_i = \mathbf{true}} w(i) \leq W$. We will show that **H** has copy complexity at most W by showing that **H** has a refutation R that uses each constraint at most W^* times. We construct R as follows:

1. Add the constraints in S to R. Recall that these constraints are equivalent to $-\sum_{i=1}^{n} x_i \geq 1 - m$.
2. For each clause $\phi_j \in \Phi$ let $T(j)$ be a literal in ϕ_j set to **true** by $\mathbf{x}$.
3. For each variable x_i, let $Pos(i) = \{\phi_j : T(j) = x_i\}$, and let $Neg(i) = \{\phi_j : T(j) = \neg x_i\}$.
4. For each variable x_i, if x_i is assigned **true** by $\mathbf{x}$, then add the constraints in $S^+_{i,|Pos(i)|}$ to R. If x_i is assigned **false** by $\mathbf{x}$, then add the constraints in $S^-_{i,|Neg(i)|}$ to R.
5. For each variable x_i set to **true** by $\mathbf{x}$, add the constraints $y_i^+ - e_{i,1} \geq 0, \ldots, e_{i,|E_i|} - g_{2 \cdot E(i,|E_i|)+1} \geq 0$ to R. Additionally, add enough copies of the constraints in S_f to cancel all of the g_l variables. From Theorem 3, this requires $\sum_{j \in E_i} 2^j = w(i)$ copies of the constraint $g_1 \geq 0$.
6. For each variable x_i set to **true** by $\mathbf{x}$ and for each $l = 1 \ldots |Pos(i)|$, let ϕ_j be the l^{th} element of $Pos(i)$. Add the constraint $z^+_{i,l} - c_j \geq 0$ to R.
7. For each variable x_i set to **false** by $\mathbf{x}$ and for each $l = 1 \ldots |Neg(i)|$, let ϕ_j be the l^{th} element of $Neg(i)$. Add the constraint $z^-_{i,l} - c_j \geq 0$ to R.
8. Add the constraints $c_1 - d_1 \geq 1$ through $c_m - d_m \geq 1$ to R.
9. For each clause ϕ_j, add the constraints $d_j - h_{j,1} \geq 0, \ldots, h_{j,|E_W|} - d_{j,2 \cdot E(W,|E_W|)+1} \geq 0$ to R. Additionally, add enough copies of the constraints in S'_j to cancel all of the $d_{j,l}$ variables. From Theorem 3, this requires $\sum_{l \in E_W} 2^l = W$ copies of the constraint $d_{j,1} \geq 0$.

It is easy to see that summing all of the constraints in R results in a contradiction of the form $0 \geq 1$. Thus, R is a refutation of **H**. Note that the constraints reused by R belong to the sets S_f and S'_j for $j = 1 \ldots m$. From Theorem 3, the constraints reused the most are the constraint $g_1 \geq 0$ and the constraints $d_{j,1} \geq 0$ for $j = 1 \ldots m$. These constraints are each used at most W times as desired.

Now assume that **H** has a refutation that uses no constraint more than W times. Thus, **H** has a refutation R such that the constraint $g_1 \geq 0$ is used $W^* \leq W$ times. We construct an assignment $\mathbf{x}$ to Φ as follows: for each variable x_i, if R contains the constraint $x_i - y_i^+ \geq 0$ or $x_i - y_i^+ - t^+_{i,1} \geq 0$, then set x_i to **true**. Otherwise set x_i to **false**.

We make the following observations about R:

1. If the constraints in S are removed from **H**, then **H** is feasible, thus these constraints must be used by R. Recall that these constraints are equivalent to $-\sum_{i=1}^{n} x_i \geq 1 - m$.

2. To get a contradiction, the defining constant of the derived constraint must be positive. Note that the only constraints with positive defining constant in $\mathbf{H}$ are of the form $c_j - d_j \geq 1$. As noted previously, eliminating d_j from each of these constraints requires W copies of the constraint $d_{j,1} \geq 0$. Thus, each of these constraints is used at most once by R. There are m such constraints, thus they must all be used by R. Consequently, the constraints in S can be each used at most once by R.
3. Consider the constraint $c_j - d_j \geq 1$. The only constraints with $-c_j$ in $\mathbf{H}$ are of the form $z_{i,l}^- - c_j \geq 0$ and $z_{i,l}^+ - c_j \geq 0$. Thus, R must contain a constraint of this form.
4. If R contains a constraint of the form $z_{i,l}^- - c_j \geq 0$, then it must contain the constraint $t_{i,l}^- - z_{i,l}^- \geq 0$ or $t_{i,l}^- - z_{i,l}^- - t_{i,l+1}^- \geq 0$. To cancel the $t_{i,l}^-$ variables, R must include the constraint $x_i - t_{i,1}^- \geq 0$. This constraint cancels the variable x_i from the constraint $-\sum_{i=1}^{n} x_i \geq 1 - m$. Thus, the constraints $x_i - y_i^+ \geq 0$ and $x_i - y_i^+ - t_{i,1}^+ \geq 0$ cannot be in R. This means that x_i is set to **false** by $\mathbf{x}$. Note that the constraint $z_{i,l}^- - c_j \geq 0$ is in $\mathbf{H}$ if and only if ϕ_j contains the literal $\neg x_i$. Thus, ϕ_j is satisfied by $\mathbf{x}$.
5. If R contains a constraint of the form $z_{i,l}^+ - c_j \geq 0$, then it must contain the constraint $t_{i,l}^+ - z_{i,l}^+ \geq 0$ or $t_{i,l}^+ - z_{i,l}^+ - t_{i,l+1}^+ \geq 0$. To cancel the $t_{i,l}^+$ variables, R must include the constraint $x_i - y_i^+ - t_{i,1}^+ \geq 0$. This means that x_i is set to **true** by $\mathbf{x}$. Note that the constraint $z_{i,l}^- - c_j \geq 0$ is in $\mathbf{H}$ if and only if ϕ_j contains the literal $\neg x_i$. Thus, ϕ_j is satisfied by $\mathbf{x}$.
6. As observed previously, canceling y_i^+ from the constraint $x_i - y_i^+ - z_{i,1}^+ \geq 0$ takes at least $w(i)$ uses of the constraint $g_1 \geq 0$. Thus, $\sum_{x_i : x_i = \mathbf{true}} w(i) \leq W^* \leq W$ as desired. □

Using Lemma 3, we now show that the CC_{Opt} problem for width 3 HCSs is **NPO-complete**.

Theorem 7. *The CC_{Opt} problem for width 3 HCSs is* **NPO-complete**.

Proof. The copy complexity of an HCS can be verified in polynomial time by providing the Farkas vector. Thus, the CC_{Opt} problem is in **NPO**. All that remains is to show **NPO-hardness**.

Let Φ be a CNF formula with m clauses over n variables where each variable x_i has weight $w(i)$. Using the construction in this section, we can construct a corresponding width 3 HCS $\mathbf{H}$. From Theorem 3, this HCS has a copy complexity of at most W if and only if Φ has a solution in which the total weight of **true** variables is at most W. This is a strict (and hence) PTAS reduction [11]. Consequently, the CC_{Opt} problem for width 3 HCSs is **NPO-complete**. □

Since the CC_{Opt} problem for width 3 HCSs is **NPO-complete**, this problem cannot be approximated to within a polynomial factor unless $\mathbf{P} = \mathbf{NP}$ [7].

5 Conclusion

In this paper, we analyzed the problem of determining bounds on the copy complexity bounds of HCSs. We showed that for any HCS, the copy complexity cannot exceed 2^{n-1}, where n is the number of variables in the HCS. We also showed that for each n, there exists a family of width 3 HCSs with copy complexity $2^{\lfloor \frac{n}{2} \rfloor}$. Additionally, we showed that the CC_D problem for width 3 HCSs is **NP-complete**.

From our perspective, the following avenues are worth pursuing:

1. The focus of this paper has been copy complexity with respect to the ADD refutation system. However, additional inference rules exist which allow for constraints to be multiplied by and divided by positive integers. We hope to replicate the analysis in this paper when we allow for the use of these additional inference rules.
2. The goal of this paper was to focus on the copy complexity of HCSs. In some refutation models, the goal is not so much to minimize the copy complexity, but to minimize the total number of **distinct** constraint replications. In other words, the first replication has a cost associated with it, but all other replications are gratis. It would be interesting to study HCSs in this model.

References

1. Ausiello, G., Crescenzi, P., Gambosi, G., Kann, V., Marchetti-Spaccamela, A., Protasi, M.: Complexity and Approximation: Combinatorial Optimization and their Approximability Properties, 1st edn. Springer, Cham (1999). https://doi.org/10.1007/978-3-642-58412-1
2. Bakhirkin, A., Monniaux, D.: Combining forward and backward abstract interpretation of horn clauses. In: Ranzato, F. (ed.) SAS 2017. LNCS, vol. 10422, pp. 23–45. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-66706-5_2
3. Bjørner, N., Gurfinkel, A., McMillan, K., Rybalchenko, A.: Horn clause solvers for program verification. In: Beklemishev, L.D., Blass, A., Dershowitz, N., Finkbeiner, B., Schulte, W. (eds.) Fields of Logic and Computation II. LNCS, vol. 9300, pp. 24–51. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-23534-9_2
4. Farkas, G.: Über die Theorie der Einfachen Ungleichungen. J. für die Reine und Angewandte Mathematik **124**(124), 1–27 (1902)
5. Impagliazzo, R., Paturi, R.: Complexity of k-sat. In: Proceedings. Fourteenth Annual IEEE Conference on Computational Complexity, pp. 237–240 (1999)
6. Impagliazzo, R., Paturi, R., Zane, F.: Which problems have strongly exponential complexity? J. Comput. Syst. Sci. **63**(4), 512–530 (2001)
7. Kann, V.: On the Approximability of NP-complete Optimization Problems. PhD thesis, Royal Institute of Technology Stockholm (1992)
8. Kleine Büning, H., Wojciechowski, P.J., Subramani, K.: New results on cutting plane proofs for Horn constraint systems. In: 39th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2019, 11–13 December, 2019, Bombay, India, pp. 43:1–43:14 (2019)
9. LiCalzi, M., Veinott, A.: Subextremal functions and lattice programming. SSRN Electron. J. **10**, 367 (2005)

10. Nemhauser, G.L., Wolsey, L.A.: Integer and Combinatorial Optimization. John Wiley & Sons, New York (1999)
11. Orponen, P., Mannila, H.: On approximation preserving reductions: Complete problems and robust measures. Technical Report, Department of Computer Science, University of Helsinki (1987)
12. Schrijver, A.: Theory of Linear and Integer Programming. John Wiley and Sons, New York (1987)