Exploiting a Hypergraph Model
for Finding Golomb Rulers

Manuel Sorge*, Hannes Moser**, Rolf Niedermeier, and Mathias Weller***

Institut für Softwaretechnik und Theoretische Informatik, TU Berlin, Berlin, Germany
{manuel.sorge, rolf.niedermeier, mathias.weller}@tu-berlin.de

Abstract Golomb rulers are special rulers where for any two marks it holds that the distance between them is unique. They find applications in positioning of radio channels, radio astronomy, communication networks, and bioinformatics. An important subproblem in constructing "compact" Golomb rulers is GOLOMB SUBRULER (GSR), which asks whether it is possible to make a given ruler Golomb by removing at most k marks. We initiate a study of GSR from a parameterized complexity perspective. In particular, we develop a hypergraph characterization of rulers and consider the construction and structure of the corresponding hypergraphs. We exploit their properties to derive polynomial-time data reduction rules that lead to a problem kernel for GSR with $O(k^3)$ marks. Finally, we provide a simplified NP-hardness construction for GSR.

Keywords: Hitting Set, NP-Hardness, Parameterized Complexity, Data Reduction, Problem Kernel, Forbidden Subgraph Characterization

1 Introduction

A *ruler* is a finite subset of $\mathbb{N}$, its elements are called *marks*. A ruler R is called *Golomb ruler* if no two pairs of marks from R have the same distance. For instance, $\{1, 2, 4, 8\}$ forms a Golomb ruler while $\{2, 3, 5, 7\}$ does not (since $7 - 5 = 5 - 3 = 2$, that is, distance 2 appears twice). The number of marks on a ruler is called its *order n* and the distance between its smallest mark (which can be assumed to be 0 without loss of generality) and its largest mark is called its *length*.[1] While it is easy to construct Golomb rulers, finding a shortest Golomb ruler for a given order n (encoded in unary) is assumed to be computationally intractable [6, 14]. However, there is no NP-hardness result for this problem so far, but due to the multitude of its practical applications, there have been several computational studies based on heuristics and massive parallelism (see [7, 8, 17] for example). The applications of Golomb ruler construction include radio frequency selection,

---

[1] The most compact Golomb ruler (measured in its length) containing four marks is $\{0, 1, 4, 6\}$.

radio astronomy, data encryption, communication networks, and bioinformatics [2, 3, 4, 14, 18]. For instance, when placing radio channels in the frequency spectrum, intermodulation interference is introduced by nonlinear behavior in transmitters and receivers. For example, three channels at frequencies a, b, c may intermodulate and create interference at the frequency $d = a + b - c$. This type of interference is avoided when placing the channels according to marks of a Golomb ruler, since then, there cannot be a channel at frequency d, because $d - a = b - c$. Currently, shortest Golomb rulers up to order $n = 26$ are known [7].

When constructing Golomb rulers one often has to place a number of marks within a set of limited possible positions. Meyer and Papakonstantinou [14] formalized this as the GOLOMB SUBRULER problem and showed its NP-hardness.[2]

> GOLOMB SUBRULER (GSR)
> **Input:** A finite ruler $R \subseteq \mathbb{N}$ and an integer $k \geq 0$.
> **Question:** Is there a Golomb ruler $R' \subseteq R$ such that $|R \setminus R'| \leq k$?

Our Results. We contribute a hypergraph characterization which helps us to show fixed-parameter tractability of GSR for parameter k and to develop efficient data reduction rules. More specifically, we provide a cubic-size problem kernel for GSR with respect to the parameter k. The kernelization result is based on structural observations of the corresponding hypergraphs and we also obtain some insight regarding forbidden subgraph characterizations of this hypergraph family. Finally, using the hypergraph notion, we provide an alternative, simpler many-one reduction for proving the NP-completeness of GSR. Due to space constraints most of the proofs and additional content is deferred to a full version of the paper.

Preliminaries. A central tool for our analysis of GSR are hypergraphs. A hypergraph basically is a system of subsets over some universe. More precisely, a *hypergraph $H = (V, E)$* consists of the *universe* or *set of vertices V* and the *set of hyperedges E*, where for each hyperedge $e \in E$, we have $e \subseteq V$. If used in context of hypergraphs, we use "edge" as synonym for "hyperedge". In particular, we work with 3,4-hypergraphs, meaning that all hyperedges have cardinality three or four. An edge of cardinality d are sometimes called d-edge. In this work, the vertices of a hypergraph will one-to-one correspond to marks on a ruler and the edges will one-to-one correspond to "conflicts" between marks, which will be defined later. We often use the corresponding terms synonymously. For a given hypergraph, we use n to denote the number of vertices and m to denote the number of hyperedges. With respect to rulers, n denotes the number of marks and m denotes the number of conflicts. If a vertex v is contained in an edge e, then e is said to be *incident* to v.

An *independent set $I \subseteq V$* of a hypergraph $H = (V, E)$ is a set of vertices such that no hyperedge $e \in E$ is a subset of I. In contrast, a *vertex cover* or,

---

[2] For brevity we reformulated the problem slightly. The original problem is to find a Golomb subruler containing at least a given number of marks. Clearly, this problem and our reformulation are equivalent under polynomial-time many-one reductions.

equivalently, a *hitting set* $C \subseteq V$ of H is a set of vertices such that $V \setminus C$ is an independent set of H. In the Hitting Set problem, a hypergraph H and an integer $l \geq 1$ is given and it is asked whether there is a hitting set in H that has cardinality at most l. We will characterize GSR as a special type of Hitting Set on 3,4-hypergraphs.

Besides hypergraph notation, we also use concepts of parameterized complexity [10, 12, 15]. A computational (typically NP-hard) problem is called *fixed-parameter tractable* with respect to a given parameter k (typically a positive integer) if instances of size n can be solved in $f(k) \cdot n^{O(1)}$ time. Herein, $f(k)$ is an arbitrary computable function. Note that fixed-parameter tractability is a stronger statement than just "solvable in polynomial time for constant parameter values" since k is not allowed to influence the degree of the polynomial.

An important concept in parameterized complexity is kernelization [5, 13]. Formally, a kernelization of a parameterized problem P is a polynomial-time algorithm that, given an instance (I, k) of P, computes an instance (I', k') of P such that both $|I'|$ and k' are bounded by a function depending only on k and such that (I', k') is a yes-instance if and only if (I, k) is a yes-instance. We call the output (I', k') a *problem kernel*.

2 Hypergraph Characterization

In this section, we provide a simple hypergraph characterization of rulers with respect to the Golomb property and consider structural properties of the implied hypergraphs. The characterization serves as basis for the succeeding sections.

Let $R \subset \mathbb{N}$ be a ruler. We say that two marks $a, b \in R$ *measure* the *distance* $|a - b|$. We say that the measurements of two pairs of marks a, b and c, d *overlap*, if the length of the ruler $\{a, b, c, d\}$ is strictly smaller than $|a - b| + |c - d|$. A *conflict* is an inclusionwise minimal non-Golomb ruler. That is, a conflict is a set of three or four marks that consists of two distinct unordered pairs of marks that measure the same distance. See also Figure 1. The *conflict hypergraph* of a ruler R is the hypergraph $H_R = (R, E)$, where E is the set of all conflicts contained in R. With respect to rulers and conflict hypergraphs, we synonymously use the terms vertices and marks, as well as edges and conflicts, respectively. Analogously to d-edges (edges of cardinality d), we speak of 3-conflicts and 4-conflicts. The following lemma is obvious.

Lemma 1. *Let R be a ruler and $H_R = (R, E)$ be its conflict hypergraph. Then R is Golomb if and only if $E = \emptyset$.*

Hypergraph Construction. We now consider the construction of conflict hypergraphs. It is obvious that they can be constructed in $O(n^4)$ time. We show that this bound can be improved to $O(n^3)$ in the worst case and this is also tight.

Instead of the trivial approach of verifying every possible tuple, one can consider the distances between marks present in the ruler and examine which of them lead to edges in the graph. Algorithm HypergraphConstruction describes such a procedure. In this algorithm we use an auxiliary map M that maps every
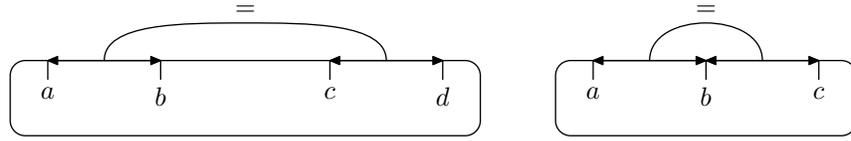
Figure 1: Two rulers with the marks a, b, c, and d, respectively. To the left, we see that the marks a and b measure the same distance as c and d. We consider this to be a conflict with respect to Golomb rulers and model it as an edge $\{a, b, c, d\}$ in the corresponding hypergraph. To the right we see a degenerated form of a conflict which leads to an edge with only three vertices.

Algorithm HypergraphConstruction: Constructing a conflict hypergraph for a given ruler

---

Input: A finite ruler $R \subset \mathbb{N}$.
Output: A hypergraph $H_R = (R, E)$.
1 Start with an empty hypergraph H with vertex set R;
2 Create an empty map M that maps integers to lists;
3 $\delta_{\max} \leftarrow \max\{x : x \in R\} - \min\{x : x \in R\}$;
4 **for** $i \in R$ **do**
5   **for** $j \in R, i < j \leq i + \delta_{max}/2$ **do**
6   Add (i, j) to the list mapped to $j - i$ in M;

7 **for** $i \in R$ **do**
8   **for** $j \in R, i < j \leq i + \delta_{max}/2$ **do**
9   **for** (k, l) *in the list mapped to* $j - i$ *in* $M, j \leq k$ **do**
10   Add the edge $\{i, j, k, l\}$ to H;

11 **return** H;

---

measurable distance to pairs of marks that measure it. First, we fill M: The first two loops iterate over distances present in R and add every pair of vertices to the entry in M corresponding to their distance. Then, for every short distance in R (every distance at most half the maximum distance in R) M contains a list with all pairs of marks that measure this distance. In the second step, we add the edges to the designated conflict hypergraph H: The last three nested loops again iterate over distances present in the ruler and simply add an edge to H for every pair of marks that measure this distance. To formally prove the correctness of Algorithm HypergraphConstruction we need the following auxiliary lemma.

Lemma 2. *Every edge in a conflict hypergraph is due to two pairs of marks that measure the same distance and the measurements do not overlap.*

This lemma also allows us to disregard distances measured by marks that are more than half the length of the ruler apart, because measurements of such long

distances must overlap. This basically gives the correctness of Algorithm HypergraphConstruction. The running time can be shown to be cubic:

Lemma 3. *Algorithm HypergraphConstruction constructs a conflict hypergraph for its input ruler in* $O(n^3)$ *time.*

Note that we only consider short distances in the loop-headers in lines 5 and 8 of Algorithm HypergraphConstruction. However, the omission of long distances does not influence the asymptotic upper bound on the running time. This is a heuristic trick that could prove useful in practice.

Unfortunately, the running time cannot be further improved because there are rulers that contain $\Omega(n^3)$ conflicts. We can show that this holds for rulers whose marks form intervals in $\mathbb{N}$ and, thus, we obtain the following theorem.

Theorem 1. *There is a hypergraph characterization for rulers such that Golomb rulers one-to-one correspond to hypergraphs without edges. The worst-case time complexity of computing the conflict hypergraph for a ruler R is* $\Theta(n^3)$.

Theorem 1 implies that GSR is fixed-parameter tractable with respect to the parameter "number k of deleted marks": By Lemma 1 Golomb rulers and only these correspond to edge-less conflict hypergraphs. Thus, the task of removing marks to obtain a Golomb subruler reduces to the task of removing vertices from a hypergraph to obtain an edge-less graph. This is exactly the HITTING SET problem and, thus, we can apply algorithms for this problem to GSR. The fastest known parameterized algorithm solving instances of HITTING SET with m edges and at most four vertices per edge runs in $O(3.076^\ell + m)$ time [9]. This implies that GSR can be solved in $O(3.076^k + n^3)$ time. However, the instances created in the reduction sketched above seem rather restricted and this might lead to speedups.

Observations on the Structure of Conflict Hypergraphs. We are interested in the structure of the constructed hypergraphs, because we would like to develop efficient algorithms exploiting the specific structure of GSR. This proves successful in that we are able to give forbidden subgraphs that we use in Section 3 to prove a polynomial-size problem kernel. However, the structure of conflict hypergraphs is also interesting on its own. In this regard, our studies merely form a starting point for further research.

At first, notice that the set of conflict hypergraphs is a strict subset of all hypergraphs with edges of size three and four. This is because the construction algorithm can be carried out using $O(n^3)$ edge additions, n being the number of marks and thus vertices. However, general 3,4-hypergraphs can have $\binom{n}{4} \in \Omega(n^4)$ edges.

It is interesting to determine which hypergraphs can and which cannot be constructed. For example, this could be done through a forbidden subgraph characterization: a set F of hypergraphs such that a 3,4-hypergraph H is a conflict hypergraph for a ruler if and only if H does not contain a hypergraph $G \in F$ as subgraph. Unfortunately, we still cannot provide a forbidden subgraph characterization. However, we make partial progress by providing some forbidden subgraphs
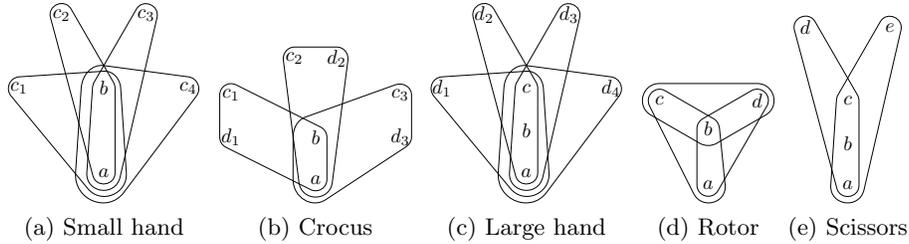
Figure 2: Forbidden subgraphs (2a, 2c, and 2d) and forbidden induced subgraphs (2b and 2e) of conflict hypergraphs of rulers. Letters (and indices) represent vertices and closed curves encircling vertices represent hyperedges.

and forbidden induced subgraphs (see Figure 2). These might be helpful in research towards a complete forbidden subgraph characterization of conflict hypergraphs, if one exists. They might also be useful in deriving more efficient algorithms for GSR.

Due to space constraints, we only prove the absence of the subgraph in Figure 2a in conflict hypergraphs. We give the basic idea for the induced subgraph shown in Figure 2b. These two forbidden subgraphs are used in our data reduction rules in Section 3. For the forbidden subgraphs in Figures 2c through 2e, we omit a formal proof and refer to Sorge [19] for details.

Lemma 4 (Forbidden subgraph "small hand"). *The graph shown in Figure 2a is a forbidden subgraph in a conflict hypergraph.*

Proof. In an edge with three marks there is one mark exactly between the other two. Let a, b be two marks on a ruler. Where can a third mark c in an edge already comprising b and a be on the ruler? Either a, b, or c can be the mark in the middle. Thus, there are at most three edges with three vertices intersecting in a and b. □

For the forbidden induced subgraph shown in Figure 2b, we first obtain an observation about conflicts that intersect in two marks.

Lemma 5. *Let $a < b$ and $c < d$ be four marks in a conflict hypergraph $H_R = (R, E)$ and let $\{a, b, c, d\} \in E$. The cases (i) $a - c = d - b$ and (ii) $a - b = c - d$ do not overlap and cover every situation. Furthermore, if there are two conflicts $e_1, e_2 \in E$, both containing a, b and corresponding to the same case, then there is another conflict $(e_1 \cup e_2) \setminus \{a, b\} \in E$.*

For a proof, one can basically consider all possible configurations of a, b, c, d on a ruler and show that in each case the implied equations of the distances can be rewritten such that other conflicts are implied. Thus, if there are three conflicts intersecting in two marks, at least two of them correspond to the same case. Hence, there is at least one additional conflict, yielding that the graph shown in Figure 2b cannot be an induced subgraph in a conflict hypergraph.

Observe that, in order for a forbidden subgraph characterization for the conflict hypergraphs to exist, the problem of deciding whether a given hypergraph is isomorphic to a conflict hypergraph has to be decidable. We note that this is the case and that it even lies in NP (details deferred to a full version of the paper). Answering the question whether this problem is NP-hard is an important challenge in this regard, as NP-hardness would rule out characterizations through finitely many subgraphs.

3 Polynomial-Time Data Reduction and Kernelization

In this section, we present a kernelization algorithm for the GOLOMB SUB-RULER (GSR) problem parameterized by the number k of deleted marks. We use the hypergraph characterization and structural observations from Section 2 to derive data reduction rules such that after at most $O(k(n + m))$ processing time, an equivalent instance with at most $O(k^3)$ marks remains.

Using the conflict hypergraphs, one can regard GSR as a special case of the HITTING SET problem. If d is the maximum number of vertices in an edge, then there is a problem kernel for HITTING SET with at most $O(k^{d-1})$ vertices [1]. However, the reduction rules used for this problem kernel are not directly applicable to our problem because the instances produced by these reduction rules may not correspond to a ruler anymore: The high-degree rule used by Abu-Khzam [1] removes edges and inserts parts of them into the hypergraph. These parts, however, can be of size one or two, violating conditions for conflict hypergraphs. Abu-Khzam additionally applies a "crown-reduction" which deletes vertices while keeping their incident edges intact. This operation is also not supported by conflict hypergraphs of rulers.

We develop alternative data reduction rules that are simpler than the ones given by Abu-Khzam. Despite this, the rules retain the problem kernel size of a cubic number of vertices due to the restricted structure of the conflict hypergraphs.

For our kernelization algorithm, we employ two modified high-degree reduction rules. When exhaustively applied, one of these rules suffices to bound the number of 3-conflicts in the conflict hypergraph, the other rule bounds the number of 4-conflicts. With the help of these two bounds, we are then able to bound the number of marks in a reduced instance. In the following description of the reduction rules, we assume that the conflict hypergraph of the input ruler has been computed, and is kept updated alongside the ruler. First we need the following simple rule.

Reduction Rule 1 (Isolated marks). *If there is a mark that is not present in any conflict, then remove it.*

It is clear that such marks never have to be deleted in order to make the input ruler conflict-free.

The next two "high-degree" rules are similar in spirit to rules of Abu-Khzam [1], but differ in decisive details.

The following reduction rule is based on the small hand forbidden subgraph (Lemma 4) and the observation that if there are more than $3k$ 3-conflicts intersecting in a mark v, then v has to be deleted: This is because deleting any other mark can destroy at most three 3-conflicts that contain v and, thus, at least $k+1$ marks would have to be deleted.

Reduction Rule 2 (High degree for 3-conflicts). *If there is a mark v that is contained in more than $3k$ 3-conflicts, then remove v from the ruler, remove any conflicts containing v and decrement k by one.*

Lemma 6. *Rule 2 is correct and one application costs $O(n+m)$ time. If Rule 2 cannot be applied to a ruler R and R can be made Golomb with at most k mark deletions, then H_R has at most $3k^2$ 3-conflicts.*

To lift the high-degree concept to 4-conflicts, we need the following auxiliary lemma. It can be seen as an analogous replacement for Lemma 4.

Lemma 7. *Let (R, k) be a yes-instance of GSR and let $a < b$ be two marks in R. The conflict hypergraph H_R has at most $3k$ 4-conflicts that intersect in a, b.*

The basic idea for the proof is to use Lemma 5 to show that if there are more than $3k$ conflicts intersecting in a, b, then the additional edges implied form a clique-like structure in which more than k marks have to be deleted. Thus, we obtain the following reduction rule:

Reduction Rule 3 (High degree for 4-conflicts). *If there is a mark v that is contained in more than $3k^2$ 4-conflicts, then remove v from the ruler, remove any conflicts containing v, and decrement k by one.*

Lemma 8. *Rule 3 is correct and one application takes $O(n+m)$ time. If Rule 3 cannot be applied to a ruler R and R can be made Golomb with at most k mark deletions, then H_R has at most $3k^3$ 4-conflicts.*

Concluding, we obtain the following theorem.

Theorem 2. GOLOMB SUBRULER *has a problem kernel with at most $9k^3 + 6k^2 + k$ marks. The conflict hypergraph of the ruler of a kernelized instance has at most $3k^3$ 4-conflicts and $3k^2$ 3-conflicts. The problem kernel can be computed in $O(k(n+m))$ time if the conflict hypergraph is given.*

Note that we can only achieve a running time of $O(k(n+m))$ if the conflict hypergraph of the given ruler is also given. In the worst case, its computation would imply an additional running time of $O(n^3)$ (Theorem 1).

4 A Simplified Hardness Construction

Meyer and Papakonstantinou [14] showed that GOLOMB SUBRULER (GSR) is NP-hard via a reduction from an NP-hard SAT variant. However, the construction of the ruler corresponding to the SAT formula is involved and hard to comprehend.
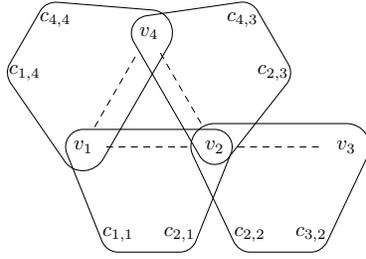
Figure 3: A graph (vertices v_1 through v_4 and dashed edges) and the corresponding hypergraph produced by Construction 1 (vertices $v_i, c_{i_j,j}$ and solid hyperedges).

Using our hypergraph characterization of rulers (Section 2), we provide a reduction from the NP-complete INDEPENDENT SET problem yielding a much simpler construction. We observe that GSR is hard even when there are no three marks that measure the same distance twice. We also note that the corresponding reduction implies a W[1]-hardness result, that is, presumable fixed-parameter intractability, for a modified version of GSR, where the size of the sought ruler depends on the number of conflicts. Both results did not directly follow from the original proof.

In INDEPENDENT SET a graph $G = (V, E)$ and an integer $l \geq 1$ are given and it is asked whether there is a vertex set $I \subseteq V$ in G such that no edge of G is contained in I and $|I| \geq l$. In HYPERGRAPH INDEPENDENT SET we simply substitute hypergraphs for graphs. For readability we opt to use the word "edges" for vertex sets of cardinality two and "hyperedges" for vertex sets of higher order in this section.

The basic idea of our reduction from INDEPENDENT SET is to output instances of HYPERGRAPH INDEPENDENT SET that constitute conflict hypergraphs for some rulers. Since the marks of a Golomb ruler R form an independent set in all conflict hypergraphs of superrulers of R, in this way one achieves a reduction from INDEPENDENT SET to GSR.

Construction 1. *Let a graph G and an integer l constitute an instance of* INDEPENDENT SET. *Construct the hypergraph H from G as follows: Add all vertices of G to H. Let $v_1, \ldots, v_n$ be the vertices of G and let $e_1, \ldots, e_m$ be the edges in G. For every edge $e_j = \{v_{i_j}, v_{k_j}\} \in E(G)$, introduce two new vertices $c_{i_j,j}, c_{k_j,j}$ into H and add the hyperedge $e_j \cup \{c_{i_j,j}, c_{k_j,j}\}$ to H. The hypergraph H and the integer $l + 2m$ constitute an instance of* HYPERGRAPH INDEPENDENT SET. *See also Figure 3.*

Lemma 9. *Construction 1 is a polynomial-time many-one reduction from* INDEPENDENT SET *to* HYPERGRAPH INDEPENDENT SET.

In order to prove NP-hardness for GSR, we now give a method to construct a ruler R from a hypergraph H produced by Construction 1 such that the conflict hypergraph H_R of R is isomorphic to H.

Construction 2. *Let H be a hypergraph derived from a graph as in Construction 1. Construct a ruler R as follows: For every vertex $v_i \in V(H)$, introduce the mark $2^{(m+2)i}$ into R, and for every vertex $c_{i,j} \in V(H)$, introduce the mark $2^{(m+2)i} + 2^j - 1$ into R.*

Lemma 10. *Construction 2 is polynomial-time computable and H is isomorphic to H_R.*

For the proof of Lemma 10 we need the following observation.

Observation 1. *The ruler $\{2^i : i \in \mathbb{N}\}$ is a Golomb ruler.*

Proof (Lemma 10). It is easy to see that Construction 2 can be carried out in polynomial time. In order to prove the second part of the lemma, we show that the function $\phi : V(H) \to V(H_R)$ that assigns each vertex v_i the mark $2^{(m+2)i}$ and each vertex $c_{i,j}$ the mark $2^{(m+2)i} + 2^j - 1$ is a hypergraph isomorphism between H and H_R. Thus, we prove that for each hyperedge in H, there is a corresponding conflict in H_R and vice-versa. For notational convenience, we denote $c_{i,0} := v_i, 1 \le i \le n$.

First consider a hyperedge in H and let this hyperedge contain the vertices $c_{i_1,0}, c_{i_1,j}, c_{i_2,0}$, and $c_{i_2,j}$. Then, the four marks $\phi(c_{i_1,0}), \phi(c_{i_1,j}), \phi(c_{i_2,0})$, and $\phi(c_{i_2,j})$ form a conflict in H_R, because

$$2^{(m+2)i_1} - (2^{(m+2)i_1} + 2^j - 1) = 2^{(m+2)i_2} - (2^{(m+2)i_2} + 2^j - 1).$$

Next, consider a conflict in H_R, that is, there are positive, not necessarily distinct integers $1 \le i_1, i_2, i_3, i_4 \le n$ and $0 \le j_1, j_2, j_3, j_4 \le m$ such that

$$\phi(c_{i_1,j_1}) - \phi(c_{i_2,j_2}) = \phi(c_{i_3,j_3}) - \phi(c_{i_4,j_4}). \tag{1}$$

Note that by allowing the variables j_k to assume the value 0 we also catch conflicts that contain marks corresponding to vertices v_{i_k}. We claim that we can assume that

$$i_1 = i_2 \qquad \text{and} \qquad i_3 = i_4. \tag{2}$$

Provided that this is the case, Equation 1 simplifies to $2^{j_1} - 2^{j_2} = 2^{j_3} - 2^{j_4}$, and this yields $J := |\{j_1, j_2, j_3, j_4\}| \le 2$, because, by Observation 1, the ruler $\{2^i : i \in \mathbb{N}\}$ is a Golomb ruler and $J = 3 \lor J = 4$ would imply that it contains a conflict. This means that either (i) $J = 1 \lor j_1 = j_2, j_3 = j_4$, or (ii) $j_1 = j_3, j_2 = j_4$. In case (i) we have that $c_{i_1,j_1} = c_{i_2,j_2}, c_{i_3,j_3} = c_{i_4,j_4}$ implying that Equation 1 does not represent a conflict in H_R. Hence, we can assume that case (ii) holds. Then, we get that either $j_1 = j_3 = 0$ or $j_2 = j_4 = 0$ as follows: For sake of contradiction, assume that $j_1 \ne 0 \ne j_2$. This implies that there are four vertices $c_{i_1,j_1}, c_{i_1,j_2}, c_{i_2,j_1}, c_{i_2,j_2}$ in H. Then, however, by Construction 1, there are two edges between the vertices v_{i_1}, v_{i_2} in the graph that H has been constructed from; this is a contradiction. Without loss of generality, let $j_1 = 0$. Thus, we get that each conflict consists of marks of the form

$$\phi(c_{i_1,0}) - \phi(c_{i_1,j_2}) = \phi(c_{i_3,0}) - \phi(c_{i_3,j_2}) \qquad \text{where} \qquad 1 \le j_2 \le m,$$

that is, each conflict represents a hyperedge in H.

We now have that ϕ is a hypergraph isomorphism if Condition 2 holds. For Condition 2, we observe that the "$2^{(m+2)i}$ parts" and the "2^j parts" of Equation 1

are independent in a sense: If we divide both sides of Equation 1 by 2^m, every 2^j part shrinks to at most 1. However, the absolute of the difference of two $2^{(m+2)i}$ parts will still be at least 4. This implies that, if Equation 1 holds, the statement $2^{(m+2)i_1} - 2^{(m+2)i_2} = 2^{(m+2)i_3} - 2^{(m+2)i_4}$ holds, too. By Observation 1 we again get that $|\{i_1, i_2, i_3, i_4\}| \leq 2$. Now, if $i_1 \neq i_2$, then $i_1 = i_3, i_2 = i_4$, and we get an equation that is equivalent to Equation 1 such that Condition 2 holds, by simply adding $\phi(c_{i_2,j_2}) - \phi(c_{i_3,j_3})$ to both sides and renaming the vertices appropriately. □

Lemma 10 implies the following theorem:

Theorem 3. Golomb Subruler *is NP-complete, even if all conflicts in the input instance are 4-conflicts.*

5 Conclusion

In this work, we continued studies concerning the algorithmic complexity of Golomb Subruler (GSR) In particular, we initiated research on its parameterized complexity (particularly, a cubic-size problem kernel was developed) and studied combinatorial properties of GSR instances, which can be considered as special Hitting Set problems. Some preliminary experimental investigations indicated that our data reduction rules and simple search tree strategies may be beneficial in practical studies for Golomb ruler construction. However, it currently seems most promising to try to combine the data reduction with known approaches such as the Distributed.net project [7].

Golomb ruler construction leads to numerous challenges for algorithmic and complexity-theoretic research. For instance, there is the unsettled computational complexity of constructing shortest Golomb rulers of order n (where input n is specified in unary) [6]. This has been open for at least nine years. Moreover there are numerous natural variations of Golomb ruler construction [14, 19]. In this paper, we focused on GSR introduced by Meyer and Papakonstantinou [14]. Even restricting attention to GSR, a number of interesting research challenges remain: Which graphs constitute a complete forbidden (induced) subgraph characterization of conflict hypergraphs (see Section 2)? Can a given hypergraph be recognized to be a conflict hypergraph in polynomial time, or is this task NP-hard? Are there other interesting (structural) parameterizations for GSR in the spirit of multivariate algorithmics [11, 16]?

References

[1] F. N. Abu-Khzam. A kernelization algorithm for d-Hitting Set. *J. Comput. System Sci.*, 76(7):524–531, 2010.

[2] W. Babcock. Intermodulation interference in radio systems. *Bell Syst. Tech. J.*, pages 63–73, 1953.

[3] G. Bloom and S. Golomb. Applications of numbered undirected graphs. *Proc. IEEE*, 65(4):562–570, 1977.

[4] E. Blum, F. Biraud, and J. Ribes. On optimal synthetic linear arrays with applications to radioastronomy. *IEEE T. Antenn. Propag.*, 22:108–109, 1974.

[5] H. L. Bodlaender. Kernelization: New upper and lower bound techniques. In *Proc. 4th IWPEC*, volume 5917 of *LNCS*, pages 17–37. Springer, 2009.

[6] A. Dimitromanolakis. Analysis of the Golomb ruler and the Sidon set problems, and determination of large, near-optimal Golomb rulers. Master's thesis, Department of Electronic and Computer Engineering, Technical University of Crete, June 2002.

[7] Distributed.net. Home page, Accessed April 2012. URL http://www.distributed.net/.

[8] A. Dollas, W. T. Rankin, and D. McCracken. A new algorithm for Golomb ruler derivation and proof of the 19 mark ruler. *IEEE T. Inform. Theory*, 44(1):379–382, 1998.

[9] M. Dom, J. Guo, F. Hüffner, R. Niedermeier, and A. Truss. Fixed-parameter tractability results for feedback set problems in tournaments. *J. Discrete Algorithms*, 8(1):76–86, 2010.

[10] R. G. Downey and M. R. Fellows. *Parameterized Complexity*. Springer, 1999.

[11] M. Fellows. Towards fully multivariate algorithmics: Some new results and directions in parameter ecology. In *Proc. 20th IWOCA*, volume 5874 of *LNCS*, pages 2–10. Springer, 2009.

[12] J. Flum and M. Grohe. *Parameterized Complexity Theory*. Springer, 2006.

[13] J. Guo and R. Niedermeier. Invitation to data reduction and problem kernelization. *ACM SIGACT News*, 38(1):31–45, 2007.

[14] C. Meyer and P. A. Papakonstantinou. On the complexity of constructing Golomb rulers. *Discrete Appl. Math.*, 157:738–748, 2008.

[15] R. Niedermeier. *Invitation to Fixed-Parameter Algorithms*. Oxford University Press, 2006.

[16] R. Niedermeier. Reflections on multivariate algorithmics and problem parameterization. In *Proc. 27th STACS*, volume 5 of *Dagstuhl Seminar Proceedings*, pages 17–32, Dagstuhl, Germany, 2010. IBFI Dagstuhl, Germany.

[17] F. Pereira, J. Tavares, and E. Costa. Golomb rulers: The advantage of evolution. In *Progress in Artificial Intelligence*, volume 2902 of *LNCS*, pages 29–42. Springer, 2003.

[18] W. T. Rankin. Optimal Golomb rulers: An exhaustive parallel search implementation. Master's thesis, Department of Electrical Engineering, Duke University, Durham, 1993. Addendum by Aviral Singh.

[19] M. Sorge. Algorithmic aspects of Golomb ruler construction. Studienarbeit, Institut für Informatik, Friedrich-Schiller-Universität Jena, Germany, 2010. Available electronically, arXiv:1005.5395v2.