Detecting Points in Integer Cones of Polytopes is Double-Exponentially Hard*

Łukasz Kowalik[†] Alexandra Lassota[‡] Konrad Majewski[§]
Michał Pilipczuk[¶] Marek Sokołowski[‖]

July 4, 2023

arXiv:2307.00406v1 [cs.DS] 1 Jul 2023

Abstract

Let d be a positive integer. For a finite set $X \subseteq \mathbb{R}^d$, we define its *integer cone* as the set $\mathsf{IntCone}(X) := \{\sum_{x \in X} \lambda_x \cdot x \mid \lambda_x \in \mathbb{Z}_{\geqslant 0}\} \subseteq \mathbb{R}^d$. Goemans and Rothvoss showed that, given two polytopes $\mathcal{P}, \mathcal{Q} \subseteq \mathbb{R}^d$ with $\mathcal{P}$ being bounded, one can decide whether $\mathsf{IntCone}(\mathcal{P} \cap \mathbb{Z}^d)$ intersects $\mathcal{Q}$ in time $\mathsf{enc}(\mathcal{P})^{2^{\mathcal{O}(d)}} \cdot \mathsf{enc}(\mathcal{Q})^{\mathcal{O}(1)}$ [J. ACM 2020], where $\mathsf{enc}(\cdot)$ denotes the number of bits required to encode a polytope through a system of linear inequalities. This result is the cornerstone of their XP algorithm for Bin Packing parameterized by the number of different item sizes.

We complement their result by providing a conditional lower bound. In particular, we prove that, unless the ETH fails, there is no algorithm which, given a bounded polytope $\mathcal{P} \subseteq \mathbb{R}^d$ and a point $q \in \mathbb{Z}^d$, decides whether $q \in \mathsf{IntCone}(\mathcal{P} \cap \mathbb{Z}^d)$ in time $\mathsf{enc}(\mathcal{P}, q)^{2^{o(d)}}$. Note that this does *not* rule out the existence of a fixed-parameter tractable algorithm for the problem, but shows that dependence of the running time on the parameter d must be at least doubly-exponential.

1 Introduction

Consider the following high-multiplicity variant of the Bin Packing problem: given a vector $s = (s_1, \dots, s_d) \in [0,1]^d$ of item sizes and a vector of multiplicities $a = (a_1, \dots, a_d) \in \mathbb{Z}^d_{\geqslant 0}$, find the smallest integer B so that the collection of items containing a_i items of size s_i, for each $i \in \{1, \dots, d\}$, can be entirely packed into B unit-size bins. In their celebrated work [5], Goemans and Rothvoss gave an algorithm for this problem with time complexity $\mathsf{enc}(s,a)^{2^{\mathcal{O}(d)}}$, where $\mathsf{enc}(s,a)$ denotes the total bitsize of the encoding of s and a in binary. In the terminology of parameterized complexity, this puts high-multiplicity Bin Packing parameterized by the number of different item sizes in the complexity class XP.

In fact, Goemans and Rothvoss studied the more general Cone and Polytope Intersection problem, defined as follows: given two polytopes $\mathcal{P}, \mathcal{Q} \subseteq \mathbb{R}^d$, where $\mathcal{P}$ is bounded, is there a point in $\mathcal{Q}$ that can be expressed as a nonnegative integer combination of integer points in $\mathcal{P}$? Goemans and Rothvoss gave an algorithm for this problem with running time $\mathsf{enc}(\mathcal{P})^{2^{\mathcal{O}(d)}} \cdot \mathsf{enc}(\mathcal{Q})^{\mathcal{O}(1)}$, where $\mathsf{enc}(\mathcal{R})$ denotes the total bitsize of the encoding of a polytope $\mathcal{R}$ through a system of linear

*This work is a part of project BOBR (ŁK, KM, MP, MS) that has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 948057). A. Lassota was supported by the Swiss National Science Foundation within the project *Complexity of Integer Programming* (207365).

[†]Institute of Informatics, University of Warsaw, Poland (kowalik@mimuw.edu.pl)
[‡]Institute of Mathematics, EPFL, Lausanne, Switzerland (alexandra.lassota@epfl.ch)
[§]Institute of Informatics, University of Warsaw, Poland (k.majewski@mimuw.edu.pl)
[¶]Institute of Informatics, University of Warsaw, Poland (michal.pilipczuk@mimuw.edu.pl)
[‖]Institute of Informatics, University of Warsaw, Poland (marek.sokolowski@mimuw.edu.pl)

inequalities. They showed that high-multiplicity Bin Packing admits a simple reduction to Cone and Polytope Intersection, where in essence, integer points in $\mathcal{P}$ correspond to possible configurations of items that fit into a single bin and $\mathcal{Q}$ is the point corresponding to all items (more precisely, $\mathcal{P} = \{\binom{x}{1} \in \mathbb{R}^{d+1}_{\geqslant 0} \mid s^T x \leqslant 1\}$ and $\mathcal{Q} = \{\binom{a}{B}\}$). In fact, Cone and Polytope Intersection is a much more versatile problem: in [5, Section 6], Goemans and Rothvoss present a number of applications of their result to other problems in the area of scheduling.

Whether the result of Goemans and Rothvoss for high-multiplicity Bin Packing can be improved to fixed-parameter tractability is considered a major problem in the area. It was asked already by Goemans and Rothvoss in [5], addressed again by Jansen and Klein in [7], and also discussed in the survey of Mnich and van Bevern [12].

In this work, we take a step into solving the complexity of this problem. We prove the following result that shows that the doubly-exponential dependence on d in the running times of algorithms for Cone and Polytope Intersection is necessary, assuming the Exponential-Time Hypothesis (ETH). The lower bound holds even for the simpler Point in Cone problem, where the polytope $\mathcal{Q}$ consists of a single integer point $q \in \mathbb{Z}^d$.

Theorem 1. *Unless the ETH fails, there is no algorithm solving Point in Cone in time* $\mathsf{enc}(\mathcal{P}, q)^{2^{o(d)}}$, *where* $\mathsf{enc}(\mathcal{P}, q)$ *is the total number of bits required to encode both* $\mathcal{P}$ *and* q.

Notice that theorem 1 does *not* rule out the possibility that there exists a fixed-parameter algorithm with running time $f(d) \cdot \mathsf{enc}(\mathcal{P}, q)^{\mathcal{O}(1)}$ for some function f. However, it shows that for this to hold, function f would need to be at least doubly exponential, assuming the ETH.

Let us briefly elaborate on our proof of theorem 1 and its relation to previous work. The cornerstone of the result of Goemans and Rothvoss is a statement called Structure Theorem, which essentially says the following: if an instance of Cone and Polytope Intersection has a solution, then it has a solution whose *support* — the set of integer points in $\mathcal{P}$ participating in the nonnegative integer combination yielding a point in $\mathcal{Q}$ — has size at most 2^{2d+1}. Moreover, except for a few outliers, this support is contained within a carefully crafted set X consisting of roughly $\mathsf{enc}(\mathcal{P})^{\mathcal{O}(d)}$ integer points within $\mathcal{P}$. In subsequent work [7], Jansen and Klein showed a more refined variant of the Structure Theorem where X is just the set of vertices of the convex hull of the integer points lying in $\mathcal{P}$; but the exponential-in-d bound on the size of the support persists. The appearance of this bound in both works [5, 7] originates in the following elegant observation of Eisenbrand and Shmonin [3]: whenever some point v can be represented as a nonnegative integer combination of integer points in $\mathcal{P}$, one can always choose such a representation of v with support of size bounded by 2^d (see [5, Lemma 3.4] for a streamlined proof). In [5, Section 8], Goemans and Rothvoss gave an example showing that the 2^d bound is tight up to a multiplicative factor of 2, thereby arguing that within their framework, one cannot hope for any substantially better bound on the support size. The main conceptual contribution of this work can be expressed as follows: the construction showing the tightness of the observation of Eisenbrand and Shmonin not only exposes a bottleneck within the support-based approach of [5, 7], but in fact can be used as a gadget in a hardness reduction proving that the doubly-exponential dependence on d in the running time is necessary for the whole problem, assuming the ETH.

Finally, we remark that tight doubly-exponential lower bounds under the ETH appear scarcely in the literature, as in reductions proving such lower bounds, the parameter of the output instance has to depend logarithmically on the size of the input instance of 3-SAT. A few examples of such lower bounds can be found here: [2, 4, 8, 9, 10, 11]; our work adds Point in Cone to this rather exclusive list.

2 Preliminaries

For a positive integer n, we denote $[n] \coloneqq \{1, 2, \ldots, n\}$ and $[n]_0 \coloneqq \{0, 1, \ldots, n-1\}$.

Euclidean spaces. Fix a positive integer d. We call the elements of $\mathbb{R}^d$ *vectors* (or *points*). Given a vector $x \in \mathbb{R}^d$, we denote its i-th coordinate (for $i \in [d]$) by $x(i)$. By $\mathbf{1}_d$, we denote the

d-dimensional vector of all ones, that is, $\mathbf{1}_d = (1, \ldots, 1) \in \mathbb{Z}^d$. When the dimension d is clear from the context, we omit it from the subscript and simply write $\mathbf{1}$ instead.

We allow vectors to be added to each other, and to be multiplied by a scalar $\lambda \in \mathbb{R}$. Both operations come from treating the space $\mathbb{R}^d$ as a linear space over $\mathbb{R}$. Given a finite set $X \subseteq \mathbb{R}^d$, we define its *integer cone* as the set

$$\mathsf{IntCone}(X) := \left\{ \sum_{x \in X} \lambda_x \cdot x \mid \lambda_x \in \mathbb{Z}^d_{\geqslant 0} \text{ for every } x \in X \right\}.$$

Polytopes. In this work a *d-dimensional polytope* is a subset of points in $\mathbb{R}^d$ satisfying a system of linear inequalities with integer coefficients, that is, a set of the form $\mathcal{P} := \{x \in \mathbb{R}^d \mid Ax \leqslant b\}$, where $A \in \mathbb{Z}^{d \times m}$ and $b \in \mathbb{Z}^m$ for some positive integer m. Then, the *encoding size of* $\mathcal{P}$, denoted $\mathsf{enc}(\mathcal{P})$, is the total number of bits required to encode the matrix A and the vector b. We say that the polytope $\mathcal{P}$ is *bounded* if there exists a number $M \in \mathbb{Z}$ such that for all $x \in \mathcal{P}$ and $i \in [d]$, we have $|x(i)| \leqslant M$.

We can now define the main problem studied in this paper, namely Point in Cone.

> Point in Cone
> **Input:** A positive integer d, a bounded polytope $\mathcal{P} \subseteq \mathbb{R}^d$ (given by a matrix $A \in \mathbb{Z}^{m \times d}$ and a vector $b \in \mathbb{Z}^m$ for some integer m), and a point $q \in \mathbb{Z}^d$.
> **Question:** Is $q \in \mathsf{IntCone}(\mathcal{P} \cap \mathbb{Z}^d)$?

As mentioned in section 1, in [5] Goemans and Rothvoss gave an algorithm for Point in Cone that runs in time $\mathsf{enc}(\mathcal{P})^{2^{\mathcal{O}(d)}} \cdot \mathsf{enc}(q)^{\mathcal{O}(1)}$. In fact, they solved the more general Cone and Polytope Intersection, where instead of a single point q, we are given a polytope $\mathcal{Q}$, and the question is whether $\mathsf{IntCone}(\mathcal{P} \cap \mathbb{Z}^d) \cap \mathcal{Q}$ is nonempty. In this case, the running time is $\mathsf{enc}(\mathcal{P})^{2^{\mathcal{O}(d)}} \cdot \mathsf{enc}(\mathcal{Q})^{\mathcal{O}(1)}$.

ETH. The Exponential-Time Hypothesis (ETH), proposed by Impagliazzo et al. [6], plays a fundamental role in providing conditional lower bounds for parameterized problems. It postulates that there exists a constant $c > 0$ such that the 3-SAT problem cannot be solved in time $\mathcal{O}(2^{cn})$, where n is the number of variables of the input formula. As proved in [6], this entails that there is no algorithm for 3-SAT with running time $2^{o(n+m)}$, where m denotes the number of clauses of the input formula; see also [1, Theorem 14.4]. We refer the reader to [1, Chapter 14] for a thorough introduction to ETH-based lower bounds within parameterized complexity.

Subset Sum. The classic Subset Sum problem asks, for a given set S of positive integers and a target integer t, whether there is a subset $S' \subseteq S$ such that $\sum_{x \in S'} = t$. The standard NP-hardness reduction from 3-SAT to Subset Sum takes an instance of 3-SAT with n variables and m clauses and outputs an equivalent instance of Subset Sum where $|S| = \mathcal{O}(n + m)$ and $t \leqslant 2^{\mathcal{O}(n+m)}$. By combining this with the $2^{o(n+m)}$-hardness for 3-SAT following from ETH, we obtain the following.

Theorem 2. *Unless the ETH fails, there is no algorithm solving Subset Sum in time $2^{o(n)}$, even under the assumption that $t \leqslant 2^{\mathcal{O}(n)}$. Here, n denotes the cardinality of the set S given on input.*

In this work, we rely on a variant of the Subset Sum problem called Subset Sum with Multiplicities. The difference between those two problems is that in the latter one, we allow the elements from the input set to be taken with any nonnegative multiplicities.

Subset Sum with Multiplicities **Input:** A set of positive integers $\{a_1, a_2, \ldots, a_n\}$ and a positive integer t. **Question:** Does there exist a sequence of n nonnegative integers $(\lambda_1, \lambda_2, \ldots, \lambda_n)$ such that $\sum_{i=1}^{n} \lambda_i \cdot a_i = t$?

The same lower bound as in Theorem 2 holds for Subset Sum with Multiplicities. This can be shown via a simple reduction from Subset Sum. As this is standard, we present the proof of the following Theorem 3 in Appendix A.

Theorem 3. *Unless the ETH fails, there is no algorithm solving Subset Sum with Multiplicities in time $2^{o(n)}$, even under the assumption that $t \leqslant 2^{\mathcal{O}(n)}$. Here, n denotes the cardinality of the set given on input.*

3 Reduction

The entirety of this section is devoted to the proof of our double-exponential hardness result: theorem 1.

The proof is by reduction from Subset Sum with Multiplicities. Let $\mathcal{I} = (\{a_1, \ldots, a_n\}, t)$ be the given instance of Subset Sum with Multiplicities. That is, we ask whether there are nonnegative integers $\lambda_1, \ldots, \lambda_n$ such that $\sum_{i=1}^{n} \lambda_i \cdot a_i = t$, where $a_1, \ldots, a_n, t$ are given positive integers. We may assume that $a_i \leqslant t$ for all $i \in [n]$ and, following on the hardness postulated by theorem 3, that $t \leqslant 2^{\mathcal{O}(n)}$.

Let $d \coloneqq \lceil \log_2(n+1) \rceil + 1$, hence d satisfies the inequality $2^d \geqslant 2n+2$ and $d = \mathcal{O}(\log n)$. Let $\chi_0, \chi_1, \ldots, \chi_{2^d-1} \in \mathbb{Z}^d$ be all $\{0,1\}$-vectors in d-dimensional space, listed in lexicographic order. Equivalently, χ_i is the bit encoding of the number i, for $i \in [2^d]_0$. Observe that we have $\chi_i + \chi_{2^d-1-i} = \mathbf{1}$ for every $i \in [2^d]_0$.

We define the set $P \coloneqq \{p_0, p_1, \ldots, p_{2^d-1}\} \subseteq \mathbb{Z}^{d+1}$ of 2^d points as follows.

$$p_i(j) = \begin{cases} \chi_i(j), & \text{for } i \in [2^d]_0 \text{ and } j \in [d]; \\ a_i, & \text{for } i \in [n] \text{ and } j = d+1; \\ 0 & \text{for } i \in [2^d]_0 - [n] \text{ and } j = d+1. \end{cases}$$

We remark that the construction of the point set P is inspired by the example of Goemans and Rothvoss provided in [5, Section 8]. First, we argue that P can be expressed as integer points in a polytope of small encoding size.

Claim 1. *There exists a bounded polytope $\mathcal{P}$ of encoding size $\mathcal{O}(n \log n \cdot \log t)$ such that $\mathcal{P} \cap \mathbb{Z}^{d+1} = P$.*

Proof of the claim. Let $\mathcal{P}$ be the polytope defined by the following inequalities.

$$0 \leqslant x(j) \leqslant 1 \quad \text{for } j \in [d], \tag{1}$$

$$0 \leqslant x(d+1) \leqslant t, \tag{2}$$

$$x(d+1) + \sum_{j\colon \chi_i(j)=0} t \cdot x(j) + \sum_{j\colon \chi_i(j)=1} t \cdot (1 - x(j)) \geqslant p_i(d+1) \quad \text{for } i \in [2^d]_0, \tag{3}$$

$$t - x(d+1) + \sum_{j\colon \chi_i(j)=0} t \cdot x(j) + \sum_{j\colon \chi_i(j)=1} t \cdot (1 - x(j)) \geqslant t - p_i(d+1) \text{ for } i \in [2^d]_0 \tag{4}$$

By (1) and (2), $\mathcal{P}$ is bounded. Also, encoding the system of all linear inequalities defining $\mathcal{P}$ takes

$$\mathcal{O}(2^d \cdot d \cdot \log t) = \mathcal{O}(n \log n \cdot \log t)$$

bits, as desired. It remains to show that $\mathcal{P} \cap \mathbb{Z}^{d+1} = P$. In what follows, when $i \in [2^d]_0$, (3.i) denotes the single inequality of the form (3) for this particular i, similarly for inequalities of the form (4).

First we show $\mathcal{P} \cap \mathbb{Z}^{d+1} \subseteq P$. Pick $x \in \mathcal{P} \cap \mathbb{Z}^{d+1}$. Since $x \in \mathbb{Z}^{d+1}$ and x satisfies (1), the first d coordinates of x form a binary encoding of a number $i^\star \in [2^d]_0$. Then, $x(j) = \chi_{i^\star}(j)$ for $j \in [d]$, hence by (3.$i^\star$), $x(d+1) \geqslant p_{i^\star}(d+1)$ and by (4.$i^\star$), $x(d+1) \leqslant p_{i^\star}(d+1)$. It follows that $x(d+1) = p_{i^\star}(d+1)$ and hence $x = p_{i^\star} \in P$, as required.

Finally we show $P \subseteq \mathcal{P} \cap \mathbb{Z}^{d+1}$. Pick $x = p_{i^\star} \in P$ for some $i^\star \in [2^d]_0$. We need to show that (1)–(4) hold for x. This is clear for (1) and (2). The inequality (3.$i^\star$) for x is just $x(d+1) \geqslant p_{i^\star}(d+1)$, and this holds since $x(d+1) = p_{i^\star}(d+1)$. We get (4.$i^\star$) analogously. Now assume $i \neq i^\star$ and let L_i be the left hand side of (3.i). Since $x(j) \in \{0,1\}$ for $j \in [d]$ and $x(d+1) \geqslant 0$, all the summands of L_i are nonnegative. Moreover, since $i \neq i^\star$, we have $x(j) \neq \chi_i(j)$ for some $j \in [d]$, and then $L_i \geqslant t \geqslant p_i(d+1)$, so the inequality (3.$i$) holds independently of the value of $x(d+1)$. Analogously, when L_i is the left hand side of (4.i), we get $L_i \geqslant 2t - x(d+1) \geqslant t \geqslant t - p_i(d+1)$, as required. $\lhd$

Let $\mathcal{P}$ be the polytope provided by Claim 1. Furthermore, let $q \in \mathbb{R}^{d+1}$ be the point defined as $q \coloneqq t \cdot \mathbf{1} = (t, t, \ldots, t)$. We consider the instance $\mathcal{I}' = (d+1, \mathcal{P}, q)$ of Point in Cone. Note that $d = \mathcal{O}(\log n)$ and $\mathsf{enc}(\mathcal{P}, q) = \mathcal{O}(n \log n \cdot \log t)$, which in turn is bounded by $\mathcal{O}(n^2 \log n)$ due to $t \leqslant 2^{\mathcal{O}(n)}$. Also, one can easily verify that $\mathcal{I}'$ can be computed from $\mathcal{I}$ in polynomial time. Now, we prove that the instance $\mathcal{I}'$ is equivalent to $\mathcal{I}$.

Claim 2. *$\mathcal{I}$ is a Yes-instance of Subset Sum with Multiplicities if and only if $\mathcal{I}'$ is a Yes-instance of Point in Cone.*

Proof of the claim. First, assume that $\mathcal{I}$ is a Yes-instance of Subset Sum with Multiplicities; that is, there are nonnegative integers $\lambda_1, \lambda_2, \ldots, \lambda_n$ such that $\sum_{i=1}^{n} \lambda_i \cdot a_i = t$. Our goal is to show that $q \in \mathsf{IntCone}(\mathcal{P} \cap \mathbb{Z}^{d+1}) = \mathsf{IntCone}(P)$. That is, we need to construct a sequence of nonnegative integers $(\lambda'_0, \lambda'_1, \ldots, \lambda'_{2^d-1})$ such that

$$\sum_{i=0}^{2^d-1} \lambda'_i \cdot p_i = q.$$

First, we set $\lambda'_i \coloneqq \lambda_i$, for $i \in [n]$. Then, we get the required value at the $(d+1)$-st coordinate, i.e.,

$$\left(\sum_{i=1}^{n} \lambda'_i \cdot p_i \right)(d+1) = t = q(d+1).$$

It remains to set the values of λ'_i for $i \in [2^d]_0 - [n]$. Note that $p_i(d+1) = 0$ for $i \in [2^d]_0 - [n]$, therefore setting those λ'_i does not affect the $(d+1)$-st coordinate of the result.

Consider an index $i \in [n]$. Recall that $\chi_i + \chi_{2^d-i-1} = \mathbf{1}$, and since $2^d \geqslant 2n+2$, we have $2^d - i - 1 \geqslant 2^d - n - 1 \geqslant n+1$. Hence, by setting $\lambda'_{2^d-i-1} \coloneqq \lambda'_i = \lambda_i$, we obtain that

$$\lambda'_i \cdot p_i + \lambda'_{2^d-i-1} \cdot p_{2^d-i-1} = (\lambda_i, \lambda_i, \ldots, \lambda_i, \lambda_i a_i).$$

By applying this procedure for every $i \in [n]$, we get a point $q' \in \mathbb{Z}^{d+1}$ of the form $(\Lambda, \Lambda, \ldots, \Lambda, t)$, where

$$\Lambda = \sum_{i=1}^{n} \lambda_i \leqslant \sum_{i=1}^{n} \lambda_i \cdot a_i = t.$$

To obtain the number t on the first d coordinates of the result, it remains to observe that $p_{2^d-1} = (1, 1, \ldots, 1, 0)$, therefore setting $\lambda'_{2^d-1} \coloneqq t - \sum_{i=1}^{n} \lambda_i$ produces the desired point q. (We set $\lambda'_i \coloneqq 0$ for all i not considered in the described procedure.)

For the other direction, suppose that $\mathcal{I}'$ is a Yes-instance of Point in Cone, that is, $q \in \mathsf{IntCone}(P)$. Then, there exist nonnegative integers λ_i (for $i \in [2^d]_0$) such that

$$\sum_{i=0}^{2^d-1} \lambda_i \cdot p_i = q.$$

Comparing the $(d+1)$-st coordinate of both sides yields the equality $\sum_{i=1}^{n} \lambda_i \cdot a_i = t$. This means that $\mathcal{I}$ is indeed a Yes-instance of Subset Sum with Multiplicities. $\lhd$

Finally, we are ready to prove Theorem 1. Suppose for contradiction that Point in Cone admits an algorithm with running time $\mathsf{enc}(\mathcal{P}, q)^{2^{o(d)}}$. As argued, given an instance $\mathcal{I}$ of Subset Sum with Multiplicities with n integers and the target integer t bounded by $2^{\mathcal{O}(n)}$, one can in polynomial time compute an equivalent instance $\mathcal{I}' = (d, \mathcal{P}, q)$ of Point in Cone with $d \leqslant \mathcal{O}(\log n)$ and $\mathsf{enc}(\mathcal{P}, q) \leqslant \mathcal{O}(n^2 \log n)$. Now, running our hypothetical algorithm on $\mathcal{I}'$ yields an algorithm for Subset Sum with Multiplicities with running time

$$\mathsf{enc}(\mathcal{P}, q)^{2^{o(d)}} = (n^2 \log n)^{2^{o(d)}} = (n^2 \log n)^{n^{o(1)}} \leqslant 2^{n^{o(1)} \cdot 3 \log n} \leqslant 2^{o(n)},$$

which contradicts theorem 3. This concludes the proof of theorem 1.

References

[1] Marek Cygan, Fedor V. Fomin, Łukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michał Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015. doi:10.1007/978-3-319-21275-3.

[2] Marek Cygan, Marcin Pilipczuk, and Michał Pilipczuk. Known algorithms for Edge Clique Cover are probably optimal. *SIAM J. Comput.*, 45(1):67–83, 2016. doi:10.1137/130947076.

[3] Friedrich Eisenbrand and Gennady Shmonin. Carathéodory bounds for integer cones. *Oper. Res. Lett.*, 34(5):564–568, 2006. doi:10.1016/j.orl.2005.09.008.

[4] Fedor V. Fomin, Petr A. Golovach, Daniel Lokshtanov, Saket Saurabh, and Meirav Zehavi. Clique-width III: Hamiltonian Cycle and the odd case of Graph Coloring. *ACM Trans. Algorithms*, 15(1):9:1–9:27, 2019. doi:10.1145/3280824.

[5] Michel X. Goemans and Thomas Rothvoss. Polynomiality for bin packing with a constant number of item types. *J. ACM*, 67(6):38:1–38:21, 2020. doi:10.1145/3421750.

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

[7] Klaus Jansen and Kim-Manuel Klein. About the structure of the integer cone and its application to bin packing. *Math. Oper. Res.*, 45(4):1498–1511, 2020. doi:10.1287/moor.2019.1040.

[8] Klaus Jansen, Kim-Manuel Klein, and Alexandra Lassota. The double exponential runtime is tight for 2-stage stochastic ILPs. *Math. Program.*, 197(2):1145–1172, 2023. doi:10.1007/s10107-022-01837-0.

[9] Dušan Knop, Michał Pilipczuk, and Marcin Wrochna. Tight complexity lower bounds for Integer Linear Programming with few constraints. *ACM Trans. Comput. Theory*, 12(3):19:1–19:19, 2020. doi:10.1145/3397484.

[10] Marvin Künnemann, Filip Mazowiecki, Lia Schütze, Henry Sinclair-Banks, and Karol Węgrzycki. Coverability in VASS revisited: Improving Rackoff's bound to obtain conditional optimality. *CoRR*, abs/2305.01581, 2023. `arXiv:2305.01581`, `doi:10.48550/arXiv.2305.01581`.

[11] Dániel Marx and Valia Mitsou. Double-exponential and triple-exponential bounds for choosability problems parameterized by treewidth. In *ICALP*, volume 55 of *LIPIcs*, pages 28:1–28:15. Schloss Dagstuhl — Leibniz-Zentrum für Informatik, 2016. `doi:10.4230/LIPIcs.ICALP.2016.28`.

[12] Matthias Mnich and René van Bevern. Parameterized complexity of machine scheduling: 15 open problems. *Comput. Oper. Res.*, 100:254–261, 2018. `doi:10.1016/j.cor.2018.07.020`.

A Subset Sum with Multiplicities

In this appendix we give a proof of theorem 3, which we recall here for convenience.

Theorem 3. *Unless the ETH fails, there is no algorithm solving* Subset Sum with Multiplicities *in time* $2^{o(n)}$*, even under the assumption that* $t \leqslant 2^{\mathcal{O}(n)}$*. Here,* n *denotes the cardinality of the set given on input.*

Proof of Theorem 3. We provide a reduction from Subset Sum to Subset Sum with Multiplicities. Let $\mathcal{I} = (\{a_1, a_2, \ldots, a_n\}, t)$ be the input instance of Subset Sum. We may assume that $a_i \leqslant t$ for all $i \in [n]$. We construct an equivalent instance $\mathcal{I}' = (\{a'_1, \ldots, a'_n, b_1, \ldots, b_n\}, t')$ of Subset Sum with Multiplicities as follows. The bit encodings of integers a'_i, b_i and t' are partitioned into three blocks B_1, B_2, B_3, where B_3 contains the least significant bits, and B_1 the most significant ones. For an integer x and a block B, we denote by $x|_B$ the integer of bit-length at most $|B|$ consisting of the bits of x at the positions within the block B. The instance $\mathcal{I}'$ is defined by the following conditions.

- Blocks B_1 and B_3 are of length n, while block B_2 is of length $\lceil \log t \rceil$.
- For $i \in [n]$,

$$a'_i|_{B_j} = \begin{cases} 2^{n-i} & \text{for } j = 1, \\ a_i & \text{for } j = 2, \\ 2^{i-1} & \text{for } j = 3; \end{cases} \qquad \text{and} \qquad b_i|_{B_j} = \begin{cases} 2^{n-i} & \text{for } j = 1, \\ 0 & \text{for } j = 2, \\ 2^{i-1} & \text{for } j = 3. \end{cases}$$

- The target integer t' is given by

$$t'|_{B_j} = \begin{cases} 2^n - 1 & \text{for } j = 1, \\ t & \text{for } j = 2, \\ 2^n - 1 & \text{for } j = 3. \end{cases}$$

Note that the instance $\mathcal{I}'$ consists of a set of $n' := 2n$ positive integers and a target integer $t' \leqslant 2^{\mathcal{O}(n)} \cdot t$. In particular, if $t \leqslant 2^{\mathcal{O}(n)}$ then also $t' \leqslant 2^{\mathcal{O}(n)}$. Clearly, $\mathcal{I}'$ can be computed from $\mathcal{I}$ in polynomial time. Next, we prove that $\mathcal{I}'$ is indeed an instance equivalent to $\mathcal{I}$.

Claim 3. *$\mathcal{I}$ is a* Yes*-instance of* Subset Sum *if and only if $\mathcal{I}'$ is a* Yes*-instance of* Subset Sum with Multiplicities*.*

Proof of the claim. ($\Longrightarrow$). Assume $\mathcal{I}$ is a Yes-instance of Subset Sum. Let $J \subseteq [n]$ be a set of indices such that $\sum_{j \in J} a_j = t$. We construct a sequence $\lambda_1, \lambda_2, \ldots, \lambda_{2n}$ of $2n$ nonnegative integers as follows. For $i \in [n]$, we set

$$\lambda_i = \begin{cases} 1 & \text{if } i \in J, \\ 0 & \text{if } i \notin J; \end{cases} \qquad \text{and} \qquad \lambda_{n+i} = \begin{cases} 0 & \text{if } i \in J, \\ 1 & \text{if } i \notin J. \end{cases}$$

Then it is easy to verify that

$$\sum_{i=1}^{n} \lambda_i \cdot a'_i + \sum_{i=1}^{n} \lambda_{n+i} \cdot b_i = t',$$

and thus the sequence $\lambda_1, \lambda_2, \ldots, \lambda_{2n}$ witnesses that $\mathcal{I}'$ is a YES-instance of SUBSET SUM WITH MULTIPLICITIES.

($\Longleftarrow$). Assume that $\mathcal{I}'$ is a YES-instance of SUBSET SUM WITH MULTIPLICITIES. Let $\lambda_1, \lambda_2, \ldots, \lambda_{2n}$ be nonnegative integers such that

$$\sum_{i=1}^{n} \lambda_i \cdot a'_i + \sum_{i=1}^{n} \lambda_{n+i} \cdot b_i = t'.$$

Let L be the left-hand side of the equation above. Comparing the least significant bit of L and t' yields $\lambda_1 + \lambda_{n+1} \equiv 1 \pmod 2$. However, if $\lambda_1 + \lambda_{n+1} \geqslant 2$, then

$$L|_{B_1} \geqslant 2 \cdot 2^{n-1} = 2^n > t'|_{B_1},$$

and consequently $L > t$, which is a contradiction. Therefore, $\lambda_1 + \lambda_{n+1} = 1$. Repeating this argument inductively for $i = 2, 3, \ldots, n$ leads us to the conclusion that the equality

$$\lambda_i + \lambda_{n+i} = 1 \tag{5}$$

holds for every $i \in [n]$. Now, define a set of indices $J \subseteq [n]$ as $J := \{i \in [n] \mid \lambda_i = 1\}$. Then, by comparing $L|_{B_2}$ and $t|_{B_2}$, we must have that

$$\sum_{j \in J} a_j = t,$$

since other terms of L do not contribute to $L|_{B_2}$ according to the equation (5). Hence $\mathcal{I}$ is a YES-instance of SUBSET SUM, as desired. $\lhd$

We are ready to conclude the proof of theorem 3. Suppose for contradiction there is an algorithm solving SUBSET SUM WITH MULTIPLICITIES in time $2^{o(n')}$ on instances with n' numbers on input and the target integer t' bounded by $2^{\mathcal{O}(n')}$. Then, as explained above, given an instance $\mathcal{I}$ of SUBSET SUM with n numbers and the target integer t bounded by $2^{\mathcal{O}(n)}$, one can in polynomial time compute an equivalent instance $\mathcal{I}'$ of SUBSET SUM WITH MULTIPLICITIES with $n' = 2n$ numbers and with target $t' \leqslant 2^{\mathcal{O}(n)} \cdot t \leqslant 2^{\mathcal{O}(n)} = 2^{\mathcal{O}(n')}$. Running the hypothetical algorithm on $\mathcal{I}'$ solves the initial instance $\mathcal{I}$ of SUBSET SUM in time

$$2^{o(n')} = 2^{o(n)},$$

which contradicts theorem 2. This finishes the proof of theorem 3. $\square$