



HAL
open science

The Maximum Zero-Sum Partition problem

Guillaume Fertin, Oscar Fontaine, Géraldine Jean, Stéphane Vialette

► **To cite this version:**

Guillaume Fertin, Oscar Fontaine, Géraldine Jean, Stéphane Vialette. The Maximum Zero-Sum Partition problem. Theoretical Computer Science, 2024, 1019, pp.114811. 10.1016/j.tcs.2024.114811 . hal-04901397

HAL Id: hal-04901397

<https://nantes-universite.hal.science/hal-04901397v1>

Submitted on 20 Jan 2025

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License



Contents lists available at ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcsThe MAXIMUM ZERO-SUM PARTITION problem ^{☆,☆☆}Guillaume Fertin ^{a,*}, Oscar Fontaine ^b, Géraldine Jean ^a, Stéphane Vialette ^c^a Nantes Université, École Centrale Nantes, CNRS, LS2N, UMR 6004, Nantes, 44000, France^b Ecole Normale Supérieure PSL, DMA, Paris, 75230, France^c LIGM, CNRS, Université Gustave Eiffel, Marne-la-Vallée, 77454, France

ARTICLE INFO

Keywords:

Zero-sum

Partition

Algorithmic complexity

Inapproximability

FPT

ABSTRACT

We study the MAXIMUM ZERO-SUM PARTITION problem (or MZSP), defined as follows: given a multiset $S = \{a_1, a_2, \dots, a_n\}$ of integers $a_i \in \mathbb{Z}^*$ (where \mathbb{Z}^* denotes the set of non-zero integers) such that $\sum_{i=1}^n a_i = 0$, find a maximum cardinality partition $\{S_1, S_2, \dots, S_k\}$ of S such that, for every $1 \leq i \leq k$, $\sum_{a_j \in S_i} a_j = 0$. Solving MZSP is useful in genomics for computing evolutionary distances between pairs of species. Our contributions are a series of algorithmic results concerning MZSP, in terms of complexity, (in)approximability, with a particular focus on the fixed-parameter tractability of MZSP with respect to either (i) the size k of the solution, (ii) the number of negative (resp. positive) values in S and (iii) the largest integer in S .

1. Introduction

In this paper, we study the MAXIMUM ZERO-SUM PARTITION problem (or MZSP), defined as follows.

MAXIMUM ZERO-SUM PARTITION (MZSP)

Instance: A multiset $S = \{a_1, a_2, \dots, a_n\}$ of numbers $a_i \in \mathbb{Z}^*$ s.t. $\sum_{i=1}^n a_i = 0$.

Output: A maximum cardinality partition $S = \{S_1, S_2, \dots, S_k\}$ of S such that, for every $1 \leq i \leq k$, $\sum_{a_j \in S_i} a_j = 0$.

This problem emerged in the context of bioinformatics [2], more precisely towards understanding large scale evolutionary events that occur in species, which we call *genome rearrangements*. In this setting, we are given two genomes g_1 and g_2 , each one representing a given species. Each genome is modeled as an ordered sequence of genes, and we give ourselves a set S of operations that allow us to modify a genome. The goal consists in finding the minimum number of operations from S that are needed to obtain g_2 , starting from g_1 . This number gives an estimate of the evolutionary distance between the two studied species; for instance, computing pairwise distances between genomes is useful to reconstruct phylogenetic trees. Genome rearrangements gave rise to many algorithmic studies and results (depending notably on the way a genome is modeled, and on the contents of set S), and we refer the interested reader to [3] for a survey.

[☆] This article belongs to Section A: Algorithms, automata, complexity and games, Edited by Paul Spirakis.

^{☆☆} A preliminary version of this paper has been published in ICS 2022 [1].

* Corresponding author.

E-mail addresses: guillaume.fertin@univ-nantes.fr (G. Fertin), oscar.fontaine@ens.psl.eu (O. Fontaine), geraldine.jean@univ-nantes.fr (G. Jean), stephane.vialette@univ-eiffel.fr (S. Vialette).

<https://doi.org/10.1016/j.tcs.2024.114811>

Received 29 August 2023; Received in revised form 21 August 2024; Accepted 27 August 2024

Available online 30 August 2024

0304-3975/© 2024 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

Table 1
Summary of our main results, in relation to parameters n, m, k, n^*, B and b .

Parameter	Results
n	Strongly NP-complete (Theorem 1) No $2^{o(n)}b^{O(1)}$ algorithm unless ETH fails (Theorem 4) FPT (Theorem 5) No approximation within ratio $O(n^{1-\epsilon})$ (Theorem 7)
m	NP-complete, even if bounded (Theorem 2)
k	NP-complete, even if bounded (Theorem 2)
m Unary encoded instance	W[1]-hard (Theorem 8) XP(Corollary 2)
k Unary encoded instance	W[1]-hard (Corollary 1) XP(Theorem 9)
n^*	No $2^{o(n^*)}b^{O(1)}$ algorithm unless ETH fails (Theorem 10) XP(Theorem 11)
B	FPT(Theorem 13)
$n^* + k$	FPT(Theorem 12)

In [2], set S is limited to a generic rearrangement operation called “weighted Double Cut and Join” (or $wDCJ$), and the minimum number of $wDCJ$ s needed to go from g_1 to g_2 is denoted $wDCJ(g_1, g_2)$. The authors derived a closed formula for $wDCJ(g_1, g_2)$, which is a sum of four terms: $wDCJ(g_1, g_2) = n - c + n_u - p$. The three first terms are polynomial-time computable from g_1 and g_2 . The relationship to the MZSP problem relies on the fourth parameter, p , which turns out to exactly correspond to the optimal solution to MZSP (where the input to MZSP is inferred, in polynomial time, from genomes g_1 and g_2). Thus, the complexity of computing $wDCJ$ directly correlates to the complexity of MZSP.

In [2], the authors mostly focused on approximating $wDCJ(g_1, g_2)$. Moreover, it turns out that MZSP has never been studied under a purely algorithmic viewpoint. Thus, in this paper, we are mainly interested in parameterized complexity, essentially because, as said above, approximations have already been addressed in [2], and also because such paradigm is a way to deal with NP-hard problems, while ensuring optimality (see e.g. [4]). In this context, the goal is to identify several candidate parameters and to study whether our problem is fixed-parameterized tractable (FPT) with respect to each of these parameters. The rationale being that if the problem is FPT for a given parameter that appears to be small in practice, then, although exponential, the corresponding algorithm can be used to determine the exact solution to the problem at hand, as it is expected to run in reasonable time.

Definitions and notations For any integer n , $\llbracket 1, n \rrbracket$ denotes the set of integers from 1 to n . Given a (multi)set S of integers and an integer p , we say that S sums to p when the sum of the elements of S is equal to p . When $p = 0$, we say that S is a zero-sum (multi)set. For any instance S of MZSP, we let neg (resp. pos) denote the number of negative (resp. positive) integers in S and $m = \min\{\text{neg}, \text{pos}\}$. We denote by n^* the number of distinct values in S , by b the number of bits needed to encode S (note that $b = \Theta(\sum_{i=1}^n \lceil \log_2(|a_i|) \rceil)$), and by $B = \max_{i \in \llbracket 1, n \rrbracket} \{|a_i|\}$. The cardinality of an optimal partition of S , i.e. the size of the solution, is denoted by k . For example, if $S = \{-7, -7, -7, -1, -1, -1, 2, 3, 3, 4, 4, 4, 4\}$, then $n = 13$, $\text{neg} = 6$, $\text{pos} = 7$, $m = 6$, $n^* = 5$, $B = 7$ and it can be seen that the optimal solution is $k = 4$: for instance, $S_1 = S_2 = \{-7, 4, 3\}$, $S_3 = \{-7, -1, 4, 4\}$ and $S_4 = \{-1, -1, 2\}$ form a solution. UNARY MZSP denotes MZSP for which unary encoding of the input instance is used. For any positive integer p , p -MZSP denotes the decision version of MZSP, in which, given p and a zero-sum integer multiset S , we ask whether there exists a zero-sum partition S' of S such that $|S'| \geq p$. We will also often use the O^* notation, as frequently done in parameterized complexity (see e.g. [4]): for a given problem whose size of the input is n and parameter is k , $O^*(f(k))$ stands for $O(f(k) \cdot \text{poly}(n))$. In other words, O^* only describes the exponential part of the running time (in k) and discards the polynomial factor (in n).

First observations Note that if we denote by $-S$ the multiset S to which all signs have been switched, then $-S$ is a valid instance for MZSP, and both S and $-S$ have the same optimum k . Consequently, any result that applies to neg (resp. pos) applies to pos (resp. neg), and thus to m . Note also that an m -size zero-sum partition of S is necessarily optimal, since at least one positive (resp. negative) element of S needs to be present in any S_i from the partition. In other words, we always have $k \leq m$. For any given $p \in \mathbb{N}^*$, a YES-instance for p -MZSP is also a YES-instance for p' -MZSP as long as $p' \in \llbracket 1, p \rrbracket$. Indeed, merging any two sets in a size- p zero-sum partition of S yields a size- $(p - 1)$ zero-sum partition of S . Finally, observe that if an integer a and its opposite $-a$ both belong to S , then there always exists an optimal solution $S = \{S_1, S_2, \dots, S_k\}$ in which $S_i = \{-a, a\}$ for some $i \in \llbracket 1, n \rrbracket$. Indeed, suppose $-a$ and a both belong to S , and observe an optimal solution $S = \{S_1, S_2, \dots, S_k\}$ in which $S_i \neq \{-a, a\}$ for every $i \in \llbracket 1, k \rrbracket$. Clearly, no S_j is such that $\{-a, a\} \subset S_j$, otherwise we could partition S_j into $\{-a, a\}$ and $S_j \setminus \{-a, a\}$, both summing to zero, contradicting the optimality of S . Thus $-a \in S_x$ and $a \in S_y$ for some $1 \leq x \neq y \leq k$. Now consider the following partition $S' = \{S'_1, S'_2, \dots, S'_k\}$ of S : (i) $S'_i = S_i$ for every $i \in \llbracket 1, k \rrbracket$ such that $i \neq x$ and $i \neq y$, (ii) $S'_x = \{-a, a\}$ and (iii) $S'_y = (S_x \cup S_y) \setminus \{-a, a\}$. Every S'_i , $i \in \llbracket 1, k \rrbracket$, sums to zero, and $|S'| = |S| = k$.

In this paper, we study the MZSP problem under an algorithmic viewpoint, and, in particular, discuss its computational complexity, approximability and fixed-parameter tractability with respect to n, n^*, m, B and k (see Table 1).

2. Computational complexity of MZSP

Theorem 1. *MZSP is strongly NP-complete, even if each S_i in the solution S contains at most four elements.*

Proof. The proof is by reduction from 3-PARTITION, which has been proved to be strongly NP-complete [5], and whose definition is as follows.

3-PARTITION

Instance: An integer C , a multiset $X = \{x_1, x_2, \dots, x_{3p}\}$ of integers such that (i) $\sum_{i=1}^{3p} x_i = C \cdot p$ and (ii) $\forall x_i \in X$, $\frac{C}{4} < x_i < \frac{C}{2}$.

Question: Does there exist a partition $\{X_1, X_2, \dots, X_p\}$ of X such that, for every $i \in \llbracket 1, p \rrbracket$, $\sum_{x_j \in X_i} x_j = C$?

Let $I = (C, X)$ be an instance of 3-PARTITION, and let S be the multiset such that $S = \{x_1, \dots, x_{3p}, -C, \dots, -C\}$, where $-C$ appears p times in S . Note that, by definition of 3-PARTITION, the sum of all elements in S is equal to zero, hence S is an instance of MZSP. We now show that $I = (C, X)$ is a YES-instance for 3-PARTITION iff MZSP (with instance S) has a solution of cardinality p .

(\Rightarrow) If I is a YES-instance for 3-PARTITION, there exists $t_j = (x_{i_{j,1}}, x_{i_{j,2}}, x_{i_{j,3}})$, $j \in \llbracket 1, p \rrbracket$, such that $x_{i_{j,1}} + x_{i_{j,2}} + x_{i_{j,3}} = C$. In particular, for every $j \in \llbracket 1, p \rrbracket$, $S_j = \{x_{i_{j,1}}, x_{i_{j,2}}, x_{i_{j,3}}, -C\}$ is a size- p partition of S in which every S_j sums to zero. Moreover, such partition is optimal: since $\text{neg} = p$, no zero-sum partition of S can contain strictly more than p sets.

(\Leftarrow) Suppose there exists a solution of MZSP of cardinality p , say $S = \{S_1, S_2, \dots, S_p\}$. Since any zero-sum subset in S contains at least one negative element from S , every S_j , $j \in \llbracket 1, p \rrbracket$, contains exactly one negative element, namely $-C$. Since, in 3-PARTITION, every x_i satisfies $\frac{C}{4} < x_i < \frac{C}{2}$, exactly 3 such elements are required to sum to C . Thus, any S_j , $j \in \llbracket 1, p \rrbracket$, contains 3 elements of the form x_i , together with $-C$. Since each S_j sums to zero, $\{x_1, \dots, x_{3p}\}$ can be partitioned in triplets, each summing to C , i.e. $I = (C, X)$ is a YES-instance for 3-PARTITION.

In unary, 3-PARTITION and MZSP are both encoded in $\Theta(p \cdot C)$ space. As 3-PARTITION is strongly NP-complete, MZSP is also strongly NP-complete. \square

Theorem 1 proves that solving MZSP when every S_i contains 4 elements is (strongly) NP-complete. Note that, consequently, this rules out parameter ‘‘maximum size of an S_i ’’ for FPT considerations.

Theorem 2. *MZSP is NP-complete, even when k and m are bounded.*

Proof. We show NP-completeness of MZSP in the specific case $m = k = 2$, by reduction from PARTITION which is known to be NP-complete [6].

PARTITION

Instance: A multiset $X = \{x_1, x_2, \dots, x_n\}$ of integers from \mathbb{N}^* .

Question: Does there exist a partition $\{X_1, X_2\}$ of X s.t. $\sum_{x_i \in X_1} x_i = \sum_{x_j \in X_2} x_j$?

Let X be an instance of PARTITION. We can always assume $\sum_{i=1}^n x_i$ to be even, otherwise we have a NO-instance. Thus assume $\sum_{i=1}^n x_i = 2N$. The MZSP instance we build from X is $S = X \cup \{-N, -N\}$. We show that X is a YES-instance for PARTITION iff MZSP yields a size-2 zero-sum partition for S .

(\Rightarrow) Suppose there exists a partition $\{X_1, X_2\}$ of X such that $\sum_{x_i \in X_1} x_i = \sum_{x_j \in X_2} x_j$. We thus have $\sum_{x_i \in X_1} x_i = \sum_{x_j \in X_2} x_j = N$, and $\{X_1 \cup \{-N\}, X_2 \cup \{-N\}\}$ is a zero-sum partition of S . Moreover, this partition is optimal since $m = \text{neg} = 2$ and $k \leq m$.

(\Leftarrow) Suppose there exists a zero-sum partition of cardinality $k = 2$ of S , say $S = \{S_1, S_2\}$. Because $\text{neg} = 2$, we know that S_1 (resp. S_2) contains exactly one negative integer; thus, both in S_1 and S_2 , this integer is $-N$. Assume $S_1 = X_1 \cup \{-N\}$ and $S_2 = X_2 \cup \{-N\}$. In that case, $\{X_1, X_2\}$ is a partition of X , and because both S_1 and S_2 sum to zero, we have $\sum_{x_i \in X_1} x_i = \sum_{x_j \in X_2} x_j = N$. Thus X is a YES-instance for PARTITION. \square

Although we just showed that MZSP is strongly NP-complete in general, and remains NP-complete when k is bounded, we can show there exists a pseudo-polynomial algorithm that solves MZSP in the case $k = 2$.

Theorem 3. *2-MZSP can be solved in pseudopolynomial time.*

Proof. Let $S = P \cup N$, where $P = \{p_1, p_2, \dots, p_{\text{pos}}\}$ (with $p_1 \leq p_2 \leq p_3 \leq \dots \leq p_{\text{pos}}$) is the set of positive numbers, and $N = \{n_1, n_2, \dots, n_{\text{neg}}\}$ is the set of negative numbers, and recall that $\text{neg} + \text{pos} = n$. The proof is by dynamic programming. More precisely, we start by filling two tables, T_{pos} and T_{neg} . We describe in more detail the contents of T_{pos} ; filling T_{neg} is achieved in a similar fashion. T_{pos} is a Boolean table with $D = \sum_{i=1}^{\text{pos}} p_i$ lines and pos columns. For any $i \in \llbracket 1, \text{pos} \rrbracket$ and any $j \in \llbracket 1, D \rrbracket$, $T_{\text{pos}}[i, j]$ is equal to `True` if there exists a subset of $\{p_1, p_2, \dots, p_i\}$ that sums to j , and `False` otherwise. T_{pos} is initialized as follows:

- for any $j \in \llbracket 1, D \rrbracket$, $T_{pos}[1, j] = \text{True}$ if $j = p_1$, and False otherwise;
- for any $i \in \llbracket 1, \text{pos} \rrbracket$, $T_{pos}[i, 1] = \text{True}$ if there exists $i' \leq i$ such that $p_{i'} = 1$, and False otherwise.

We then fill the rest of table T_{pos} using the following dynamic programming formula: for any $i \in \llbracket 2, \text{pos} \rrbracket$ and $j \in \llbracket 2, D \rrbracket$:

- $T_{pos}[i, j] = \text{True}$ if (a) $T_{pos}[i-1, j-p_i] = \text{True}$ or (b) $T_{pos}[i-1, j] = \text{True}$
- $T_{pos}[i, j] = \text{False}$ otherwise

Case (a) above corresponds to the case where p_i is used to obtain a sum equal to j , while (b) corresponds to the case where p_i is not present in the sum. Clearly, T_{pos} is filled in time $O(\text{pos} \cdot D)$. Similarly, table T_{neg} is a Boolean table with neg lines and D columns (recall that, by definition, $\sum_{i=1}^{\text{neg}} n_i = -D$), and $T_{neg}[i, j]$ is set to True if there exists a subset of $\{n_1, n_2, \dots, n_i\}$ that exactly sums to $-j$, and to False otherwise. T_{neg} is filled in a similar fashion as T_{pos} , and the time complexity to fill it is $O(\text{neg} \cdot D)$. The above described algorithm thus takes $O(nD)$ time and space. We now claim that S is a YES-instance for 2-MZSP iff there exists a $j < D$ such that $T_{pos}[\text{pos}, j] = T_{neg}[\text{neg}, j] = \text{True}$.

(\Rightarrow) Suppose S is a YES-instance for 2-MZSP. Then S can be partitioned in two zero-sums multisets S_1 and S_2 . Let $D_1 > 0$ be the sum of positive values in S_1 , and note that $D_1 < D$ by hypothesis. Since S_1 sums to zero, $-D_1$ is the sum of the negative values in S_1 . Thus, we necessarily have $T_{pos}[\text{pos}, D_1] = T_{neg}[\text{neg}, D_1] = \text{True}$.

(\Leftarrow) Assume $T_{pos}[\text{pos}, X] = T_{neg}[\text{neg}, X] = \text{True}$ for some value $X < D$. By definition of T_{pos} (resp. T_{neg}), we can identify a subset S_1^+ (resp. S_1^-) of positive (resp. negative) values of S , whose sum is X (resp. $-X$). Thus $S_1 = S_1^+ \cup S_1^-$ is a zero-sum subset of S . Since $X < D$, we know that $S_1 \subset S$. Thus $S_2 = S - S_1$ is non-empty, and is also a zero-sum subset of S , as both S and S_1 sum to zero. We thus conclude that $\{S_1, S_2\}$ is a zero-sum partition of S . \square

The following result gives two lower bounds on the time to solve MZSP, both based on the Exponential-Time Hypothesis (ETH, see e.g. [7] for a definition). Recall that b is the size of the input S , assuming it is binary encoded.

Theorem 4. *Unless ETH fails, MZSP cannot be solved (i) in $2^{o(n)} \cdot b^{O(1)}$ or (ii) in $2^{o(\sqrt{b})}$.*

Proof. We reduce from the SUBSET-SUM problem, which is defined as follows.

SUBSET-SUM

Instance: A multiset $X = \{x_1, x_2, \dots, x_n\}$ of integers from \mathbb{N}^* , an integer C .

Question: Does there exist a subset $P \subseteq X$ such that $\sum_{x_i \in P} x_i = C$?

It is known from [8] that, unless ETH fails, it is not possible to solve SUBSET-SUM in time $2^{o(n)}\beta^{O(1)}$, or in time $2^{o(\sqrt{\beta})}$ where β is the number of bits that are necessary to binary encode an instance $I = (X, C)$ of SUBSET-SUM. Our reduction is as follows: let $I = (X, C)$ be an instance of SUBSET-SUM and $N = \sum_{i=1}^n x_i$.

We define our MZSP instance S to be the multiset $S = X \cup \{-C, -N + C\}$. We now show that S has a zero-sum partition of cardinality 2 iff I is a YES-instance for SUBSET-SUM. Note that the optimal value k for MZSP of S cannot exceed 2, as $m = 2$.

(\Rightarrow) Suppose there exists a zero-sum partition $\{S_1, S_2\}$ of S . Necessarily, S_1 (resp. S_2) contains exactly one negative integer. Assume $wlog -C \in S_1$. Thus $S_1 = P \cup \{-C\}$, where P is a subset of X . Since S_1 sums to zero, we conclude that P sums to C . Hence $I = (X, C)$ is a YES-instance for SUBSET-SUM.

(\Leftarrow) Suppose $I = (X, C)$ is a YES instance for SUBSET-SUM. Thus there exists $P \subseteq X$ such that P sums to C . Thus $S_1 = \{P \cup \{-C\}\}$ is a zero-sum multiset from S , and consequently $S_2 = \{S \setminus P \cup \{-N + C\}\}$ is also a zero-sum multiset from S . Since S_1 and S_2 partition S , we conclude that $k = 2$ is optimal, since $m = 2$ and $k \leq m$.

The set X in the instance $I = (X, C)$ of SUBSET-SUM contains n integers, while S , our instance of MZSP, contains $n + 2$ integers. Hence ETH, based on n , also holds for MZSP. The number β of bits to encode $I = (X, C)$ is $\beta = (\sum_{i=1}^n \lceil \log_2 x_i \rceil) + \lceil \log_2 C \rceil$, while S is encoded using $b = \beta + \lceil \log_2 (N - C) \rceil$ bits. Since $N - C$ does not exceed $\sum_{i=1}^n x_i$, we have $b = \Theta(\beta)$, which proves the result. \square

We now show that the above ETH bound based on n is essentially tight.

Theorem 5. *MZSP is solvable in $O^*(2^n)$.*

Proof. We solve MZSP by dynamic programming. Given an instance S of MZSP, we create a dynamic programming 1-dimensional table T indexed by the subsets of S . We set $T[\emptyset]$ to 0. Then, for increasing $i \in \llbracket 1, n \rrbracket$, and for every size- i subset P_i of S , we fill $T[P_i]$ using the following rule:

$$T[P_i] = \begin{cases} \max_{a \in P_i} \{T[P_i \setminus \{a\}]\} & \text{if } P_i \text{ does not sum to } 0 \\ \max_{a \in P_i} \{T[P_i \setminus \{a\}]\} + 1 & \text{otherwise} \end{cases}$$

The optimal value k for MZSP is then found in $T[S]$, and an optimal zero-sum partition of S can be found, from that value, by traceback.

The space and time complexity of the above algorithm is $O^*(2^n)$, since it takes polynomial time to fill any of the 2^n elements in T .

It remains to show correctness. For this, for any subset P of S , we denote by k_P the cardinality of a maximum zero-sum subpartition of P , where the term *subpartition* describes a partition of a subset of P . Our goal is to show that for any P , $T[P] = k_P$. This is done by induction on $i = |P|$. When $i = 0$, this trivially holds as $T[\emptyset]$ is set to 0. Suppose now that for some $i \in \llbracket 0, n-1 \rrbracket$, any P such that $|P| = i$ satisfies $T[P] = k_P$. Let us now observe a set P of cardinality $i + 1$. If P does not sum to zero, let us consider a maximum cardinality zero-sum subpartition of P , say (A_1, \dots, A_{k_P}) . Since P does not sum to zero, there exists $a \in P$ such that $a \notin \bigcup_{i=1}^{k_P} A_i$. Thus (A_1, \dots, A_{k_P}) is a maximum cardinality zero-sum subpartition of $P \setminus \{a\}$, otherwise this would contradict the cardinality maximality of (A_1, \dots, A_{k_P}) . Thus $k_{P \setminus \{a\}} = k_P$. Since $T[P \setminus \{a\}] = k_{P \setminus \{a\}}$, we have $T[P] \geq k_P$ by definition of $T[P]$. Conversely, if $a \in P$ and $(A_1, \dots, A_{k_{P \setminus \{a\}}})$ is a zero-sum subpartition of $P \setminus \{a\}$, then $(A_1, \dots, A_{k_{P \setminus \{a\}}})$ is a zero-sum subpartition of P . Thus $k_{P \setminus \{a\}} \leq k_P$, which means $T[P \setminus \{a\}] \leq k_P$ and thus implies $T[P] \leq k_P$. Altogether, we have $T[P] = k_P$. Now if P sums to zero, let (A_1, \dots, A_{k_P}) be a maximum cardinality zero-sum partition of P . We thus have $P = \bigcup_{j=1}^{k_P} A_j$. Let $a \in A_{k_P}$ (note that a exists, since A_{k_P} is non-empty). Thus (A_1, \dots, A_{k_P-1}) is a maximum zero-sum subpartition of $P \setminus \{a\}$, and hence $k_P = k_{P \setminus \{a\}} + 1$, which implies $T[P] \geq k_P$. Conversely, if $a \in P$, then $a \in A_j$ for a given $j \in \llbracket 1, k_P \rrbracket$. Using similar arguments as previously, we can prove that $k_P = k_{P \setminus \{a\}} + 1$, and thus $T[P] \leq k_P$. Altogether, we have $T[P] = k_P$. We conclude that $T[P] = k_P$ for any $P \subseteq S$. In particular, $T[S]$ contains a maximum cardinality zero-sum partition of S . By operating a traceback in T , the sought partition can be found in polynomial time, which solves MZSP. \square

The previous theorem is based on the fact that the number of distinct subsets in S is upper bounded by $O(2^n)$. It is also possible to upper bound this number by a function of b , the number of bits needed to binary encode S .

Theorem 6. MZSP is solvable in $2^{O(\frac{b}{\log b})}$.

Proof. Let us partition S into $S_p = \{a_i \in S \text{ s.t. } |a_i| \leq \sqrt{b}\}$ and $S_q = \{a_i \in S \text{ s.t. } |a_i| > \sqrt{b}\}$. Let us also denote, for any multiset E , by $\mathcal{P}(E)$ the set of subsets of E (e.g. $\mathcal{P}(\{2, 3, 3\}) = \{\emptyset, \{2\}, \{3\}, \{2, 3\}, \{3, 3\}, \{2, 3, 3\}\}$). In that case, we have $|\mathcal{P}(S_p)| \leq (b+1)^{2\sqrt{b}}$: indeed, by definition any element $a \in S_p$ satisfies $|a| \leq \sqrt{b}$. Moreover, a appears at most b times in S_p , since any a needs at least one bit to be encoded, while b bits are enough to encode S . Thus S_p contains at most $2\sqrt{b}$ different numbers, each of them appearing at most b times in S_p . Hence $|\mathcal{P}(S_p)| \leq (b+1)^{2\sqrt{b}} \leq 2^{2\sqrt{b} \log(b+1)} \leq 2^{2\frac{b}{\log b}}$. On the other hand, S_q contains elements of size at least \sqrt{b} . Then S_q cannot be of cardinality greater than $2\frac{b}{\log b}$, otherwise encoding S_q would require more than b bits; thus $|\mathcal{P}(S_q)| \leq 2^{2\frac{b}{\log b}}$. Since S_p and S_q form a partition of S , we have that $|\mathcal{P}(S)| = |\mathcal{P}(S_p)| \cdot |\mathcal{P}(S_q)| \leq 2^{4\frac{b}{\log b}}$. Since the dynamic programming algorithm from the proof of Theorem 5 solves MZSP and has running time in $O^*(|\mathcal{P}(S)|)$, we conclude that MZSP can be solved in $2^{O(\frac{b}{\log b})}$, which proves the theorem. \square

We now end this section by turning our attention to the inapproximability of MZSP, in Theorem 7 below.

Theorem 7. Unless $P=NP$, MZSP cannot be approximated within ratio $O(n^{1-\epsilon})$ for any $\epsilon > 0$.

Proof. As for Theorem 2, we prove the result by reduction from PARTITION, which is known to be NP-complete [6]. Let $X = \{x_1, x_2, \dots, x_\ell\}$ be an instance of PARTITION, and let $\sum_{i=1}^\ell x_i = 2N$ with $N \geq 1$. We can indeed assume $\sum_{i=1}^\ell x_i$ to be non-zero and even, otherwise the problem is trivially answered. The reduction is as follows: let $q \geq 1$ be any integer, and let us recursively build a set $\{p_0, p_1, \dots, p_q\}$ of integers. More precisely, we set $p_0 = 1$, and $p_i = (2iN + 1)p_{i-1}$ for any $i \in \llbracket 1, q \rrbracket$. We note that for any $0 \leq j < i \leq q$, p_j divides p_i . Based on X and on the values p_0, p_1, \dots, p_q , we now construct the multiset $S = (\bigcup_{i=0}^q \{p_i X, -N p_i\}) \cup \{-N \sum_{i=0}^q p_i\}$ where, for any $i \in \llbracket 0, q \rrbracket$, $p_i X$ denotes the values obtained by multiplying each element of X by p_i . It can be seen that S sums to zero, and is thus a valid instance of MZSP. It can also be seen that the above reduction takes polynomial time as long as q remains polynomial in the input size of PARTITION. Intuitively, the above reduction consists in ‘‘copying’’, a certain amount of times, an (expanded) instance X of PARTITION, so that the solution size of MZSP on S increases, while maintaining the property that the different ‘‘expanded copies’’ of X in S do not mutually interact.

Let us now prove correctness of our reduction, by showing the following: (i) X is a YES-instance for PARTITION iff (ii) MZSP for S yields a partition of cardinality $q + 2$ iff (iii) MZSP for S yields a partition of cardinality 2.

(i) \Rightarrow (ii) Suppose X is a YES-instance for PARTITION. Then there exists $P \subset X$ such that $\sum_{a \in P} a = N$. By construction, for every $i \in \llbracket 0, q \rrbracket$, $\sum_{a \in P} p_i a = N p_i$. Hence, for every negative number $-s$ in S (s being either $-N p_i$ for some $i \in \llbracket 0, q \rrbracket$, or $-N \sum_{i=0}^q p_i$), it is possible to find a subset of S summing to $N p_i$, and moreover any pair of such sets is mutually disjoint. Hence S can be partitioned into zero-sum subsets, and the cardinality of such a partition is $q + 2$.

(ii) \Rightarrow (iii) If MZSP for S yields a partition of cardinality $q + 2$, and since $q + 2 \geq 2$ then, by merging any $q + 1$ sets in this partition, we obtain a size-2 zero-sum partition.

((iii) \Rightarrow (i)) Suppose there exists a zero-sum partition of MZSP for S , of cardinality 2. In that case, there exists a non-empty zero-sum (multi)set $P \subset S$ that does not contain the negative integer $-N \sum_{i=0}^q p_i$. Let us denote by i_0 the smallest index $i \in \llbracket 0, q \rrbracket$ such that $-N p_i$ belongs to P . Note that for all $i \in \llbracket 0, i_0 - 1 \rrbracket$, we have $p_i X \cap P = \emptyset$: indeed, suppose by contradiction that this is not the case, and let A be the sum of the elements of $(\bigcup_{i=0}^{i_0-1} p_i X) \cap P$. Then we have $A \leq (1 + p_1 + \dots + p_{i_0-1}) \cdot 2N$, hence $A \leq i_0 \cdot 2N \cdot p_{i_0-1}$, which yields $A < p_{i_0}$. In particular, p_{i_0} does not divide A , since $A \neq 0$. As p_{i_0} divides every other element of P , we conclude that $A = 0$, which is the sought contradiction.

Now, let us consider $P' = P \bmod p_{i_0+1}$. From the above, the only elements from P that induce non zero elements in P' are the elements of $p_{i_0} X \cap P$, together with $-N p_{i_0}$. We thus conclude there exists a (multi)set $K \subset X$ such that $\sum_{a \in K} p_{i_0} a \equiv N p_{i_0} \bmod p_{i_0+1}$. Since $p_{i_0+1} > 2N p_{i_0}$, $\sum_{a \in K} p_{i_0} a = N p_{i_0}$, and thus $\sum_{a \in K} a = N$. In other words, we have a YES-instance for PARTITION.

Now we have proved correctness of our reduction, let us turn to proving our inapproximability result. Let ϵ be any strictly positive value, and suppose that there exists an approximation algorithm \mathcal{A} for MZSP, of ratio $\rho = O(n^{1-\epsilon})$ with $n = |S|$. Take now an instance X of PARTITION, and recall that $\ell = |X|$. Let C be a constant such that $\rho \leq C n^{1-\epsilon}$ for sufficiently large n . We let $q = \max(C^{1/1-\epsilon} - 1, (C^{1/1-\epsilon}(\ell + 1) + 1)^{1/\epsilon-1} - 1)$, and we proceed with the above mentioned reduction by building the MZSP instance S based on X and on parameter q . We have that $n = |S| = (\ell + 1)(q + 1) + 1$.

Then, $\frac{q+1}{C^{1-\epsilon}} \geq 1$ and $(q + 1)^{\frac{1}{1-\epsilon}-1} - C^{1/1-\epsilon}(\ell + 1) \geq 1$ which yields $\frac{q+1}{C^{1-\epsilon}} \cdot ((q + 1)^{\frac{1}{1-\epsilon}-1} - C^{1/1-\epsilon}(\ell + 1)) \geq 1$ and $C((\ell + 1)(q + 1) + 1)^{1-\epsilon} \leq q + 1$. We thus conclude that $C n^{1-\epsilon} \leq q + 1$. We now apply \mathcal{A} on S , and solve it polynomially within factor $\rho \leq C n^{1-\epsilon}$. Thus we obtain $\rho \leq q + 1$, hence $\rho < q + 2$. However, we know from the above that if we have a YES-instance for PARTITION, then there exists a zero-sum partition of S of cardinality $q + 2$. In that case, the solution provided by the approximation algorithm \mathcal{A} is a zero-sum partition of S of cardinality $c \geq \frac{q+2}{\rho} > 1$. Conversely, if \mathcal{A} provides a zero-sum partition of cardinality $c > 1$, then such zero-sum partition shows that X is a YES-instance for PARTITION.

Altogether, if there exists an approximation algorithm \mathcal{A} for MZSP of ratio $\rho = O(n^{1-\epsilon})$, it is possible to polynomially solve PARTITION: a contradiction, unless $P = NP$, to the fact that PARTITION is NP-complete [6]. \square

3. Parameterized complexity of MZSP

The previous section was devoted to providing results on the computational complexity of MZSP. In this section, we discuss the parameterized complexity of MZSP, essentially focusing on parameters k, m, n^* and B , which were defined in Section 1.

Parameters k and m We first consider fixed-parameterized complexity of MZSP with respect to the size k of the solution. On the way, we will also discuss parameter $m = \min\{\text{neg}, \text{pos}\}$, as we always have $m \geq k$. By Theorem 2, we know that, unless $P = NP$, MZSP is not FPT with respect to parameter k (resp. m), since MZSP is NP-complete even in the case where both these values are constant. The following theorem and corollary show W[1]-hardness of UNARY MZSP with respect to the same parameters.

Theorem 8. UNARY MZSP parameterized by m is W[1]-hard.

Theorem 8 implies the following corollary, as we always have $m \geq k$.

Corollary 1. UNARY MZSP parameterized by k is W[1]-hard.

Proof of Theorem 8. Let us assume that integers are encoded in unary. We reduce from UNARY BIN-PACKING, which is known to be W[1]-hard with respect to parameter “size of the solution” [9], to UNARY MZSP. We first recall the definition of BIN-PACKING (presented here in its decision version):

BIN-PACKING

Instance: a multiset of strictly positive integers $P = \{w_1, \dots, w_n\}$, an integer W , an integer t .

Question: Does there exist a partition $\{J_1, \dots, J_t\}$ of P such that $\sum_{w_j \in J_i} w_j \leq W$ for every $i \in \llbracket 1, t \rrbracket$?

As mentioned above, UNARY BIN-PACKING is BIN-PACKING in which all integers are assumed to be encoded in unary; besides, UNARY BIN-PACKING, parameterized by the number t of bins, is known to be W[1]-hard [9]. Let $I = (P, W, t)$ be an instance of UNARY BIN-PACKING. Moreover, assume $\sum_{i=1}^n w_i = tW$, since UNARY BIN-PACKING remains W[1]-hard parameterized by the number t of bins under this condition [9]. Let us now construct the following instance of MZSP: $S = \{w_1, \dots, w_n, \underbrace{-W, \dots, -W}_t\}$. Note that we have

$m = t$. We now show that MZSP admits a t -size zero-sum partition iff $I = (P, W, t)$ is a YES-instance for UNARY BIN-PACKING.

(\Rightarrow) Suppose (P, W, t) is a YES-instance for UNARY BIN-PACKING. Thus there exists a partition $\{J_1, \dots, J_t\}$ of P such that $\sum_{w_j \in J_i} w_j \leq W$ for every $i \in \llbracket 1, t \rrbracket$. However, since we assume $\sum_{i=1}^n w_i = tW$, we conclude that every $J_i, i \in \llbracket 1, t \rrbracket$, is such that $\sum_{w_j \in J_i} w_j = W$. Hence, $\{A_1, A_2, \dots, A_t\}$, where $A_i = J_i \cup \{-W\}$ for every $i \in \llbracket 1, t \rrbracket$, is a t -size zero-sum partition of S .

(\Leftarrow) Conversely, suppose there exists a t -size zero-sum partition $\{A_1, A_2, \dots, A_t\}$ of S . Since S contains exactly t negative numbers, this implies that every $A_i, i \in \llbracket 1, t \rrbracket$, contains exactly one occurrence of $-W$. Thus $\{J_1, J_2, \dots, J_t\}$, where $J_i = A_i \setminus \{-W\}$ for every

Algorithm 1 XP algorithm for solving MZSP, parameterized by k .

```

1: for every  $k$ -tuple  $u = (u_1, \dots, u_k)$  of  $S$  do
2:   for every  $v \in \llbracket -nB, nB \rrbracket^k$  do
3:      $T[v] \leftarrow \text{False}$ 
4:   end for
5:    $T[u] \leftarrow \text{True}$ 
6:   for every  $a \in S \setminus u$  do
7:     for every  $v \in \llbracket -nB, nB \rrbracket^k$  do
8:       if  $T[v] = \text{True}$  then
9:         for all  $i \in \llbracket 1, k \rrbracket$  do
10:             $T[v + a \cdot e_i] \leftarrow \text{True}$ 
11:          end for
12:        end if
13:      if  $T[0, \dots, 0] = \text{True}$  then
14:        return True
15:      end if
16:    end for
17:  end for
18: end for
19: return False

```

$i \in \llbracket 1, t \rrbracket$, is a partition of P . Moreover, since each A_i sums to zero, we know that each J_i sums to W , which ensures that (P, W, t) is a YES-instance for UNARY BIN-PACKING.

The above reduction is a valid parameterized reduction, since parameter t for UNARY BIN-PACKING is strictly equal to parameter m for UNARY MZSP. Moreover, the instance S of UNARY MZSP that we built satisfies $k = m$. \square

Theorem 8 suggests that, even encoded in unary, MZSP admits no FPT algorithm parameterized by k . What we can show, in the following theorem, is that MZSP encoded in unary is in XP when parameterized by k .

Theorem 9. *UNARY MZSP is in XP parameterized by k .*

Proof. We show here that MZSP can be solved in $O(kn^{k+1}(2nB)^k)$. Let S be an instance of MZSP. Recall that $n = |S|$, B is the maximum taken over the absolute values of all elements of S , and k is the size of the solution. We create a Boolean array T indexed by $\llbracket -nB, nB \rrbracket^k$. We then apply Algorithm 1, which explicitly computes a size- k zero-sum partition of S , while forcing each set of this partition to be non-empty. More precisely, Algorithm 1 consists in iteratively computing the k -tuple (v_1, \dots, v_k) such that there is a subpartition A_1, \dots, A_k in which each A_i sums to v_i . We start by initialising table T Lines 2–4. We want the A_i s to be non-empty, so we require that they contain at least one element. We choose a u_i for each and initialise $T[u]$ to True (see Line 5). Then, to increment table T , we need to put every element a of S in one of the A_i s. Thus we try to assign a to each of them, and see to which value A_i sums. This is done in Lines 6–12; note that $e_i = (0, 0, \dots, 0, 1, 0, \dots, 0)$ denotes the vector of length k where the unique non-zero coordinate is 1 and appears at position i . We only need to check, Line 13, if we have been able to obtain $(0, \dots, 0)$ this way. Finally, we need to iterate the process for each possible initial choice of u , as done in Line 1.

The time complexity of Algorithm 1 is $O(kn^{k+1}(2nB)^k \log B)$, and its space complexity is $O((2nB)^k)$.

Let us now show its correctness, by showing that a size- k zero-sum partition of S exists iff Algorithm 1 returns True . We will use the following property: $T[v] = \text{True}$ iff there is a size- k subpartition (A_1, \dots, A_k) of S such that each A_i sums to v_i . We prove this property by induction on $t := |A_1| + |A_2| + \dots + |A_k|$. Then, by definition, the property is satisfied for $t \leq k$. Let us now assume that $t > k$. If $T[v] = \text{True}$, then there exists $a \in S$ and $i \in \llbracket 1, k \rrbracket$ such that $T[v - a \cdot e_i] = \text{True}$. By induction, there exists (A_1, \dots, A_k) a subpartition of S such that each A_j sums to v_j for $j \neq i$ and A_i sums to $v_i - a$. Then $(A_1, \dots, A_i \cup \{a\}, \dots, A_k)$ satisfies the property for v . Conversely, suppose that (A_1, \dots, A_k) is a subpartition of S such that each A_i sums to v_i and $t > k$. Then there is an index $i \in \llbracket 1, k \rrbracket$ such that A_i contains at least two numbers. Let a be one of them. Then $(A_1, \dots, A_i \setminus \{a\}, \dots, A_k)$ is a subpartition of S summing to $v - a \cdot e_i$. Then $T(v - a \cdot e_i) = \text{True}$ by induction. Then $T[v] = \text{True}$.

From the above property, we have that $T[0] = \text{True}$ (and thus Algorithm 1 returns True) iff there is a size- k zero-sum subpartition of S , which is equivalent to the existence a size- k zero-sum partition of S . \square

Since we always have $m \geq k$, Theorem 9 implies the following corollary.

Corollary 2. *UNARY MZSP is in XP when parameterized by m .*

*Parameter number of distinct values n^** We now discuss parameter n^* , for which we first provide a complexity lower bound based on ETH. Recall that b is the (binary encoded) size of the input instance S .

Theorem 10. *Unless ETH fails, MZSP cannot be solved in $2^{o(n^*)}b^{O(1)}$.*

Proof. The proof is based on a combination of (i) a reduction from 3-SAT to SUBSET-SUM presented in [8] and inspired from [10] and (ii) the reduction from SUBSET-SUM to MZSP from proof of Theorem 4. More precisely, starting from any instance of 3-SAT with n variables and m clauses, an instance of SUBSET-SUM containing $n' = 2n + 2m$ integers, among which $2n + m$ are pairwise distinct, is constructed. Moreover, in proof of Theorem 4, the instance S of MZSP built from SUBSET-SUM contains $n'' = n' + 2 = 2n + 2m + 2$ integers, among which $n^* = 2n + m + 2$ are pairwise distinct. Under ETH, 3-SAT cannot be solved in $2^{o(n)}$. If we combine this information with the sparsification method [11] (which allows to consider only 3-SAT instances for which $m = O(n)$) and the above argument, we conclude that, under ETH, MZSP cannot be solved in $2^{o(n^*)}b^{O(1)}$. \square

Concerning parameterized complexity with respect to n^* , we suspect MZSP to be W[1]-hard parameterized by n^* , but the question remains open. Meanwhile, we are able to prove (see Theorem 11) that the problem is in XP with respect to n^* . The rationale for this result is that the multisets that constitute any maximum zero-sum partition of S are few, and that we can efficiently compute them. In order to prove Theorem 11, we need to introduce several definitions, and first prove two propositions (Propositions 1 and 2).

Let us suppose that S is a multiset containing n^* distinct values, denoted a_1, \dots, a_{n^*} . We introduce several notions: given any multiset \mathcal{M} built from a_1, \dots, a_{n^*} , we call *multiplicity multiset* of \mathcal{M} the multiset $\{u_1, u_2, \dots, u_{n^*}\}$ representing the multiplicities of each a_i in \mathcal{M} : more precisely, for any $i \in [1, n^*]$, $u_i \in \mathbb{N}$ is the number of times a_i appears in \mathcal{M} . With this notation, we can define a partial order \leq on multiplicity multisets as follows: let $u = \{u_1, u_2, \dots, u_{n^*}\}$ and $v = \{v_1, v_2, \dots, v_{n^*}\}$ be two multiplicity multisets; we write $u \leq v$ whenever $u_i \leq v_i$ for every $i \in [1, n^*]$. Now let $s = \{s_1, s_2, \dots, s_{n^*}\}$ be the multiplicity multiset of S . We first define two sets, respectively named K and D : K is the set of irreducible multiplicity multisets of S leading to zero-sum subsets of S , and D is the set of all zero-sum subsets of S . In the following, for simplicity we write $u = 0$ for any vector u whenever all its coordinates are equal to 0. Formally, K and D are defined as follows:

$$K = \left\{ u \in \mathbb{N}^{n^*} \left| \sum_{i=1}^{n^*} u_i a_i = 0, u \neq 0 \text{ and } \forall v \in \mathbb{N}^{n^*}, v \leq u \text{ and } \sum_{i=1}^{n^*} v_i a_i = 0 \Rightarrow v = 0 \text{ or } v = u \right. \right\} \text{ and}$$

$$D = \left\{ u \in \mathbb{N}^{n^*} \left| u \leq s, u \neq 0 \text{ and } \sum_{i=1}^{n^*} u_i a_i = 0 \right. \right\}.$$

Any maximum zero-sum partition of S is induced by elements of $K \cap D$ only. We define a third set Z as follows: $Z = \left\{ u \in \mathbb{N}^{n^*} \left| u \neq 0 \text{ and } \sum_{i=1}^{n^*} u_i a_i = 0 \right. \right\}$. Note that both K and D are included in Z . We are now interested in two properties, related to the above definitions. We begin with Proposition 1.

Proposition 1. *Let U be a subset of Z such that $K \cap D \subseteq U$. There exists a size- p zero-sum partition of S iff there exist $\ell \geq p$ elements of U , say u^1, u^2, \dots, u^ℓ such that $\sum_{i=1}^\ell u^i = s$.*

Proof. Let $U \subseteq Z$ such that $K \cap D \subseteq U$. Suppose there exists a size- p zero-sum partition of S , and let A_1, \dots, A_k be a maximum zero-sum partition of S , thus with $k \geq p$. Let us denote by u^1, u^2, \dots, u^k the multiplicity multisets of resp. A_1, \dots, A_k . By definition, $A_i \subseteq S$ for any $i \in [1, k]$, and thus $u^i \in D$. Moreover, suppose u^i is not irreducible, i.e. $u^i \notin K$. Then there exists $C \neq \emptyset$, $C \subset A_i$ such that C sums to zero. In particular, it is possible to replace A_i by both C and $A_i \setminus C$ in the maximum zero-sum partition of S , which contradicts maximality. We thus conclude that $u^i \in K$, and thus that $u^i \in K \cap D$. Consequently, we have that $u^i \in U$. Hence, u^1, u^2, \dots, u^k are $k \geq p$ elements of U such that $\sum_{i=1}^k u^i = s$.

Conversely, suppose there exists $\ell \geq p$ elements u^1, u^2, \dots, u^ℓ of U such that $\sum_{i=1}^\ell u^i = s$. For any $i \in [1, \ell]$, let us denote by A_i the multiset associated to u^i , and for every $j \in [1, n^*]$, let u_j^i denote the j -th coordinate of vector u^i . Then A_i sums to zero, since $\sum_{j=1}^{n^*} u_j^i a_j = 0$. Moreover, the A_i s form a partition of S since for every $j \in [1, n^*]$, $\sum_{i=1}^\ell u_j^i a_j = s_j$. Thus A_1, \dots, A_ℓ is a size- ℓ zero-sum partition of S . Since it is always possible to merge several zero-sum sets to create another zero-sum set, we conclude there exists a zero-sum partition of S , whose cardinality is $p \leq \ell$. \square

For solving p -MZSP, it thus suffices to compute $K \cap D$, and to test whether it is possible to reach s , using p or more elements of $K \cap D$. Before that, we discuss the maximum cardinalities of K and D (see Proposition 2), which will be useful to evaluate the time complexity to generate these sets.

Proposition 2. $D \subseteq [0, n]^{n^*}$ and $K \subseteq [0, n^*B - 1]^{n^*}$.

Proof. The first result is straightforward: since for any $j \in [1, n^*]$, we have $s_j \leq n$, then for any $u = \{u_1, u_2, \dots, u_{n^*}\} \in D$ and for any $j \in [1, n^*]$, we have $u_j \leq n$. The second result is by contradiction. Let $u \in Z$ with $u = \{u_1, u_2, \dots, u_{n^*}\}$, and let us suppose that there exists an $i \in [1, n^*]$ such that $u_i \geq n^*B$. We will show that, in that case, $u \notin K$. Recall that the n^* distinct values in S are denoted a_1, a_2, \dots, a_{n^*} , and let us denote by I the set of indices $j \in [1, n^*]$ for which a_j and a_i have same sign. Then we have $\sum_{j \in I} u_j |a_j| = \sum_{j \notin I} u_j |a_j|$.

Consequently, $n^*B|a_i| \leq u_i |a_i| \leq \sum_{j \notin I} u_j |a_j|$. In particular, there exists $j \notin I$ such that $u_j |a_j| \geq B|a_i|$. Thus there exists two indices i and j , whose corresponding values a_i and a_j have opposite signs, and such that $u_i \geq B \geq |a_j|$ and $u_j \geq \frac{B|a_i|}{|a_j|} \geq |a_i|$. Since $a_i a_j - a_j a_i =$

Algorithm 2 XP algorithm for solving MZSP, parameterized by n^* .

```

1: Compute  $D$ 
2: Initialize  $T$ 
3: for every  $i \in \llbracket 1, n \rrbracket$  do
4:   for every  $u \in D$  do
5:     for every  $v \in D$  do
6:       if  $v + u \in D$  then
7:          $T[v + u] = \max(T[v] + 1, T[v + u])$ 
8:       end if
9:     end for
10:  end for
11: end for
12: return  $T[s]$ 

```

0, the n^* -tuple $v = \{v_1, v_2, \dots, v_{n^*}\}$ is such that $v_i = |a_j|$, $v_j = |a_i|$ and $v_\ell = 0$ for all $\ell \neq i, j$ satisfies $\sum_{\ell=1}^{n^*} v_\ell a_\ell = 0$. Moreover, $v \leq u$, $u \neq 0$ and $v \neq 0$ since $n^*B > |a_j|$. Thus we conclude that u is not irreducible, i.e. $u \notin K$, which proves the proposition. \square

In order to compute D (resp. K), it thus suffices to generate each element of $\llbracket 0, n \rrbracket^{n^*}$ (resp. $\llbracket 0, n^*B - 1 \rrbracket^{n^*}$), and check for each of them whether it belongs to D (resp. K). For each element of $\llbracket 0, n \rrbracket^{n^*}$, checking its membership to D can be achieved in $O(n^2 \log B)$, thus D can be computed in $O(n^{n^*+2} \log(B))$. Concerning K , testing if an element of $\llbracket 0, n^*B - 1 \rrbracket^{n^*}$ sums to zero can be done in $O(n^* \log(n^*B))$, and by dynamic programming, we can check if it is irreducible in $O((n^*B)^{n^*})$; thus K can be computed in $O((n^*B)^{2n^*})$.

The set K (and its computation) will be useful later for proving Theorem 13. In the following, we first focus on set D , whose cardinality is denoted c_D . Indeed, starting from D , Algorithm 2 shows that MZSP can be solved in $O(nc_D^2 n^* \log n + n^{n^*+2} \log B)$. Since, by Proposition 2 above, c_D is in $O(n^{n^*})$, this shows that MZSP is XP relatively to parameter n^* , as stated by the following theorem.

Theorem 11. *MZSP is in XP when parameterized by n^* .*

Proof. We provide an algorithm that runs in time $O(nc_D^2 n^* \log n + n^{n^*+2} \log B)$. The proof derives from Algorithm 2, in which T is an array indexed by the elements of D and which is initialized as follows: $T[0] = 0$, and for every other vector $v \in D$, $T[v] = -\infty$. We also recall that s is the multiplicity multiset of S .

The idea behind Algorithm 2 is to understand every multiset as its multiplicity multiset. As seen before, we can efficiently compute D . Then, we iteratively compute the largest number of elements of D that we can sum to obtain v . To do so, for each u in D , we say that the largest number of elements of D that we can sum to obtain $v + u$ is at most the same number as for v plus one for each v in D . This is what is done Lines 4 – 10 of Algorithm 2. We know that we will need at most n elements of D to sum to s , so we only need to do this operation n times, which explains Line 3.

Clearly, Algorithm 2 runs in $O(nc_D^2 n^* \log n)$ where $c_D = |D|$, since nc_D^2 additions on vectors are realized, each taking $O(n^* \log n)$ time. To this complexity, $O(n^{n^*+2} \log B)$ should be added for the precomputation of D . We now show that Algorithm 2 is correct. At the end of the algorithm, for any $v \in D$, $T[v]$ represents the largest number of elements of D that we can sum to obtain v . Let us denote by w_v this value (thus $T[v] = w_v$). For any $i \in \llbracket 1, n \rrbracket$, let $\mathcal{P}(i)$ be the following property: for every $v \in D$, $T[v] = w_v$ if $w_v \in \llbracket 0, i \rrbracket$. Our goal is to prove, by induction on i , that $\mathcal{P}(i)$ holds for any $i \in \llbracket 1, n \rrbracket$. First, $\mathcal{P}(0)$ is true since $w_v = 0$ implies $v = 0$. Now let $i \geq 0$, and let us assume $\mathcal{P}(i)$ holds. Let $v \in D$. If $w_v \leq i$, then by induction hypothesis, we have $T[v] = w_v$. If not, then there exist $u \in D$ such that $w_{v-u} = i$, and hence $T[v-u] = i$. Hence, by construction of T , $T[v] = i + 1 = w_v$, which consequently proves that $\mathcal{P}(i+1)$ holds. By induction, $\mathcal{P}(i)$ holds for any $i \in \llbracket 1, n \rrbracket$. In particular, $T[s]$ represents the largest number of elements from D that can be summed in order to obtain s . By Proposition 1, there exists ℓ elements of D whose sum is s iff there exists a zero-sum partition of S , of cardinality ℓ . Thus the cardinality of a maximum zero-sum partition of S is $T[s]$, which shows correctness of Algorithm 2. \square

As mentioned before, we conjecture MZSP to be W[1]-hard parameterized by n^* . In contrast, we have the following result.

Theorem 12. *MZSP is FPT when parameterized by $n^* + k$.*

Proof. In order to prove the result, we model the MZSP problem as an Integer Linear Program (ILP). Since ILP is FPT parameterized by the number of its variables (see e.g. [12,13]), it suffices that our ILP formulation uses a number of variables that is only a function of n^* and k to prove the result.

Let S be an instance of MZSP, let us call the n^* distinct values of S a_1, \dots, a_{n^*} , and let us assume that s_1, \dots, s_{n^*} are their respective multiplicities in S .

The ILP instance we construct uses kn^* variables, namely $x_{i,j}$ with $i \in \llbracket 1, n^* \rrbracket$ and $j \in \llbracket 1, k \rrbracket$. Each such variable $x_{i,j}$ indicates the number of times value a_i is present in a zero-sum subset S_j of a partition of S .

MZSP (ILP model):

$$\begin{array}{ll}
\text{C. 1} & x_{i,j} \geq 0 \quad \forall 1 \leq i \leq n^*, \forall 1 \leq j \leq k \\
\text{C. 2} & \sum_{j=1}^k x_{i,j} = s_i \quad \forall 1 \leq i \leq n^* \\
\text{C. 3} & \sum_{i=1}^{n^*} a_i x_{i,j} = 0 \quad \forall 1 \leq j \leq k \\
\text{C. 4} & \sum_{i=1}^{n^*} x_{i,j} > 0 \quad \forall 1 \leq j \leq k
\end{array}$$

Constraints C. 1 trivially impose that each variable (representing the number of occurrences of value a_i in S_j) belongs to \mathbb{N} . Constraints C. 2 impose that each value a_i appears exactly s_i times in the partition of S . Constraints C. 3 impose that each S_j in the partition of S sums to zero. Finally, constraints C. 4 impose that each S_j is non-empty.

Since solving any ILP is FPT parameterized by the number of its variables [12,13], and since we have kn^* variables in our model, we conclude MZSP is FPT parameterized by $n^* + k$. More precisely, any ILP on x variables can be solved in $O^*(x^{2.5x+o(x)})$ [12]. Thus MZSP can be solved in $O^*((kn^*)^{2.5kn^*+o(kn^*)})$. \square

Parameter maximum absolute value B Recall that B is the greatest integer (in absolute value) in an instance S of MZSP.

Theorem 13. *MZSP is FPT when parameterized by B .*

Proof. In order to prove the result, we provide an ILP model for our problem. We will then show that the number of variables of our ILP is a function of n^* and B only, which, combined with the fact that ILP is FPT parameterized by its number of variables [12,13] and the fact that $n^* \leq 2B$, allows us to conclude. Given an integer k , we are interested in solving k -MZSP, which asks whether a size- k zero-sum partition of S exists. Let (S, k) be an instance of k -MZSP. Let us number the n^* distinct values in S a_1, \dots, a_{n^*} and let s_1, \dots, s_{n^*} be their respective multiplicities in S . Let $c_K = |K|$, where K is the set defined previously, and let us compute K – recall that c_K , by Proposition 2 and the discussion that follows, satisfies $c_K = O((n^*B)^{n^*})$, and that K can be computed in $O((n^*B)^{2n^*})$. Our ILP is based on the following c_K variables x_u , $u \in K$, where x_u represents the number of times element u appears in a zero-sum partition of S of cardinality at least k . The ILP formulation of the problem is as follows.

 k -MZSP (ILP model):

$$\begin{array}{ll}
\text{C. 1} & \forall u \in K \quad x_u \geq 0 \\
\text{C. 2} & \forall i \in [1, n^*] \quad \sum_{u \in K} u_i x_u = s_i \\
\text{C. 3} & \sum_{u \in K} x_u \geq k
\end{array}$$

We now show correctness of our ILP model, by proving that there exists a zero-sum partition of MZSP of cardinality k iff the above ILP formulation admits a solution.

(\Rightarrow) Suppose S admits a size- k zero-sum partition. Then, by Proposition 1, we know there exist $\ell \leq k$ elements of K which sum to s , that we will call u^1, \dots, u^ℓ . For $u \in K$, let x_u denote the number of times u appears in (u_1, \dots, u_ℓ) . Then, by definition, $\forall u \in K, x_u \geq 0$, $\sum_{u \in K} x_u u = s$ and $\sum_{u \in K} x_u = \ell \geq k$. Thus our ILP formulation admits a solution.

(\Leftarrow) Conversely, suppose there exists x_u for $u \in K$, which is a solution to the above ILP formulation. Let us build $(u^i)_{i \in [1, \ell]}$, where element u appears exactly x_u times. Then, from C. 2, $\sum_{i=1}^{\ell} u^i = s$. Moreover, from C. 3, $\ell \geq k$. Thus, from Proposition 1, there exists a size- k zero-sum partition of S .

Since $n^* \leq 2B$, and since ILP, parameterized by the number x of variables, is FPT and can be solved in $O^*(x^{2.5x+o(x)})$ [12,13], and since here $x = c_K = O((n^*B)^{n^*})$, the result follows. \square

4. Conclusion

We provided diverse algorithmic results concerning the MZSP problem, including hardness, (in)approximability and fixed-parameterized complexity considerations with respect to parameters n, m, k, n^* and B . Some questions about MZSP remain unanswered. In particular, we conjecture MZSP to be W[1]-hard parameterized by the number n^* of distinct values in S ; (dis)proving it remains open.

CRedit authorship contribution statement

Guillaume Fertin: Writing – review & editing, Writing – original draft, Supervision, Methodology, Formal analysis, Conceptualization. **Oscar Fontaine:** Writing – review & editing, Writing – original draft, Methodology, Formal analysis, Conceptualization. **Géraldine Jean:** Writing – review & editing, Writing – original draft, Methodology, Formal analysis, Conceptualization. **Stéphane Vialette:** Writing – review & editing, Writing – original draft, Methodology, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We thank Ton van der Zanden for his suggestions, in particular those concerning parameter b .

References

- [1] G. Fertin, O. Fontaine, G. Jean, S. Vialette, The maximum zero-sum partition problem, in: S. Hsieh, L. Hung, R. Klasing, C. Lee, S. Peng (Eds.), *New Trends in Computer Technologies and Applications - 25th International Computer Symposium, ICS 2022*, in: *Communications in Computer and Information Science*, vol. 1723, Springer, 2022, pp. 73–85.
- [2] G. Fertin, G. Jean, E. Tannier, Algorithms for computing the double cut and join distance on both gene order and intergenic sizes, *Algorithms Mol. Biol.* 12 (1) (2017) 16:1–16:11.
- [3] G. Fertin, A. Labarre, I. Rusu, E. Tannier, S. Vialette, *Combinatorics of Genome Rearrangements*, Computational Molecular Biology, MIT Press, 2009.
- [4] M. Cygan, F.V. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, S. Saurabh, *Parameterized Algorithms*, Springer, 2015.
- [5] M.R. Garey, D.S. Johnson, *Computers and Intractability: a Guide to the Theory of NP-Completeness*, Bell Telephone Laboratories, 1979.
- [6] R.M. Karp, *Reducibility Among Combinatorial Problems*, Springer US, Boston, MA, 1972, pp. 85–103.
- [7] R. Impagliazzo, R. Paturi, On the complexity of k -SAT, *J. Comput. Syst. Sci.* 62 (2) (2001) 367–375.
- [8] K. Jansen, F. Land, K. Land, Bounding the running time of algorithms for scheduling and packing problems, in: F. Dehne, R. Solis-Oba, J.-R. Sack (Eds.), *Algorithms and Data Structures*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, pp. 439–450.
- [9] K. Jansen, S. Kratsch, D. Marx, I. Schlotter, Bin packing with fixed number of bins revisited, *J. Comput. Syst. Sci.* 79 (1) (2013) 39–49.
- [10] I. Wegener, *Complexity Theory*, Springer, 2005.
- [11] R. Impagliazzo, R. Paturi, F. Zane, Which problems have strongly exponential complexity?, *J. Comput. Syst. Sci.* 63 (4) (2001) 512–530.
- [12] A. Frank, É. Tardos, An application of simultaneous Diophantine approximation in combinatorial optimization, *Combinatorica* 7 (1) (1987) 49–65.
- [13] H.W. Lenstra, Integer programming with a fixed number of variables, *Math. Oper. Res.* 8 (4) (1983) 538–548.