Generic-Case Complexity of the Multiple Subset Sum Problem

Alexandr Seliverstov
Institute for Information Transmission Problems of the Russian Academy of Sciences (Kharkevich Institute),
Moscow, Russia

PCA 2022

The subset sum problem is NP-complete. A commonly held view was that its worst-case complexity cannot be sub-exponential.

Moreover, if we restrict our computations by so-called linear machines, then the problem is proved hard (K. Meer, 1992).

Generic-case complexity of a decision problem is sub-exponential when the set of hard inputs is negligible (or empty), but almost all inputs can be solved in sub-exponential time. Moreover, the negligible set containing hard inputs can be discerned explicitly. Such algorithms are also known as deterministic errorless heuristics.

An example of fast generic-case algorithm is the condensation method for computing determinants (C.L. Dodgson, 1866). For general matrices, the method is very nice. Nevertheless, if some intermediate matrix contains a zero entry, then the algorithm can fails.

By means of variable elimination, searching for a $\{0,1\}$ solution to a system of m linearly independent linear equations in n variables is reduced to a parallel check whether a $\{0,1\}$ solution to a subsystem in $n-m$ variables can be extended to a $\{0,1\}$ solution to the whole system of equations in n variables. Hence, the initial problem is polynomial-time solvable when the difference between the number of variables and the number of linearly independent equations is bounded by a function of the type $n-m=O(\log_2 n)$.

Let us consider generic-case complexity when both m and difference $n-m$ are sufficiently large. For example, if $n=2m$, then almost all instances are decidable in $O(2^{\sqrt{n}\log^c n})$ operations over the field of coefficients.

The Kuzyurin Algorithm

Let us denote by A a $m \times n$ matrix with nonnegative entries and by $\mathbf{b}$ a column. One can enumerate all $\{0,1\}$ solutions to the system of inequalities $A\mathbf{x} \leq \mathbf{b}$ using dynamic programming. If $m > 9 \log_2 n$ and some assumptions about the distribution of the entry values hold, then the average number of $\{0,1\}$ solutions is polynomially bounded. Therefore, all solutions can be found in average polynomial time. The proof is based on the tail bounds of the binomial distribution. (N. N. Kuzyurin, 1994).

Next, one can verify whether a $\{0,1\}$ solution to the system of equations $A\mathbf{x} = \mathbf{b}$ exists.

The crucial limitation on the applicability of the Kuzyurin algorithm is the requirement of nonnegativity of the matrix entries. Of course, any system of linear equations can be reduced to another system with nonnegative coefficients, but the distribution is warped.

Low-density Problems

Let the density of an instance of the subset sum problem with positive integer coefficients a_k be defined by

$$\varrho = \frac{n}{\log_2 \max_k a_k}.$$

A polynomial-time algorithm is known for solving almost all instances of sufficiently low density $\varrho < 0.9408$ using a subroutine for finding the shortest nonzero vector in a lattice (J.C. Lagarias & A.M. Odlyzko, 1985) and (M.J. Coster, A. Joux, B.A. LaMacchia, A.M. Odlyzko, C.P. Schnorr, J. Stern, 1992).

Next, the multiple low-density problems are considered too (Y. Pan & F. Zhang, 2016), where

$$\varrho = \frac{n}{m \log_2 \max_k a_k}.$$

.

Our Main Algorithm

Within the context of the generic-case complexity, we consider machines having three halting states. So, the machine not only rejects or accepts an input, but it can also halt in the vague halting state. The latter means denial of response. But such a failure is possible on a small fraction of inputs (A.N. Rybalov, 2020).

Given three positive integers d, n, and m and a system of linear equations $x_j = \ell_j(1, x_1, \ldots, x_{n-m})$, where $n-m+1 \leq j \leq n$ and every $\ell_j(x_0, \ldots, x_{n-m})$ is a linear form.
Question. Is there a $\{0, 1\}$ solution to the system?
if there exist numbers λ_{ik} and λ_{ij} such that

$$\sum_t \left(\sum_{k=1}^{n-m} \lambda_{tk} x_k (x_k - x_0) + \sum_{j=n-m+1}^{n} \lambda_{tj} \ell_j (\ell_j - x_0) \right) g_t = x_0^d,$$

where g_t is the t-th monomial of degree $d-2$ in variables x_0, ..., x_{n-m}
then the machine **rejects** the input
else the machine halts in the **vague** halting state.

The algorithm uses as a subroutine the calculation of the rank of a matrix whose columns correspond to monomials of degree d.
The rank of an $N \times N$ matrix can be computed in $O\left(\log^2 N\right)$ arithmetic operations using a polynomial number of processors.

A.L. Chistov, *Fast parallel calculation of the rank of matrices over a field of arbitrary characteristic*. In: L. Budach (eds) Fundamentals of Computation Theory. FCT 1985. Lecture Notes in Computer Science, vol 199. Springer, Berlin, Heidelberg (1985), pp. 63–69.

On the other hand, the characteristic polynomial and the rank of the matrix are computable in $O\left(N^{\omega}\right)$ operations, where $2 \leqslant \omega \leqslant 3$ denotes the exponent of matrix multiplication, e.g., $\omega = 2.3728596$.

O.N. Pereslavtseva, *Calculation of the characteristic polynomial of a matrix*, Discrete Mathematics and Applications, **21**:1 (2011), 109–128.

H.Y. Cheung, T.C. Kwok, L.C. Lau, *Fast matrix rank algorithms and applications*, Journal of the ACM, **60**:5 (2013), 31.

V. Neiger, C. Pernet, *Deterministic computation of the characteristic polynomial in the time of matrix multiplication*, Journal of Complexity, **67** (2021), 101572.

Example. Let us consider a straight line L in the projective plane. There are four $\{0,1\}$ points with homogeneous coordinates $(1:0:0)$, $(1:0:1)$, $(1:1:0)$, and $(1:1:1)$, respectively. Our goal is a condition that no $\{0,1\}$-point belongs to L.

Theorem. *The straight line L does not pass through any $\{0,1\}$ point in the projective plane if and only if there is a cubic curve C such that its intersection $L \cap C$ consists of only one point at infinity.*

Thus, for $n = 2$ and $m = 1$, one can use the algorithm with $d = 3$.

Theorem. *There exist both constant c and machine with the vague halting state such that for all positive integers $d \geq 2$, n, and $m < n$ satisfying the inequality $(n - m + d)(n - m + d - 1) \leq md(d - 1)$ and for every m-tuple of linear forms $\ell_j(x_0, \ldots, x_{n-m})$, where $n - m < j \leq n$, the machine either rejects the input or halts in the vague halting state in $O(n^{cd})$ arithmetic operations.*
If the machine rejects the input, then there is no $\{0, 1\}$ solution to the system of all equations of the type $x_j = \ell_j(1, x_1, \ldots, x_{n-m})$.
Moreover, for every applicable integers d, n, and m, there exists a nonzero polynomial of degree at most $n^2(n - m + 1)^{2d-4}$ in coefficients of all the linear forms ℓ_j such that if the machine halts in the vague halting state, then the polynomial vanishes.

Roughly speaking, the algorithm checks whether there exists a hypersurface passing through each $\{0, 1\}$ point but not intersecting the given affine subspace. Therefore, for given d, if the algorithm accepts a subsystem, then it accepts the whole system too.

A special case $d = 2$ was previously published (S., 2021).

If $n - m = O(\sqrt{n})$, then one can use a constant d. Thus, generic-case complexity is polynomial, cf. (S., 2021).

Theorem. *There exist both constant c and machine with the vague halting state such that for all positive integers $n > m \geq 4 \log_2^4 n$ and for every m-tuple of linear forms $\ell_j(x_0, \ldots, x_{n-m})$, where $n - m < j \leq n$, the machine either rejects the input or halts in the vague halting state in $O\left(2^{cn/\log n}\right)$ arithmetic operations.*
If the machine rejects the input, then there is no $\{0, 1\}$ solution to the system of all equations of the type $x_j = \ell_j(1, x_1, \ldots, x_{n-m})$.
Moreover, for every applicable integers n and m, if all coefficients of forms ℓ_j picked independently and uniformly at random from a set of cardinality $(1/\varepsilon)4^{\lceil n/\log_2 n \rceil}$, then the machine halts in the vague halting state with probability at most ε.

The proof uses the Schwartz–Zippel lemma as well as special bases for the space of forms in variables x_0, . . . , x_{n-m}.

Let us denote $s = n - m$.

Lemma. *Given a general set of linear forms $\ell_j(x_0, \ldots, x_s)$, where $s < j \leq n$. If the inequality $(s+d)(s+d-1) \leq (n-s)d(d-1)$ holds, then every form of degree d in variables x_0, ..., x_s is equal to a linear combination of forms of the type $\ell_j^2 g$, where $s < j \leq n$ and g is a monomial of degree $d-2$.*

The inequality $(s+d)(s+d-1) \leq (n-s)d(d-1)$ is equivalent to

$$\frac{(s+d)!}{s!d!} \leq (n-s)\frac{(s+d-2)!}{s!(d-2)!},$$

where the dimension of the space of all forms of degree d is on the left hand and the dimension of the space of forms of the type $\ell_j^2 g$ is on the right hand.

Open question: What can we do having a sparse system of forms?

References

Dodgson C.L. Condensation of determinants, being a new and brief method for computing their arithmetical values.
Proceedings of the Royal Society of London, 1866, vol. 15, pp. 150–155.

Kuzyurin N.N. An algorithm that is polynomial in the mean in integer linear programming (in Russian).
Sibirskii Zhurnal Issledovaniya Operatsii, 1994, vol. 1, no. 3, pp. 38–48.

Meer K. A note on a $P \neq NP$ result for a restricted class of real machines.
Journal of Complexity, 1992, vol. 8, no. 4, pp. 451–453.

Rybalov A.N. On generic complexity of the subset sum problem for semigroups of integer matrices (in Russian).
Prikladnaya Diskretnaya Matematika, 2020, no. 50, pp. 118–126.

Seliverstov A.V. Binary solutions to large systems of linear equations (in Russian). *Prikladnaya Diskretnaya Matematika*, 2021, vol. 52, pp. 5–15.

Thank you!