Generic complexity of algorithmic problems over Brandt semigroups

Alexander Rybalov
Sobolev Institute of Mathematics, Pevtsova street, 13, Omsk, 644043, Russia
E-mail: alexander.rybalov@gmail.com

Abstract. In this paper we present a generic polynomial algorithm for the problem about the sum of subset over Brandt semigroup with index set $\mathbb{N}$ over any finitely defined group with polynomially decidable word problem. The problem about the sum of subset is a classical combinatorial problem, studied for many decades. This problem is very popular in cryptography, where there are many cryptosystems based on it. Myasnikov, Nikolaev, and Ushakov in 2015 introduced analogs of the problem about the sum of subset for arbitrary groups (semigroups). There are Brandt semigroups for which this problem is NP-complete. Also we propose a polynomial generic algorithm for the problem of identity checking over all finite Brandt monoids $B_{1,n}$. This problem is co-NP-complete in the classical sense. Generic algorithms solve problems for almost all inputs with respect to some natural measures on the set of inputs. Supported by Russian Science Foundation, grant 18-71-10028.

1. Introduction
In this paper we study the generic complexity of two algorithmic problems over Brandt semigroups and monoids. These semigroups played central role in studying of identities over semigroups.

The first problem is the problem about the sum of subset, a classical combinatorial problem, studied for many decades. It can be formulated as follows. Given a set of natural numbers $A = \{a_1, \ldots, a_n\}$ and a natural number $S$. All numbers are written in binary coding. It is necessary to decide does exists a subset of set $A$, which sum is $S$. This problem is related to another classic optimization problem – the knapsack problem and it is sometimes considered as a special case of the knapsack problem. The problem about the sum of subset is very popular in cryptography, where cryptosystems [1, 2] based on it have been proposed. Myasnikov, Nikolaev, and Ushakov [4] introduced analogs of the knapsack and problem about the sum of subsets for arbitrary groups (semigroups). This is a generalization of the classic problems, since in the classical case the input data is taken from the group of integers $\mathbb{Z}$ with the addition operation given by the infinite system of generators $\{2^m, m = 0, 1, 2, \ldots\}$. They explored the computational complexity of these problems for various groups: polynomial solvability is proved for hyperbolic groups, NP-completeness is proved for Baumslag-Solitar groups.

We present a generic polynomial algorithm for the problem about the sum of subset over Brandt semigroup with index set $\mathbb{N}$ over any finitely defined group with polynomially decidable word problem. There are such Brandt semigroups for which this problem is NP-complete. For

1 Supported by Russian Science Foundation, grant 18-71-10028.
example, Brandt semigroup over the Baumslag-Solitar group $BS(1, 2)$. Since there is no effective polynomial algorithm for it provided $P \neq NP$, it is actual to apply the generic approach [3] to this algorithmic problem. In the frameworks of generic approach an algorithm solves a problem for almost all inputs with respect to some natural measures on the set of inputs. This treatment may be fruitful for hard in the classical sense algorithmic problems.

The second problem is the problem of identity checking over various finite structures. There are many results about computational complexity of this problem for finite groups, semigroups, fields and rings – see [5] for references. Polynomial decidability has been proven for the following finite structures:

(i) associative nilpotent rings,
(ii) nilpotent groups,
(iii) commutative semigroups,
(iv) aperiodic finite 0-simple semigroups,
(v) monoids with less than 6 elements.

The co-NP-completeness has been proven for the following finite structures:

(i) associative non-nilpotent rings,
(ii) non-solvable finite groups,
(iii) some classes of matrix semigroups,
(iv) 6-element Brandt monoid $B^1_2$,
(v) some 0-simple semigroups.

Special role here has the 6-element Brandt monoid $B^1_2$. This is the smallest monoid with hard problem of identity checking. In [5] a generic polynomial algorithm was proposed for finite monoids, which contains elements with period larger than 1. But it cannot work in the Brandt monoid $B^1_2$. In this paper we propose a polynomial generic algorithm for the problem of identity checking over all finite Brandt monoids $B^1_n$.

2. Generic algorithms

Suppose $I$ is the set of all possible inputs and $I_n$ is the set of all such inputs of size $n$. Define for a subset $A \subseteq I$ the sequence

$$\rho_n(A) = \frac{|A_n|}{|I_n|}, \quad n = 1, 2, 3, \ldots$$

where $A_n = A \cap I_n$. Here for a finite set $A$ we denote by $|A|$ its cardinality. We will call by asymptotic density of set $A$ the limit (in the case if it exists)

$$\rho(A) = \lim_{n \to \infty} \rho_n(A).$$

The set $A$ is generic if $\rho(A) = 1$ and negligible if $\rho(A) = 0$.

Algorithm $A : I \to J \cup \{?\}$ is called generic if

(i) algorithm $A$ stops on every input from the set $I$,
(ii) $\{x \in I : A(x) \neq ?\}$ is generic set.

We talk that generic algorithm $A : A \to B \cup \{?\}$ computes a function $f : A \to B$ if

$$\forall x \in A \ A(x) \neq ? \Rightarrow f(x) = A(x).$$

A problem of recognition of a set $A \subseteq I$ is decidable generically in polynomial time if there is a polynomial generic algorithm, computing the characteristic function of $A$. 
3. The Brandt semigroups and monoids

Let \( G \) be a group, \( I \) a set with at least 2 elements, and \( 0 \notin G \cup I \). Define a operation of multiplication on the set \( B(G, I) = I \times G \times I \cup \{0\} \) in the following manner:

\[
(i, g, j)(k, h, l) = \begin{cases} 
(i, gh, l), & \text{if } j = k, \\
0, & \text{otherwise,}
\end{cases}
\]

for all \( i, j, k, l \in I \) and all \( g, h \in G \). Also define \( 0x = 0 \) and \( x0 = 0 \) for all \( x \in B(G, I) \). The set \( B(G, I) \) with defined operation of multiplication is called the Brandt semigroup over the group \( G \) with index set \( I \).

We are interested in Brandt semigroup \( B(G, \mathbb{N}) \) with index set \( \mathbb{N} \) of natural numbers. Also we will use Brandt semigroup \( B(E, \mathbb{N}) \) over the trivial one-element group \( E \), which we will denote by \( B(\mathbb{N}) \) and its elements are just pairs \((i, j)\) of natural numbers. In addition, we define \( B_n \) as Brandt semigroup with index set \( \{1, 2, \ldots, n\} \) over the trivial one-element group \( E \). Brandt monoid \( B_1^n \) is the Brandt semigroup \( B_n \) with joined unit 1.

4. The subset problem over Brandt semigroups

Now formulate the problem about the sum of subset over Brandt semigroup \( B(G, \mathbb{N}) \). For a given input

\[
\alpha = ((a_1, g_1, b_1), \ldots, (a_n, g_n, b_n); (a, g, b)),
\]

where \( a_i, b_i \leq n, |g_i| \leq n, i = 1, \ldots, n \) and \( a, b \leq n, |g| \leq n, \) decide do there exist \( 1 \leq i_1 < i_2 < \ldots < i_k \leq n \) such that

\[
(a_{i_1}, g_{i_1}, b_{i_1})(a_{i_2}, g_{i_2}, b_{i_2}) \ldots (a_{i_k}, g_{i_k}, b_{i_k}) = (a, g, b).
\]

The number \( n \) is the size of input \( \alpha \). It follows from the definition of multiplication in Brandt semigroup, that this equality holds if and only if hold the following both equalities:

\[
(a_{i_1}, b_{i_1})(a_{i_2}, b_{i_2}) \ldots (a_{i_k}, b_{i_k}) = (a, b)
\]

in semigroup \( B(\mathbb{N}) \) and

\[
g_{i_1}g_{i_2} \ldots g_{i_n} = g
\]

in the group \( G \). If \( G \) is finitely defined group with polynomially decidable word problem, then the problem about the sum of subset over \( B(G, \mathbb{N}) \) is in class NP.

**Lemma 1.** If the problem about the sum of subset over semigroup \( B(G, \mathbb{N}) \) is polynomially decidable, then the problem about the sum of subset over group \( G \) is polynomially decidable.

**Proof.** To decide the problem about the sum of subset over group \( G \) in polynomial time using polynomial algorithm for problem about the sum of subset over \( B(G, \mathbb{N}) \) we need to translate every input

\[
(g_1, g_2, \ldots, g_n; g)
\]

for \( G \) to the following equivalent input

\[
((1, g_1, 1), (1, g_2, 1), \ldots, (1, g_n, 1); (1, g, 1))
\]

for \( B(G, \mathbb{N}) \).

Since the problem about the sum of subset over the Baumslag-Solitar group \( BS(1, 2) \) is NP-complete, Lemma 1 implies that the problem about the sum of subset over Brandt semigroup \( B(BS(1, 2), \mathbb{N}) \) is NP-complete too.
Lemma 2. The problem about the sum of subset over $B(\mathbb{N})$ is decidable in polynomial time.

Proof. Algorithm $\mathcal{A}$, deciding the problem about the sum of subset over $B(\mathbb{N})$, computes on the input

$$\alpha = ((a_1, b_1), \ldots, (a_n, b_n); (a, b))$$

in the following manner.

(i) If there is a pair $(a_k, b_k)$ from $\alpha$ such that $a_k = a$ and $b_k = b$, terminate all working recursive calls of algorithm $\mathcal{A}$ and output $1$.

(ii) Find all $(a_{i_1}, b_{i_1}), \ldots, (a_{i_k}, b_{i_k})$ such that $a_{i_j} = a$, $j = 1, \ldots, k$.

(iii) If there are no such pairs, terminate the current recursive call of algorithm $\mathcal{A}$ and output "No solutions".

(iv) Run recursively algorithm $\mathcal{A}$ on every input

$$\alpha' = ((a_{i_{m+1}}, b_{i_{m+1}}), (a_{i_{m+2}}, b_{i_{m+2}}), \ldots, (a_n, b_n); (b_m, b))$$

on which algorithm $\mathcal{A}$ has not been run before.

(v) If all recursive calls of algorithm $\mathcal{A}$ stopped and gave out "No solutions", output $0$.

To show that algorithm $\mathcal{A}$ is polynomial, it is enough to get a polynomial bound on the number of recursive calls of $\mathcal{A}$. This number is not larger than the number of different inputs of type

$$\alpha' = ((a_m, b_m), (a_{m+1}, b_{m+1}), \ldots, (a_n, b_n); (a', b))$$

There are $n$ variants for the pairs before the last pair and $n$ variants to choose element $a'$. So we have the upper bound $n^2$.

Let

$$\alpha = ((a_1, b_1), \ldots, (a_n, b_n); (a, b))$$

be an input of the problem about the sum of subset over $B(\mathbb{N})$. A solution for problem about the sum of subset over $B(\mathbb{N})$ from input $\alpha$ is the list

$$\tau = (a_{i_1}, b_{i_1}), (a_{i_2}, b_{i_2}), \ldots, (a_{i_k}, b_{i_k}), \ 1 \leq i_1 < i_2 < \cdots < i_k \leq n,$$

such that

$$(a_{i_1}, b_{i_1})(a_{i_2}, b_{i_2}) \cdots (a_{i_k}, b_{i_k}) = (a, b).$$

It is obvious that it holds $a_{i_1} = a$, $b_{i_1} = a_{i_2}$, $\ldots$, $b_{i_{k-1}} = a_{i_k}$ and $b_{i_k} = b$. For an input $\alpha$ denote by $\tau(\alpha)$ the set of all solutions from $\alpha$.

For every $n$ let $I_n$ be the set of all inputs of size $n$ for the problem about the sum of subset over $B(\mathbb{N})$.

Lemma 3. $|I_n| = n^{2n+2}$.

Proof. Every input from $I_n$ has a form

$$\alpha = ((a_1, b_1), \ldots, (a_n, b_n); (a, b)).$$

There $n^2$ variants to choose every of $n + 1$ pairs in $\alpha$. So we have $n^{2n+2}$ inputs in $I_n$.

Now define the following set

$$A_n = \{ \alpha \in I_n : |\tau(\alpha)| > n^2 \}.$$
Lemma 4. For every $n$ it is hold

$$\frac{|A_n|}{|I_n|} < \frac{1}{n}.$$  

Proof. Consider the following number

$$\Sigma_n = \sum_{\alpha \in I_n} |\tau(\alpha)|.$$  

Suppose, by contradiction, that

$$\frac{|A_n|}{|I_n|} > \frac{1}{n},$$  

Whence

$$|A_n| > \frac{|I_n|}{n} = n^{2n+1}.$$  

And

$$\Sigma_n = \sum_{\alpha \in I_n} |\tau(\alpha)| > |A_n|n^2 > n^{2n+3}. \tag{1}$$  

For a solution $\tau$ denote by $\alpha(\tau)$ the set of all inputs, including solution $\tau$. Denote by $S_n$ the set of all possible solutions included in all inputs from $I_n$. Define the following number

$$\Omega_n = \sum_{\tau \in S_n} |\alpha(\tau)|.$$  

It is not hard to see that $\Sigma_n = \Omega_n$. Indeed, consider the list of all inputs of size $n$, which have solutions, in which every input $\alpha$ is duplicated $|\tau(\alpha)|$ times. The number of elements in this list is $\Sigma_n$. Re-group inputs in this list into groups associated by every possible solution $\tau$. The number of elements in this list is $\Omega_n$.

Denote by $S_{n,k}$ the set of all solutions from $S_n$ with $k$ elements. Split the number $\Omega_n$ in the following way:

$$\Omega_n = \Omega_{n,1} + \Omega_{n,2} + \ldots + \Omega_{n,n},$$

where

$$\Omega_{n,k} = \sum_{\tau \in S_{n,k}} |\alpha(\tau)|.$$  

Every input $\alpha \in \alpha(\tau)$, where $\tau \in S_{n,k}$ has a form

$$((a_1, b_1), \ldots, (a_n, b_n); (a, b))$$  

and includes a solution of form

$$(a, b_{i_1}), (b_{i_1}, b_{i_2}), (b_{i_2}, b_{i_3}), \ldots, (b_{i_{k-1}}, b), \ 1 \leq i_1 < i_2 < \ldots < i_k \leq n.$$  

There are $C^k_n$ variants to choose places $i_1, \ldots, i_k$ for the solution in the input. There are $n^{k-1}$ variants to choose elements $b_{i_1}, \ldots, b_{i_{k-1}}$ in the solution. There are $n^{2(n-k)}$ variants to choose another elements in the input. And there are $n^2$ variants for the pair $(a, b)$. So we can count

$$\Omega_{n,k} = C^k_n n^{k-1} n^{2(n-k)} n^2 = C^k_n n^{2n-k+1} < n^k n^{2n-k+1} = n^{2n+1}.$$  

And

$$\Sigma_n = \Omega_n = \Omega_{n,1} + \Omega_{n,2} + \ldots + \Omega_{n,n} < n^{2n+2}.$$  

But this contradicts the bound (1).
Lemma 5. There is a generic polynomial algorithm $B$, which for almost every input $\alpha$ of the problem about the sum of subset over $B(\mathbb{N})$ outputs $\tau(\alpha)$.

Proof. Algorithm $B$ will output solutions and will use the counter of solutions $k$. Algorithm $B$ computes on the input $\alpha = (a_1, \ldots, a_n; a)$ of size $n$ in the following manner. Before the start it assign $k = 0$.

(i) If $k = n^2$ terminate all recursive calls and output "?".
(ii) If $n = 1$ and $a_1 = a$ output $a_1$.
(iii) Run polynomial algorithm $A$ from Lemma 2 on inputs $\alpha_1 = (a_1, \ldots, a_n; a)$ and $\alpha_2 = (a_2, \ldots, a_n; a')$. Here $a' = (k, l)$, where $a_1 = (j, k)$ and $a = (j, l)$.
(iv) If $A(\alpha_1) = 1$ and $A(\alpha_2) = 0$, then output $B(\alpha_1)$.
(v) If $A(\alpha_1) = 0$ and $A(\alpha_2) = 1$, then output $(a_1, B(\alpha_2))$.
(vi) If $A(\alpha_1) = 1$ and $A(\alpha_2) = 1$, then increase $k := k + 1$ and output $B(\alpha_1)$ and $(a_1, B(\alpha_2))$.

This algorithm is polynomial because the number of branching is bounded by $n^2$ and depth of every branching is bounded by $n$. Genercity of algorithm $B$ follows from Lemma 4.

Now we have all tools to prove one of the main results.

Theorem 1. Let $G$ be a finitely defined group with word problem, decidable in polynomial time. Then the problem about the sum of subset over Brandt semigroup $B(G, \mathbb{N})$ is generically decidable in polynomial time.

Proof. A polynomial generic algorithm $C$, deciding the problem about the sum of subset over $B(G, \mathbb{N})$, computes on input $\alpha = ((a_1, g_1, b_1), \ldots, (a_n, g_n, b_n); (a, g, b))$ in the following manner.

(i) Run generic polynomial algorithm $B$ from Lemma 5 on the input

$$\alpha' = ((a_1, b_1), \ldots, (a_n, b_n); (a, b))$$

for the problem about the sum of subset over semigroup $B(\mathbb{N})$.

(ii) If $B(\alpha') = ?$, then output "?".

(iii) If algorithm $B$ output a list of all solutions for semigroup $B(\mathbb{N})$, then try all these solutions for the problem about the sum of subset over group $G$. It can be done in polynomial time, because the number of solutions is bounded by $n^2$ and the word problem for $G$ is polynomially decidable.

5. The problem of identity checking over Brandt monoids $B_n^1$

Suppose $M$ is a finite monoid and $X = \{x_1, x_2, \ldots\}$. Define $X_n = \{x_1, \ldots, x_n\}$. Any finite word $t$ over alphabet $X_n$, where $n = |t|$, is called term over monoid $M$. Define a function $t : M^n \to M$ for every term $t$ of size $n$ in the following manner. The value $t(a_1, \ldots, a_n) \in M^n$ can be computed by replacement of element $a_i$ instead of every variable $x_i$ in word $t$ for every $i = 1, \ldots, n$. An identity over monoid $M$ is such equality of terms $t_1 = t_2$, where $|t_1| = |t_2| = n$, that $t_1(a_1, \ldots, a_n) = t_2(a_1, \ldots, a_n)$ for all $(a_1, \ldots, a_n) \in M^n$. The problem of identity checking in monoid $M$ can be stated in the following manner. For a given pair of terms $t_1, t_2$ such that $|t_1| = |t_2| = n$, decide is it true that $t_1 = t_2$ is an identity in monoid $M$?

We will denote by $\mathcal{T}$ the set of terms and by $\mathcal{PT}$ the set of pairs of terms of the same size, i.e.

$$\mathcal{PT} = \{(t_1, t_2) : t_1, t_2 \in \mathcal{T}, |t_1| = |t_2|\}.$$
Lemma 6. For any \( n \) it holds \( \mathcal{PT}_n = n^{2n} \).

Proof. There are \( n \) variants to choose a variable from the set \( \{x_1, \ldots, x_n\} \) for any of \( n \) places in the left term. Therefore there are \( n^n \) variants to choose the left term. And the same number of variants to choose the right term. Finally we have \( n^{2n} \) variants. \( \Box \)

Following [6], for a term \( t(x_1, \ldots, x_n) \) define the following bipartite graph \( B(t) \). It has \( n \) upper vertices \( u(x_1), \ldots, u(x_n) \) and \( n \) bottom vertices \( b(x_1), \ldots, b(x_n) \). Each variable \( x_i \) corresponds to two vertices \( u(x_i) \) and \( b(x_i) \). Vertices \( u(x_i) \) and \( b(x_j) \) are connected by an edge in \( B(t) \) if and only if there is a subword \( x_i x_j \) in the term \( t \). Denote by

\[
C(B(t)) = \{C_1, C_2, \ldots, C_k\}
\]

the set of all connected components (given by sets of vertices) of the graph \( B(t) \). We will say that a connected component \( C_k \) of \( B(t) \) depends on variable \( x_i \) if \( C_k \) has vertex \( u(x_i) \) or vertex \( b(x_i) \).

For example the following picture is the graph \( B(t) \) of term \( t = x_1 x_3 x_1 x_2 x_5 x_6 x_5 \).

There are 8 connected components in this graph:

(i) \( \{u(x_1), b(x_2), b(x_3)\} \),
(ii) \( \{b(x_1), u(x_3)\} \),
(iii) \( \{u(x_2), b(x_5), u(x_6)\} \),
(iv) \( \{u(x_5), b(x_6)\} \),
(v) \( \{u(x_4)\} \),
(vi) \( \{b(x_4)\} \),
(vii) \( \{u(x_7)\} \),
(viii) \( \{b(x_7)\} \).

Seif and Szabo [6] (Proposition 4.12) proved that \( t_1 = t_2 \) is identity over Brandt semigroup \( B_n \) if and only if

(i) \( C(B(t_1)) = C(B(t_2)) \),
(ii) connected component of \( B(t_1) \), containing \( u(x_i) \), where \( x_i \) is the first letter of \( t_1 \), is the same as connected component of \( B(t_2) \), containing \( u(x_j) \), where \( x_j \) is the first letter of \( t_2 \),
(iii) connected component of \( B(t_1) \), containing \( b(x_k) \), where \( x_k \) is the last letter of \( t_1 \), is the same as connected component of \( B(t_2) \), containing \( b(x_l) \), where \( x_l \) is the last letter of \( t_2 \).
Note that sets $C(B(t_1))$ and $C(B(t_2))$ can be constructed in polynomial time, therefore the problem of identity checking in Brandt semigroup $B_n$ is polynomial time decidable. But, as proved by Klima and Seif (see [5] for references), the problem of identity checking over Brandt monoid $B_n^1$, $n \geq 2$, is co-NP-complete. Thus there is no effective polynomial algorithm for the problem of identity checking over Brandt monoid $B_n^1$, $n \geq 2$, provided P $\neq$ NP.

**Lemma 7.** The set $S \subset T$ of terms $t$ such that $B(t)$ has only connected components, depending on one variable, is negligible.

**Proof.** If a term $t$ contains more than one variable, then $t$ has a subword $x_i x_j$ and $B(t)$ contains an edge connecting $u(x_i)$ and $b(x_j)$. So $B(t)$ contains a connected component, depending on more than one variable. Therefore $S$ consists of all terms from one variables. It is easy to see that this set is negligible.

**Lemma 8.** For every term $t(x_1, \ldots, x_n)$ of size $n > 2$ there is no connected component in $C(B(t))$, depending on $n$ variables.

**Proof.** Suppose $n > 2$ and there is a connected component in $C(B(t))$, depending on all $n$ variables. It implies that some variable must be contained twice in $t$. Hence there is a variable $x_i$ that not contained in $t$ and every component in $C(B(t))$ does not depend on $x_i$.

Define the following set

$$\mathcal{EPT} = \{(t_1, t_2) \in \mathcal{PT} : C(B(t_1)) = C(B(t_2))\}.$$

**Lemma 9.** The set $\mathcal{EPT}$ is negligible.

**Proof.** Fix a term $t_1(x_1, \ldots, x_n)$ of size $n$. Suppose

$$C(B(t_1)) = \{C_1, C_2, \ldots, C_k\}.$$

Denote by $E(t_1)$ the set of terms $t_2(x_1, \ldots, x_n)$ of size $n$ such that $C(B(t_1)) = C(B(t_2))$.

Suppose $i \neq j$ and $i, j \leq n$. Define by $\varphi_{ij}$ the map from $\mathcal{PT}_n$ to $\mathcal{PT}_n$ which transpose variables $x_i$ and $x_j$ in every term. By Lemmas 7 and 8, we can assume that component $C_1$ depends on $m$ variables, where $2 \leq m < n$. Suppose $C_1$ depends on variable $x_i$ and does not depend on variable $x_j$. Denote by $I$ all such pairs $i, j$. Then it is not hard to see that:

(i) $E(t_1) \cap \varphi_{ij}(E(t_1)) = \emptyset$ for all $i, j \in I$,

(ii) moreover $\varphi_{ij}(E(t_1)) \cap \varphi_{kl}(E(t_1)) = \emptyset$ for different pairs $i, j \in I$ and $k, l \in I$,

(iii) the map $\varphi_{ij}$ is a bijection between $E(t_1)$ and $\varphi_{ij}(E(t_1))$, so $|E(t_1)| = |\varphi_{ij}(E(t_1))|$.

This implies

$$E(t_1) \cup \bigcup_{(i,j) \in I} \varphi_{ij}(E(t_1)) \subseteq T_n$$

and

$$|T_n| \geq |E(t_1)| + \sum_{(i,j) \in I} |\varphi_{ij}(E(t_1))| = (|I| + 1)|E(t_1)|. \tag{2}$$

Note that $|I| \geq \frac{n}{2}$. Indeed, consider two cases. If $m \leq \frac{n}{2}$, then we can fix $i$ such that $C_1$ depends on $x_i$ and we have at least $\frac{n}{2}$ variants to choose $j$ such that $C_1$ does not depend on $x_j$. So $|I| \geq \frac{n}{2}$ in this case. If $m > \frac{n}{2}$ then we can fix $j$ such that $C_1$ does not depend on $x_j$ and we have at least $\frac{n}{2}$ variants to choose $i$ such that $C_1$ depends on $x_i$. So again $|I| \geq \frac{n}{2}$.

Now (2) implies

$$|E(t_1)| \leq \frac{|T_n|}{\left(\frac{n}{2} + 1\right)}$$

and therefore the set $\mathcal{EPT}$ is negligible. \qed
Now we have all tools to prove the second result.

**Theorem 2.** The problem of identity checking over the Brandt monoid $B_n^1$ is decidable generically in polynomial time.

**Proof.** A polynomial algorithm $A$, deciding our problem, computes on an input $(t_1, t_2) \in PT$ in the following manner.

(i) Construct $B(t_1)$ and $B(t_2)$. These graphs can be constructed in polynomial time.

(ii) If $B(t_1) = B(t_2)$ then output "YES". In this case substitution of 1 in any subset of variables preserves equality of subgraphs of $B(t_1)$ and $B(t_2)$. So terms $t_1$ and $t_2$ are equivalent over $B_n^1$.

(iii) If $C(B(t_1)) \neq C(B(t_2))$ then output "NO". In this case terms $t_1$ and $t_2$ are not equivalent over $B_n$ by Theorem of Seif and Szabo [6]. Therefore they are not equivalent over $B_n^1$ too.

(iv) If $C(B(t_1)) = C(B(t_2))$ then output "?". This case is negligible by Lemma 9.

Lemma 9 implies, that our algorithm $A$ is generic.

[1] Hellman M and Merkle R 1978 Hiding Information and Signatures in Trapdoor Knapsacks *IEEE Transactions Information Theory* **24** (5) pp 525–530

[2] Chor B and Rivest R 1988 A knapsack-type public key cryptosystem based on arithmetic in finite fields *IEEE Transactions on Information Theory* **34** (5) pp 901–909

[3] Kapovich I and Myasnikov A and Schupp P and Shpilrain V 2003 Generic-case complexity, decision problems in group theory and random walks *Journal of Algebra* **264** (2) pp 665–694

[4] Miasnikov A and Nikolaev A and Ushakov A 2015 Knapsack problems in groups *Mathematics of Computation* **84** pp 987–1016

[5] Rybalov A 2020 A generic algorithm for the identity problem in finite groups and monoids *Journal of Physics: Conference Series* **1546** 012101 pp 1–6

[6] Seif S and Szabo C 2006 Computational complexity of checking identities in 0-simple semigroups and matrix semigroups over finite fields *Semigroup Forum* **72** (2) pp 207–222