Logical Methods in Computer Science
Volume 15, Issue 1, 2019, pp. 11:1–11:48
https://lmcs.episciences.org/

Submitted Mar. 22, 2018
Published Feb. 13, 2019

THE SUBPOWER MEMBERSHIP PROBLEM FOR FINITE ALGEBRAS WITH CUBE TERMS

ANDREI BULATOV [a], PETER MAYR [b], AND ÁGNES SZENDREI [b]

[a] School of Computing Science, Simon Fraser University, Burnaby BC, Canada V5A 1S6
e-mail address: abulatov@sfu.ca

[b] Department of Mathematics, University of Colorado, Boulder CO, USA, 80309-0395
e-mail address: {agnes.szendrei,peter.mayr}@colorado.edu

Abstract. The subalgebra membership problem is the problem of deciding if a given element belongs to an algebra given by a set of generators. This is one of the best established computational problems in algebra. We consider a variant of this problem, which is motivated by recent progress in the Constraint Satisfaction Problem, and is often referred to as the Subpower Membership Problem (SMP). In the SMP we are given a set of tuples in a direct product of algebras from a fixed finite set $\mathcal{K}$ of finite algebras, and are asked whether or not a given tuple belongs to the subalgebra of the direct product generated by a given set.

Our main result is that the subpower membership problem $\mathrm{SMP}(\mathcal{K})$ is in P if $\mathcal{K}$ is a finite set of finite algebras with a cube term, provided $\mathcal{K}$ is contained in a residually small variety. We also prove that for any finite set of finite algebras $\mathcal{K}$ in a variety with a cube term, each one of the problems $\mathrm{SMP}(\mathcal{K})$, $\mathrm{SMP}(\mathbb{HSK})$, and finding compact representations for subpowers in $\mathcal{K}$, is polynomial time reducible to any of the others, and the first two lie in NP.

1. Introduction

The subalgebra membership problem is one of the most well established and most thoroughly studied algorithmic problems in algebra. In this problem we are given a (finite) set of elements, called generators, of an algebraic structure (briefly: algebra), and we are asked to check whether or not a given element belongs to the subalgebra generated by the generators. This question has been studied within the general algebraic theory [18, 19, 13, 22], and it is also an important problem in computational group theory [25, 2, 14] where it is referred to as the subgroup membership problem. In addition to its theoretical interest, the subalgebra membership problem for general algebraic structures has found applications in some learning

Key words and phrases: membership problem, direct products, few subpowers, cube term, residually small variety.

This material is based upon work supported by the Austrian Science Fund (FWF) grant no. P24285, the National Science Foundation grant no. DMS 1500254, the Hungarian National Foundation for Scientific Research (OTKA) grant no. K104251 and K115518, and an NSERC Discovery Grant.

DOI:10.23638/LMCS-15(1:11)2019

algorithms [8, 4, 15], and the subgroup membership problem has found applications in areas such as cryptography [23, 24].

The subgroup membership problem was the first to be studied. The celebrated algorithm by Sims [25] decides, for any given set of permutations $a_1, \dots, a_k$ on a finite set X, whether or not a given permutation b belongs to the subgroup of the full symmetric group $\mathbf{S}_X$ generated by $a_1, \dots, a_k$. This algorithm has been widely used, and a number of improvements have been suggested [13, 17, 21]. While Sims' algorithm is quite efficient, Kozen [19] proved that if the group $\mathbf{S}_X$ is replaced by the full transformation semigroup on X, the subalgebra membership problem is PSPACE-complete.

Kozen [18] also considered the subalgebra membership problem for infinite algebras. Clearly, in this case the elements of the algebras in question must be efficiently represented. The natural approach is to consider algebras that are finitely presented by a finite set of generators and finitely many defining relations. Kozen suggested a polynomial time algorithm for this version of the problem.

Interest in the subalgebra membership problem has been renewed in relation to the study of the Constraint Satisfaction Problem (CSP). Recall that in a CSP the goal is to assign values from a certain domain to variables subject to specified constraints. Representation of the constraints is a very important issue in the CSP research, especially from the practical perspective, given the wide variety of constraint types considered. Unfortunately, if we do not restrict ourselves to a specific constraint type, choices are limited. The standard approach in the general case (in finite domain CSPs) is to represent constraints by constraint relations, that is, to give an explicit list of all the allowed combinations of values. This is clearly not space efficient, therefore more concise representations of general constraints are highly sought.

The algebraic approach to the CSP may offer a possible method for representing general constraints in a better way. Under this approach the constraint relations are considered along with their *polymorphisms*, that is, operations on the set of values which preserve all constraint relations. Hence, each constraint relation can be viewed as (the underlying set of) a subalgebra of a direct power of a certain finite algebra associated to the problem; namely, the algebra on the set of values whose operations are all polymorphisms described above. Therefore, knowing the polymorphisms, each constraint relation can be given by a set of generators, which is often much smaller than the constraint relation (i.e., the generated subalgebra) itself. In a number of cases the representation of a constraint relation by a set of generators is exponentially more concise (in the arity of the constraint) than listing all of its elements. This is especially valuable when the constraint relations have high arity. But even if we are not in one of those lucky cases, savings may be substantial.

The main difficulty with representations of this kind, however, is that although they are very space efficient, they can be inefficient in other aspects. For example, it is not clear whether or not it is possible to efficiently check that a tuple belongs to a constraint relation given by generators, but that check is unavoidable if we want to test if some mapping is a solution to a CSP. Thus we are led to the following variation of the subalgebra membership problem suggested by Willard [27]. Let $\mathbf{A}$ be a finite algebra with finitely many fundamental operations. The *subpower membership problem* (SMP) for $\mathbf{A}$ is the following decision problem:

SMP($\mathbf{A}$):

- INPUT: $a_1, \ldots, a_k, b \in \mathbf{A}^n$.
- QUESTION: Is b in the subalgebra of $\mathbf{A}^n$ generated by $\{a_1, \ldots, a_k\}$?

Observe that this problem is different from the problems studied in [13, 19], since the finite algebra $\mathbf{A}$ is fixed and subalgebras are generated in direct powers of $\mathbf{A}$.

A naive algorithm for solving SMP($\mathbf{A}$) is to compute and list all elements of the subalgebra $\mathbf{B}$ of $\mathbf{A}^n$ generated by $\{a_1, \ldots, a_k\}$, and then check whether b is in $\mathbf{B}$. Since the size of the input $a_1, \ldots, a_k, b$ is $(k+1)n$, while the best upper bound for the size of $\mathbf{B}$ is $|A|^n$, and for many algebras $\mathbf{A}$, some subalgebras $\mathbf{B}$ of $\mathbf{A}^n$ (say, $\mathbf{B} = \mathbf{A}^n$ itself) have generating sets of size k bounded by a polynomial of n, the time complexity of the naive algorithm is exponential. It turns out that without further restrictions on $\mathbf{A}$, one cannot do better than the naive algorithm: it follows from the main result of M. Kozik [20] that there exists a finite algebra $\mathbf{A}$ such that the subpower membership problem for $\mathbf{A}$ is EXPTIME-complete.

In contrast, for finite algebras $\mathbf{A}$ in many familiar classes, the subpower membership problem is in P, that is, there is a polynomial time algorithm for solving the problem. For example, if $\mathbf{A}$ is a group, then a variant of Sims' algorithm (cf. [13]) solves the problem in polynomial time. Other simple algorithms work if $\mathbf{A}$ is a finite lattice or a finite lattice with additional operations (use the Baker–Pixley Theorem [1]) or if $\mathbf{A}$ is a finite semilattice. A recent result of A. Bulatov, M. Kozik, P. Mayr, and M. Steindl [6] extends this observation on semilattices to any finite commutative semigroup $\mathbf{A}$ by showing that if $\mathbf{A}$ embeds into a direct product of a Clifford semigroup and a nilpotent semigroup, then the subpower membership problem for $\mathbf{A}$ is in P, and for all other finite commutative semigroups $\mathbf{A}$ the problem is NP-complete.

Extending the result on groups mentioned earlier R. Willard [27] proved that the subpower membership problem is in P for every finite algebra $\mathbf{A}$ that is an expansion of a group by multilinear operations. In particular, this is the case for every finite ring, finite module, and finite K-algebra. It is not known whether this statement remains true if the word 'multilinear' is omitted.

Returning to the CSP motivation of the subpower membership problem, the case in which representing constraint relations by generators is the most advantageous, is when the finite algebra $\mathbf{A}$ associated to the CSP has the property that all subalgebras of finite powers $\mathbf{A}^n$ of $\mathbf{A}$ admit small generating sets; here 'small' means that the size is bounded by $p(n)$ for some polynomial p depending only on $\mathbf{A}$. These algebras have been intensively studied within the theory of constraint satisfaction problems, and constitute one of the two known major classes of CSPs for which general polynomial time algorithms were found several years ago [5, 15]. It was proved in [3] that a finite algebra $\mathbf{A}$ has the aforementioned property on small generating sets for subalgebras of powers if and only if $\mathbf{A}$ *has few subpowers* in the following sense: there is a polynomial q depending only on $\mathbf{A}$ such that $\mathbf{A}^n$ has at most $2^{q(n)}$ subalgebras. This number is indeed 'small', as there exist finite algebras $\mathbf{A}$ (e.g., semilattices or unary algebras) for which the number of subalgebras of $\mathbf{A}^n$ is doubly exponential in n.

Algebras with few subpowers were also characterized in [3] by the existence of a *cube term* (to be defined in Section 2.2). Cube terms generalize *Mal'tsev terms* and *near unanimity terms*, which have been studied in algebra since the 60's and 70's, and more recently have played significant roles in isolating important properties of constraints in CSPs.

The principal research problem addressed in this paper is the following.

Question 1 [15]**.** Is the subpower membership problem for $\mathbf{A}$ in P if $\mathbf{A}$ is a finite algebra with a cube term?

The answer is known to be 'yes' if $\mathbf{A}$ has a near unanimity term (by the Baker–Pixley Theorem [1]), and the answer is not known in general, if $\mathbf{A}$ has a Mal'tsev term; in fact, as we mentioned earlier, the answer is not known even if $\mathbf{A}$ is a finite group expanded by further operations. We will now briefly discuss what the main difficulty seems to be.

Recall from [3] that if a finite algebra $\mathbf{A}$ has a cube term, then for every subalgebra $\mathbf{B}$ of any finite power $\mathbf{A}^n$ of $\mathbf{A}$ *there exists* a sort of canonical set of generators, called a *compact representation* in [5, 3, 15] or a *frame* in [10]. Compact representations played a crucial role in the CSP applications of finite algebras $\mathbf{A}$ with cube terms (see [15]). For the subpower membership problem SMP($\mathbf{A}$), a natural approach to answering the question "Is b in the subalgebra $\mathbf{B}$ of $\mathbf{A}^n$ generated by $a_1, \dots, a_k$?" would be to first find a compact representation for $\mathbf{B}$, and then use it to answer the question. The main components of a compact representation for $\mathbf{B}$ are *witnesses for forks.* For a coordinate position $i \in \{1, \dots, n\}$ and for two elements $c, d \in A$, the tuples $(c_1, \dots, c_n), (d_1, \dots, d_n)$ in $\mathbf{B}$ are said to witness that the pair (c, d) is a fork in $\mathbf{B}$ in position i if $c_1 = d_1, \dots, c_{i-1} = d_{i-1}$ and $c_i = c, d_i = d$. It turns out (see, e.g., Theorem 4.16 and the paragraph preceding Lemma 4.11) that in order to give a positive answer to Question 1 it would be sufficient to be able to decide in polynomial time — using only the specified generators $a_1, \dots, a_k$ — whether or not a given pair $(c, d) \in A^2$ is a fork in $\mathbf{B}$ in a given position i (and if it is, find witnesses for it in $\mathbf{B}$). This, however, appears to be a significant problem; cf. Theorems 4.12, 4.13, and 4.16.

Question 1 will remain unresolved in this paper, but we will prove (see Theorem 6.5) that the answer to Question 1 is YES in an important special case, namely when $\mathbf{A}$ belongs to a residually small variety (to be defined in Section 2.3). For a finite algebra $\mathbf{A}$ with a cube term this condition is equivalent to the requirement that there exists a natural number c such that every algebra in the variety generated by $\mathbf{A}$ is a subalgebra of a product of algebras of size less than c. For example, a finite algebra $\mathbf{A}$ belongs to a residually small variety if it is a lattice, a module, a group with abelian Sylow subgroups, or a commutative ring with 1 whose Jacobson radical J satisfies $J^2 = 0$. Although subpower membership for all finite groups, rings and lattices is already known to be tractable by various algorithms, we give a new unified approach that can handle all of them and also more general algebras in residually small varieties. So our algorithm is a first step towards finding a polynomial time algorithm for all algebras with a cube term.

The proof of Theorem 6.5 does not use compact representations; rather, it relies on a structure theorem proved in [16] for the subalgebras of finite powers of an algebra $\mathbf{A}$ with a cube term (or equivalently, parallelogram term). The application of this structure theorem leads us to considering subalgebras of finite products $\mathbf{S}_1 \times \cdots \times \mathbf{S}_n$ where the factors $\mathbf{S}_1, \dots, \mathbf{S}_n$ come from the finite collection $\mathbb{HSA}$ of homomorphic images of subalgebras of $\mathbf{A}$. As a consequence, it is natural for us to expand the scope of the *subpower membership problem*, and define it for any finite set $\mathcal{K}$ of finite algebras as follows:

SMP($\mathcal{K}$):

- INPUT: $a_1, \dots, a_k, b \in \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ with $\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$.
- QUESTION: Is b in the subalgebra of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ generated by $\{a_1, \dots, a_k\}$?

There is an important issue, which is raised by our passage from SMP($\mathbf{A}$) to SMP($\mathbb{HSA}$): Does the subpower membership problem get harder when $\mathbf{A}$ is replaced by the set $\mathbb{SA}$ of all its subalgebras or by the set $\mathbb{HA}$ of all its homomorphic images? It is easy to see that for

any finite algebra $\mathbf{A}$, SMP($\mathbf{A}$) and SMP($\mathbb{S}\mathbf{A}$) are essentially the same problem. However, this is not the case for homomorphic images. A surprising result of M. Steindl [26] shows that there exists a 10-element semigroup $\mathbf{S}$ with a 9-element quotient semigroup $\overline{\mathbf{S}}$ such that SMP($\mathbf{S}$) is in P while SMP($\overline{\mathbf{S}}$) is NP-complete. This shows that the problem SMP($\mathbb{H}\mathbb{S}\mathbf{A}$) may be harder than SMP($\mathbf{A}$) (provided $\mathsf{P} \neq \mathsf{NP}$), and therefore poses the following question for us:

Question 2. Are the problems SMP($\mathcal{K}$) and SMP($\mathbb{H}\mathbb{S}\mathcal{K}$) polynomial time equivalent if $\mathcal{K}$ is a finite set of finite algebras in a variety with a cube term?

We will prove (see Theorem 4.10) that the answer to Question 2 is YES. The proof uses the techniques of compact representations mentioned above. We will also show that, given a finite set of finite algebras $\mathcal{K}$ in a variety with a cube term, the subpower membership problem SMP($\mathcal{K}$) and the problem of finding compact representations for subalgebras of products of algebras in $\mathcal{K}$ are polynomial time reducible to each other (Theorem 4.16), and the two problems are in NP and $\mathsf{FP}^{\mathsf{NP}}$, respectively (Theorems 4.13 and 4.16).

Our results also yield new information on another problem that is closely related to SMP, namely the *subpower intersection problem* for a finite set $\mathcal{K}$ of finite algebras which is defined as follows:

SIP($\mathcal{K}$):

- INPUT: $a_1, \dots, a_k, b_1, \dots, b_\ell \in \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ with $\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$.
- OUTPUT: Generators for the intersection of the subalgebras of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ generated by $\{a_1, \dots, a_k\}$ and by $\{b_1, \dots, b_\ell\}$.

Assume $\mathcal{K}$ is a finite set of finite algebras in a variety with a cube term. Then it turns out that SIP($\mathcal{K}$) has a polynomial time reduction to SMP($\mathcal{K}$). Indeed, given compact representations for two subalgebras $\mathbf{B}$, $\mathbf{C}$ of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ with $\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$, Dalmau's algorithm [15] yields a compact representation for $\mathbf{B} \cap \mathbf{C}$ in polynomial time. Hence SIP($\mathcal{K}$) reduces to finding compact representations, which is polynomial time equivalent to SMP($\mathcal{K}$) (Theorem 4.16). For the converse, we do not know whether SMP($\mathcal{K}$) reduces to SIP($\mathcal{K}$) in general. However, it clearly does for idempotent algebras because for any elements $a_1, \dots, a_k, b$ in $\mathbf{A}_1 \times \cdots \times \mathbf{A}_k$, the subalgebra generated by $\{a_1, \dots, a_k\}$ contains b if and only if that subalgebra has a nonempty intersection with $\{b\}$, the subalgebra generated by b. Hence for a finite set of finite algebras $\mathcal{K}$ in an idempotent variety with a cube term the problems SMP($\mathcal{K}$), SIP($\mathcal{K}$), and finding compact representations are all polynomial time reducible to each other.

2. Preliminaries

In this section we summarize the concepts and facts from universal algebra that will be used throughout the paper. For more details the reader is referred to [7].

For every natural number m, we will use the notation $[m]$ for the set $\{1, 2, \dots, m\}$. The collection of all k-element subsets of a set S will be denoted by $\binom{S}{k}$.

2.1. **Algebras and varieties.** An *algebraic language* is a set F of function symbols, together with a mapping $F \to \{0, 1, \dots\}$ which assigns an *arity* k_f to every symbol $f \in F$. An *algebraic structure* (or briefly *algebra*) *in the language* F is a pair $\mathbf{A} = (A, F^{\mathbf{A}})$ where A is a nonempty set, called the *universe* of $\mathbf{A}$, and $F^{\mathbf{A}}$ is a set of operations $f^{\mathbf{A}} \colon A^{k_f} \to A$ on A, indexed by elements of F, called the *(basic) operations* of $\mathbf{A}$. We will say that $\mathbf{A} = (A; F^{\mathbf{A}})$ is a *finite algebra* if A is finite, and $\mathbf{A}$ has a *finite language* if F is finite.

Let $\mathbf{A}$ be an algebra in the language F. If t is a *term* in the language F (as defined in first order logic), we will write $t(x_1, \dots, x_n)$ to indicate that all variables occurring in t are among $x_1, \dots, x_n$ (but some of the variables $x_1, \dots, x_n$ may not occur in t). We may refer to a term $t(x_1, \dots, x_n)$ as an *n-ary term*. Each such term t induces an n-ary *term operation* $t^{\mathbf{A}} \colon A^n \to A$ of $\mathbf{A}$. Thus, the term operations of $\mathbf{A}$ are exactly those operations that can be obtained from the basic operations of $\mathbf{A}$ and from projection operations by composition. A function $g \colon A^n \to A$ is called an n-ary *polynomial operation* of $\mathbf{A}$ if it has the form $g(x_1, \dots, x_n) = t^{\mathbf{A}}(x_1, \dots, x_n, c_1, \dots, c_k)$ for some $(n + k)$-ary term operation $t^{\mathbf{A}}$ and some elements $c_1, \dots, c_k$ of $\mathbf{A}$.

As in the last two paragraphs, algebras will usually be denoted by boldface letters, and their universes by the same letters in italics. However, we will omit the superscript $\mathbf{A}$ from $F^{\mathbf{A}}$, $f^{\mathbf{A}}$, and $t^{\mathbf{A}}$ whenever there is no danger of confusion.

Terms in a language F encode all computations that are possible in any algebra $\mathbf{A}$ of the language F. Since we are interested in the complexity of certain computations, we will now discuss what measure of complexity we will use for terms. Assuming that the language F is finite, and hence there is a constant bound on the arities of all symbols in F, the complexity of a term t can be captured by the number of function symbols in t, or equivalently, by the number of nodes in the term tree of t. This number will be referred to as *the length of* t. If such a term t is short, say, is polynomially long in some parameter, then it can be efficiently evaluated in the straightforward way in any algebra, for any given input. However, even if the term itself is long, it may admit a more concise representation by an *F-circuit*, i.e., a circuit with gates whose types correspond to the symbols in F. Note that a circuit may evaluate identical subterms of t only once, and pass on the value to several gates in the circuit. This is why we will consider the number of gates in such a circuit to measure the amount of computation needed to evaluate t, which may be exponentially smaller than the length of t. More formally, let $\mathsf{Circ}_F(t)$ denote a minimal F-circuit representing t, that is, no two gates compute identical subterms. We define the *size of* $\mathsf{Circ}_F(t)$ to be the number of gates in $\mathsf{Circ}_F(t)$, and will use this number to measure the time needed to evaluate t. For $F = \{f\}$ a singleton, we also abbreviate $\{f\}$-circuit to f-circuit. Further we will omit the reference to F and write $\mathsf{Circ}(t)$ when the set of gate types is clear from context.

Let $\mathbf{A}, \mathbf{B}$, and $\mathbf{A}_j$ $(j \in J)$ be algebras in the same language F. A mapping $\varphi \colon A \to B$ is said to be a *homomorphism* $\varphi \colon \mathbf{A} \to \mathbf{B}$ if

$$\varphi(f^{\mathbf{A}}(a_1, \dots, a_{k_f})) = f^{\mathbf{B}}(\varphi(a_1), \dots, \varphi(a_{k_f})) \quad \text{for all } f \in F \text{ and } a_1, \dots, a_{k_f} \in A.$$

An invertible (i.e., bijective) homomorphism is called an *isomorphism*. We say that $\mathbf{B}$ is a *homomorphic image* of $\mathbf{A}$ if there exists an onto homomorphism $\mathbf{A} \to \mathbf{B}$. Furthermore, we say that $\mathbf{B}$ is a *subalgebra* of $\mathbf{A}$, and write $\mathbf{B} \leq \mathbf{A}$, if $B \subseteq A$ and the inclusion map $B \to A$ is a homomorphism $\mathbf{B} \to \mathbf{A}$; equivalently, $\mathbf{B}$ is a subalgebra of $\mathbf{A}$ iff $B \subseteq A$ and $f^{\mathbf{B}}$ is the restriction of $f^{\mathbf{A}}$ to the set B for all $f \in F$. The *direct product* $\prod_{j \in J} \mathbf{A}_j$ of the algebras $\mathbf{A}_j$ $(j \in J)$ is the unique algebra $\mathbf{A}$ with universe $A := \prod_{j \in J} A_j$ for which all projection maps $\mathrm{proj}_\ell \colon A \to A_\ell$, $(a_j)_{j \in J} \mapsto a_\ell$ $(\ell \in J)$ are homomorphisms; equivalently, $\mathbf{A}$ is the direct

product of the algebras $\mathbf{A}_j$ ($j \in J$) iff $\mathbf{A}$ has universe $A := \prod_{j\in J} A_j$, and its operations $f^{\mathbf{A}}$ act coordinatewise, via $f^{\mathbf{A}_j}$ in coordinate $j \in J$, for all $f \in F$. A subalgebra $\mathbf{B}$ of a direct product $\prod_{j\in J} \mathbf{A}_j$ will be called a *subdirect subalgebra* if $\mathrm{proj}_\ell(B) = A_\ell$ for every $\ell \in J$.

A *congruence* of $\mathbf{A}$ is the kernel of a homomorphism with domain $\mathbf{A}$; equivalently, a congruence of $\mathbf{A}$ is an equivalence relation on A that is invariant under the operations of $\mathbf{A}$. If ϑ is a congruence of $\mathbf{A}$, then there is a unique algebra, denoted $\mathbf{A}/\vartheta$, whose universe is the set A/ϑ of equivalence classes of ϑ, and whose operations are defined so that the natural map $\nu\colon A \to A/\vartheta$, $a \mapsto a/\vartheta$ is a homomorphism $\mathbf{A} \to \mathbf{A}/\vartheta$. Here a/ϑ denotes the equivalence class of ϑ containing a. The algebra $\mathbf{A}/\vartheta$ is called a *quotient algebra* of $\mathbf{A}$. By the first isomorphism theorem ([7, Theorem II.6.12]), if $\varphi\colon \mathbf{A} \to \mathbf{B}$ is an onto homomorphism with kernel ϑ, then $\mathbf{B}$ is isomorphic to $\mathbf{A}/\vartheta$.

For any algebra $\mathbf{A}$, the congruences of $\mathbf{A}$ form a lattice with respect to inclusion, which is called the *congruence lattice* of $\mathbf{A}$, and is denoted by $\mathrm{Con}(\mathbf{A})$. The *meet* operation $\wedge$ of $\mathrm{Con}(\mathbf{A})$ is simply the set-theoretic intersection of congruences, and the *join* operation $\vee$ is the transitive closure of the union of congruences. The top and bottom elements of $\mathrm{Con}(\mathbf{A})$ are denoted 1 and 0, respectively. For $\alpha, \beta \in \mathrm{Con}(\mathbf{A})$ with $\alpha \leq \beta$ we define the *interval* $[\![\, \alpha, \beta \,]\!]$ by $[\![\, \alpha, \beta \,]\!] := \{\gamma \in \mathrm{Con}(\mathbf{A}) : \alpha \leq \gamma \leq \beta\}$.

A congruence $\vartheta \in \mathrm{Con}(\mathbf{A})$ is said to be *completely meet irreducible* if for any family $\{\vartheta_j : j \in J\}$ of congruences of $\mathbf{A}$, $\vartheta = \bigwedge_{j\in J} \vartheta_j$ implies that $\vartheta = \vartheta_j$ for some $j \in J$. If ϑ is completely meet irreducible and $\vartheta \neq 1$, then ϑ has a unique *upper cover* ϑ^+ in $\mathrm{Con}(\mathbf{A})$, that is, there is a unique congruence $\vartheta^+ \in \mathrm{Con}(\mathbf{A})$ such that $\vartheta < \vartheta^+$ and $[\![\, \vartheta, \vartheta^+ \,]\!] = \{\vartheta, \vartheta^+\}$. We will use the notation $\mathrm{Irr}(\mathbf{A})$ for the set of all completely meet irreducible congruences of $\mathbf{A}$, excluding 1. An algebra $\mathbf{A}$ is said to be *subdirectly irreducible* if $0\,(\neq 1)$ is completely meet irreducible in $\mathrm{Con}(\mathbf{A})$. In this case the unique minimal non-trivial congruence 0^+ is called the *monolith* of $\mathbf{A}$. It follows from the second isomorphism theorem ([7, Theorem II.6.15]) that the subdirectly irreducible quotients of $\mathbf{A}$ are exactly the algebras $\mathbf{A}/\sigma$, $\sigma \in \mathrm{Irr}(\mathbf{A})$.

Let ϑ be a congruence of an algebra $\mathbf{A}$. We will often write $a \equiv_\vartheta b$ instead of $(a, b) \in \vartheta$. If $\mathbf{B}$ is a subalgebra of $\mathbf{A}$, we will say that $\mathbf{B}$ is *saturated with respect to* ϑ, or $\mathbf{B}$ is a *ϑ-saturated subalgebra* of $\mathbf{A}$, if $b \in B$ and $b \equiv_\vartheta a$ imply $a \in B$ for all $a \in A$. In other words, $\mathbf{B}$ is ϑ-saturated if and only if its universe is a union of ϑ-classes of $\mathbf{A}$. For an arbitrary subalgebra $\mathbf{B}$ of $\mathbf{A}$ there exists a smallest ϑ-saturated subalgebra of $\mathbf{A}$ that contains $\mathbf{B}$, which we denote by $\mathbf{B}[\vartheta]$; the universe of $\mathbf{B}[\vartheta]$ is $B[\vartheta] := \bigcup_{b\in B} b/\vartheta$. Denoting the restrictions of ϑ to $\mathbf{B}$ and $\mathbf{B}[\vartheta]$ by $\vartheta_{\mathbf{B}}$ and $\vartheta_{\mathbf{B}[\vartheta]}$, respectively, we have by the third isomorphism theorem ([7, Theorem II.6.18]) that the map $\mathbf{B}/\vartheta_{\mathbf{B}} \to \mathbf{B}[\vartheta]/\vartheta_{\mathbf{B}[\vartheta]}$, $b/\vartheta_{\mathbf{B}} \mapsto b/\vartheta_{\mathbf{B}[\vartheta]} (= b/\vartheta)$ is an isomorphism.

For a product $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ of algebras and for any set $I \subseteq [n]$, the projection homomorphism

$$\mathbf{A}_1 \times \cdots \times \mathbf{A}_n = \prod_{i\in[n]} \mathbf{A}_i \to \prod_{i\in I} \mathbf{A}_i, \qquad (a_i)_{i\in[n]} \mapsto (a_i)_{i\in I}$$

will be denoted by proj_I. For a subalgebra $\mathbf{B}$ or for an element b of $\prod_{i\in[n]} \mathbf{A}_i$, we will write $\mathbf{B}|_I$ or $b|_I$ for $\mathrm{proj}_I(\mathbf{B})$ or $\mathrm{proj}_I(b)$, respectively. If $I = \{j_1, \ldots, j_k\}$, then the notation $|_{\{j_1,\ldots,j_k\}}$ will be simplified to $|_{j_1,\ldots,j_k}$.

For any class $\mathcal{K}$ of algebras in the same language, $\mathbb{H}\mathcal{K}$, $\mathbb{S}\mathcal{K}$, and $\mathbb{P}\mathcal{K}$ denote the classes of all homomorphic images, subalgebras, and direct products of members of $\mathcal{K}$, respectively.

A *variety* is a class of algebras in the same language that is closed under taking direct products, subalgebras, and homomorphic images. By Tarski's Theorem ([7, Theorem II.9.5]),

the smallest variety containing a class $\mathcal{K}$ of algebras in the same language equals $\mathbb{HSP}\mathcal{K}$. Birkhoff's Theorem ([7, Theorem 11.9]) characterizes varieties as classes of algebras defined by identities. In more detail, an *identity* in a language F is an expression of the form $s(x_1,\dots,x_n) = t(x_1,\dots,x_n)$ where $s(x_1,\dots,x_n)$ and $t(x_1,\dots,x_n)$ are terms in the language F. An algebra $\mathbf{A}$ in the language F *satisfies an identity* $s(x_1,\dots,x_n) = t(x_1,\dots,x_n)$ if the n-ary term operations $s^{\mathbf{A}}$ and $t^{\mathbf{A}}$ are equal. Birkhoff's Theorem is the statement that for a fixed language, a class $\mathcal{V}$ of algebras is a variety if and only if there exists a set Σ of identities such that an algebra lies in $\mathcal{V}$ if and only if it satisfies the identities in Σ.

Occasionally, when we want to describe sets of identities where the identities have a similar form, it will be convenient to represent the identities in matrix form as follows. For any class $\mathcal{K}$ of algebras in the same language, and for any term $t = t(x_1,\dots,x_n)$, if M is an $m \times n$ matrix of variables and $\vec{v}$ is an $m \times 1$ matrix of variables,

$$\mathcal{K} \models t(M) = \vec{v} \tag{2.1}$$

will denote that the m identities represented by the rows in (2.1) are true in $\mathcal{K}$. For example,

$$\mathcal{K} \models t\begin{pmatrix} x & x & y \\ y & x & x \end{pmatrix} = \begin{pmatrix} y \\ y \end{pmatrix} \tag{2.2}$$

expresses that the identities $t(x,x,y) = y$ and $t(y,x,x) = y$ hold in $\mathcal{K}$, that is, t is a *Mal'tsev* term for $\mathcal{K}$.

2.2. **Cube Terms and Parallelogram Terms.** Let us fix an integer d (> 1). A d-*cube term* for $\mathcal{K}$ is a term t satisfying a set of identities of the form (2.1) in two variables x, y, where M is a matrix with d rows such that every column of M contains at least one x, and $\vec{v}$ consists of y's only. As (2.2) shows, a Mal'tsev term is a 2-cube term.

As we mentioned in the introduction, cube terms were introduced in [3] to show that a finite algebra $\mathbf{A}$ has few subpowers if and only if $\mathbf{A}$ has a cube term. More manageable terms that are equivalent to cube terms (e.g., edge terms, star terms) were also found in [3]. In this paper we will use another family of equivalent terms, called parallelogram terms, which were introduced in [16].

Let m and n be positive integers and let $\mathsf{d} = m + n$. An (m,n)-*parallelogram term* for $\mathcal{K}$ is a $(\mathsf{d}+3)$-ary term $P_{m,n}$ such that

$$\mathcal{K} \models P_{m,n}\left(\begin{array}{ccc|ccccccccc} x & x & y & z & y & \cdots & y & y & \cdots & y & y \\ x & x & y & y & z & & y & y & & y & y \\ & \vdots & & \vdots & & \ddots & & & & & \vdots \\ x & x & y & y & y & & z & y & & y & y \\ \hline y & x & x & y & y & & y & z & & y & y \\ & \vdots & & \vdots & & & & & \ddots & & \vdots \\ y & x & x & y & y & & y & y & & z & y \\ y & x & x & y & y & \cdots & y & y & \cdots & y & z \end{array}\right) = \begin{pmatrix} y \\ y \\ \vdots \\ y \\ y \\ \vdots \\ y \\ y \end{pmatrix}. \tag{2.3}$$

Here the rightmost block of variables is a $\mathsf{d} \times \mathsf{d}$ array, the upper left block is $m \times 3$ and the lower left block is $n \times 3$.

It is easy to see from these definitions that an (m,n)-parallelogram term that is independent of its last d variables is a Mal'tsev term, and an (m,n)-parallelogram term that is independent of its first 3 variables is a d-ary near unanimity term.

The theorem below summarizes the facts we will need later on about cube terms and parallelogram terms.

Theorem 2.1 (See [3, 16])**.** *Let $\mathcal{V}$ be a variety, and let $\mathsf{d}\,(>1)$ be an integer.*

(1) *The following conditions are equivalent:*
   (a) *$\mathcal{V}$ has a d-cube term,*
   (b) *$\mathcal{V}$ has an (m,n)-parallelogram term for all $m,n\geq 1$ with $m+n=\mathsf{d}$,*
   (c) *$\mathcal{V}$ has an (m,n)-parallelogram term for some $m,n\geq 1$ with $m+n=\mathsf{d}$.*

(2) *If $\mathcal{V}$ has a cube term, then $\mathcal{V}$ is congruence modular.*

In statement (2) the phrase $\mathcal{V}$ is *congruence modular* means that the congruence lattice of every algebra in $\mathcal{V}$ is *modular*; see, [11]. The equivalence of the conditions (a)–(c) in statement (1) follows by combining results from [3, Theorem 4.4] and [16, Theorem 3.5]. Proofs for statement (2) can be found in [3, Theorem 2.7] and [9, Theorem 3.2].

In view of the equivalence of conditions (a) and (c) in Theorem 2.1, when we consider classes of algebras with a d-cube term, we will work with a $(1,\mathsf{d}-1)$-parallelogram term $P=P_{1,\mathsf{d}-1}$, and we will also use the following terms derived from P:

$$\begin{aligned} s(x_1,\dots,x_{\mathsf{d}}) &:= P(x_1,x_2,x_2,x_1,\dots,x_{\mathsf{d}}), \\ p(x,u,y) &:= P(x,u,y,x,y,\dots,y). \end{aligned} \tag{2.4}$$

For any class $\mathcal{K}$ of algebras with a $(1,\mathsf{d}-1)$-parallelogram term P one can easily deduce from the $(1,\mathsf{d}-1)$-parallelogram identities that

$$\begin{aligned} \mathcal{K} \;\models\; & y = p(x,x,y), \\ & p(x,y,y) = s(x,y,y,\dots,y), \\ & s(y,x,y,\dots,y) = y, \\ & \qquad\vdots \\ & s(y,y,y,\dots,x) = y. \end{aligned} \tag{2.5}$$

To simplify notation, we also define

$$x^y := p(x,y,y)\ (= s(x,y,\dots,y)), \tag{2.6}$$

and

$$s^\ell(x_1,\dots,x_{\mathsf{d}}) := s(s^{\ell-1}(x_1,x_2\dots,x_{\mathsf{d}}),x_2,\dots,x_{\mathsf{d}}) \quad \text{for all } \ell\geq 1, \tag{2.7}$$

where $s^0 := x_1$. So, $s^1 = s$ and s^ℓ is the ℓ-th iterate of s in the first variable.

2.3. **Congruence Modular Varieties: the Commutator and Residual Smallness.** Let $\mathcal{V}$ be an arbitrary congruence modular variety. There is a well-behaved commutator operation $[\ ,\ ]$ on the congruence lattices of algebras in any such variety $\mathcal{V}$, which extends — and shares many important properties of — the group theoretic commutator for normal subgroups of groups. For the definition and basic properties of the commutator operation in $\mathcal{V}$ the reader is referred to [11]. A congruence $\alpha\in\mathrm{Con}(\mathbf{A})$ of an algebra $\mathbf{A}\in\mathcal{V}$ is called *abelian* if $[\alpha,\alpha]=0$, and the *centralizer* of a congruence $\alpha\in\mathrm{Con}(\mathbf{A})$, denoted $(0:\alpha)$, is the largest congruence $\gamma\in\mathrm{Con}(\mathbf{A})$ such that $[\alpha,\gamma]=0$.

Recall that $\mathcal{V}$ has a *difference term* (see [11, Theorem 5.5]), which we will denote by d. In the last two sections of this paper we will need some properties of abelian congruences, which can be summarized informally as follows: the difference term d induces abelian groups on

the blocks of all abelian congruences α of all algebras $\mathbf{A} \in \mathcal{V}$; moreover, the term operations of $\mathbf{A}$ are 'linear between the blocks' of α with respect to these abelian groups. The theorem below gives a more precise formulation of these facts.

Theorem 2.2 [11, Section 9]**.** *Let $\mathcal{V}$ be a congruence modular variety with a difference term d, let $\mathbf{A} \in \mathcal{V}$, and let α be an abelian congruence of $\mathbf{A}$.*

(1) *For every $o \in A$, the α-class containing o forms an abelian group $(o/\alpha; +_o, -_o, o)$ with zero element o for the operations $+_o$ and $-_o$ defined by*

$$x +_o y := d(x, o, y) \quad \text{and} \quad -_o x := d(o, x, o) \quad \text{for all } x, y \in o/\alpha.$$

(2) *For every term $g(x_1, \dots, x_k)$ in the language of $\mathcal{V}$, for arbitrary elements $o_1, \dots, o_k, o \in A$ such that $g(o_1, \dots, o_k) \equiv_\alpha o$, and for any tuple $(a_1, \dots, a_k) \in (o_1/\alpha) \times \cdots \times (o_k/\alpha)$,*

$$g(a_1, a_2, \dots, a_k) = g(a_1, o_2, \dots, o_k) +_o g(o_1, a_2, o_3, \dots, o_k) +_o \dots \\ +_o g(o_1, \dots, o_{k-1}, a_k) -_o (k-1) g(o_1, o_2, \dots, o_k).$$

We will refer to the abelian groups described in statement (1) as the *induced abelian groups* on the α-classes of $\mathbf{A}$.

For a cardinal c, a variety $\mathcal{V}$ is called *residually less than c* if every subdirectly irreducible algebra in $\mathcal{V}$ has cardinality $< c$; $\mathcal{V}$ is called *residually small* if it is residually less than some cardinal.

Theorem 2.3 [11]**.** *If $\mathbf{A}$ is a finite algebra that generates a congruence modular variety $\mathcal{V}(\mathbf{A})$, then the following conditions are equivalent:*

(1) *$\mathcal{V}(\mathbf{A})$ is residually small.*
(2) *$\mathcal{V}(\mathbf{A})$ is residuall less than some natural number.*
(3) *For every subdirectly irreducible algebra $\mathbf{S} \in \mathbb{HS}(\mathbf{A})$ with abelian monolith μ, the centralizer $(0 : \mu)$ of μ is an abelian congruence of $\mathbf{S}$.*

Proof. By [11, Theorem 10.15], (1) and (2) are both equivalent to the condition that the commutator identity $[x \wedge y, y] = x \wedge [y, y]$ holds in the congruence lattice of every subalgebra of $\mathbf{A}$. The latter is equivalent to (3) by [12, p. 422]. □

Now let $\mathcal{K} = \{\mathbf{A}_1, \dots, \mathbf{A}_n\}$ be a finite set of finite algebras in a congruence modular variety, and let $\mathcal{V}(\mathcal{K})$ be the variety generated by $\mathcal{K}$. Then $\mathcal{V}(\mathcal{K})$ is the join of the varieties $\mathcal{V}(\mathbf{A}_i)$ for $i \leq n$. Hence, by [11, Theorem 11.1], $\mathcal{V}(\mathcal{K})$ is residually small iff $\mathcal{V}(\mathbf{A}_i)$ is residually small for all $i \leq n$. Thus we get the following corollary.

Corollary 2.4. *Let $\mathcal{K}$ be a finite set of finite algebras in a congruence modular variety. The variety generated by $\mathcal{K}$ is residually small if and only if for every subdirectly irreducible algebra $\mathbf{S} \in \mathbb{HS}(\mathcal{K})$ with abelian monolith μ, the centralizer $(0 : \mu)$ of μ is an abelian congruence of $\mathbf{S}$.*

In conclusion of this section, we introduce a slightly technical notion, *similarity*, which is an important equivalence relation on the class of subdirectly irreducible algebras in a congruence modular variety and will play a role in Sections 5–6. However, a reader of this paper may safely regard the definition of similarity as a 'black box', since similarity only occurs as a result of using the structure theorem from [16] (see Theorem 5.2), and in the algorithms in Section 6, similarity is applied only to a finite set of finite subdirectly irreducible algebras, therefore checking similarity between them does not affect the complexity of the algorithms.

To define similarity, let $\mathcal{V}$ be a congruence modular variety. *Similarity* is a binary relation defined on the class of subdirectly irreducible algebras in $\mathcal{V}$. The definition may be found in [11, Definition 10.7], and shows that similarity is an equivalence relation. To decide whether two subdirectly irreducible algebras in $\mathcal{V}$ are similar it is more convenient to use the following characterization given in [11, Theorem 10.8]: two subdirectly irreducible algebras $\mathbf{B}, \mathbf{C} \in \mathcal{V}$ are similar if and only if there exists an algebra $\mathbf{E} \in \mathcal{V}$ (which can be taken to be a subdirect subalgebra of $\mathbf{B} \times \mathbf{C}$) and there exist congruences $\beta, \gamma, \delta, \varepsilon \in \mathrm{Con}(\mathbf{E})$ such that $\mathbf{E}/\beta \cong \mathbf{B}$, $\mathbf{E}/\gamma \cong \mathbf{C}$ and there is a projectivity $[\![\beta, \beta^+]\!] \searrow [\![\delta, \varepsilon]\!] \nearrow [\![\gamma, \gamma^+]\!]$ in $\mathrm{Con}(\mathbf{E})$, where β^+ and γ^+ are the unique upper covers of β and γ, respectively. Here $[\![\beta, \beta^+]\!] \searrow [\![\delta, \varepsilon]\!] \nearrow [\![\gamma, \gamma^+]\!]$ denotes that $\beta \wedge \varepsilon = \delta, \beta \vee \varepsilon = \beta^+$ and $\varepsilon \wedge \gamma = \delta, \varepsilon \vee \gamma = \gamma^+$.

3. Compact Representations

Throughout this section we will use the following global assumptions:

Assumption 3.1.

- $\mathcal{V}$ *is a variety with a* d*-cube term (*$\mathsf{d} > 1$*),*
- P *is a* $(1, \mathsf{d}-1)$*-parallelogram term in* $\mathcal{V}$ *(the existence of such a term is ensured by Theorem 2.1), and*
- $s(x_1, \dots, x_\mathsf{d})$*,* $p(x, u, y)$*,* x^y*, and* $s^\ell(x_1, \dots, x_\mathsf{d})$ *are the terms defined in* (2.4) *and* (2.6)–(2.7)*.*

We will not assume that the algebras we are considering are finite, or have a finite language. Therefore, unless finiteness is assumed explicitly, all statements and results hold for arbitrary algebras.

Terms in the language of $\mathcal{V}$ which can be expressed using P only, will be referred to as *P-terms*. For example, the terms $s(x_1, \dots, x_\mathsf{d})$, $p(x, u, y)$, x^y, and $s^\ell(x_1, \dots, x_\mathsf{d})$ are P-terms. By a *P-subalgebra* of an algebra $\mathbf{A} \in \mathcal{V}$ we mean a subalgebra of the reduct of $\mathbf{A}$ to the language $\{P\}$. We will say that an algebra $\mathbf{A} \in \mathcal{V}$ is *P-generated* by $R \subseteq A$ if R is a generating set for the reduct of $\mathbf{A}$ to the language $\{P\}$; or equivalently, if every element of $\mathbf{A}$ is of the form $t(r_1, \dots, r_m)$ for some $m \geq 0$, some elements $r_1, \dots, r_m \in R$, and some m-ary P-term t. The P-subalgebra of an algebra $\mathbf{A} \in \mathcal{V}$ generated by a set $S\,(\subseteq A)$ will be denoted by $\langle S \rangle_P$. Recall that circuit representations of P-terms are called *P-circuits*, as they use only gates of type P.

Now we will introduce a variant of the concept of 'compact representation' from [3]. One difference is that we will use a less restrictive notion of 'fork' than 'minority fork', because we want to avoid assuming finiteness of the algebras considered unless finiteness is necessary for the conclusions. Another difference is that we will consider subalgebras of products of algebras, rather than subalgebras of powers of a single algebra. Let $\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{V}$, and let $B \subseteq A_1 \times \cdots \times A_n$. For $i \in [n]$ and $\gamma, \delta \in A_i$ we will say that (γ, δ) is a *fork in the i-th coordinate of B* if there exist $b, \widehat{b} \in B$ such that

$$b|_{[i-1]} = \widehat{b}|_{[i-1]} \quad \text{and} \quad b|_i = \gamma, \;\; \widehat{b}|_i = \delta. \tag{3.1}$$

The set of all forks in the i-th coordinate of B will be denoted by $\textsc{fork}_i(B)$. Tuples $b, \widehat{b} \in B$ satisfying (3.1) will be referred to as *witnesses* for the fork $(\gamma, \delta) \in \textsc{fork}_i(B)$. For each i and B as above and for every positive integer e, we define

$$\textsc{fork}_i^e(B) := \{(\gamma, \delta^{\gamma^e}) : (\gamma, \delta) \in \textsc{fork}_i(B)\}$$

where δ^{γ^e} is a short notation for $(\dots((\delta^\gamma)^\gamma)\dots)^\gamma$ with e occurrences of γ. The elements of $\mathrm{FORK}_i^e(B)$ will be called *e-derived forks in the i-th coordinate of B*. In the case when $e = 1$ we will use the notation $\mathrm{FORK}_i'(B)$ instead of $\mathrm{FORK}_i^1(B)$, and will call the elements of $\mathrm{FORK}_i'(B)$ *derived forks in the i-th coordinate of B*. The next lemma shows that derived forks are indeed forks, and they are 'transferable', which does not hold for forks in general.

Lemma 3.2. *If $\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{V}$ and $\mathbf{B}$ is a P-subalgebra of $\mathbf{A}_1 \times \dots \times \mathbf{A}_n$, then*

(1) $\mathrm{FORK}_i(B) \supseteq \mathrm{FORK}_i'(B) \supseteq \dots \supseteq \mathrm{FORK}_i^e(B) \supseteq \mathrm{FORK}_i^{e+1}(B) \supseteq \dots$ *for all $i \in [n]$ and $e \geq 1$; moreover,*

(2) *for every $(\gamma, \delta) \in \mathrm{FORK}_i'(B)$ and for every $b \in B$ with $b|_i = \gamma$, there is an element $\widehat{b} \in B$ such that (3.1) holds, that is, b and $\widehat{b}$ witness that $(\gamma, \delta) \in \mathrm{FORK}_i(B)$.*

Proof. For (2) let $(\gamma, \delta) \in \mathrm{FORK}_i'(B)$ and let $b \in B$ with $b|_i = \gamma$. Then there exists $(\gamma, \beta) \in \mathrm{FORK}_i(B)$ such that $\delta = \beta^\gamma$. Let $c, \widehat{c} \in B$ be witnesses for $(\gamma, \beta) \in \mathrm{FORK}_i(B)$; thus, $c|_{[i-1]} = \widehat{c}|_{[i-1]}$ and $c|_i = \gamma$, $\widehat{c}|_i = \beta$. It follows from the identities in (2.5) that for the element $\widehat{b} := p(\widehat{c}, c, b) \in B$ we have

$$\widehat{b}|_{[i-1]} = p(\widehat{c}|_{[i-1]}, c|_{[i-1]}, b|_{[i-1]}) = b|_{[i-1]},$$

and

$$\widehat{b}|_i = p(\widehat{c}|_i, c|_i, b|_i) = p(\beta, \gamma, \gamma) = \beta^\gamma = \delta.$$

This proves (2), and also the inclusion $\mathrm{FORK}_i(B) \supseteq \mathrm{FORK}_i'(B)$ in (1). The inclusion $\mathrm{FORK}_i^e(B) \supseteq \mathrm{FORK}_i^{e+1}(B)$ for any $e \geq 1$ follows by the same argument, using $\delta = \beta^{\gamma^{e+1}} = (\beta^{\gamma^e})^\gamma$ and β^{γ^e} in place of $\delta = \beta^\gamma$ and β. □

The following 'weak transitivity rule' for forks will also be useful.

Lemma 3.3. *Let $\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{V}$, and let $\mathbf{B}$ be a P-subalgebra of $\mathbf{A}_1 \times \dots \times \mathbf{A}_n$. If (γ, δ) and (β, δ) are forks in $\mathrm{FORK}_i(B)$ witnessed by the pairs $(v, \widehat{v})$ and $(u, \widehat{u})$ in $\mathbf{B}$, respectively, then the pair*

$$\Big(p(p(v, \widehat{v}, \widehat{u}), p(v, \widehat{v}, \widehat{v}), v), \quad p(u, v, v)\Big) \tag{3.2}$$

in $\mathbf{B}$ is a witness for the fork $(\gamma, \beta^\gamma) \in \mathrm{FORK}_i(B)$.

Proof. The choice of $u, \widehat{u}, v, \widehat{v}$ implies that $u|_{[i-1]} = \widehat{u}|_{[i-1]}$, $v|_{[i-1]} = \widehat{v}|_{[i-1]}$, and $u|_i = \beta$, $\widehat{u}|_i = \delta = \widehat{v}|_i$, $v|_i = \gamma$. Hence,

$$\begin{aligned} &p(p(v, \widehat{v}, \widehat{u}), p(v, \widehat{v}, \widehat{v}), v)|_{[i-1]} \\ &\quad = p(p(v|_{[i-1]}, \widehat{v}|_{[i-1]}, \widehat{u}|_{[i-1]}), p(v|_{[i-1]}, \widehat{v}|_{[i-1]}, \widehat{v}|_{[i-1]}), v|_{[i-1]}) \\ &\quad = p(u|_{[i-1]}, v|_{[i-1]}, v|_{[i-1]}) = p(u, v, v)|_{[i-1]} \end{aligned}$$

and

$$\begin{aligned} p(p(v, \widehat{v}, \widehat{u}), p(v, \widehat{v}, \widehat{v}), v)|_i &= p(p(\gamma, \delta, \delta), p(\gamma, \delta, \delta), \gamma) = \gamma, \\ p(u, v, v)|_i &= p(\beta, \gamma, \gamma) = \beta^\gamma. \end{aligned}$$

Clearly, $u, \widehat{u}, v, \widehat{v} \in \mathbf{B}$ implies that the pair (3.2) also lies in $\mathbf{B}$, so the proof of the lemma is complete. □

Definition 3.4. For two sets $B, R \subseteq A_1 \times \dots \times A_n$, we will say that R is a (d, e)-*representation for B* if the following three conditions are met:

(1) $R \subseteq B$;

(2) $R|_I = B|_I$ for all $I \subseteq [n]$ with $|I| < \mathsf{d}$;
(3) $\text{FORK}_i(R) \supseteq \text{FORK}_i^e(B)$ for all $i \in [n]$.
If $e = 1$ and the parameter d of the cube term of $\mathcal{V}$ is clear from the context, then reference to (d, e) will be omitted.

In the special case when $\mathbf{A}_1, \dots, \mathbf{A}_n$ are members of a fixed finite set $\mathcal{K}$ of finite algebras in $\mathcal{V}$, and the maximum size of an algebra in $\mathcal{K}$ is $\mathsf{a}_\mathcal{K}$, then it is easy to see that every set $B \subseteq A_1 \times \cdots \times A_n$ has a (d, e)-representation R of size

$$|R| \leq \sum_{I \subseteq [n], |I| < d} \big|B|_I\big| + \sum_{i \in [n]} \text{FORK}_i^e(B) \leq \binom{n}{\mathsf{d}-1} \mathsf{a}_\mathcal{K}^{\mathsf{d}-1} + 2n\mathsf{a}_\mathcal{K}^2.$$

A (d, e)-representation R for B of size $|R| \leq \binom{n}{\mathsf{d}-1}\mathsf{a}_\mathcal{K}^{\mathsf{d}-1} + 2n\mathsf{a}_\mathcal{K}^2$ is called a *compact* (d, e)-*representation* for B.

Remark 3.5. If $\mathcal{K}$ is a finite set of finite algebras in $\mathcal{V}$, then there exists an $e \geq 1$ such that $\mathcal{K} \models (x^{y^e})^{y^e} = x^{y^e}$. For such an e, the terms $d(x, y) = y^{x^e}$, $p(z, y, x)^{x^{e-1}}$ and $s^e(x_1, \dots, x_\mathsf{d})$ satisfy the identities in [3, Lemma 2.13]. Consequently, for this e, our e-derived forks and (compact) (d, e)-representations are exactly the minority forks (called minority indices), and the (compact) representations with minority forks defined in [3, Definitions 3.1–3.2].

It was proved in [3, Lemma 3.4, Theorem 3.6] that for any subalgebra $\mathbf{B}$ of a finite power of a finite algebra $\mathbf{A} \in \mathcal{V}$, a compact representation (with minority forks) generates $\mathbf{B}$. The next theorem is essentially the same result, with several significant differences, which we will discuss in Remark 3.7 below.

Theorem 3.6. *Let e be a fixed positive integer. For every integer $n \geq \mathsf{d}$ there exists a P-term T_n satisfying the following conditions:*

(1) *If $\mathbf{B}$ is a P-subalgebra of a product $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ of finitely many algebras $\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{V}$, and R is a (d, e)-representation for $\mathbf{B}$, then every element of $\mathbf{B}$ is produced by a single application of T_n to some elements of R. Consequently, $\mathbf{B}$ is P-generated by R.*
(2) *The size of the P-circuit $\mathsf{Circ}(T_n)$ is $O(en^{\mathsf{d}+1})$, and there is an algorithm that runs in time $O(en^{\mathsf{d}+1})$, which outputs $\mathsf{Circ}(T_n)$ for any given e and n.*

Note that the P-terms T_n depend on the parameter e, but to simplify notation, we decided to suppress this dependence in the notation of T_n (and in the notation of the terms needed to build up T_n, see Lemma 3.8). In this paper we will use Theorem 3.6 for $e = 1$ and $e = 2$ only.

Remark 3.7. The main differences between [3, Lemma 3.4, Theorem 3.6] and Theorem 3.6 are as follows:

- As we explained at the beginning of this section, we consider subalgebras of direct products of finitely many different algebras rather than just subalgebras of finite direct powers of the same algebra. In addition, we don't assume that the algebras are finite, therefore we use a less restrictive notion of a fork than the notion used in [3], cf. Remark 3.5.
- Given e, we produce a P-term T_n for every n, which describes a uniform way for obtaining each element of a P-subalgebra $\mathbf{B}$ of a product $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ in one step from members of an arbitrary (d, e)-representation for $\mathbf{B}$. Since we are interested in algorithmic applications, the advantage of finding these terms T_n explicitly is that — via the minimal circuit representation of T_n — we can get a useful upper bound on the length of computations needed to generate elements of $\mathbf{B}$ from members of a (d, e)-representation for $\mathbf{B}$.

First we will prove the following lemma.

Lemma 3.8. *Let e be a fixed positive integer. For every integer $n \geq \mathsf{d}$ there exists a P-term $t_n = t_n(x, y, z, \overline{w_I})$, where $\overline{w_I} := (w_I)_{I \in \binom{[n]}{\mathsf{d}-1}}$ is a tuple of variables indexed by all $(\mathsf{d}-1)$-element subsets of $[n]$, such that the following hold:*

(1) *For every subset R of a product $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ with $\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{V}$, and for every tuple $b = (b_1, \dots, b_{n-1}, \gamma)$ in $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$, if*
   (a) *for each $I \in \binom{[n]}{\mathsf{d}-1}$ the set R contains a tuple b^I satisfying $b^I|_I = b|_I$, and*
   (b) *for some element $b' = (b_1, \dots, b_{n-1}, \beta)$ of the P-subalgebra $\mathbf{R}^* := \langle R \rangle_P$ of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$, the set R contains tuples $u = (u_1, \dots, u_{n-1}, \gamma)$ and $\widehat{u} = (u_1, \dots, u_{n-1}, \beta^{\gamma^e})$ which are witnesses for the fork $(\gamma, \beta^{\gamma^e}) \in \mathrm{FORK}_n(R)$,*

   then
$$b = t_n\big(b', \widehat{u}, u, \overline{b^I}\big) \quad \text{where} \quad \overline{b^I} := (b^I)_{I \in \binom{[n]}{\mathsf{d}-1}}, \tag{3.3}$$
   and therefore b is in $\mathbf{R}^*$.

(2) *The size of the P-circuit $\mathsf{Circ}(t_n)$ is $O(en^{\mathsf{d}})$, and there is an algorithm that runs in time $O(en^{\mathsf{d}})$, which outputs $\mathsf{Circ}(t_n)$ for any given e and n.*

Proof. We use a dynamic programming approach to build the tuple b and keep a record of the sequence of operations we perform so that we obtain the term t_n in the end. For every $\ell \in [n] \setminus [\mathsf{d}-2]$ and for every set $V' \in \binom{[\ell]}{\mathsf{d}-1}$ we construct a P-term $t_{\ell,V'}$ which 'approximates' t_n in the sense that for $t_{\ell,V'}$ in place of t_n the equality (3.3) holds in all coordinates in $V' \cup ([n] \setminus [\ell])$ (but may fail in other coordinates).

Claim 3.9. For $\ell = n, n-1, \dots, \mathsf{d}-1$ and for every set $V := V' \cup ([n] \setminus [\ell])$ with $V' \in \binom{[\ell]}{\mathsf{d}-1}$ there exists a P-term $t_{\ell,V'} = t_{\ell,V'}(x, y, z, \overline{w_I}^V)$ with $\overline{w_I}^V := (w_I)_{I \in \binom{V}{\mathsf{d}-1}}$ such that whenever the algebras $\mathbf{A}_1, \dots, \mathbf{A}_n$, $\mathbf{R}^*$ and the elements b, b^I $\big(I \in \binom{[n]}{\mathsf{d}-1}\big)$, b', u, $\widehat{u}$ satisfy the assumptions of Lemma 3.8 (1), we have that
$$b|_V = t_{\ell,V'}\big(b', \widehat{u}, u, \overline{b^I}^V\big)|_V \quad \text{where} \quad \overline{b^I}^V := (b^I)_{I \in \binom{V}{\mathsf{d}-1}}.$$

Proof of Claim 3.9. We proceed by induction on $n - \ell$. For $n = \ell$ we have that $V = V' \in \binom{[n]}{\mathsf{d}-1}$. So we can choose $t_{n,V'}(x, y, z, w_V) := w_V$, and our claim is trivial.

Assume now that $\ell < n$ and that our claim is true for $\ell + 1$. To prove the claim for ℓ, let $V = V' \cup ([n] \setminus [\ell])$ with $V' = \{i_1, \dots, i_{\mathsf{d}-1}\} \in \binom{[\ell]}{\mathsf{d}-1}$, $i_1 < \cdots < i_{\mathsf{d}-1}$. For $j \in [\mathsf{d}-1]$ let $V'_j = \{i_1, \dots, i_{j-1}, i_{j+1}, \dots, i_{\mathsf{d}-1}, \ell+1\}$ and $V_j := V \setminus \{i_j\} = V'_j \cup ([n] \setminus [\ell+1])$. We will prove that the P-term
$$\begin{aligned} t_{\ell,V'}\big(x, y, z, \overline{w_I}^V\big) := P\Big(& s^{e+1}\big(x, (t_{\ell+1,V'_j}(x, y, z, \overline{w_I}^{V_j}))_{j \in [\mathsf{d}-1]}\big), \\ & p\big(y, z, t_{\ell+1,V'_1}(x, y, z, \overline{w_I}^{V_1})\big), t_{\ell+1,V'_1}(x, y, z, \overline{w_I}^{V_1}), x, \big(t_{\ell+1,V'_j}(x, y, z, \overline{w_I}^{V_j})\big)_{j \in [\mathsf{d}-1]}\Big) \end{aligned} \tag{3.4}$$
has the desired properties, where the P-terms $t_{\ell+1,V'_j}$ $(j \in [\mathsf{d}-1])$ are supplied by the induction hypothesis. To prove this, let $\mathbf{A}_1, \dots, \mathbf{A}_n$, $\mathbf{R}^*$, and b, b^I $\big(I \in \binom{[n]}{\mathsf{d}-1}\big)$, b', u, $\widehat{u}$ satisfy the assumptions of Lemma 3.8 (1). By the induction hypothesis, the elements $b^{\ell+1,V'_j} := t_{\ell+1,V'_j}\big(b', \widehat{u}, u, \overline{b^I}^{V_j}\big)$ satisfy the condition $b^{\ell+1,V'_j}|_{V_j} = b|_{V_j}$ for all $j \in [\mathsf{d}-1]$; that

is, $b^{\ell+1,V'_j}|_V$ has the form

$$b^{\ell+1,V'_j}|_V = (b_{i_1}, \dots, b_{i_{j-1}}, \zeta_j, b_{i_{j+1}}, \dots, b_{i_{\mathsf{d}-1}}, b_{\ell+1}, \dots, b_{n-1}, \gamma)$$

for some element $\zeta_j \in A_{i_j}$ in the j-th coordinate. To compute $t_{\ell,V'}(b', \widehat{u}, u, \overline{b^I}^V)|_V$, we start with evaluating the first two arguments of P in the expression for $t_{\ell,V'}$ in (3.4).

Using the identities for s in (2.5) we get that

$$s\big(b', \big(t_{\ell+1,V'_j}(b', \widehat{u}, u, \overline{b^I}^{V_j})\big)_{j\in[\mathsf{d}-1]}\big)\big|_V = s(b'|_V, b^{\ell+1,V'_1}|_V, \dots, b^{\ell+1,V'_{\mathsf{d}-1}}|_V)$$
$$= s\begin{pmatrix} b_{i_1} & \zeta_1 & b_{i_1} & \dots & b_{i_1} \\ b_{i_2} & b_{i_2} & \zeta_2 & \dots & b_{i_2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ b_{i_{\mathsf{d}-1}} & b_{i_{\mathsf{d}-1}} & b_{i_{\mathsf{d}-1}} & \dots & \zeta_{\mathsf{d}-1} \\ b_{\ell+1} & b_{\ell+1} & b_{\ell+1} & \dots & b_{\ell+1} \\ \vdots & \vdots & \vdots & & \vdots \\ b_{n-1} & b_{n-1} & b_{n-1} & \dots & b_{n-1} \\ \beta & \gamma & \gamma & \dots & \gamma \end{pmatrix} = \begin{pmatrix} b_{i_1} \\ b_{i_2} \\ \vdots \\ b_{i_{\mathsf{d}-1}} \\ b_{\ell+1} \\ \vdots \\ b_{n-1} \\ \beta\gamma \end{pmatrix}.$$

By repeating the same computation e more times so that every time the tuple just obtained is placed in the first argument of s in the next computation, we obtain that

$$s^{e+1}\big(b', \big(t_{\ell+1,V'_j}(b', \widehat{u}, u, \overline{b^I}^{V_j})\big)_{j\in[\mathsf{d}-1]}\big)\big|_V = s^{e+1}(b'|_V, b^{\ell+1,V'_1}|_V, \dots, b^{\ell+1,V'_{\mathsf{d}-1}}|_V)$$
$$= \begin{pmatrix} b_{i_1} \\ b_{i_2} \\ \vdots \\ b_{i_{\mathsf{d}-1}} \\ b_{\ell+1} \\ \vdots \\ b_{n-1} \\ \beta\gamma^{e+1} \end{pmatrix}. \quad (3.5)$$

The identities for p in (2.5) yield that

$$p\big(\widehat{u}, u, t_{\ell+1,V'_1}(b', \widehat{u}, u, \overline{b^I}^{V_1})\big)|_V = p\big(\widehat{u}|_V, u|_V, b^{\ell+1,V'_1}|_V\big)$$
$$= p\begin{pmatrix} u_{i_1} & u_{i_1} & \zeta_1 \\ u_{i_2} & u_{i_2} & b_{i_2} \\ \vdots & \vdots & \vdots \\ u_{i_{\mathsf{d}-1}} & u_{i_{\mathsf{d}-1}} & b_{i_{\mathsf{d}-1}} \\ u_{\ell+1} & u_{\ell+1} & b_{\ell+1} \\ \vdots & \vdots & \vdots \\ u_{n-1} & u_{n-1} & b_{n-1} \\ \beta\gamma^e & \gamma & \gamma \end{pmatrix} = \begin{pmatrix} \zeta_1 \\ b_{i_2} \\ \vdots \\ b_{i_{\mathsf{d}-1}} \\ b_{\ell+1} \\ \vdots \\ b_{n-1} \\ \beta\gamma^{e+1} \end{pmatrix}. \quad (3.6)$$

Combining (3.5) and (3.6) with the definition of $t_{\ell,V'}$, and using the $(1,\mathsf{d}-1)$-parallelogram identities, we obtain that

$$t_{\ell,V'}(b',\widehat{u},u,\overline{b^I}^V)|_V$$
$$= P\begin{pmatrix} b_{i_1} & \zeta_1 & \zeta_1 & b_{i_1} & \zeta_1 & b_{i_1} & \dots & b_{i_1} \\ b_{i_2} & b_{i_2} & b_{i_2} & b_{i_2} & b_{i_2} & \zeta_2 & \dots & b_{i_2} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ b_{i_{\mathsf{d}-1}} & b_{i_{\mathsf{d}-1}} & b_{i_{\mathsf{d}-1}} & b_{i_{\mathsf{d}-1}} & b_{i_{\mathsf{d}-1}} & b_{i_{\mathsf{d}-1}} & \dots & \zeta_{\mathsf{d}-1} \\ b_{\ell+1} & b_{\ell+1} & b_{\ell+1} & b_{\ell+1} & b_{\ell+1} & b_{\ell+1} & \dots & b_{\ell+1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & & \vdots \\ b_{n-1} & b_{n-1} & b_{n-1} & b_{n-1} & b_{n-1} & b_{n-1} & \dots & b_{n-1} \\ \beta^{\gamma^{e+1}} & \beta^{\gamma^{e+1}} & \gamma & \beta & \gamma & \gamma & \dots & \gamma \end{pmatrix} = \begin{pmatrix} b_{i_1} \\ b_{i_2} \\ \vdots \\ b_{i_{\mathsf{d}-1}} \\ b_{\ell+1} \\ \vdots \\ b_{n-1} \\ \gamma \end{pmatrix} = b|_V.$$

This completes the proof of Claim 3.9. ◇

The term $t_n := t_{\mathsf{d}-1,[\mathsf{d}-1]}$ constructed in Claim 3.9 for

$$V = [n] = \{1,\dots,\mathsf{d}-1\} \cup ([n] \setminus [\mathsf{d}-1])$$

clearly satisfies the requirement in statement (1) of Lemma 3.8.

To estimate the size of $\mathsf{Circ}(t_n)$, note that the term t_n has $\sum_{\ell=\mathsf{d}-1}^{n-1}\binom{\ell}{\mathsf{d}-1} = O(n^{\mathsf{d}})$ distinct subterms of the form $t_{\ell,V'}$ where $\ell \in \{\mathsf{d}-1,\dots,n-1\}$ and $V' \in \binom{[\ell]}{\mathsf{d}-1}$. By (3.4), obtaining each one of these terms $t_{\ell,V'}$ from variables and subterms of the form $t_{\ell+1,W}$ requires one application of each of s^{e+1}, p, and P, that is, altogether $e+3$ applications of P. Hence $\mathsf{Circ}(t_n)$ has size $O(en^{\mathsf{d}})$. This proves the first part of statement (2) of Lemma 3.8. The second part is an immediate consequence of this estimate and the definition of t_n. □

Now let $n \geq \mathsf{d}$, and let $t_n = t_n(x,y,z,\overline{w_I})$ be the P-term from Lemma 3.8. The analogous P-terms for $\mathsf{d} \leq m \leq n$ are $t_m = t_m(x,y,z,\overline{w_I}^{[m]})$ with $\overline{w_I}^{[m]} := (w_I)_{I\in\binom{[m]}{\mathsf{d}-1}}$. We will use these terms to define new P-terms

$$T_m(z^{(\mathsf{d})},\widehat{z}^{(\mathsf{d})},\dots,z^{(m)},\widehat{z}^{(m)},\overline{w_I}^{[m]})$$

for each $m = \mathsf{d}-1,\mathsf{d},\dots,n$ by recursion as follows: $T_{\mathsf{d}-1}(w_{[\mathsf{d}-1]}) := w_{[\mathsf{d}-1]}$, and for all m with $\mathsf{d} \leq m \leq n$,

$$T_m(z^{(\mathsf{d})},\widehat{z}^{(\mathsf{d})},\dots,z^{(m)},\widehat{z}^{(m)},\overline{w_I}^{[m]})$$
$$:= t_m\big(T_{m-1}(z^{(\mathsf{d})},\widehat{z}^{(\mathsf{d})},\dots,z^{(m-1)},\widehat{z}^{(m-1)},\overline{w_I}^{[m-1]}),z^{(m)},\widehat{z}^{(m)},\overline{w_I}^{[m]}\big).$$

In particular, for $m=n$, the term is $T_n(z^{(\mathsf{d})},\widehat{z}^{(\mathsf{d})},\dots,z^{(n)},\widehat{z}^{(n)},\overline{w_I})$, because $\overline{w_I}^{[n]} = \overline{w_I}$.

Note that by this definition, T_n has $n-\mathsf{d}+1$ subterms of the form T_m $(m=\mathsf{d},\dots,n)$ other than the variable $T_{\mathsf{d}-1}$. Furthermore, each T_m is obtained from T_{m-1} by applying t_m to T_{m-1} and some variables. Since by Lemma 3.8, the P-circuit $\mathsf{Circ}(t_m)$ has size $O(em^{\mathsf{d}})$, we get that

$$\text{the } P\text{-circuit } \mathsf{Circ}(T_n) \text{ has size } O(en^{\mathsf{d}+1}). \tag{3.7}$$

Lemma 3.10. *Let e be a fixed positive integer. Assume that $\mathbf{A}_1,\dots,\mathbf{A}_n \in \mathcal{V}$ $(n \geq \mathsf{d})$, and let b be an element and R a subset of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ such that*

(a) *for each $I \in \binom{[n]}{\mathsf{d}-1}$ the set R contains a tuple b^I satisfying $b^I|_I = b|_I$, and*

(b) *for every m with $\mathsf{d} \leq m \leq n$ the set R contains tuples $u^{(m)}, \widehat{u}^{(m)}$ which are witnesses for the fork $(\gamma, \beta^{\gamma^e}) \in \mathrm{FORK}_m(R)$ where*

$$\gamma := b|_m \quad \text{and} \quad \beta := T_{m-1}(u^{(\mathsf{d})}, \widehat{u}^{(\mathsf{d})}, \dots, u^{(m-1)}, \widehat{u}^{(m-1)}, \overline{b^I}^{[m-1]})|_m. \tag{3.8}$$

Then the following equalities hold:

$$b|_{[m]} = T_m(u^{(\mathsf{d})}, \widehat{u}^{(\mathsf{d})}, \dots, u^{(m)}, \widehat{u}^{(m)}, \overline{b^I}^{[m]})|_{[m]} \quad \text{for all } m = \mathsf{d}-1, \mathsf{d}, \dots, n; \tag{3.9}$$

in particular,

$$b = T_n(u^{(\mathsf{d})}, \widehat{u}^{(\mathsf{d})}, \dots, u^{(n)}, \widehat{u}^{(n)}, \overline{b^I}). \tag{3.10}$$

Proof. Let $\mathbf{R}^*$ denote the P-subalgebra of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ generated by R. For $m = \mathsf{d}-1, \mathsf{d}, \dots, n$ let

$$b^{(m)} := T_m(u^{(\mathsf{d})}, \widehat{u}^{(\mathsf{d})}, \dots, u^{(m)}, \widehat{u}^{(m)}, \overline{b^I}^{[m]}).$$

We will proceed by induction to prove that (3.9) holds, that is, $b|_{[m]} = b^{(m)}|_{[m]}$ for all $m = \mathsf{d}-1, \mathsf{d}, \dots, n$. Then, for the case when $m = n$, (3.9) yields the equality (3.10).

To start the induction, let $m := \mathsf{d}-1$. Since $T_{\mathsf{d}-1}(w_{[\mathsf{d}-1]}) = w_{[\mathsf{d}-1]}$, we have that $b^{(\mathsf{d}-1)} = b^{[\mathsf{d}-1]}$, so $b|_{[\mathsf{d}-1]} = b^{(\mathsf{d}-1)}|_{[\mathsf{d}-1]}$ is clearly true by assumption (a).

Now assume that $m \geq \mathsf{d}$ and that the equality $b|_{[m-1]} = b^{(m-1)}|_{[m-1]}$ holds. Then the tuples $b|_{[m]} \in \mathbf{A}_1 \times \cdots \times \mathbf{A}_m$ and $b^{(m-1)}|_{[m]} \in \mathbf{R}^*|_{[m]}$ have the form $(b_1, \dots, b_{m-1}, \gamma)$ and $(b_1, \dots, b_{m-1}, \beta)$, respectively, where γ and β are defined by (3.8). Assumption (b) implies that R contains tuples $u^{(m)}, \widehat{u}^{(m)} \in R$ which are witnesses for the fork $(\gamma, \beta^{\gamma^e}) \in \mathrm{FORK}_m(R)$. Then $u^{(m)}|_{[m]}$ and $\widehat{u}^{(m)}|_{[m]}$ are in $R|_{[m]}$, and they have the form $u^{(m)}|_{[m]} = (u_1, \dots, u_{m-1}, \gamma)$ and $\widehat{u}^{(m)}|_{[m]} = (u_1, \dots, u_{m-1}, \beta^{\gamma^e})$ for some $u_i \in \mathbf{A}_i$ $(i \in [m-1])$. The fact that R satisfies assumption (a) also implies that the elements $b^I|_{[m]} \in R|_{[m]}$ have the property $(b^I|_{[m]})|_I = (b|_{[m]})|_I$ for all $I \in \binom{[m]}{\mathsf{d}-1}$.

This shows that the assumptions (a)–(b) of Lemma 3.8 (1) hold for

- the subset $R|_{[m]}$ and element $b|_{[m]} = (b_1, \dots, b_{m-1}, \gamma)$ of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_m$,
- the element $b^{(m-1)}|_{[m]} = (b_1, \dots, b_{m-1}, \beta)$ in $\mathbf{R}^*|_{[m]}$, and
- the elements $u^{(m)}|_{[m]} = (u_1, \dots, u_{m-1}, \gamma)$, $\widehat{u}^{(m)}|_{[m]} = (u_1, \dots, u_{m-1}, \beta^{\gamma^e})$, and $b^I|_{[m]}$ $\big(I \in \binom{[m]}{\mathsf{d}-1}\big)$ in $R|_{[m]}$.

Thus, Lemma 3.8 (1) — combined with the definitions of $b^{(m-1)}$ and T_m — implies that

$$\begin{aligned}
b|_{[m]} &= t_m(b^{(m-1)}|_{[m]}, \widehat{u}^{(m)}|_{[m]}, u^{(m)}|_{[m]}, \overline{b^I|_{[m]}}^{[m]}) \\
&= t_m(b^{(m-1)}, \widehat{u}^{(m)}, u^{(m)}, \overline{b^I}^{[m]})|_{[m]} \\
&= t_m\big(T_{m-1}(u^{(\mathsf{d})}, \widehat{u}^{(\mathsf{d})}, \dots, u^{(m-1)}, \widehat{u}^{(m-1)}, \overline{b^I}^{[m-1]}), \widehat{u}^{(m)}, u^{(m)}, \overline{b^I}^{[m]}\big)|_{[m]} \\
&= T_m(u^{(\mathsf{d})}, \widehat{u}^{(\mathsf{d})}, \dots, u^{(m)}, \widehat{u}^{(m)}, \overline{b^I}^{[m]})|_{[m]},
\end{aligned}$$

which is what we wanted to prove. □

Lemma 3.11. *Let e be a fixed positive integer, and let $\mathbf{B}$ be a P-subalgebra of a product $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ of finitely many algebras $\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{V}$. If R is a (d, e)-representation for $\mathbf{B}$, then for every element b of $\mathbf{B}$,*

(1) *there exist elements* b^I $\left(I \in \binom{[n]}{\mathsf{d}-1}\right)$ *and* $u^{(m)}, \widehat{u}^{(m)}$ $(m = \mathsf{d}, \dots, n)$ *in* R *such that conditions* (a)–(b) *of Lemma 3.10 are satisfied,*
(2) *hence the equality* $b = T_n(u^{(\mathsf{d})}, \widehat{u}^{(\mathsf{d})}, \dots, u^{(n)}, \widehat{u}^{(n)}, \overline{b^I})$ *in* (3.10) *holds for* b.

Proof. Let R be a (d, e)-representation for $\mathbf{B}$, and let $b \in B$. As before, let $\mathbf{R}^*$ denote the P-subalgebra of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_\mathbf{n}$ generated by R. Our statements will follow from Lemma 3.10 if we show that b and R satisfy the assumptions (a)–(b) of that lemma. Condition (a) clearly follows from our assumptions that $b \in B$ and R is a (d, e)-representation for $\mathbf{B}$. To verify condition (b) we proceed by induction on m to show that for every m $(\mathsf{d} \le m \le n)$

(b)$_m$ R contains tuples $u^{(m)}, \widehat{u}^{(m)}$ witnessing the fork $(\gamma, \beta^{\gamma^e}) \in \mathrm{FORK}_m(R)$ where γ and β are defined by (3.8).

Let $\mathsf{d} \le m \le n$, and assume that condition (b)$_i$ holds for $i = \mathsf{d}, \dots, m-1$; note that this assumption is vacuously true for the base case $m = \mathsf{d}$. Our goal is to show that (b)$_m$ also holds. Let

$$b^{(m-1)} := T_{m-1}(u^{(\mathsf{d})}, \widehat{u}^{(\mathsf{d})}, \dots, u^{(m-1)}, \widehat{u}^{(m-1)}, \overline{b^I}^{[m-1]}),$$

and let γ and β be defined by (3.8); that is, $\gamma = b|_m$ and $\beta = b^{(m-1)}|_m$. Since $b^{(m-1)}$ involves only the elements b^I $\left(I \in \binom{[m-1]}{\mathsf{d}-1}\right)$ and $u^{(\mathsf{d})}, \widehat{u}^{(\mathsf{d})}, \dots, u^{(m-1)}, \widehat{u}^{(m-1)}$ of R, and since our induction hypothesis ensures that these elements satisfy the assumptions of Lemma 3.10, we get from Lemma 3.10 that $b|_{[m-1]} = b^{(m-1)}|_{[m-1]}$. Here $b \in B$ and $b^{(m-1)} \in R^* \subseteq B$, therefore b and $b^{(m-1)}$ are witnesses in $\mathbf{B}$ for the fork $(\gamma, \beta) \in \mathrm{FORK}_m(B)$. Hence, by Lemma 3.2, $(\gamma, \beta^{\gamma^e}) \in \mathrm{FORK}^e_m(B)$. So, the fact that R is a (d, e)-representation for $\mathbf{B}$ implies that $(\gamma, \beta^{\gamma^e}) \in \mathrm{FORK}_m(R)$. Thus, (b)$_m$ holds, as we wanted to show. □

Proof of Theorem 3.6. Statement (1) of Theorem 3.6 is an immediate consequence of Lemma 3.11. Statement (2) of Theorem 3.6 follows directly from (3.7) and the definition of T_n. □

We close this section by discussing the following question: given a generating set for a subalgebra $\mathbf{B}$ of a product $\mathbf{B}_1 \times \cdots \times \mathbf{B}_n$ (in a variety with a cube term) and a product congruence $\theta = \theta_1 \times \cdots \times \theta_n$ of $\mathbf{B}_1 \times \cdots \times \mathbf{B}_n$ (where $\theta_i \in \mathrm{Con}(\mathbf{B}_i)$ for all $i \in [n]$), how can one construct a generating set for the θ-saturation $\mathbf{B}[\theta]$ of $\mathbf{B}$? This is a crucial step for giving a positive answer to Question 2 in the introduction, as we explain now.

Suppose that we are given a subalgebra $\mathbf{D}$ of $\mathbf{B}_1/\theta_1 \times \cdots \times \mathbf{B}_n/\theta_n$ by a generating set $\{c_1, \dots, c_k\}$, along with another element v of $\mathbf{B}_1/\theta_1 \times \cdots \times \mathbf{B}_n/\theta_n$, and our task is to decide whether or not $v \in \mathbf{D}$, but we are only allowed to do computations in $\mathbf{B}_1 \times \cdots \times \mathbf{B}_n$. For the product congruence $\theta = \theta_1 \times \cdots \times \theta_n$ on $\mathbf{B}_1 \times \cdots \times \mathbf{B}_n$, we can use the natural isomorphism

$$\mathbf{B}_1/\theta_1 \times \cdots \times \mathbf{B}_n/\theta_n \cong (\mathbf{B}_1 \times \cdots \times \mathbf{B}_n)/\theta, \quad (b_1/\theta_1, \dots, b_n/\theta_n) \mapsto (b_1, \dots, b_n)/\theta$$

to view the elements $c_1, \dots, c_k, v$ and the subalgebra $\mathbf{D}$ of $\mathbf{B}_1/\theta_1 \times \cdots \times \mathbf{B}_n/\theta_n$ as those of $(\mathbf{B}_1 \times \cdots \times \mathbf{B}_n)/\theta$. Now choose representatives $a_1, \dots, a_k, u\,(\in B_1 \times \cdots \times B_n)$ from the θ-classes $c_1, \dots, c_k, v$, respectively, and let $\mathbf{B}$ be the subalgebra of $\mathbf{B}_1 \times \cdots \times \mathbf{B}_n$ generated by the set $\{a_1, \dots, a_k\}$. Since the definition of $\mathbf{B}[\theta]$ in Section 2 implies that $\mathbf{B}[\theta]$ is the full inverse image of $\mathbf{D}$ under the natural homomorphism

$$\mathbf{B}_1 \times \cdots \times \mathbf{B}_n \twoheadrightarrow (\mathbf{B}_1 \times \cdots \times \mathbf{B}_n)/\theta \cong \mathbf{B}_1/\theta_1 \times \cdots \times \mathbf{B}_n/\theta_n,$$

we get that $v \in \mathbf{D}$ if and only if $u \in \mathbf{B}[\theta]$. Hence, we can decide whether v is in $\mathbf{D}$ by a subpower membership calculation in $\mathbf{B}_1 \times \cdots \times \mathbf{B}_n$, provided we can construct a generating

set for $\mathbf{B}[\theta]$ from the generating set $\{a_1, \dots, a_k\}$ of $\mathbf{B}$. We now give such a generating set explicitly.

Theorem 3.12. *Let $\mathbf{B}$ be a subalgebra of $\mathbf{B}_1 \times \cdots \times \mathbf{B}_n$ with $\mathbf{B}_1, \dots, \mathbf{B}_n \in \mathcal{V}$ ($n \geq \mathsf{d}$), let $\theta_i \in \mathrm{Con}(\mathbf{B}_i)$ for all $i \in [n]$, and let $\theta := \theta_1 \times \cdots \times \theta_n \in \mathrm{Con}(\mathbf{B}_1 \times \cdots \times \mathbf{B}_n)$. Assume that G is a generating set for $\mathbf{B}$, and $\Lambda, \Phi \subseteq B[\theta]$ satisfy the following conditions:*

(1) *For every $I \in \binom{[n]}{\mathsf{d}-1}$ and for every $\overline{b} \in B|_I$ and $\overline{c} \in B[\theta]|_I$ with $\overline{b} \equiv_{\theta|_I} \overline{c}$ there exist $b_{I,\overline{b}} \in B$ and $c_{I,\overline{c}} \in \Lambda$ such that*

$$b_{I,\overline{b}}|_I = \overline{b}, \quad c_{I,\overline{c}}|_I = \overline{c}, \quad \text{and} \quad b_{I,\overline{b}} \equiv_\theta c_{I,\overline{c}}.$$

(2) *For every $i \in [n]$ and $(\beta, \gamma) \in \theta_i$ with $\beta \in B|_i$, there exist $b_{i,\beta} \in B$ and $c_{i,\gamma} \in \Phi$ such that*

$$b_{i,\beta}|_i = \beta, \quad c_{i,\gamma}|_i = \gamma, \quad b_{i,\beta}|_{[i-1]} = c_{i,\gamma}|_{[i-1]}, \quad \text{and} \quad b_{i,\beta} \equiv_\theta c_{i,\gamma}.$$

Then $G \cup \Lambda \cup \Phi$ is a generating set for $\mathbf{B}[\theta]$.

Proof. Let $\mathbf{C}$ denote the subalgebra of $\mathbf{B}_1 \times \cdots \times \mathbf{B}_n$ generated by $G \cup \Lambda \cup \Phi$. Clearly, $\mathbf{B} \leq \mathbf{C} \leq \mathbf{B}[\theta]$. To show that $\mathbf{C} = \mathbf{B}[\theta]$, choose an arbitrary element d in $\mathbf{B}[\theta]$. Then there exists b in $\mathbf{B}$ such that $b \equiv_\theta d$. Using that Λ satisfies condition (1), choose and fix elements $b^I \in B$ and $d^I \in \Lambda$ such that

$$b^I|_I = b|_I, \quad d^I|_I = d|_I, \quad \text{and} \quad b^I \equiv_\theta d^I \quad \text{for each} \quad I \in \tbinom{[n]}{\mathsf{d}-1}.$$

We will use the P-terms in Lemma 3.10 with $e = 2$ to show that

$$d = T_n(v^{(\mathsf{d})}, \widehat{v}^{(\mathsf{d})}, \dots, v^{(n)}, \widehat{v}^{(n)}, \overline{d^I}^{[n]}) \tag{3.11}$$

holds for appropriately chosen elements $v^{(\mathsf{d})}, \widehat{v}^{(\mathsf{d})}, \dots, v^{(n)}, \widehat{v}^{(n)}$ in $\mathbf{C}$. Since all d^I are in Λ ($\subseteq \mathbf{C}$), this will show that $d \in \mathbf{C}$, and hence will complete the proof of the theorem.

By the definition of the term $T_{\mathsf{d}-1} := w^{[\mathsf{d}-1]}$ and by the choice of $b^{[\mathsf{d}-1]}$ and $d^{[\mathsf{d}-1]}$ we have that

$$b|_{[\mathsf{d}-1]} = b^{[\mathsf{d}-1]}|_{[\mathsf{d}-1]}, \quad d|_{[\mathsf{d}-1]} = d^{[\mathsf{d}-1]}|_{[\mathsf{d}-1]},$$

and

$$T_{\mathsf{d}-1}(b^{[\mathsf{d}-1]}) = b^{[\mathsf{d}-1]} \equiv_\theta d^{[\mathsf{d}-1]} = T_{\mathsf{d}-1}(d^{[\mathsf{d}-1]}).$$

Now we proceed by induction to show that for every $m = \mathsf{d}, \dots, n$ there exist tuples $u^{(\mathsf{d})}, \widehat{u}^{(\mathsf{d})}, \dots, u^{(m)}, \widehat{u}^{(m)}$ in $\mathbf{B}$ and tuples $v^{(\mathsf{d})}, \widehat{v}^{(\mathsf{d})}, \dots, v^{(m)}, \widehat{v}^{(m)}$ in $\mathbf{C}$ such that

$$b|_{[m]} = T_m(u^{(\mathsf{d})}, \widehat{u}^{(\mathsf{d})}, \dots, u^{(m)}, \widehat{u}^{(m)}, \overline{b^I}^{[m]})|_{[m]}, \tag{3.12}$$

$$d|_{[m]} = T_m(v^{(\mathsf{d})}, \widehat{v}^{(\mathsf{d})}, \dots, v^{(m)}, \widehat{v}^{(m)}, \overline{d^I}^{[m]})|_{[m]}, \tag{3.13}$$

and

$$u^{(j)} \equiv_\theta v^{(j)}, \quad \widehat{u}^{(j)} \equiv_\theta \widehat{v}^{(j)} \qquad \text{for all } j = \mathsf{d}, \dots, m, \tag{3.14}$$

hence also

$$\begin{aligned}(B \ni)\ T_m(u^{(\mathsf{d})}, \widehat{u}^{(\mathsf{d})}, \dots, u^{(m)}, \widehat{u}^{(m)}, \overline{b^I}^{[m]}) \\ \equiv_\theta T_m(v^{(\mathsf{d})}, \widehat{v}^{(\mathsf{d})}, \dots, v^{(m)}, \widehat{v}^{(m)}, \overline{d^I}^{[m]})\ (\in C).\end{aligned} \tag{3.15}$$

Then, equality (3.13) for $m = n$ yields the desired equality (3.11).

Our induction hypothesis is that the statement in the preceding paragraph is true for $m-1$, that is, there exist tuples $u^{(\mathsf{d})}, \widehat{u}^{(\mathsf{d})}, \dots, u^{(m-1)}, \widehat{u}^{(m-1)}$ in $\mathbf{B}$ and tuples $v^{(\mathsf{d})}, \widehat{v}^{(\mathsf{d})}, \dots, v^{(m-1)}, \widehat{v}^{(m-1)}$ in $\mathbf{C}$ such that (3.12)–(3.15) hold for $m-1$ in place of m. To simplify notation, let

$$b^{(m-1)} := T_{m-1}(u^{(\mathsf{d})}, \widehat{u}^{(\mathsf{d})}, \dots, u^{(m-1)}, \widehat{u}^{(m-1)}, \overline{b^I}^{[m-1]}),$$
$$d^{(m-1)} := T_{m-1}(v^{(\mathsf{d})}, \widehat{v}^{(\mathsf{d})}, \dots, v^{(m-1)}, \widehat{v}^{(m-1)}, \overline{d^I}^{[m-1]}).$$

Then we have that $b^{(m-1)}|_{[m-1]} = b|_{[m-1]}$ and $d^{(m-1)}|_{[m-1]} = d|_{[m-1]}$. Let $\gamma := b|_m$, $\beta := b^{(m-1)}|_m$, $\tau := d|_m$, and $\sigma := d^{(m-1)}|_m$. If we can show the existence of a pair of witnesses $u^{(m)}, \widehat{u}^{(m)}$ in $\mathbf{B}$ for the fork $(\gamma, \beta^\gamma) \in \mathrm{FORK}_m(B)$ and a pair of witnesses $v^{(m)}, \widehat{v}^{(m)}$ in $\mathbf{C}$ for the fork $(\tau, \sigma^\tau) \in \mathrm{FORK}_m(C)$ such that $u^{(m)} \equiv_\theta v^{(m)}$ and $\widehat{u}^{(m)} \equiv_\theta \widehat{v}^{(m)}$, then (3.14)–(3.15) will follow for m, and by Lemma 3.10, (3.12)–(3.13) will also hold for m.

To prove the existence of such $u^{(m)}, \widehat{u}^{(m)}$ and $v^{(m)}, \widehat{v}^{(m)}$, notice first that our induction hypothesis that (3.15) holds for $m-1$ in place of m has the effect that $(B \ni)\ b^{(m-1)} \equiv_\theta d^{(m-1)}\ (\in C)$, so in particular, $(B|_m \ni)\ \beta \equiv_{\theta_m} \sigma$. By our choice of b and d, we also have that $(B|_m \ni)\ \gamma = b|_m \equiv_{\theta_m} d|_m = \tau$. Thus, by our assumption (2), there exist witnesses $b_\beta := b_{m,\beta}$ and $c_\sigma := c_{m,\sigma} \in \Phi$ for the fork $(\beta, \sigma) \in \mathrm{FORK}_m(C)$ such that

$$b_\beta|_m = \beta, \quad c_\sigma|_m = \sigma, \quad b_\beta|_{[m-1]} = c_\sigma|_{[m-1]}, \quad \text{and} \quad b_\beta \equiv_\theta c_\sigma. \tag{3.16}$$

There also exist witnesses $b_\gamma := b_{m,\gamma} \in B$ and $c_\tau := c_{m,\tau} \in \Phi$ for the fork $(\gamma, \tau) \in \mathrm{FORK}_m(C)$ such that

$$b_\gamma|_m = \gamma, \quad c_\tau|_m = \tau, \quad b_\gamma|_{[m-1]} = c_\tau|_{[m-1]}, \quad \text{and} \quad b_\gamma \equiv_\theta c_\tau. \tag{3.17}$$

Let $b_{\beta^\gamma} := p(b_\beta, b_\gamma, b_\gamma)$, $c_{\beta^\tau} := p(b_\beta, c_\tau, c_\tau)$, and $c_{\sigma^\tau} := p(c_\sigma, c_\tau, c_\tau)$. Clearly, $b_{\beta^\gamma} \in B$ and $c_{\beta^\tau}, c_{\sigma^\tau} \in C$. Furthermore, the relations in (3.16)–(3.17) imply that

$$\begin{aligned} &b_{\beta^\gamma}|_m = \beta^\gamma, \quad c_{\beta^\tau}|_m = \beta^\tau, \quad c_{\sigma^\tau}|_m = \sigma^\tau, \\ &\qquad b_{\beta^\gamma}|_{[m-1]} = c_{\beta^\tau}|_{[m-1]} = c_{\sigma^\tau}|_{[m-1]}, \quad \text{and} \quad b_{\beta^\gamma} \equiv_\theta c_{\beta^\tau} \equiv_\theta c_{\sigma^\tau}. \end{aligned} \tag{3.18}$$

It follows that c_{β^τ} and c_{σ^τ} are witnesses in $\mathbf{C}$ for the fork $(\beta^\tau, \sigma^\tau) \in \mathrm{FORK}_m(C)$.

Since b and $b^{(m-1)}$ are witnesses in B for the fork $(\gamma, \beta) \in \mathrm{FORK}_m(B) \subseteq \mathrm{FORK}_m(C)$, we can apply Lemma 3.3 first to (τ, γ) and (β, γ) to obtain witnesses

$$z := p(p(c_\tau, b_\gamma, b), p(c_\tau, b_\gamma, b_\gamma), c_\tau)\ (\in C)$$
$$\widehat{z} := p(b^{(m-1)}, c_\tau, c_\tau)\ (\in C)$$

for the fork $(\tau, \beta^\tau) \in \mathrm{FORK}_m(C)$. The analogous construction for the forks (γ, γ) and (β, γ) yields witnesses

$$w := p(p(b_\gamma, b_\gamma, b), p(b_\gamma, b_\gamma, b_\gamma), b_\gamma) = p(b, b_\gamma, b_\gamma)\ (\in B)$$
$$\widehat{w} := p(b^{(m-1)}, b_\gamma, b_\gamma)\ (\in B)$$

for the fork $(\gamma, \beta^\gamma) \in \mathrm{FORK}_m(B)$. Since $w, \widehat{w}$ are obtained from $z, \widehat{z}$ by replacing c_τ with b_γ, the relation $b_\gamma \equiv_\theta c_\tau$ in (3.17) implies that $z \equiv_\theta w$ and $\widehat{z} \equiv_\theta \widehat{w}$. Applying Lemma 3.3 again, now to the forks (τ, β^τ) and $(\sigma^\tau, \beta^\tau)$, we obtain witnesses

$$v^{(m)} := p(p(z, \widehat{z}, c_{\beta^\tau}), p(z, \widehat{z}, \widehat{z}), z)\ (\in C)$$
$$\widehat{v}^{(m)} := p(c_{\sigma^\tau}, z, z)\ (\in C)$$

for the fork $(\tau, \sigma^{\tau^2}) = (\tau, (\sigma^\tau)^\tau) \in \text{FORK}_m(C)$. Similarly, for the corresponding forks (γ, β^γ) and $(\beta^\gamma, \beta^\gamma)$, we get witnesses

$$u^{(m)} := p(p(w, \widehat{w}, b_{\beta^\gamma}), p(w, \widehat{w}, \widehat{w}), w) \ (\in B)$$
$$\widehat{u}^{(m)} := p(b_{\beta^\gamma}, w, w) \ (\in B)$$

for the fork $(\gamma, \beta^{\gamma^2}) = (\gamma, (\beta^\gamma)^\gamma) \in \text{FORK}_m(B)$. Since $u^{(m)}$ and $\widehat{u}^{(m)}$ are obtained from $v^{(m)}$ and $\widehat{v}^{(m)}$ by replacing z with w, $\widehat{z}$ with $\widehat{w}$, and $c_{\sigma^\tau}, c_{\beta^\tau}$ with b_{β^γ}, the relations $z \equiv_\theta w$, $\widehat{z} \equiv_\theta \widehat{w}$ proved earlier, and the relations $b_{\beta^\tau} \equiv_\theta c_{\beta^\tau} \equiv_\theta c_{\sigma^\tau}$ in (3.18) imply that $v^{(m)} \equiv_\theta u^{(m)}$ and $\widehat{v}^{(m)} \equiv_\theta \widehat{u}^{(m)}$. This completes the induction for (3.12)–(3.15) and the proof of the theorem. □

4. Algorithms Involving Compact Representations

Throughout this section we will work under the following global assumptions.

Assumption 4.1.

- *$\mathcal{V}$ is a variety in a finite language F with a d-cube term ($\mathsf{d} > 1$),*
- *P is a $(1, \mathsf{d}-1)$-parallelogram term in $\mathcal{V}$ (the existence of such a term is ensured by Theorem 2.1), and*
- *$\mathcal{K}$ is a finite set of finite algebras in $\mathcal{V}$.*

Our aim in this section is to use the results of Section 3 to show that

- ▶ the problems SMP($\mathcal{K}$) and SMP($\mathbb{HSK}$) are polynomial time equivalent (Theorem 4.10),
- ▶ SMP($\mathcal{K}$) is in NP (Theorem 4.13), and
- ▶ SMP($\mathcal{K}$) and the problem of finding compact representations for algebras in $\mathbb{SP}_{\text{fin}}\mathcal{K}$ given by their generators are polynomial time reducible to each other (Theorem 4.16).

Recall that if $\mathbf{B}$ is a subalgebra of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ with $\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$, then a representation for $\mathbf{B}$ is a subset R of B satisfying conditions (2)–(3) of Definition 3.4 (with $e = 1$). Condition (2) in the definition makes sure that for every $I \subseteq [n]$ with $|I| = \min\{n, \mathsf{d}-1\}$ and for every tuple $\bar{b} \in B|_I$ there exists an element $r_{I,\bar{b}}$ in R such that $r_{I,\bar{b}}|_I = \bar{b}$. Similarly, condition (3) makes sure that for every derived fork $(\gamma, \delta) \in \text{FORK}'_i(B)$ ($i \in [n]$) there exist elements $u_{i,\gamma,\delta}, \widehat{u}_{i,\gamma,\delta}$ in R which witness that $(\gamma, \delta) \in \text{FORK}_i(R)$. For some of the algorithms we are going to discuss, it will be convenient to fix these choices for witnesses in R. This motivates our definition of a standardized representation below.

Before the definition, let us note that if $n < \mathsf{d}$, condition (2) forces $R = B$, so finding representations for $\mathbf{B}$ in this case is trivial. Therefore, we will assume that $n \geq \mathsf{d}$. Then, by condition (2), we have $R|_{[\mathsf{d}-1]} = B|_{[\mathsf{d}-1]}$, hence one can easily find witnesses among the elements $r_{[\mathsf{d}-1],\bar{b}} \in R$ for the forks in coordinates $i \in [\mathsf{d}-1]$. Therefore, witnesses for forks are of interest only in coordinates $i = \mathsf{d}, \mathsf{d}+1, \dots, n$. (See also Lemma 3.11.)

Definition 4.2. For a subalgebra $\mathbf{B}$ of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ with $\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$ ($n \geq \mathsf{d}$), we will call a representation R for $\mathbf{B}$ a *standardized representation for* $\mathbf{B}$ if the following conditions are satisfied:

(1) For every $I \subseteq [n]$ with $|I| = \mathsf{d}-1$ and for every tuple $\bar{b} \in B|_I$, an element $r_{I,\bar{b}}$ of R is fixed so that $r_{I,\bar{b}}|_I = \bar{b}$; this element will be referred to as *the designated witness in R for $\bar{b} \in R|_I$*.

(2) For every m ($\mathsf{d} \leq m \leq n$) there is a set F_m with $\textsc{fork}'_m(B) \subseteq F_m \subseteq \textsc{fork}_m(R)$ such that for every $(\gamma, \delta) \in F_m$ a pair $(u_{m,\gamma,\delta}, \widehat{u}_{m,\gamma,\delta})$ in R^2 is fixed which witnesses that $(\gamma, \delta) \in \textsc{fork}_m(R)$; these elements $u_{m,\gamma,\delta}, \widehat{u}_{m,\gamma,\delta}$ will be referred to as *the designated witnesses in R for $(\gamma, \delta) \in \textsc{fork}_m(R)$*.
(3) Every element of R has at least one designation.

We will refer to the designated witnesses in (1) and (2) as *local witnesses for* $\mathbf{B}$ and *fork witnesses for* $\mathbf{B}$*, respectively.*

Formally, a standardized representation for $\mathbf{B}$ is the representation R together with a *designation function*

$$\Big(\big\{(I, \bar{b}) : I \in \tbinom{[n]}{\mathsf{d}-1}, \bar{b} \in B|_I\big\} \sqcup \bigcup_{m=\mathsf{d}}^{n} \big(\{m\} \times F_m \times [2]\big)\Big) \to R,$$
$$(I, \bar{b}) \mapsto r_{I,\bar{b}},\ (m, (\gamma, \delta), 1) \mapsto u_{m,\gamma,\delta},\ (m, (\gamma, \delta), 2) \mapsto \widehat{u}_{m,\gamma,\delta} \quad (4.1)$$

where the notation is the same as in Definition 4.2 (1)–(2), and $\sqcup$ denotes disjoint union. Condition (3) in the definition requires the designation function to be onto. For our purposes in this paper it will be sufficient to treat the designation function of a standardized representation informally, as we did in Definition 4.2.

Remark 4.3. It is clear that under the assumptions of Definition 4.2 a standardized representation for $\mathbf{B}$ is compact. Moreover, it follows from Lemma 3.11 that if R is a standardized representation for $\mathbf{B}$, then the equality (3.10) in Lemma 3.10 (with $e = 1$) holds provided

- $\overline{b^I}$ is an enumeration of the elements b^I of R designated to witness $b^I|_I = b|_I$ for all $I \in \binom{[n]}{\mathsf{d}-1}$; and
- for every $\mathsf{d} \leq m \leq n$, the pair $(u^{(m)}, \widehat{u}^{(m)})$ is the designated witness in R for the fork $(\gamma, \beta^\gamma) \in \textsc{fork}_m(R)$ where γ and β are determined by (3.8).

Now we are ready to describe in more detail what we mean by the problem of finding compact representations for algebras in $\mathbb{SP}_{\text{fin}}\mathcal{K}$ given by their generators:

CompactRep($\mathcal{K}$):

- INPUT: $a_1, \dots, a_k \in \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ with $\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$ ($n \geq \mathsf{d}$).
- OUTPUT: A standardized representation for the subalgebra of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ generated by $\{a_1, \dots, a_k\}$.

In Subsections 4.3 and 4.5 below we will discuss two different algorithms for solving CompactRep($\mathcal{K}$). The idea of the first algorithm (Algorithm 4) will be key to showing in Subsection 4.4 that SMP($\mathcal{K}$) is in NP. The second algorithm (Algorithm 5), on the other hand, will use SMP($\mathcal{K}$) to solve CompactRep($\mathcal{K}$). Prior to our first algorithm for CompactRep($\mathcal{K}$) we will introduce two auxiliary algorithms (Subsection 4.1), and then prove in Subsection 4.2 that the problems SMP($\mathcal{K}$) and SMP($\mathbb{HSK}$) are polynomial time equivalent.

4.1. **Two auxiliary algorithms.** When we construct standardized representations, the following concepts will be useful.

Definition 4.4. Let $\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$ with $n \geq \mathsf{d}$, and let $R \subseteq \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$. We will call R a *partial standardized representation* if every element of R is designated to witness either $\bar{b} \in R|_I$ for some $I \in \binom{[n]}{\mathsf{d}-1}$ or a fork in $\textsc{fork}_m(R)$ for some m ($\mathsf{d} \leq m \leq n$).

If R is a partial standardized representation that is contained in a subalgebra $\mathbf{B}$ of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$, we may refer to R as a *partial standardized representation for* $\mathbf{B}$. If, in addition, R satisfies condition (1) of Definition 4.2, we will say that R *contains a full set of designated local witnesses for* $\mathbf{B}$.

Similarly to standardized representations, a partial standardized representation is formally defined to be a set R satisfying the requirements of Definition 4.4, together with a *designation function*, which is a function mapping onto R, similar to (4.1), but the domain is a subset of the set

$$\Big\{(I, \bar{b}) : I \in \tbinom{[n]}{\mathsf{d}-1}, \bar{b} \in \prod_{i \in I} \mathbf{A}_i\Big\} \sqcup \bigcup_{m=\mathsf{d}}^{n} \big(\{m\} \times A_m^2 \times [2]\big).$$

If R is a partial standardized representation for a subalgebra $\mathbf{B}$ of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$, then the domain of its designation function is a subset of the domain indicated in (4.1). Although we will usually treat the designation functions of partial standardized representations R informally, we will consider the size of R to be the size of the domain of its designation function (not just the cardinality of the set R). In particular, if R is a partial standardized representation for a subalgebra $\mathbf{B}$ of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ such that R contains a full set of designated local witnesses for $\mathbf{B}$, then its size is $\Theta(n^{\mathsf{d}-1})$.

Definition 4.5. Let $\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$ with $n \geq \mathsf{d}$, let $R \subseteq \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ be a partial standardized representation, and let $b \in \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$. We will say that b *is representable by* R if the following conditions are met:

(1) for each $I \in \binom{[n]}{\mathsf{d}-1}$, R contains elements b^I designated to witness $b^I|_I = b|_I$, and
(2) for every $\mathsf{d} \leq m \leq n$, R contains designated witnesses $u^{(m)}, \widehat{u}^{(m)}$ for the fork (γ, β^γ) where β, γ are as defined in (3.8) (with $e = 1$).

We will say that b *is completely representable by* R if condition (1) and the following stronger version of (2) hold:

$(2)_\mathrm{c}$ for every $\mathsf{d} \leq m \leq n$, R contains designated witnesses $u^{(m)}, \widehat{u}^{(m)}$ and $v^{(m)}, \widehat{v}^{(m)}$ for the forks (γ, β^γ) and (γ, β) in $\textsc{fork}_m(R)$ where β, γ are as defined in (3.8) (with $e = 1$).

The following fact is an easy consequence of Lemma 3.10.

Corollary 4.6. *Let $R \subseteq \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ ($\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$, $n \geq \mathsf{d}$) be a partial standardized representation. If an element $b \in \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ is representable by R, then the equality in* (3.10) *holds for b (with $e = 1$), and hence b is in the P-subalgebra of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ generated by R.*

The proof of Lemma 3.10 can easily be turned into a polynomial time algorithm for solving the following problem.

IsRepresentable($\mathcal{K}$):

- INPUT: $b \in \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ ($n \geq \mathsf{d}$) and a partial standardized representation $R \subseteq \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ ($\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$) such that R contains elements b^I designated to witness $b^I|_I = b|_I$ for each $I \in \binom{[n]}{\mathsf{d}-1}$.
- OUTPUT: (YES, $\emptyset$, S) or (NO, S', S) where $S', S\,(\subseteq \langle R \cup \{b\}\rangle_P)$ are lists of designated witnesses (missing from R) for the derived forks and for the forks that are not derived forks, respectively, so that b becomes representable(completely representable) by the partial standardized representation $R \cup S'$ ($R \cup S' \cup S$, respectively).

Note that the designated witnesses (for forks that are not derived forks) collected in the set S do not play a role in determining whether or not b is representable, but they will be useful in other algorithms that call IsRepresentable($\mathcal{K}$).

Lemma 4.7. *Algorithm 1 solves* IsRepresentable($\mathcal{K}$) *in polynomial time.*

Algorithm 1 For IsRepresentable($\mathcal{K}$)

Require: $b \in \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ and a partial standardized representation $R \subseteq \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ ($\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$, $n \geq \mathsf{d}$) such that R contains elements b^I designated to witness $b^I|_I = b|_I$ for each $I \in \binom{[n]}{\mathsf{d}-1}$

Ensure: (YES, $\emptyset$, S) or (NO, S', S) where $S', S\,(\subseteq \langle R \cup \{b\}\rangle_P)$ are lists of designated witnesses (missing from R) for the derived forks and for the forks that are not derived forks, respectively, so that b becomes representable (completely representable) by the partial standardized representation $R \cup S'$ ($R \cup S' \cup S$, respectively)

1: $S := \emptyset$, $S' := \emptyset$,
2: $b^{(\mathsf{d}-1)} := b^{[\mathsf{d}-1]}$
3: **for** $m = \mathsf{d}, \dots, n$ **do**
4: $\beta := b^{(m-1)}|_m$, $\gamma := b|_m$, $c := p(b^{(m-1)}, b, b)$
5: **if** $R \cup S'$ has no designated witnesses for $(\gamma, \beta^\gamma) \in \mathrm{FORK}_m(R \cup S')$ **then**
6: add b, c to S' as designated witnesses for $(\gamma, \beta^\gamma) \in \mathrm{FORK}_m(R \cup S')$
7: **end if**
8: **if** $R \cup S' \cup S$ has no designated witnesses for $(\gamma, \beta) \in \mathrm{FORK}_m(R \cup S' \cup S)$ **then**
9: add $b, b^{(m-1)}$ to S as designated witnesses for $(\gamma, \beta) \in \mathrm{FORK}_m(R \cup S' \cup S)$
10: **end if**
11: let $u, \widehat{u} \in R \cup S'$ be the designated witnesses for $(\gamma, \beta^\gamma) \in \mathrm{FORK}_m(R \cup S')$
12: $b^{(m)} = t_m(b^{(m-1)}, \widehat{u}, u, \overline{b^I}^{[m]})$
13: **end for**
14: **if** $S' = \emptyset$ **then**
15: **return** (YES, S', S)
16: **else**
17: **return** (NO, S', S)
18: **end if**

Proof. First we prove that Algorithm 1 is correct. In Step 2, $b^{[\mathsf{d}-1]} \in R$ is the designated witness for $b^{[\mathsf{d}-1]}|_{[\mathsf{d}-1]} = b|_{[\mathsf{d}-1]}$. The loop in Steps 3–13 follows the induction in the proof of Lemma 3.10 (with $e = 1$). For each $m = \mathsf{d}, \dots, n$, if $b^{(m-1)}$, S' and S (with $S' = S = \emptyset$ for $m = \mathsf{d}$ by Step 1) have been constructed such that $S', S \subseteq \langle R \cup \{b\}\rangle_P$, $b^{(m-1)} \in \langle R \cup S'\rangle_P$,and

$b^{(m-1)}|_{[m-1]} = b|_{[m-1]}$, then to construct $b^{(m)} = t_m(b^{(m-1)}, \widehat{u}, u, \overline{b^I}^{[m]})$ we need designated witnesses u, $\widehat{u}$ for the fork $(\gamma, \beta^\gamma) \in \text{FORK}_m(R \cup S')$ where $\gamma = b|_m$ and $\beta = b^{(m-1)}|_m$. Furthermore, for b to be completely representable by $R \cup S' \cup S$, we need designated witnesses v, $\widehat{v}$ for the fork $(\gamma, \beta) \in \text{FORK}_m(R \cup S' \cup S)$.

Clearly, $b, b^{(m-1)}$ witness the fork $(\gamma, \beta) \in \text{FORK}_m(\langle R \cup \{b\}\rangle_B)$. It follows that b and $c = p(b^{(m-1)}, b, b)$ witness the derived fork $(\gamma, \beta^\gamma) \in \text{FORK}'_m(\langle R \cup \{b\}\rangle_P)$, because $c|_{[m-1]} = p(b^{(m-1)}, b, b)|_{[m-1]} = p(b, b, b)|_{[m-1]} = b|_{[m-1]}$ and $b|_m = \gamma$, $c|_m = p(b^{(m-1)}, b, b)|_m = p(\beta, \gamma, \gamma) = \beta^\gamma$. Therefore, if the required designated witnesses $u, \widehat{u}$ do not exist in $R \cup S'$, the algorithm (in Steps 5–7) correctly adds b, c to be the designated witnesses. Similarly, if the required designated witnesses $v, \widehat{v}$ do not exist in $R \cup S' \cup S$, the algorithm (in Steps 8–10) correctly adds $b, b^{(m-1)}$ to be the designated witnesses. Notice also that the updated versions S' and S maintain the property $S', S \subseteq \langle R \cup \{b\}\rangle_P$ when the loop starts over (if $m < n$) or ends (if $m = n$) after computing $b^{(m)}$ in Step 12.

Thus, the standardized representations $R \cup S'$ and $R \cup S' \cup S$ satisfy the requirements that b is representable by $R \cup S'$ and completely representable by $R \cup S' \cup S$. Since new designated fork witnesses (at most one pair for each $m = \mathsf{d}, \dots, n$) are added to S' or S only if such witnesses do not exist, but are necessary for the (complete) representability of b, we see that b is representable by R if and only if $S' = \emptyset$. This proves the correctness of Algorithm 1.

To estimate the running time of Algorithm 1, notice that Step 1 requires constant time, while Steps 2 and 14–18 can be done in time $O(n|R|)$ and $O(n|S| + n|S'|)$, respectively, where $|R| + |S| + |S'| \leq O(n^{\mathsf{d}-1})$. Finally, by Lemma 3.8 (2) (with $e = 1$), if b is representable by R, the computations in the loop in Steps 3–13 require $O(n^{\mathsf{d}+1})$ applications of P. So Steps 3–16 (including the search for witnesses for forks in R) can be done in $O(n^{\mathsf{d}+2})$ time. If b is not representable by R, essentially the same computation is performed, so the bound $O(n^{\mathsf{d}+2})$ applies in this case as well. This proves that Algorithm 1 runs in time $O(n^{\mathsf{d}+2})$. □

Remark 4.8. The definition of the P-term T_n shows that Algorithm 1 may be viewed as doing the following two computations **simultaneously**, when it is run on the inputs b and R:

- it evaluates the P-term T_n (for $e = 1$) step-by-step on the tuple $\overline{b^I}$ of designated local witnesses in R and on appropriately chosen designated fork witnesses $u^{(\mathsf{d})}, \widehat{u}^{(\mathsf{d})}, \dots, u^{(n)}, \widehat{u}^{(n)}$ to obtain the elements
$$b^{(m)} = T_m(u^{(\mathsf{d})}, \widehat{u}^{(\mathsf{d})}, \dots, u^{(m)}, \widehat{u}^{(m)}, \overline{b^I}^{[m]}), \quad m = \mathsf{d}, \dots, n,$$
which, for $m = n$, yields the desired representation
$$b = T_n(u^{(\mathsf{d})}, \widehat{u}^{(\mathsf{d})}, \dots, u^{(n)}, \widehat{u}^{(n)}, \overline{b^I})$$
of b;
- at each step, to be able to evaluate T_m, it produces (namely, it finds in $R \cup S'$ or, if missing, adds to S') the designated fork witnesses $u^{(m)}, \widehat{u}^{(m)}$ for derived forks needed for the evaluation of T_m, and does the same with the designated fork witnesses for the corresponding non-derived forks.

So, in addition to some book-keeping to ensure that the necessary designated fork witnesses (and nothing else) get into the sets S' and S, the computations done by Algorithm 1 in the algebra $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ can be performed by a P-circuit $\mathsf{Circ}(T_n)$ for T_n, which is expanded at each node that produces an element $b^{(m-1)}$ ($m = \mathsf{d}, \dots, n$) by a single P-gate that also

computes $p(b^{(m-1)}, b, b)$ (see the definition of p in (2.4)). We will denote this expanded circuit by $\mathsf{Circ}^+(T_n)$. Since $\mathsf{Circ}^+(T_n)$ is obtained from $\mathsf{Circ}(T_n)$ by adding at most $n - \mathsf{d} + 1$ P-gates, it follows from (3.7) (and $e = 1$) that $\mathsf{Circ}^+(T_n)$ has size $O(n^{\mathsf{d}+1}) + O(n) = O(n^{\mathsf{d}+1})$.

Now we define another problem which will be useful in constructing standardized representations for subalgebras of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ ($\mathbf{A}_1 \ldots, \mathbf{A}_n \in \mathcal{K}$), and also for the polynomial time reduction of SMP($\mathbb{HSK}$) to SMP($\mathcal{K}$):

LocalRep($\mathcal{K}$):

- INPUT: $a_1, \ldots, a_k \in \mathbf{B}_1 \times \cdots \times \mathbf{B}_n$ with $\mathbf{B}_1, \ldots, \mathbf{B}_n \in \mathbb{S}\mathcal{K}$ ($n \geq \mathsf{d}$), and $\theta = \theta_1 \times \cdots \times \theta_n$ with $\theta_i \in \mathrm{Con}(\mathbf{B}_i)$ for all $i \in [n]$. (Let $\mathbf{B}$ denote the subalgebra of $\mathbf{B}_1 \times \cdots \times \mathbf{B}_n$ generated by $\{a_1, \ldots, a_k\}$.)
- OUTPUT: Partial standardized representation R for the θ-saturation $\mathbf{B}[\theta]$ of $\mathbf{B}$ such that R is a full set of designated local witnesses for $\mathbf{B}[\theta]$, and whenever $r_{I,\bar{b}}, r_{I,\bar{d}} \in R$ are designated to witness $r_{I,\bar{b}}|_I = \bar{b}$ and $r_{I,\bar{d}}|_I = \bar{d}$ where $\bar{b} \in \mathbf{B}|_I$ and $\bar{b} \equiv_{\theta|_I} \bar{d}$, then $r_{I,\bar{b}} \equiv_\theta r_{I,\bar{d}}$.

In particular, if $\theta_i = 0$ for all $i \in [n]$, then the latter condition holds vacuously for R, therefore the output of LocalRep($\mathcal{K}$) is the subset R of a standardized representation for $\mathbf{B}$, which is a full set of designated local witnesses for $\mathbf{B}$.

Lemma 4.9. *Algorithm 2 solves* LocalRep($\mathcal{K}$) *in polynomial time.*

Algorithm 2 For LocalRep($\mathcal{K}$)

Require: $a_1, \ldots, a_k \in \mathbf{B}_1 \times \cdots \times \mathbf{B}_n$ with $\mathbf{B}_1, \ldots, \mathbf{B}_n \in \mathbb{S}\mathcal{K}$, and $\theta = \theta_1 \times \cdots \times \theta_n$ with $\theta_i \in \mathrm{Con}(\mathbf{B}_i)$ for all $i \in [n]$ ($n \geq \mathsf{d}$). (Let $\mathbf{B}$ denote the subalgebra of $\mathbf{B}_1 \times \cdots \times \mathbf{B}_n$ generated by $\{a_1, \ldots, a_k\}$.)

Ensure: R is a partial standardized representation for $\mathbf{B}[\theta]$ such that R is a full set of designated local witnesses for $\mathbf{B}[\theta]$, and whenever $r_{I,\bar{b}}, r_{I,\bar{d}} \in R$ are designated to witness $r_{I,\bar{b}}|_I = \bar{b}$ and $r_{I,\bar{d}}|_I = \bar{d}$ where $\bar{b} \in \mathbf{B}|_I$ and $\bar{b} \equiv_{\theta|_I} \bar{d}$, then $r_{I,\bar{b}} \equiv_\theta r_{I,\bar{d}}$.

1: $R := \emptyset$
2: **for** $I \in \binom{[n]}{\mathsf{d}-1}$ **do**
3: generate $\mathbf{B}|_I$ by $\{a_1|_I, \ldots, a_k|_I\}$, and simultaneously,
4: **for** each $\bar{b} \in \mathbf{B}|_I \setminus R|_I$ **do**
5: find an element $r_{I,\bar{b}} \in \mathbf{B}$ such that $r_{I,\bar{b}}|_I = \bar{b}$, include it in R, and designate it to witness $\bar{b} \in R|_I$
6: **for** all $\bar{d} \in \prod_{i \in I} \mathbf{B}_i$ such that $\bar{d} \neq \bar{b}$ and $\bar{d}|_j \equiv_{\theta_j} \bar{b}|_j$ for all $j \in I$ **do**
7: include in R the tuple $r_{I,\bar{d}}$ defined by $r_{I,\bar{d}}|_I = \bar{d}$ and $r_{I,\bar{d}}|_{[n]\setminus I} = r_{I,\bar{b}}|_{[n]\setminus I}$ as a designated witness for $\bar{d} \in R|_I$
8: **end for**
9: **end for**
10: **end for**

Proof. The correctness of Algorithm 2 is straightforward to check. To estimate its running time, notice that Step 1 requires constant time, and in Steps 3–9, each subalgebra $\mathbf{B}|_I$ can be generated in a constant number of steps that depends on $\mathcal{K}$ only (and is independent of the size of the input). Therefore, the running time of the algorithm is determined by the number

$\binom{n}{\mathsf{d}-1} = O(n^{\mathsf{d}-1})$ of iterations of the outermost loop (Steps 2–10) and the time needed for computing the designated witnesses that are added to R in each iteration (Steps 3–9) of the loop, which is bounded above by $O(n)$. Thus, Algorithm 2 runs in $O(n^{\mathsf{d}})$ time. □

4.2. **The equivalence of** SMP($\mathcal{K}$) **and** SMP($\mathbb{HSK}$)**.** In this subsection we prove that the answer to Question 2 in the Introduction is YES.

Theorem 4.10. *The decision problems* SMP($\mathcal{K}$) *and* SMP($\mathbb{HSK}$) *are polynomial time equivalent.*

Algorithm 3 Reduction of SMP($\mathbb{HSK}$) to SMP($\mathcal{K}$)

Require: $c_1, \dots, c_k, c_{k+1} \in \mathbf{C}_1 \times \cdots \times \mathbf{C}_n$ with $\mathbf{C}_1, \dots, \mathbf{C}_n \in \mathbb{HSK}$, $n \geq \mathsf{d}$
Ensure: Is c_{k+1} in the subalgebra $\mathbf{D}$ of $\mathbf{C}_1 \times \cdots \times \mathbf{C}_n$ generated by $c_1, \dots, c_k$?
1: **for** $i = 1, \dots, n$ **do**
2: find $\mathbf{A}_i \in \mathcal{K}$, $\mathbf{B}_i \leq \mathbf{A}_i$ and $\theta_i \in \mathrm{Con}(\mathbf{B}_i)$ such that $\mathbf{C}_i = \mathbf{B}_i/\theta_i$
3: **end for**
4: **for** $j = 1, \dots, k+1$ **do**
5: find $a_j \in B_1 \times \cdots \times B_n$ such that $a_j/(\theta_1 \times \cdots \times \theta_n) = c_j$
6: **end for**
7: $G := \{a_1, \dots, a_k\}$ (let $\mathbf{B}$ denote the subalgebra of $\mathbf{B}_1 \times \cdots \times \mathbf{B}_n$ generated by G)
8: Run LocalRep($\mathcal{K}$) with the input $a_1, \dots, a_k \in \mathbf{B}_1 \times \cdots \times \mathbf{B}_n$, $\theta_i \in \mathrm{Con}(\mathbf{B}_i)$ $(i \in [n])$ to get output R (a set; the designations will not play a role)
9: $\Lambda := R$, $\Phi := \emptyset$
10: **for** $i = 1, \dots, n$ **do**
11: generate $\mathbf{B}|_i$ by $G|_i = \{a_1|_i, \dots, a_k|_i\}$, and simultaneously,
12: **for** each new β in $\mathbf{B}|_i$ **do**
13: find an element $b_{i,\beta}$ generated by $a_1, \dots, a_k$ satisfying $b_{i,\beta}|_i = \beta$
14: **for** all $\gamma \equiv_{\theta_i} \beta$ $(\gamma \in B_i)$ **do**
15: add to Φ the tuple $c_{i,\gamma}$ defined by $c_{i,\gamma}|_i = \gamma$ and $c_{i,\gamma}|_{[n]\setminus\{i\}} = b_{i,\beta}|_{[n]\setminus\{i\}}$
16: **end for**
17: **end for**
18: **end for**
19: run SMP($\mathcal{K}$) with the input $G \cup \Lambda \cup \Phi \subseteq \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ and $a_{k+1} \in \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ $(\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K})$, to get an answer $\mathsf{A} =$ YES or $\mathsf{A} =$ NO
20: **return** A

Proof. SMP($\mathcal{K}$) is a subproblem of SMP($\mathbb{HSK}$), so SMP($\mathcal{K}$) is clearly polynomial time reducible to SMP($\mathbb{HSK}$). For the converse we will show that Algorithm 3 reduces SMP($\mathbb{HSK}$) to SMP($\mathcal{K}$) in polynomial time.

In Steps 1–3 Algorithm 3 finds the algebras $\mathbf{A}_i \in \mathcal{K}$, their subalgebras $\mathbf{B}_i$ and their congruences θ_i such that the algebras $\mathbf{C}_i$ in the input are $\mathbf{C}_i = \mathbf{B}_i/\theta_i$ $(i \in [n])$. In Steps 4–6 tuples $a_1, \dots, a_{k+1} \in \mathbf{B}_1 \times \cdots \times \mathbf{B}_n$ $(\leq \mathbf{A}_1 \times \cdots \times \mathbf{A}_n)$ are found such that for the product congruence $\theta := \theta_1 \times \cdots \times \theta_n$ we have $c_j = a_j/\theta$ for all $j \in [k+1]$. Thus, as we explained in the paragraph preceding Theorem 3.12, the set $G = \{a_1, \dots, a_k\}$ obtained in Step 7 is a generating set for a subalgebra $\mathbf{B}$ of $\mathbf{B}_1 \times \cdots \times \mathbf{B}_n$ with the following properties: $\mathbf{B}[\theta]/\theta \cong \mathbf{D}$, and $c_{k+1} \in \mathbf{D}$ if and only if $a_{k+1} \in \mathbf{B}[\theta]$. Therefore, Algorithm 3 gives the correct answer in

Steps 19–20, provided the set $G \cup \Lambda \cup \Phi$ produced earlier in the process is a generating set for $\mathbf{B}[\theta]$.

By Theorem 3.12 it suffices to check that the set Λ constructed in Steps 8–9 satisfies condition (1), while the set Φ constructed in Steps 9–18 satisfies condition (2) in Theorem 3.12. For Λ this is straightforward to check. For Φ note that, given $i \in [n]$ and $\beta \in B|_i$ as in the loop 10–18, the tuple $b_{i,\beta}$ obtained in line 13 belongs to $\mathbf{B}$. Hence, when the **for** loop in Steps 14–16 is performed for $\gamma = \beta$, we get the tuple $c_{i,\beta} = b_{i,\beta} \in B$, which is added to Φ. This tuple can serve as the tuple denoted $b_{i,\beta}$ in condition (2) for every $c_{i,\gamma}$ added to Φ in Steps 14–16. This shows that the set Φ constructed in Steps 9–18 satisfies condition (2) in Theorem 3.12, and hence finishes the proof of the correctness of Algorithm 3.

Steps 1–7 run in $O(kn)$ time, Steps 8–9 in $O(n^{\mathsf{d}})$ time, while Steps 10–18 in $O(n^2)$ time. So, the reduction of $\mathrm{SMP}(\mathbb{HS}\mathcal{K})$ to $\mathrm{SMP}(\mathcal{K})$ takes $O(kn^{\mathsf{d}})$ time. For an input of size $O(kn)$ of $\mathrm{SMP}(\mathbb{HS}\mathcal{K})$ we get an input of size $O(kn^{\mathsf{d}})$ for $\mathrm{SMP}(\mathcal{K})$. □

4.3. **Finding compact representations: A direct algorithm.** Our aim in this subsection is to present an algorithm for CompactRep($\mathcal{K}$), which does not rely on $\mathrm{SMP}(\mathcal{K})$. The idea of the algorithm will be used in the next subsection to prove that $\mathrm{SMP}(\mathcal{K}) \in \mathsf{NP}$. A different algorithm for CompactRep($\mathcal{K}$), which does rely on $\mathrm{SMP}(\mathcal{K})$, will be presented in Subsection 4.5.

Recall from Definitions 3.4 and 4.2 that if $\mathbf{B}$ is a subalgebra of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ ($\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$, $n \geq \mathsf{d}$), then for a set $R \subseteq \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ to be a standardized representation for $\mathbf{B}$, it has to satisfy three conditions: (i) $R \subseteq \mathbf{B}$ and each element in R has a designation, (ii) R contains a full set of local witnesses for $\mathbf{B}$, and (iii) R contains witnesses for all derived forks (and possible some non-derived forks) of $\mathbf{B}$ in coordinates $\geq \mathsf{d}$. Algorithm 2 (with all $\theta_i = 0$) yields, in polynomial time, a partial standardized representation R_0 consisting of a full set of designated local witnesses for $\mathbf{B}$. Therefore, the task we are left with is to find a way to expand R_0 to R so that both conditions (i) and (iii) are met. To enforce (i) without relying on $\mathrm{SMP}(\mathcal{K})$ we have to make sure that every new element we are adding to R is obtained by applying operations of $\mathbf{B}$ to elements of R or the given generators of $\mathbf{B}$. To achieve (iii), we need to ensure that we add sufficiently many elements to get designated witnesses for all derived forks of $\mathbf{B}$ in coordinates $\geq \mathsf{d}$. The next lemma shows how this can be done.

Lemma 4.11. *Let $\mathbf{B}$ be a subalgebra of a product $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ with $\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$, $n \geq \mathsf{d}$, and assume that $R_1, R_2 \subseteq \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ are partial standardized representations which satisfy conditions* (1)–(3) *below:*

(1) *$R_1 \subseteq R_2 \subseteq \mathbf{B}$ and the designation function for R_2 extends the designation function for R_1.*
(2) *Every element of $\mathbf{B}$ is completely representable by R_1 (in particular, R_1 contains a full set of designated local witnesses for $\mathbf{B}$).*
(3) *R_2 contains designated witnesses for the forks that are obtained by the 'weak transitivity rule' (cf. Lemma 3.3) from forks witnessed in R_1; that is, for every i ($\mathsf{d} \leq i \leq n$), if R_1 has designated witnesses for $(\gamma, \delta), (\beta, \delta) \in \mathrm{FORK}_i(R_1)$, then $(\gamma, \beta^\gamma) \in \mathrm{FORK}_i(R_2)$ and R_2 has designated witnesses for it.*

Then R_2 is a standardized representation for $\mathbf{B}$.

Proof. Let $R_1, R_2 \subseteq \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ be partial standardized representations, which satisfy conditions (1)–(3). Conditions (1)–(2) imply that R_2 is a partial standardized representation for $\mathbf{B}$, which contains a full set of designated local witnesses for $\mathbf{B}$. Therefore, it remains to show that R_2 contains designated witnesses for all derived forks of $\mathbf{B}$ in coordinates $\geq \mathsf{d}$.

To prove this, let $m \geq \mathsf{d}$, and let $(\gamma, \sigma) \in \textsc{fork}'_m(B)$. Then there exists $(\gamma, \beta) \in \textsc{fork}_m(B)$ such that $\sigma = \beta^\gamma$. Hence, there exist $b, b' \in B$ such that $b|_{[m-1]} = b'|_{[m-1]}$ and $b|_m = \beta$, $b'|_m = \gamma$. By assumption (2), both b and b' are completely representable by R_1. Since $b|_{[m-1]} = b'|_{[m-1]}$, it follows from Lemma 3.10 that

$$b^{(m-1)} = T_{m-1}(u^{(\mathsf{d})}, \widehat{u}^{(\mathsf{d})}, \dots, u^{(m-1)}, \widehat{u}^{(m-1)}, \overline{b^I}^{[m-1]}) = (b')^{(m-1)}.$$

Let $\delta := b^{(m-1)}|_m \big(= (b')^{(m-1)}|_m\big)$. Since b and b' are completely representable by R_1, R_1 contains designated witnesses (u, u') and (v, v') for the forks $(\beta, \delta), (\gamma, \delta) \in \textsc{fork}_m(R_1)$. Now, assumption (3) makes sure that in this situation, R_2 contains designated witnesses for the fork $(\gamma, \beta^\gamma) = (\gamma, \sigma)$. This completes the proof. □

Recall that FNP denotes the function problem version of NP. More precisely, a binary relation $R(x, y)$, where the size of y is polynomial in the size of x, is in FNP if and only if there is a deterministic polynomial time algorithm that can determine whether $R(x, y)$ holds given both x and y.

Theorem 4.12. *Algorithm 4 solves* CompactRep($\mathcal{K}$) *in polynomial time by making multiple calls to an oracle* NeedMoreForkWitnesses($\mathcal{K}$) *in* FNP.

Proof. First we define the oracle.

NeedMoreForkWitnesses($\mathcal{K}$):

- INPUT: Partial standardized representation $R \subseteq \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ ($\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$, $n \geq \mathsf{d}$) such that R contains designated local witnesses for the subalgebra $\mathbf{R}^*$ of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ generated by R
- OUTPUT:
  $\big(\text{YES}, f, \big((u_j^{(\mathsf{d})}, \widehat{u}_j^{(\mathsf{d})}, \dots, u_j^{(n)}, \widehat{u}_j^{(n)}, \overline{w_j^I}\big)\big)_{j=1}^{\ell}, b_1, \dots, b_\ell, b, \widetilde{R}\big)$ where
  (1) $f \in F$ is an ℓ-ary operation symbol (for some ℓ),
  (2) each pair $(u_j^{(m)}, \widehat{u}_j^{(m)})$ ($j \in [\ell]$, $\mathsf{d} \leq m \leq n$) is designated to witness a fork in $\textsc{fork}_m(R)$,
  (3) the tuples $\overline{w_j^I} = (w_j^I)_{I \in \binom{[n]}{I}}$ ($j \in [\ell]$) are such that each w_j^I is a designated local witness of $b_j|_I \in R|_I$,
  (4) $b_j = T_n(u_j^{(\mathsf{d})}, \widehat{u}_j^{(\mathsf{d})}, \dots, u_j^{(n)}, \widehat{u}_j^{(n)}, \overline{w_j^I})$ (with $e = 1$) for each $j \in [\ell]$,
  (5) $b = f(b_1, \dots, b_\ell)$,
  (6) b is not completely representable by R,
  (7) $\widetilde{R} = R \cup S' \cup S$ is the partial standardized representation obtained from the output of IsRepresentable($\mathcal{K}$) run on the input b, R (hence, b is completely representable by $\widetilde{R}$).
  NO, if f, $(u_j^{(\mathsf{d})}, \widehat{u}_j^{(\mathsf{d})}, \dots, u_j^{(n)}, \widehat{u}_j^{(n)}, \overline{w_j^I})$ ($j \in [\ell]$), $b_1, \dots, b_\ell$, and b with these properties do not exist.

To show that NeedMoreForkWitnesses($\mathcal{K}$) is in FNP, we have to verify that the length of the output $(\text{YES}, \dots, \widetilde{R})$ is bounded by a polynomial of the length of the input R, and there is a polynomial time algorithm which, when R and $(\text{YES}, \dots, \widetilde{R})$ are both given, determines whether or not conditions (1)–(7) hold. The first requirement holds, because

the length of the output is dominated by the length of $\widetilde{R}$, and both R and $\widetilde{R}$ have length $\Theta(n^{\mathsf{d}})$, as they are partial standardized representations for $\mathbf{R}^*$ which contain a full set of designated local witnesses for $\mathbf{R}^*$.

To prove that the second requirement holds as well, suppose that we are given a partial standardized representation $R \subseteq \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ ($\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$, $n \geq \mathsf{d}$) such that R contains designated local witnesses for $\mathbf{R}^*$, and we are given a tuple

$$\big(f, \big((u_j^{(\mathsf{d})}, \widehat{u}_j^{(\mathsf{d})}, \dots, u_j^{(n)}, \widehat{u}_j^{(n)}, \overline{w_j^I}\big)\big)_{j=1}^{\ell}, b_1, \dots, b_\ell, b, \widetilde{R}\big).$$

Conditions (1)–(3) can be checked in $O(1) + O(n^2) + O(n^{\mathsf{d}}) = O(n^{\mathsf{d}})$ time as ℓ is bounded by a constant that depends only on $\mathcal{K}$. By (3.7), the circuit complexity of T_n (with $e = 1$) is $O(n^{\mathsf{d}+1})$, hence checking condition (4) requires $\ell \cdot O(n^{\mathsf{d}+2}) = O(n^{\mathsf{d}+2})$ time. Condition (5) can be checked in $O(n)$ time. Finally, conditions (6)–(7) can be checked by running Algorithm 1 for IsRepresentable($\mathcal{K}$) with input b and R, which requires $O(n^{\mathsf{d}+2})$ time, and comparing $\widetilde{R}$ with its output. Note here that b and R form a correct input for Algorithm 1 because, by construction, $b \in \mathbf{R}^*$, so our requirement on R guarantees that appropriate designated local witnesses for b exist in R.

Algorithm 4 Algorithm for CompactRep($\mathcal{K}$) (direct)

Require: $a_1, \dots, a_k \in \mathbf{A}_1 \times \dots \mathbf{A}_n$ with $\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$ ($n \geq \mathsf{d}$)
Ensure: Standardized representation R for the subalgebra $\mathbf{B}$ of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ generated by $a_1, \dots, a_k$

1: Run Algorithm 2 for LocalRep($\mathcal{K}$) with input $a_1, \dots, a_k \in \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ and $\theta_i = 0$ ($i \in [n]$) to get output R
2: **for** $b \in \{a_1, \dots, a_k\}$ **do**
3: run Algorithm 1 for IsRepresentable($\mathcal{K}$) with input b and R to get output (YES, S', S) or (NO, S', S)
4: $R := R \cup S' \cup S$
5: **end for**
6: **while** NeedMoreForkWitnesses($\mathcal{K}$) returns (YES, $\dots, \widetilde{R}$) **do**
7: $R := \widetilde{R}$
8: **end while**
9: **for** $m = \mathsf{d}, \dots, n$ **do**
10: **for** all $(\gamma, \delta), (\beta, \delta) \in \mathrm{FORK}_m(R)$ which have designated witnesses $(v, \widehat{v})$, $(u, \widehat{u})$, respectively, in R **do**
11: **if** R contains no designated witnesses for (γ, β^γ) **then**
12: add $p(p(v, \widehat{v}, \widehat{u}), p(v, \widehat{v}, \widehat{v}), v)$ and $p(u, v, v)$ to R and designate them to witness the fork $(\gamma, \beta^\gamma) \in \mathrm{FORK}_m(R)$
13: **end if**
14: **end for**
15: **end for**
16: **return** R

Now we turn to the analysis of Algorithm 4. To prove its correctness, let $a_1, \dots, a_k \in \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ ($\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$) be an input for CompactRep($\mathcal{K}$) with $n \geq \mathsf{d}$, and let $\mathbf{B}$ denote the subalgebra of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ generated by $a_1, \dots, a_k$. We have to show that the set R returned in Step 16 of Algorithm 4 is a standardized representation for $\mathbf{B}$.

Notice that during Algorithm 4 we only add elements to R, and every element added to R has a designation. Therefore, at every point in the algorithm, the current version of R is a partial standardized representation, which extends all previous versions. The description of LocalRep($\mathcal{K}$) implies that the version of R produced in Step 1, which we will denote by R_0, is a standardized representation for $\mathbf{B}$ that is a full set of designated local witnesses for $\mathbf{B}$. In particular, $R_0 \subseteq \mathbf{B}$.

Next we want to argue that

(1) throughout all iterations of the **for** loop in Steps 2–5 and the **while** loop in Steps 6–8, R satisfies
$$R_0 \subseteq R \subseteq \mathbf{B}; \tag{4.2}$$
(2) in every iteration of these two loops, the algorithms for IsRepresentable($\mathcal{K}$) and NeedMoreForkWitnesses($\mathcal{K}$), respectively, have a correct input; and
(3) Algorithm 4 exits the **while** loop, i.e., NeedMoreForkWitnesses($\mathcal{K}$) answers NO for the first time, after polynomially many iterations.

Statements (1)–(2) follow by induction on the number of iterations by observing that (4.2) holds for $R = R_0$ at the start of the first iteration of the **for** loop; moreover, the descriptions of IsRepresentable($\mathcal{K}$) and NeedMoreForkWitnesses($\mathcal{K}$) ensure that whenever (4.2) holds at the start of an iteration of the **for** loop, R together with $b \in \{a_1, \dots, a_k\}$ is a correct input for IsRepresentable($\mathcal{K}$), and the new version of R produced also satisfies (4.2); similarly, whenever (4.2) holds at the start of an iteration of the **while** loop, R is a correct input for NeedMoreForkWitnesses($\mathcal{K}$), and the new version of R produced also satisfies (4.2).

For (3), notice that the **for** loop is repeated exactly k times, and the **while** loop is iterated as long as new designated pairs of fork witnesses can be added to R by such an iteration. By statement (1) above, all these pairs of fork witnesses are witnessing forks in $\mathbf{B}$ in coordinates $\mathsf{d}, \dots, n$, therefore their number is at most $\mathsf{a}_{\mathcal{K}}^2(n - \mathsf{d})$, where $\mathsf{a}_{\mathcal{K}} := \max\{|\mathbf{A}| : \mathbf{A} \in \mathcal{K}\}$. Hence, after at most $\mathsf{a}_{\mathcal{K}}^2(n - \mathsf{d})$ iterations of the **while** loop in Steps 6–8, we have to get a NO answer, and exit the loop.

Let R_1 denote the version of R after Step 8, and let R_2 denote the R returned at the end of Algorithm 4 in Step 16. We will prove that R_2 is a standardized representation for $\mathbf{B}$ by showing that R_1 and R_2 satisfy conditions (1)–(3) of Lemma 4.11. Clearly, in Steps 9–15 of Algorithm 4 exactly those new fork witnesses are added to R_1 which yield R_2 so that condition (3) of Lemma 4.11 is satisfied. Since (4.2) holds for $R = R_1$, and every element $R_2 \setminus R_1$ is obtained from elements of R_1 by applying the term p, we have that $R_0 \subseteq R_1 \subseteq R_2 \subseteq \mathbf{B}$. This proves condition (1) of Lemma 4.11. It remains to prove condition (2).

Let $\overline{R}_1$ denote the set of all elements of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ that are completely representable by R_1. By Corollary 4.6, $R_1 \subseteq \mathbf{B}$ implies that $\overline{R}_1 \subseteq \mathbf{B}$. Next we want to prove that $\overline{R}_1$ is in fact a subalgebra of $\mathbf{B}$. Let $f \in F$ be an operation symbol, say f is ℓ-ary, let $b_1, \dots, b_\ell \in \overline{R}_1$, and let $b := f(b_1, \dots, b_\ell)$. We need to show that $b \in \overline{R}_1$. Since $b_1, \dots, b_\ell \in \overline{R}_1$, i.e., $b_1, \dots, b_\ell$ are completely representable by R_1, there exist $(u_j^{(\mathsf{d})}, \widehat{u}_j^{(\mathsf{d})}, \dots, u_j^{(n)}, \widehat{u}_j^{(n)}, \overline{w_j^I})$ for $j \in [\ell]$ such that conditions (2)–(4) in the description of NeedMoreForkWitnesses($\mathcal{K}$) hold for $R = R_1$. By the choice of f and definition of b above, we have that actually, all conditions (1)–(5) in the description of NeedMoreForkWitnesses($\mathcal{K}$) hold for $R = R_1$. Since R_1 is the version of R in Step 8 of Algorithm 4, the last iteration of the **while** loop ran with $R = R_1$ and produced the answer NO. Thus, b must be completely representable by R_1, which

proves that $b \in \overline{R}_1$. Finally, notice that the **for** loop in Steps 2–5 of Algorithm 4 ensure that the generators $a_1, \dots, a_k$ of $\mathbf{B}$ belong to $\overline{R}_1$. Thus, $\overline{R}_1 = \mathbf{B}$, so condition (2) of Lemma 4.11 also holds. This completes the proof of the correctness of Algorithm 4.

To estimate the running time of Algorithm 4, recall that Algorithm 2 runs in $O(n^{\mathsf{d}})$ time, so this is the running time of Step 1. In Steps 2–5, Algorithm 1 is called k times, and each iteration runs in $O(n^{\mathsf{d}+2})$ time, therefore the completion of Steps 2–5 requires $O(kn^{\mathsf{d}+2})$ time. As we saw earlier in this proof, in Steps 6–8 the oracle NeedMoreForkWitnesses($\mathcal{K}$) is called $O(n)$ times, therefore the running time of these steps is $O(n)$. Finally, in Steps 9–15, the double **for** loop is iterated $O(n)$ times, and each iteration requires $O(n)$ time. Therefore, Steps 9–15 require $O(n^2)$ time. Hence, the overall running time of Algorithm 4 is $O(kn^{\mathsf{d}+2})$. □

4.4. SMP($\mathcal{K}$) **is in** NP**.** The main result of this subsection is the following theorem.

Theorem 4.13. *Let $a_1, \dots, a_k, b \in \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ ($\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$) be an input for* SMP($\mathcal{K}$)*. If b is in the subalgebra of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ generated by $\{a_1, \dots, a_k\}$, then there exists a term t in the language F of $\mathcal{K}$ such that*

- *$t(a_1, \dots, a_k) = b$ and*
- *$\mathsf{Circ}_F(t)$ has size $O(kn^{\mathsf{d}+2})$.*

Consequently, SMP($\mathcal{K}$) $\in$ NP.

The proof relies on Lemma 4.14 below, which shows that by recording the computations Algorithm 4 performs on a given input $a_1, \dots, a_k$, we can build a polynomial size circuit that produces a standardized representation for the algebra with generators $a_1, \dots, a_k$. The precise statement is as follows.

Lemma 4.14. *Let $a_1, \dots, a_k \in \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ with $\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$ ($n \geq \mathsf{d}$), and let $\mathbf{B}$ be the subalgebra of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ generated by $\{a_1, \dots, a_k\}$. There exists an F-circuit C with the following properties:*

(1) *C has k input nodes and $O(n^{\mathsf{d}-1})$ labeled output nodes such that for the input $a_1, \dots, a_k$, the collection of outputs of C — together with the labeling of the outputs — is a standardized representation R for $\mathbf{B}$ computed by Algorithm 4.*
(2) *The size of C is $O(kn^{\mathsf{d}+2})$.*

Proof. C will be constructed in five phases, the first four of which produce $F \cup \{P\}$-circuits, which correspond to Step 1, Steps 2–5, Steps 6–8 and Steps 9–15 of Algorithm 4.

In Step 1, i.e., during the execution of Algorithm 2 on the input $a_1, \dots, a_k$, for every choice of $I \in \binom{[n]}{\mathsf{d}-1}$ and $\bar{b} \in \mathbf{B}|_I$, the designated local witness $r_{I,\bar{b}} \in R$ is obtained as $r_{I,\bar{b}} := t_{I,\bar{b}}(a_1, \dots, a_k)$ for a term $t_{I,\bar{b}}$ (in the language F) such that $\bar{b} = t_{I,\bar{b}}(a_1|_I, \dots, a_k|_I)$. Since $\left|\mathbf{B}|_I\right| \leq \mathsf{a}_{\mathcal{K}}^{\mathsf{d}-1}$ for every such I, where $\mathsf{a}_{\mathcal{K}} := \max\{|\mathbf{A}| : \mathbf{A} \in \mathcal{K}\}$ is a constant independent of $a_1, \dots, a_k$, the terms $t_{I,\bar{b}}$ can be chosen to be of constant size ($\leq \mathsf{a}_{\mathcal{K}}^{\mathsf{d}-1}$). Therefore, an F-circuit C_1 which, for the input $a_1, \dots, a_k$, computes the partial standardized representation obtained in Step 1 of Algorithm 4 (i.e., a full set of designated local witnesses for $\mathbf{B}$) can be constructed as follows: C_1 is the disjoint union of the F-circuits $\mathsf{Circ}(t_{I,\bar{b}})$, except that they all have the same input nodes for $a_1, \dots, a_k$. The output $r_{I,\bar{b}}$ of each $\mathsf{Circ}(t_{I,\bar{b}})$ is an output of C_1 as well, with the appropriate label recording its designation. Clearly, C_1 has size $O(n^{\mathsf{d}-1})$.

In Steps 2–5, Algorithm 1 is run on each input tuple $b \in \{a_1, \dots, a_k\}$ together with the partial standardized representation R constructed by that point in the algorithm (which contains a full set of designated local witnesses for $\mathbf{B}$ by Step 1). By Remark 4.8, each of these k applications of Algorithm 1 is a computation that can be performed by the P-circuit $\mathsf{Circ}^+(T_n)$ on designated local witnesses in R and on appropriately chosen designated fork witnesses, some of which are added to R during the computation. Therefore, an $F \cup \{P\}$-circuit C_2 which, for the inputs $a_1, \dots, a_k$, computes the partial standardized representation for $\mathbf{B}$ obtained in Step 5 of Algorithm 4 can be constructed as follows: add to C_1 k disjoint copies of the P-circuit $\mathsf{Circ}^+(T_n)$ described in Remark 4.8, one for each choice of $b \in \{a_1, \dots, a_k\}$, so that their input nodes are the appropriate designated local witnesses and fork witnesses computed earlier or during the evaluation of T_n as described in Remark 4.8, and they have appropriately labeled output nodes for every pair of designated fork witnesses added to R. It follows from Remark 4.8 that C_2 has size $O(n^{\mathsf{d}-1}) + k \cdot O(n^{\mathsf{d}+1}) = O(kn^{\mathsf{d}+1})$.

Similarly, in Steps 6–8, in every iteration of the **while** loop where the output of NeedMoreForkWitnesses($\mathcal{K}$) has the form

$$X = \big(\text{YES}, f, \big((u_j^{(\mathsf{d})}, \widehat{u}_j^{(\mathsf{d})}, \dots, u_j^{(n)}, \widehat{u}_j^{(n)}, \overline{w_j^I}\big)\big)_{j=1}^{\ell}, b_1, \dots, b_\ell, b, \widetilde{R}\big)$$

such that conditions (1)–(7) in the description of NeedMoreForkWitnesses($\mathcal{K}$) are satisfied, the fork witnesses $u_j^{(m)}, \widehat{u}_j^{(m)}$ and local witnesses w_j^I had already been included in the current version of R, $b_1, \dots, b_\ell$ and b can be computed as stated in conditions (4)–(5), and Algorithm 1 can be run on b (and the current version of R) to find the fork witnesses that have to be added to R to get $\widetilde{R}$ (the new R). We saw in the proof of Theorem 4.12 that the **while** loop is iterated at most $O(n)$ times. Therefore, an $F \cup \{P\}$-circuit C_3 which, for the input $a_1, \dots, a_k$, computes the partial standardized representation for $\mathbf{B}$ obtained in Step 8 of Algorithm 4 can be constructed as follows: add to C_2, one after the other, as many as necessary (at most $O(n)$) circuits of the following form:

- ℓ disjoint copies of the P-circuit $\mathsf{Circ}(T_n)$ whose inputs are (already computed) fork and local witnesses $u_j^{(\mathsf{d})}, \widehat{u}_j^{(\mathsf{d})}, \dots, u_j^{(n)}, \widehat{u}_j^{(n)}, \overline{w_j^I}$ ($j \in [\ell]$), and
- whose outputs $b_1, \dots, b_\ell$ are the inputs for an f-gate, which produces b,
- followed by a copy of the P-circuit $\mathsf{Circ}^+(T_n)$ from Remark 4.8 with appropriately labeled output nodes for every pair of designated fork witnesses added to R, as described in Remark 4.8.

Since the P-circuits $\mathsf{Circ}(T_n)$ and $\mathsf{Circ}^+(T_n)$ have size $O(n^{\mathsf{d}+1})$ by (3.7) and Remark 4.8, and the arities ℓ of the operation symbols f are bounded by a constant independent of the input, we get that the size of C_3 is

$$O(kn^{\mathsf{d}+1}) + O(n) \cdot \big(\ell \cdot O(n^{\mathsf{d}+1}) + 1 + O(n^{\mathsf{d}+1})\big) = O(kn^{\mathsf{d}+2}).$$

Finally, to compute the additional fork witnesses included in R in Steps 9–15 of Algorithm 4, we have to add at most $O(n)$ many P-circuits to C_3, each one of size 1 or 3 (see the definition of p in (2.4)).

The resulting $F \cup \{P\}$-circuit, C_4, is therefore of size $O(kn^{\mathsf{d}+2})$. Finally, since P is a term in F, gates of type P can be easily eliminated from C_4 by replacing each of them with the corresponding F-circuit (of constant size). Thus we obtain an F-circuit C from C_4 which satisfies conditions (1)–(2) of the lemma. □

Proof of Theorem 4.13. To prove the first statement of the theorem, consider an input $a_1, \dots, a_k, b \in \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ ($\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$) for SMP($\mathcal{K}$), let $\mathbf{B}$ denote the subalgebra of

$\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ generated by $\{a_1, \dots, a_k\}$, and assume that $b \in \mathbf{B}$. For $n < \mathsf{d}$ the assertion of the theorem is trivial — by the same argument that we used in the second paragraph of the proof of Lemma 4.14 — since $|\mathbf{B}| \leq \mathsf{a}_{\mathcal{K}}^{\mathsf{d}-1}$ for a constant $\mathsf{a}_{\mathcal{K}}$ that depends only on $\mathcal{K}$. Therefore, we will assume from now on that $n \geq \mathsf{d}$.

Let C be a circuit satisfying properties (1)–(2) of Lemma 4.14, and let R denote the standardized representation for $\mathbf{B}$ that is computed by C from $a_1, \dots, a_k$. As we noted in Remark 4.3, Lemma 3.11 shows that if $b \in \mathbf{B}$, then the equality (3.10),

$$b = T_n(u^{(\mathsf{d})}, \widehat{u}^{(\mathsf{d})}, \dots, u^{(n)}, \widehat{u}^{(n)}, \overline{b^I}),$$

holds with all arguments of T_n members of R; namely, each pair $u^{(m)}, \widehat{u}^{(m)}$ $(m = \mathsf{d}, \dots, n)$ is designated to witness a derived fork in the m-th coordinate, and the tuple $\overline{b^I}$ $(I \in \binom{[n]}{\mathsf{d}-1})$ consists of some designated local witnesses for $\mathbf{B}$. Hence, C has a subcircuit U which outputs exactly the arguments of T_n in the above expression for b. Since C is an F-circuit of size $O(kn^{d+2})$, so is U. Identifying the outputs of U with the inputs of an F-circuit for T_n, which is obtained from its P-circuit by replacing each P-gate by the corresponding F-circuit, we get an F-circuit for a term t such that $t(a_1, \dots, a_k) = b$. The size of $\mathsf{Circ}_F(T_n)$ is constant times the size of $\mathsf{Circ}_P(T_n)$, which is $O(n^{d+1})$ by (3.7) (for $e = 1$). Therefore the size of $\mathsf{Circ}_F(t)$ is $O(kn^{\mathsf{d}+2}) + O(n^{\mathsf{d}+1}) = O(kn^{\mathsf{d}+2})$. This proves the first statement of Theorem 4.13.

By this statement, we have a polynomial size certificate for "b is in the subalgebra generated by $\{a_1, \dots, a_k\}$" for every input $a_1, \dots, a_k, b$ for $\mathrm{SMP}(\mathcal{K})$. This proves that $\mathrm{SMP}(\mathcal{K})$ lies in NP. □

Remark 4.15. In Theorem 4.13, the term t witnessing that b belongs to the subalgebra generated by $a_1, \dots, a_k$ clearly has depth polynomial in n. In the case that the d-cube term is a Mal'tsev term, this was already observed in [22, Theorem 2.2]. If the d-cube term is a near-unanimity term, then the fact that b belongs to the subalgebra generated by $a_1, \dots, a_k$ can be witnessed by polynomially many terms of constant length by the Baker–Pixley Theorem [1]. It is however open in the general case, whether a polynomial bound can be imposed on the length of a representation of t by a term rather than by a circuit, as in Theorem 4.13.

4.5. Finding compact representations: The relationship between $\mathrm{SMP}(\mathcal{K})$ and $\textsc{CompactRep}(\mathcal{K})$.

The main result of this section is the following theorem.

Theorem 4.16. *The problems* $\mathrm{SMP}(\mathcal{K})$ *and* $\textsc{CompactRep}(\mathcal{K})$ *are polynomial time reducible to one another. In more detail, Algorithm 5 solves* $\textsc{CompactRep}(\mathcal{K})$ *in polynomial time by repeated calls of* $\mathrm{SMP}(\mathcal{K})$*, and Algorithm 6 solves* $\mathrm{SMP}(\mathcal{K})$ *(for inputs satisfying* $n \geq \mathsf{d}$*) in polynomial time by a single call of* $\textsc{CompactRep}(\mathcal{K})$.

Proof. As we mentioned earlier, $\mathrm{SMP}(\mathcal{K})$ for inputs $a_1, \dots, a_k, b$ that are n-tuples with $n < \mathsf{d}$ can be solved in constant time, since the algebra $\mathbf{B}$ generated by $a_1, \dots, a_k$ has size bounded by a constant that depends only on $\mathcal{K}$. Therefore it will not restrict generality to assume $n \geq \mathsf{d}$ for the inputs of $\mathrm{SMP}(\mathcal{K})$ in Algorithm 6, and the first statement of Theorem 4.16 will follow from the second statement on Algorithms 5 and 6.

First we will discuss Algorithm 5. To prove its correctness, note that Step 1 produces a partial standardized representation R_0 for $\mathbf{B}$, which is a full set of designated local witnesses for $\mathbf{B}$. Elements without designations are not added to $R = R_0$ later on in Algorithm 5, therefore we will be done if we show that Steps 3–16 add to R a pair of designated witnesses

Algorithm 5 Reduction of CompactRep($\mathcal{K}$) to SMP($\mathcal{K}$)

Require: $a_1, \dots, a_k \in \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ with $\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$, $n \geq \mathsf{d}$
Ensure: Standardized representation R for the subalgebra $\mathbf{B}$ of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ generated by $a_1, \dots, a_k$

1: Run Algorithm 2 for LocalRep($\mathcal{K}$) with input $a_1, \dots, a_k \in \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ and $\theta_i = 0$ ($i \in [n]$) to get output R_0
2: $R := R_0$
3: **for** $i = \mathsf{d}, \dots, n$ and $\gamma \in \mathbf{B}|_i$ **do**
4: find $b \in R_0$ with $b|_i = \gamma$
5: **for** $\beta \in \mathbf{B}|_i$ **do**
6: let $c \in \mathbf{A}_1 \times \cdots \times \mathbf{A}_i$ be such that $c|_{[i-1]} = b|_{[i-1]}$ and $c|_i = \beta^\gamma$
7: run SMP($\mathcal{K}$) with input $a_1|_{[i]}, \dots, a_k|_{[i]}, c \in \mathbf{A}_1 \times \cdots \times \mathbf{A}_i$
8: **if** answer is YES **then**
9: **for** $j = i+1, \dots, n$ **do**
10: find $c_j \in A_j$ such that SMP($\mathcal{K}$) with input $a_1|_{[j]}, \dots, a_k|_{[j]}, (c, c_j) \in \mathbf{A}_1 \times \cdots \times \mathbf{A}_j$ answers YES
11: $c := (c, c_j)$
12: **end for**
13: add b, c to R, and designate them to witness the fork $(\gamma, \beta^\gamma) \in \text{FORK}_i(R)$.
14: **end if**
15: **end for**
16: **end for**
17: **return** R

Algorithm 6 Reduction of SMP($\mathcal{K}$) to CompactRep($\mathcal{K}$)

Require: $a_1, \dots, a_k, b \in \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ with $\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$, $n \geq \mathsf{d}$
Ensure: Is b in the subalgebra $\mathbf{B}$ of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ generated by $a_1, \dots, a_k$?

1: Run CompactRep($\mathcal{K}$) with input $a_1, \dots, a_k$, and let R be its output (a standardized representation for $\mathbf{B}$)
2: **for** $I \in \binom{[n]}{\mathsf{d}-1}$ **do**
3: **if** R contains no designated witness for $b|_I$ **then**
4: **return** NO
5: **end if**
6: **end for**
7: Run Algorithm 1 for IsRepresentable($\mathcal{K}$) with input b, R, to get output (A, S', S) with $\mathsf{A} = \text{YES}$ or $\mathsf{A} = \text{NO}$
8: **return** A

(from $\mathbf{B}$) for all derived forks of $\mathbf{B}$ in coordinates $\geq \mathsf{d}$, and possibly for some non-derived forks as well, but nothing else is added to R.

Lines 3–7 show that the loop in Steps 3–16 examines each pair $(\gamma, \beta) \in \mathbf{B}|_i \times \mathbf{B}|_i$ for every $i = \mathsf{d}, \dots, n$, finds $b = (b_1, \dots, b_n) \in R_0 \, (\subseteq R \subseteq B)$ such that $b_i = \gamma$, and checks — using SMP($\mathcal{K}$) — whether or not the tuple $c = (b_1, \dots, b_{i-1}, \beta^\gamma)$ is in the subalgebra $\mathbf{B}|_{[i]}$ generated by the elements $a_1|_{[i]}, \dots, a_k|_{[i]}$. If the answer is YES, then B contains a tuple of the form $(c, c_{i+1}, \dots, c_n) = (b_1, \dots, b_{i-1}, \beta^\gamma, c_{i+1}, \dots, c_n)$ for some $c_j \in A_j$ $(j = i+1, \dots, n)$,

which will be found, coordinate-by-coordinate, by repeated applications of SMP($\mathcal{K}$) in Steps 9–12. It is clear that in this case b, c are in $\mathbf{B}$ and witness that (γ, β^γ) is a fork in B, so these witnesses are correctly added to R in Step 13. This is the only step when R changes, so this shows that the R returned in Step 17 is a partial standardized representation for $\mathbf{B}$.

It remains to show that R contains designated witnesses for all derived forks of $\mathbf{B}$ in coordinates $\geq \mathsf{d}$. Let $\mathsf{d} \leq i \leq n$, and let $(\gamma, \beta^\gamma) \in \textsc{fork}'_i(B)$ with $(\gamma, \beta) \in \textsc{fork}_i(B)$. By statement (2) of Lemma 3.2, for every choice of $b \in \mathbf{B}$ with $b|_i = \gamma$ there exists an element $\widehat{b}$ such that b and $\widehat{b}$ are witnesses for the derived fork $(\gamma, \beta^\gamma) \in \textsc{fork}'_i(B)$, that is, such that $\widehat{b}|_{[i-1]} = b|_{[i-1]}$ and $\widehat{b}|_i = \beta^\gamma$. Hence, for the b and for the i-tuple $c = \widehat{b}|_{[i]}$ found in Steps 4 and 6 of Algorithm 5, SMP($\mathcal{K}$) in Step 7 has to give the answer YES, and the n-tuple c produced in Steps 8–12 and added to R in Step 13, along with b, to witness $(\gamma, \beta^\gamma) \in \textsc{fork}'_i(R)$ is such a $\widehat{b}$. This finishes the proof of the correctness of Algorithm 5.

In Step 1, Algorithm 2 runs in $O(n^{\mathsf{d}})$ time. Steps 3–16 require running SMP($\mathcal{K}$) $O(n^2)$ times on inputs not larger than the input for Algorithm 5, and adding one pair of witnesses to R no more than $O(n)$ times. This show that Algorithm 5 reduces CompactRep($\mathcal{K}$) to SMP($\mathcal{K}$) in $O(n^{\mathsf{d}})$ time.

Now we turn to Algorithm 6. The algorithm starts with a call to CompactRep($\mathcal{K}$) to compute a standardized representation R for the algebra $\mathbf{B}$ generated by the input tuples $a_1, \dots, a_k$. A necessary condition for the input tuple b to be in $\mathbf{B}$ is that $b|_I \in \mathbf{B}|_I$ for all $I \in \binom{[n]}{\mathsf{d}-1}$. Since R contains a full set of designated local witnesses for $\mathbf{B}$, b will satisfy this necessary condition if and only if R contains designated local witnesses for all projections $b|_I$ $\left(I \in \binom{[n]}{\mathsf{d}-1}\right)$ of b. This is being checked in Steps 2–6 of Algorithm 6; if the condition fails for some $I \in \binom{[n]}{\mathsf{d}-1}$, the algorithm returns the correct answer NO, meaning, $b \notin \mathbf{B}$.

If the algorithm passes Steps 2–6 without returning NO, then b, R is a correct input for IsRepresentable($\mathcal{K}$), which checks in Step 7 whether b is representable by R. Since every tuple representable by R must be in $\mathbf{B}$, and conversely, by Remark 4.3, every element of $\mathbf{B}$ is representable by R, we get that the YES/NO answer provided by IsRepresentable($\mathcal{K}$) is the correct answer to SMP($\mathcal{K}$) for the given input. This shows the correctness of Algorithm 6.

Steps 2–6 of Algorithm 6 run in $O(n^{\mathsf{d}-1})$ time, while IsRepresentable($\mathcal{K}$) in Step 7 requires $O(n^{\mathsf{d}+2})$ time. Thus, Algorithm 6 reduces SMP($\mathcal{K}$) to CompactRep($\mathcal{K}$) in $O(n^{\mathsf{d}+2})$ time. □

5. Structure Theory and the Subpower Membership Problem

Our global assumption for this section is the following:

Assumption 5.1. *$\mathcal{V}$ is a variety with a d-cube term, or equivalently, with a $(1, \mathsf{d}-1)$-parallelogram term ($\mathsf{d} > 1$).*

As in Section 3, we do not assume that the algebras we are considering are finite, or have a finite language, so the results in this section hold for arbitrary algebras.

One of the main results of [16] is a structure theorem for the critical subalgebras of finite powers of algebras with a cube (or parallelogram) term. In this section we adapt this structure theorem to find a new representation (different from compact representations) for subalgebras of products of algebras in a variety $\mathcal{V}$ with a cube term. In the next section, this representation will be used to prove the main result of the paper.

To restate the results from [16] that we need here, we introduce some terminology and notation. Let $\mathbf{R}$ be a subalgebra of a product $\mathbf{A}^{(1)} \times \cdots \times \mathbf{A}^{(n)}$ of some algebras $\mathbf{A}^{(1)}, \dots, \mathbf{A}^{(n)} \in \mathcal{V}$. Let $\mathbf{A}_i := \mathbf{R}|_i$ for each $i \in [n]$, and let $\mathbf{C} := \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$. So, $\mathbf{R}$ is a *subdirect subalgebra* of $\mathbf{C}$.

We say that $\mathbf{R}$ is a *critical subalgebra* of $\mathbf{A}^{(1)} \times \cdots \times \mathbf{A}^{(n)}$ if it has the following two properties:

- $\mathbf{R}$ is *completely $\cap$-irreducible* in the lattice of subalgebras of $\mathbf{A}^{(1)} \times \cdots \times \mathbf{A}^{(n)}$, that is, whenever $\mathbf{R}$ is an intersection of a family of subalgebras $\mathbf{S}_j$ ($j \in J$) of $\mathbf{A}^{(1)} \times \cdots \times \mathbf{A}^{(n)}$, we have that $\mathbf{R} = \mathbf{S}_j$ for some $j \in J$; and
- $\mathbf{R}$ is *directly indecomposable* in the following sense: $[n]$ cannot be partitioned into two nonempty sets I and J such that $\mathbf{R}$ and $\mathbf{R}|_I \times \mathbf{R}|_J$ differ only by a permutation of coordinates.

Note that if $\mathbf{R}$ is completely $\cap$-irreducible, it will not have a direct decomposition $\mathbf{R}|_I \times \mathbf{R}|_J$ (up to a permutation of coordinates) where $\mathbf{R}|_I$ and $\mathbf{R}|_J$ are both proper subalgebras of the corresponding products $\prod_{i\in I} \mathbf{A}^{(i)}$ and $\prod_{i\in J} \mathbf{A}^{(i)}$, respectively. However, $\mathbf{R}$ may be completely $\cap$-irreducible and still directly decomposable as $\mathbf{R}|_I \times \mathbf{R}|_J$ (up to a permutation of coordinates) if, say, $\mathbf{R}|_J = \prod_{i\in J} \mathbf{A}^{(i)}$ and $\mathbf{R}|_I$ is a completely $\cap$-irreducible subalgebra of $\prod_{i\in I} \mathbf{A}^{(i)}$.

Now let us assume that $\mathbf{R}$ is a critical subalgebra of $\mathbf{A}^{(1)} \times \cdots \times \mathbf{A}^{(n)}$. Continuing with the notation introduced in the second to last paragraph, let us choose and fix $\delta_i \in \mathrm{Con}(\mathbf{A}_i)$ ($i \in [n]$) such that $\delta := \delta_1 \times \cdots \times \delta_n$ is the largest product congruence of $\mathbf{C}$ with the property that $\mathbf{R}$ is a δ-saturated subalgebra of $\mathbf{C}$. (Such a congruence exists, because the join of product congruences of $\mathbf{C}$ is a product congruence, and if $\mathbf{R}$ is saturated with respect to a family of congruences of $\mathbf{C}$, then it is saturated with respect to their join.) With this notation, let $\overline{\mathbf{R}} := \mathbf{R}/\delta_{\mathbf{R}}$, and let $\overline{\mathbf{A}}_i := \mathbf{A}_i/\delta_i$ ($i \in [n]$); we call $\overline{\mathbf{R}}$ *the reduced representation of* $\mathbf{R}$.

Theorems 2.5 and 4.1 of [16] yield a structure theorem for the critical subalgebras of finite powers $\mathbf{A}^n$ of an arbitrary algebra $\mathbf{A} \in \mathcal{V}$. The relevant proofs in [16], namely the proofs of Theorem 2.5 (and its preparatory Lemmas 2.1, 2.3, 2.4) and Theorem 3.6 (part (3), implication $\Rightarrow$), carry over without any essential changes to the more general situation when instead of subalgebras of powers $\mathbf{A}^n$ with $\mathbf{A} \in \mathcal{V}$ we consider subalgebras of products $\mathbf{A}^{(1)} \times \cdots \times \mathbf{A}^{(n)}$ with $\mathbf{A}^{(1)}, \dots, \mathbf{A}^{(n)} \in \mathcal{V}$. Thus, we get the theorem below, where we state only those parts of the structure theorem that we need later on, retaining the numbering from [16, Theorem 2.5]. The superscript $^\flat$ in $(6)^\flat$ indicates that instead of the original condition (6) we state a weaker condition which is sufficient for our purposes.

Theorem 5.2 (Cf. [16])**.** *Let $\mathbf{A}^{(1)}, \dots, \mathbf{A}^{(n)} \in \mathcal{V}$, and let $\overline{\mathbf{R}}$ be the reduced representation of a critical subalgebra $\mathbf{R}$ of $\mathbf{A}^{(1)} \times \cdots \times \mathbf{A}^{(n)}$. If $n \geq \mathsf{d}$, then the following hold.*

(1) *$\overline{\mathbf{R}} \leq \prod_{i=1}^n \overline{\mathbf{A}}_i$ is a representation of $\overline{\mathbf{R}}$ as a subdirect product of subdirectly irreducible algebras $\overline{\mathbf{A}}_i$.*

$(6)^\flat$ *$\overline{\mathbf{A}}_i$ and $\overline{\mathbf{A}}_j$ are similar for any $i, j \in [n]$ (see Section 2.3).*

(7) *If $n > 2$, then each $\overline{\mathbf{A}}_i$ has abelian monolith μ_i ($i \in [n]$).*

(8) *For the centralizers $\rho_\ell := (0 : \mu_\ell)$ of the monoliths μ_ℓ ($\ell \in [n]$), the image of the composite map*

$$\overline{\mathbf{R}} \overset{\mathrm{proj}_{ij}}{\to} \overline{\mathbf{A}}_i \times \overline{\mathbf{A}}_j \twoheadrightarrow \overline{\mathbf{A}}_i/\rho_i \times \overline{\mathbf{A}}_j/\rho_j.$$

is the graph of an isomorphism $\overline{\mathbf{A}}_i/\rho_i \to \overline{\mathbf{A}}_j/\rho_j$ for any distinct $i, j \in [n]$.

Note that the homomorphism in part (8) is the same as in [16, Theorem 2.5 (8)]; the slightly different description presented here will be more convenient later on.

Now we are ready to discuss our representation theorem. Let $\mathbf{B}_1, \dots, \mathbf{B}_n$ be nontrivial algebras in $\mathcal{V}$, and let $\mathbf{B}$ be a subdirect subalgebra of $\mathbf{B}_1 \times \cdots \times \mathbf{B}_n$.

First, we replace each $\mathbf{B}_j$ ($j \in [n]$) by its image under the embedding

$$\mathbf{B}_j \hookrightarrow \prod_{\sigma \in \mathrm{Irr}(\mathbf{B}_j)} \mathbf{B}_j/\sigma, \qquad x_j \mapsto (x_j/\sigma)_{\sigma \in \mathrm{Irr}(\mathbf{B}_j)}; \tag{5.1}$$

thus, each $\mathbf{B}_j$ is replaced by a subdirect product of all of its subdirectly irreducible quotients $\mathbf{B}_j/\sigma$. To set up a more convenient notation, let

$$W := \{(j, \sigma) : j \in [n], \sigma \in \mathrm{Irr}(\mathbf{B}_j)\},$$

and for each $j \in [n]$, let $W_j := \{j\} \times \mathrm{Irr}(\mathbf{B}_j)$; thus, $W = W_1 \cup \cdots \cup W_n$. Furthermore, for each $w = (j, \sigma) \in W$ let $\widehat{\mathbf{B}}_w := \mathbf{B}_j/\sigma$, and for every element x_j in $\mathbf{B}_j$ let $\widehat{x}_w := x_j/\sigma$. Then the product of the embeddings (5.1) for all $j \in [n]$ yields an embedding

$$\widehat{\ }\colon \prod_{j \in [n]} \mathbf{B}_j \hookrightarrow \prod_{w \in W} \widehat{\mathbf{B}}_w, \qquad x = (x_j)_{j \in [n]} \mapsto \widehat{x} := (\widehat{x}_w)_{w \in W}.$$

We will denote the image of $\mathbf{B}$ under this embedding $\widehat{\ }$ by $\widehat{\mathbf{B}}$. By construction, $\widehat{\mathbf{B}}$ is a subdirect product of the subdirectly irreducible algebras $\widehat{\mathbf{B}}_w$ ($w \in W$); in particular, $\widehat{\mathbf{B}}_w = \widehat{\mathbf{B}}|_w$ for all $w \in W$. For each $w \in W$ let μ_w denote the monolith of $\widehat{\mathbf{B}}_w$ and ρ_w its centralizer $(0 : \mu_w)$.

Next we define a relation $\sim$ on W as follows: we require $\sim$ to be reflexive, and for distinct $v, w \in W$ we define $v \sim w$ to hold if and only if

- the subdirectly irreducible algebras $\widehat{\mathbf{B}}_v$ and $\widehat{\mathbf{B}}_w$ are similar with abelian monoliths μ_v and μ_w, and
- the image of $\widehat{\mathbf{B}}|_{vw}$ under the natural map $\widehat{\mathbf{B}}_v \times \widehat{\mathbf{B}}_w \twoheadrightarrow (\widehat{\mathbf{B}}_v/\rho_v) \times (\widehat{\mathbf{B}}_w/\rho_w)$ is the graph of an isomorphism $\widehat{\mathbf{B}}_v/\rho_v \to \widehat{\mathbf{B}}_w/\rho_w$.

It is easy to see that $\sim$ is an equivalence relation on W.

Our representation theorem describes the algebra $\mathbf{B}$ in terms of its image $\widehat{\mathbf{B}}$, namely, it shows that $\widehat{\mathbf{B}}$ is determined by its projections onto small sets of coordinates (i.e., small subsets of W) and by its projections onto the blocks of $\sim$. A block of $\sim$ may be large, but the image of $\widehat{\mathbf{B}}$ under a projection onto a block of $\sim$ has a special structure.

Theorem 5.3. *Let $\mathcal{V}$ be a variety with a d-cube term, let $\mathbf{B}_1, \dots, \mathbf{B}_n$ be nontrivial algebras in $\mathcal{V}$, and let $\mathbf{B}$ be a subdirect subalgebra of $\mathbf{B}_1 \times \cdots \times \mathbf{B}_n$. Furthermore, let W, $\widehat{\ }$, $\widehat{\mathbf{B}}_w$ ($w \in W$), $\widehat{\mathbf{B}}$, and $\sim$ be as defined above. Then, for any tuple $c \in \mathbf{B}_1 \times \cdots \times \mathbf{B}_n$, the following conditions are equivalent:*

(1) $c \in \mathbf{B}$.
(2) *c satisfies*
   - *$c|_I \in \mathbf{B}|_I$ for all $I \subseteq [n]$ with $|I| < \max\{\mathsf{d}, 3\}$, and*
   - *$\widehat{c}|_U \in \widehat{\mathbf{B}}|_U$ for all blocks U ($\subseteq W$) of $\sim$ of size $|U| \geq \max\{\mathsf{d}, 3\}$.*
(3) *$\widehat{c}|_U \in \widehat{\mathbf{B}}|_U$ for all $U \subseteq W$ such that*
   - *$|U| < \mathsf{m} \cdot \max\{\mathsf{d}, 3\}$ where $\mathsf{m} = \max\{|\mathrm{Irr}(\mathbf{B}_j)| : j \in [n]\}$, or*
   - *U is a block of $\sim$ of size $|U| \geq \mathsf{m} \cdot \max\{\mathsf{d}, 3\}$.*

Remark 5.4. The equivalence of conditions (1) and (3) in Theorem 5.3 can be restated as follows: $\widehat{\mathbf{B}}$ is the intersection of the subalgebras

$$\mathrm{proj}_U^{-1}\big(\mathrm{proj}_U(\widehat{\mathbf{B}})\big) = \mathrm{proj}_U^{-1}\big(\widehat{\mathbf{B}}|_U\big)$$

of $\prod_{w\in W}\widehat{\mathbf{B}}_w$ as U runs over the subsets of W listed in (3).

Proof of Theorem 5.3. Since $\widehat{\ }$ and $|_U$ ($U \subseteq W$) are homomorphisms, it is clear that $c \in \mathbf{B}$ implies $\widehat{c}|_U \in \widehat{\mathbf{B}}|_U$ for all $U \subseteq W$. This proves (1) $\Rightarrow$ (3).

For the implication (3) $\Rightarrow$ (2), assume that (3) holds. Then $\widehat{c}|_U \in \widehat{\mathbf{B}}|_U$ for all blocks U of $\sim$, so the second statement in (2) holds. To establish the first statement, choose $I \subseteq [n]$ such that $|I| < \max\{\mathsf{d}, 3\}$, and let $W_I := \bigcup_{j\in I} W_j$. Since the product of the isomorphisms $\mathbf{B}_j \to \widehat{\mathbf{B}}|_{W_j}$, $x_j \mapsto (\widehat{x}_w)_{w\in W_j}$ (induced by the embeddings in (5.1)) yields an isomorphism

$$\prod_{j\in I}\mathbf{B}_j \to \prod_{j\in I}\widehat{\mathbf{B}}|_{W_j}\Big(\leq \prod_{w\in W_I}\widehat{\mathbf{B}}_w\Big), \quad (x_j)_{j\in I} \mapsto (x_w)_{w\in W_I} \qquad (x_j \in \mathbf{B}_j),$$

which maps $\mathbf{B}|_I$ onto $\widehat{\mathbf{B}}|_{W_I}$, we get that $c|_I \in \mathbf{B}|_I$ holds if and only if $\widehat{c}|_{W_I} \in \widehat{\mathbf{B}}|_{W_I}$. The latter follows from assumption (3), because $|W_I| \leq \sum_{j\in I}|W_j| = \sum_{j\in I}|\operatorname{Irr}(\mathbf{B}_j)| \leq \mathsf{m}|I|$. This completes the proof of (3) $\Rightarrow$ (2).

The remaining implication (2) $\Rightarrow$ (1) is the main statement in Theorem 5.3, which we will prove now. Assume that $c \notin \mathbf{B}$, but $c|_I \in \mathbf{B}|_I$ for all $I \subseteq [n]$ with $|I| < \max\{\mathsf{d}, 3\}$. We have to show that $\widehat{c}|_U \notin \widehat{\mathbf{B}}|_U$ for some block U ($\subseteq W$) of $\sim$ of size $|U| \geq \max\{\mathsf{d}, 3\}$.

Using the assumption $c \notin \mathbf{B}$ and Zorn's Lemma, we first choose and fix a subalgebra $\mathbf{M}$ of $\mathbf{B}_1 \times \cdots \times \mathbf{B}_n$ containing $\mathbf{B}$, which is maximal for the property that it fails to contain c. Then $\mathbf{M}$ is completely $\cap$-irreducible in the lattice of subalgebras of $\mathbf{B}_1 \times \cdots \times \mathbf{B}_n$. Let $\{T_1, \dots, T_\ell\}$ be a partition of $[n]$ such that $\mathbf{M}|_{T_i}$ is directly indecomposable for every $i \in [\ell]$, and $\mathbf{M}$ differs from $\mathbf{M}|_{T_1} \times \cdots \times \mathbf{M}|_{T_\ell}$ by a permutation of variables only; we will denote this fact by $\mathbf{M} \approx \mathbf{M}|_{T_1} \times \cdots \times \mathbf{M}|_{T_\ell}$. We must have $c|_T \notin \mathbf{M}|_T$ for at least one block $T := T_i$, because otherwise $\mathbf{M} \approx \mathbf{M}|_{T_1} \times \cdots \times \mathbf{M}|_{T_\ell}$ would imply that $c \in \mathbf{M}$, contradicting the choice of $\mathbf{M}$. Let us fix such a T for the rest of the proof. Note that $|T| > 1$, because $\mathbf{M}$ is a subdirect product of $\mathbf{B}_1, \dots, \mathbf{B}_n$ (as $\mathbf{B} \leq \mathbf{M}$), so we have $c|_j \in \mathbf{B}_j = \mathbf{M}|_j$ for every one-element set $\{j\} \subseteq [n]$.

It follows from $\mathbf{M} \approx \mathbf{M}|_{T_1} \times \cdots \times \mathbf{M}|_{T_\ell}$ that $\mathbf{M}$ is the intersection of two subalgebras of $\mathbf{B}_1 \times \cdots \times \mathbf{B}_n$, as shown below:

$$\mathbf{M} \approx \Big(\mathbf{M}|_T \times \prod_{j\in[n]\setminus T}\mathbf{B}_j\Big) \cap \Big(\prod_{j\in T}\mathbf{B}_j \times \mathbf{M}|_{[n]\setminus T}\Big).$$

Since $\mathbf{M}$ is a completely $\cap$-irreducible subalgebra of $\mathbf{B}_1 \times \cdots \times \mathbf{B}_n$, we get that $\mathbf{M} \approx \mathbf{M}|_T \times \prod_{j\in[n]\setminus T}\mathbf{B}_j$ or $\mathbf{M} \approx \prod_{j\in T}\mathbf{B}_j \times \mathbf{M}|_{[n]\setminus T}$. The latter is impossible, because $\mathbf{M}|_T \subsetneq \prod_{j\in T}\mathbf{B}_j$ (as witnessed by $c|_T$). Hence, for the subalgebra $\mathbf{R} := \mathbf{M}|_T$ of $\prod_{j\in T}\mathbf{B}_j$ we get that $\mathbf{M} \approx \mathbf{R} \times \prod_{j\in[n]\setminus T}\mathbf{B}_j$. Furthermore, $\mathbf{R}$ is both completely $\cap$-irreducible (because $\mathbf{M}$ is) and directly indecomposable (by construction), so $\mathbf{R}$ is a critical subalgebra of $\prod_{j\in T}\mathbf{B}_j$. Our construction also implies that $\mathbf{B}|_T \leq \mathbf{R}$ and $c|_T \notin \mathbf{R}$. Hence, $c|_T \notin \mathbf{B}|_T$. Thus, our assumption that $c|_I \in \mathbf{B}|_I$ holds for all $I \subseteq [n]$ with $|I| < \max\{\mathsf{d}, 3\}$ forces that $|T| \geq \max\{\mathsf{d}, 3\}$.

Now we can apply Theorem 5.2 to the algebras $\mathbf{B}_t$ ($t \in T$) in $\mathcal{V}$, and the critical subalgebra $\mathbf{R}$ of $\prod_{t\in T}\mathbf{B}_t$ where the number of factors in the product is $|T| \geq \max\{\mathsf{d}, 3\}$.

Since $\mathbf{R}$ is a subdirect product of the algebras $\mathbf{B}_t$ ($t \in T$), the reduced representation $\overline{\mathbf{R}}$ of $\mathbf{R}$ is the quotient algebra $\mathbf{R}/\delta_{\mathbf{R}}$ where $\delta = \prod_{t\in T} \delta_t$ ($\delta_t \in \mathrm{Con}(\mathbf{B}_t)$) is the largest product congruence of $\prod_{t\in T} \mathbf{B}_t$ for which $\mathbf{R}$ is δ-saturated. Let $\overline{\mathbf{B}}_t := \mathbf{B}_t/\delta_t$ for every $t \in T$. So, the conclusions of Theorem 5.2 can be restated as follows:

(1) $\overline{\mathbf{R}} \le \prod_{t\in T} \overline{\mathbf{B}}_t$ is a representation of $\overline{\mathbf{R}}$ as a subdirect product of subdirectly irreducible algebras $\overline{\mathbf{B}}_t$.

(6)$^\flat$ $\overline{\mathbf{B}}_s$ and $\overline{\mathbf{B}}_t$ are similar for any $s, t \in T$.

(7) Each $\overline{\mathbf{B}}_t$ has abelian monolith μ_t ($t \in T$).

(8) For the centralizers $\rho_\ell := (0 : \mu_\ell)$ of the monoliths μ_ℓ ($\ell \in T$), the image of the composite map

$$\overline{\mathbf{R}} \overset{\mathrm{proj}_{st}}{\to} \overline{\mathbf{B}}_s \times \overline{\mathbf{B}}_t \twoheadrightarrow \overline{\mathbf{B}}_s/\rho_s \times \overline{\mathbf{B}}_t/\rho_t \tag{5.2}$$

is the graph of an isomorphism $\overline{\mathbf{B}}_s/\rho_s \to \overline{\mathbf{B}}_t/\rho_t$ for any distinct $s, t \in T$.

By conclusion (1), we have for each $t \in T$ that $\overline{\mathbf{B}}_t = \mathbf{B}_t/\delta_t$ is subdirectly irreducible, so $\delta_t \in \mathrm{Irr}(\mathbf{B}_t)$ and $(t, \delta_t) \in W$. Hence, the algebra $\overline{\mathbf{B}}_t = \mathbf{B}_t/\delta_t$ is one of the subdirect factors of $\widehat{\mathbf{B}}$, namely, $\overline{\mathbf{B}}_t = \mathbf{B}_t/\delta_t = \widehat{\mathbf{B}}_w$ for $w = (t, \delta_t)$. This implies also that μ_t, ρ_t are the congruences of $\overline{\mathbf{B}}_t = \widehat{\mathbf{B}}_w$ that we denoted earlier by μ_w, ρ_w.

Let $\widehat{T} := \{(t, \delta_t) : t \in T\}$. Next we want to show that any two elements of $\widehat{T}$ are related by $\sim$. Let $s, t \in T$, and let $v := (s, \delta_s)$, $w := (t, \delta_t)$. As we noticed in the preceding paragraph, we have that $\overline{\mathbf{B}}_s = \widehat{\mathbf{B}}_v$, $\mu_s = \mu_v$, $\rho_s = \rho_v$, and $\overline{\mathbf{B}}_t = \widehat{\mathbf{B}}_w$, $\mu_t = \mu_w$, $\rho_t = \rho_w$. If $s = t$, then $v = w$, and hence $v \sim w$ holds because $\sim$ is reflexive. So, assume from now on that $s \ne t$. Hence $v \ne w$. In this case, checking whether $v \sim w$ holds involves two conditions. One is that the subdirectly irreducible algebras $\widehat{\mathbf{B}}_v$ and $\widehat{\mathbf{B}}_w$ are similar with abelian monoliths μ_v and μ_w, which follows from conclusions (6)$^\flat$–(7). The other is that the image of $\widehat{\mathbf{B}}|_{vw}$ under the natural map $\varphi\colon \widehat{\mathbf{B}}_v \times \widehat{\mathbf{B}}_w \twoheadrightarrow (\widehat{\mathbf{B}}_v/\rho_v) \times (\widehat{\mathbf{B}}_w/\rho_w)$, is the graph of an isomorphism $\widehat{\mathbf{B}}_v/\rho_v \to \widehat{\mathbf{B}}_w/\rho_w$. We will establish this property by proving that $\widetilde{\mathbf{B}}_{vw} := \varphi(\widehat{\mathbf{B}}|_{vw})$ is equal to the image of $\overline{\mathbf{R}}$ under the homomorphism in (5.2), which is $\varphi(\overline{\mathbf{R}}|_{st})$, because the map $\twoheadrightarrow$ in (5.2) is φ. Let $\widetilde{\mathbf{R}}_{st} := \varphi(\overline{\mathbf{R}}|_{st})$. Since $\widehat{\mathbf{B}}|_{vw}$ is a subdirect product of $\widehat{\mathbf{B}}_v$ and $\widehat{\mathbf{B}}_w$, we get that $\widetilde{\mathbf{B}}_{vw}$ is a subdirect product of $\widehat{\mathbf{B}}_v/\rho_v$ and $\widehat{\mathbf{B}}_w/\rho_w$. By the construction of $\mathbf{R}$, we have that $\mathbf{R} \ge \mathbf{B}|_T$, and $\ge$ is preserved under the natural homomorphism

$$\prod_{r\in T} \mathbf{B}_r \twoheadrightarrow \prod_{r\in T} \mathbf{B}_r/\delta_r = \prod_{u\in\widehat{T}} \widehat{\mathbf{B}}_u. \tag{5.3}$$

The images of $\mathbf{R}$ and $\mathbf{B}|_T$ under this homomorphism are $\overline{\mathbf{R}}$ and $\widehat{\mathbf{B}}|_{\widehat{T}}$, respectively, hence we conclude that $\overline{\mathbf{R}} \ge \widehat{\mathbf{B}}|_{\widehat{T}}$. Projecting further onto the coordinates s, t in T, and the corresponding coordinates $v = (s, \delta_s)$, $w = (t, \delta_t)$ in $\widehat{T}$, we get that $\overline{\mathbf{R}}|_{st} \ge \widehat{\mathbf{B}}|_{vw}$. Hence, it follows that $\widetilde{\mathbf{R}}_{st} = \varphi(\overline{\mathbf{R}}|_{st}) \ge \varphi(\widehat{\mathbf{B}}|_{vw}) = \widetilde{\mathbf{B}}_{vw}$. By conclusion (8) above, $\widetilde{\mathbf{R}}_{st}$ is the graph of an isomorphism $\overline{\mathbf{B}}_s/\rho_s \to \overline{\mathbf{B}}_t/\rho_t$, or equivalently, the graph of an isomorphism $\widehat{\mathbf{B}}_v/\rho_v \to \widehat{\mathbf{B}}_w/\rho_w$. Combining this fact with the earlier observation that $\widetilde{\mathbf{B}}_{vw}$ is a subdirect product of $\widehat{\mathbf{B}}_v/\rho_v$ and $\widehat{\mathbf{B}}_w\rho_w$, we obtain that $\widetilde{\mathbf{R}}_{st}$ and $\widetilde{\mathbf{B}}_{vw}$ must be equal. In particular, $\widetilde{\mathbf{B}}_{vw}$ is the graph of an isomorphism $\widehat{\mathbf{B}}_v/\rho_v \to \widehat{\mathbf{B}}_w/\rho_w$, and hence $v \sim w$.

Our arguments in the last two paragraphs show that $\widehat{T}$ is contained in one of the blocks U of $\sim$. We have $|U| \ge |\widehat{T}| = |T| \ge \max\{\mathtt{d}, 3\}$. It remains to verify that $\widehat{c}|_U \notin \widehat{\mathbf{B}}|_U$.

Assume, for a contradiction, that $\widehat{c}|_U \in \widehat{\mathbf{B}}|_U$. Then projecting further to $\widehat{T} \subseteq U$ yields that $\widehat{c}|_{\widehat{T}} \in \widehat{\mathbf{B}}|_{\widehat{T}}$. As we saw earlier, $\widehat{\mathbf{B}}|_{\widehat{T}} \leq \overline{\mathbf{R}}$, therefore we get that the tuple $\widehat{c}|_{\widehat{T}} = (c_r/\delta_r)_{r\in T}$ lies in $\overline{\mathbf{R}}$. Hence the tuple $c|_T = (c_r)_{r\in T}$ lies in the full inverse image of $\overline{\mathbf{R}}$ under the natural homomorphism (5.3). This inverse image is $\mathbf{R}$, because $\overline{\mathbf{R}} = \mathbf{R}/\delta_{\mathbf{R}}$ with $\delta = \prod_{r\in T} \delta_r$, and $\mathbf{R}$ is δ-saturated in $\prod_{r\in T} \mathbf{B}_r$. Thus, we obtain that $c|_T \in \mathbf{R}$, which is impossible, since $\mathbf{R}$ was chosen so that $c|_T \notin \mathbf{R}$. This contradiction proves that $\widehat{c}|_U \notin \widehat{\mathbf{B}}|_U$, and completes the proof of Theorem 5.3. □

6. Algorithms Based on Theorem 5.3

Throughout this section we will work under the following global assumptions.

Assumption 6.1.
- *$\mathcal{V}$ is a variety in a finite language with a d-cube term ($\mathsf{d} > 1$),*
- *$\mathcal{K}$ is a finite set of finite algebras in $\mathcal{V}$.*

Definition 6.2. Let $a_1, \dots, a_k, b \in \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ ($\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$) be an input for $\mathrm{SMP}(\mathcal{K})$ where $a_r = (a_{r1}, \dots, a_{rn})$ ($r \in [k]$) and $b = (b_1, \dots, b_n)$. We call this input d-*coherent* if the following conditions are satisfied:

(1) $n \geq \max\{\mathsf{d}, 3\}$;
(2) $a_1, \dots, a_k$ generate a subdirect subalgebra of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$;
(3) $\mathbf{A}_1, \dots, \mathbf{A}_n$ are similar subdirectly irreducible algebras, and each $\mathbf{A}_\ell$ has abelian monolith μ_ℓ;
(4) $b|_I$ is in the subalgebra of $\prod_{i\in I} \mathbf{A}_i$ generated by $\{a_1|_I, \dots, a_k|_I\}$ for all $I \subseteq [n]$, $|I| < \max\{\mathsf{d}, 3\}$; and
(5) for the centralizers $\rho_\ell := (0 : \mu_\ell)$ of the monoliths μ_ℓ, the subalgebra of $\mathbf{A}_i/\rho_i \times \mathbf{A}_j/\rho_j$ generated by $\{(a_{1i}/\rho_i, a_{1j}/\rho_j), \dots, (a_{ki}/\rho_i, a_{kj}/\rho_j)\}$ is the graph of an isomorphism $\mathbf{A}_i/\rho_i \to \mathbf{A}_j/\rho_j$ for any distinct $i, j \in [n]$.

Definition 6.3. We define $\mathrm{SMP}_{\mathsf{d}\text{-coh}}(\mathcal{K})$ to be the restriction of $\mathrm{SMP}(\mathcal{K})$ to d-coherent inputs.

It is clear from Definition 6.2 that d-coherence for inputs of $\mathrm{SMP}(\mathcal{K})$ can be checked in polynomial time.

Theorem 6.4. *The decision problems* $\mathrm{SMP}(\mathcal{K})$ *and* $\mathrm{SMP}_{\mathsf{d}\text{-}coh}(\mathbb{HSK})$ *are polynomial time equivalent.*

Proof. By Theorem 4.10, $\mathrm{SMP}(\mathcal{K})$ is polynomial time equivalent to $\mathrm{SMP}(\mathbb{HSK})$. Clearly, $\mathrm{SMP}_{\mathsf{d}\text{-coh}}(\mathbb{HSK})$ is polynomial time reducible to $\mathrm{SMP}(\mathbb{HSK})$, because it is a subproblem of $\mathrm{SMP}(\mathbb{HSK})$. Therefore it only remains to show that Algorithm 7 reduces $\mathrm{SMP}(\mathcal{K})$ to $\mathrm{SMP}_{\mathsf{d}\text{-coh}}(\mathbb{HSK})$ in polynomial time.

The correctness of Algorithm 7 is based on Theorem 5.3. Let $\mathbf{B}$ denote the subalgebra of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ generated by $b_1, \dots, b_k$. Without loss of generality we may assume that $n \geq \mathsf{d}$. In Steps 1–5, it is checked whether $b_{k+1}|_I$ is contained in the projection $\mathbf{B}|_I$ of $\mathbf{B}$ for each $I \in \binom{[n]}{\mathsf{d}-}$. If not, then clearly $b_{k+1} \notin \mathbf{B}$ and the algorithm correctly returns the answer NO in Step 4. In Step 6 the algebras $\mathbf{B}_i := \mathbf{B}|_i$ are computed for every $i \in [n]$. Note that by this time, b_{k+1} had passed the tests in Step 4, which implies that $b_{k+1}|_i \in \mathbf{B}_i$ holds for every $i \in [n]$. Therefore, when in Step 7 all coordinates with $|B_i| = 1$ are omitted from the input

Algorithm 7 Reduction of $\mathrm{SMP}(\mathcal{K})$ to $\mathrm{SMP}_{\mathsf{d}\text{-coh}}(\mathbb{HSK})$

Require: $b_1, \dots, b_k, b_{k+1} \in \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ with $\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$ $(n \geq \mathsf{d})$
Ensure: Is b_{k+1} in the subalgebra of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ generated by $b_1, \dots, b_k$?
1: $\mathsf{d}^- := \max(\mathsf{d}-1, 2)$,
2: **for** $I \in \binom{[n]}{\mathsf{d}^-}$ **do**
3: generate $\mathbf{B}|_I$ by $b_1|_I, \dots, b_k|_I$
4: **return** NO if $b_{k+1}|_I \notin \mathbf{B}|_I$
5: **end for**
6: generate $\mathbf{B}_i$ by $b_1|_i, \dots, b_k|_i$ for $i \in [n]$
7: omit all coordinates i from the input for which $|\mathbf{B}_i| = 1$ (but keep earlier notation)
8: **return** YES if $n \leq \mathsf{d}^-$
9: $W := \{(j, \sigma) : j \in [n], \sigma \in \mathrm{Irr}(\mathbf{B}_j)\}$
10: $\widehat{\mathbf{B}}_w := \mathbf{B}_j/\sigma$, $\mu_w :=$ monolith of $\widehat{\mathbf{B}}_w$, $\rho_w := (0 : \mu_w)$ for each $w = (j, \sigma) \in W$
11: $\widehat{b}_i := (\widehat{b}_{iw})_{w \in W}$ where $\widehat{b}_{iw} := b_i|_j/\sigma$ for each $i \in [k+1]$ and $w = (j, \sigma)$
12: **for** distinct $v, w \in W$ **do**
13: generate $\widetilde{\mathbf{B}}_{vw}$ by the pairs $(\widehat{b}_{iv}/\rho_v, \widehat{b}_{iw}/\rho_w)$, $i \in [k]$
14: **end for**
15: compute the equivalence relation $\sim$ on W determined by the following condition: for distinct $v, w \in W$,

$$v \sim w \quad \Leftrightarrow \quad \begin{cases} \mu_v \leq \rho_v,\ \mu_w \leq \rho_w,\ \widehat{B}_v \text{ and } \widehat{B}_w \text{ are similar, and} \\ \widetilde{\mathbf{B}}_{vw} \text{ is the graph of an isomorphism } \widehat{\mathbf{B}}_v/\rho_v \to \widehat{\mathbf{B}}_w/\rho_w \end{cases}$$

16: find the equivalence classes $E_1, \dots, E_m$ of $\sim$ of size $> \mathsf{d}^-$
17: **for** $\lambda = 1, \dots, m$ **do**
18: run $\mathrm{SMP}_{\mathsf{d}\text{-coh}}(\mathbb{HSK})$ with input $\widehat{b}_1|_{E_\lambda}, \dots, \widehat{b}_k|_{E_\lambda}, \widehat{b}_{k+1}|_{E_\lambda} \in \prod_{w \in E_\lambda} \widehat{\mathbf{B}}_w$ to get answer $\mathsf{A} \in \{\mathrm{YES}, \mathrm{NO}\}$
19: **return** NO if $\mathsf{A} = \mathrm{NO}$
20: **end for**
21: **return** YES

tuples (but the earlier notation is kept for simplicity), this deletion of the trivial coordinates does no affect whether or not $b_{k+1} \in \mathbf{B}$. It follows that by the end of Step 7 we have that
(1) $\mathbf{B}$ is a subdirect subalgebra of $\mathbf{B}_1 \times \cdots \times \mathbf{B}_n$ where $\mathbf{B}_1, \dots, \mathbf{B}_n$ are nontrivial, and
(2) $b_{k+1}|_I \in \mathbf{B}|_I$ for all $I \in \binom{[n]}{\mathsf{d}^-}$.

By (1) the algebra $\mathbf{B}$ satisfies the assumptions of Theorem 5.3, and (2) shows that the tuple $c := b_{k+1} \in \mathbf{B}_1 \times \cdots \times \mathbf{B}_n$ satisfies the first part of condition (2) in Theorem 5.3.

In Steps 9–16, Algorithm 7 computes the data needed to check whether or not $c = b_{k+1}$ satisfies the second part of condition (2) as well. In more detail, the algorithm computes the set W, the subdirectly irreducible algebras $\widehat{\mathbf{B}}_w$ $(w \in W)$, their congruences μ_w (the monolith) and $\rho_w = (0 : \mu_w)$, the images $\widehat{b}_i$ of the input tuples b_i $(i \in [k+1])$ under the homomorphism $\widehat{\ }$, and finally, the equivalence relation $\sim$ on W, and its equivalence classes $E_1, \dots, E_m$ of size $> \mathsf{d}^-$. All computations follow the definitions exactly, except Step 15. We explain now why the conditions used in Step 15 to compute $\sim$ are equivalent to the conditions in the definition of $\sim$ (stated right before Theorem 5.3):

- The inclusion $\mu_w \leq \rho_w$ ($w \in W$) is equivalent to the condition that μ_w is abelian, because the inclusion is true if μ_w is abelian, whereas $\rho_w = 0$ (and hence the inclusion fails) if μ_w is nonabelian.
- As discussed in the proof of Theorem 5.3, $\widetilde{B}_{vw}$ is the image of $\widehat{B}|_{vw}$ under the natural map $\widehat{\mathbf{B}}_v \times \widehat{\mathbf{B}}_w \twoheadrightarrow (\widehat{\mathbf{B}}_v/\rho_v) \times (\widehat{\mathbf{B}}_w/\rho_w)$.

Since $c = b_{k+1}$ satisfies the first part of condition (2) in Theorem 5.3, it follows from the equivalence of conditions (1) and (2) in Theorem 5.3 that $b_{k+1} \in \mathbf{B}$ if and only if $c = b_{k+1}$ satisfies the second part of condition (2) as well, that is,

(3) $\widehat{b}_{k+1}|_{E_\lambda} \in \widehat{\mathbf{B}}|_{E_\lambda}$ for all $\lambda \in [m]$.

This is clearly equivalent to the condition that

(3)′ $\widehat{b}_{k+1}|_{E_\lambda}$ belongs to the subalgebra of $\prod_{w \in E_\lambda} \widehat{B}_w$ generated by the tuples $\widehat{b}_1|_{E_\lambda}, \dots, \widehat{b}_k|_{E_\lambda}$ for all $\lambda \in [m]$.

It follows from the construction that for each $\lambda \in [m]$, $\widehat{b}_1|_{E_\lambda}, \dots, \widehat{b}_k|_{E_\lambda}, \widehat{b}_{k+1}|_{E_\lambda}$ is a d-coherent input for $\mathrm{SMP}(\mathbb{HSK})$, so condition (3)′ can be checked using $\mathrm{SMP}_{\mathsf{d}\text{-coh}}(\mathbb{HSK})$. This is exactly what Algorithm 7 does in Steps 17–20, and returns the correct answer: YES if (3)′ holds (including the case when $m = 0$), and NO otherwise. This completes the proof of the correctness of Algorithm 7.

Now we show that Algorithm 6 reduces $\mathrm{SMP}(\mathcal{K})$ to $\mathrm{SMP}_{\mathsf{d}\text{-coh}}(\mathbb{HSK})$ in polynomial time. Clearly, Steps 1, 8, and 21 require constant time. Since each $\mathbf{B}_i$ ($i \in [n]$) is a subalgebra of some member of $\mathcal{K}$, we have $|\mathbf{B}_i| \leq \mathsf{a}_\mathcal{K}$ where the constant $\mathsf{a}_\mathcal{K}$ is independent of the input. The parameter d is also independent of the input, and so is $\mathsf{s} := \max\{|\operatorname{Irr}(\mathbf{A})| : \mathbf{A} \in \mathbb{SK}\}$. It follows that $|W| \leq n\mathsf{s}$ and that each iteration of the **for** loops in Steps 2–5 and 12–14 require time $O(k)$. Hence, Steps 2–5, 6–7, 9–10, 11, and 12–14 run in $O(kn^{\mathsf{d}^-})$, $O(kn)$, $O(n)$, $O(kn)$, and $O(kn^2)$ time, respectively.

In Step 15, to determine whether $v \sim w$ holds for a particular pair of elements $v, w \in W$ requires constant time, because the condition only involves data on algebras in $\mathbb{HSK}$ and on products of two such algebras. (In particular, recall from Section 2 that similarity of $\widehat{B}_v$ and $\widehat{B}_w$ can be checked by looking at congruences of subalgebras of $\widehat{B}_v \times \widehat{B}_w$.) Thus, Step 15 runs in $O(n^2)$ time as does Step 16. Since $E_1, \dots, E_m$ are disjoint subsets of W and $|W| \leq n\mathsf{s}$, we get that $m \leq n\mathsf{s}$ and each E_λ has size $|E_\lambda| \leq n\mathsf{s}$. Thus, in Steps 17–20, $\mathrm{SMP}_{\mathsf{d}\text{-coh}}(\mathbb{HSK})$ has to be run at most $O(n)$ times, and the input size of each run is $O(kn)$, approximately the same as the size of the original input.

This proves that Algorithm 7 reduces $\mathrm{SMP}(\mathcal{K})$ to $\mathrm{SMP}_{\mathsf{d}\text{-coh}}(\mathbb{HSK})$ in polynomial time. □

Theorem 6.5. *If, in addition to Assumption 6.1, the variety $\mathcal{V}$ is residually small, then*

$$\mathrm{SMP}(\mathcal{K}) \in \mathsf{P}.$$

Proof. By Theorem 6.4, $\mathrm{SMP}(\mathcal{K})$ is polynomial time equivalent to $\mathrm{SMP}_{\mathsf{d}\text{-coh}}(\mathbb{HSK})$. Also, if $\mathcal{K}$ is finite, so is $\mathbb{HSK}$. Therefore it suffices to prove that $\mathrm{SMP}_{\mathsf{d}\text{-coh}}(\mathcal{K}) \in \mathsf{P}$ holds for every $\mathcal{K}$ as in the theorem. We will prove this by showing that, under the assumptions of the theorem, Algorithm 8 solves $\mathrm{SMP}_{\mathsf{d}\text{-coh}}(\mathcal{K})$ in polynomial time.

First we discuss the correctness of Algorithm 8. Let $a_1, \dots, a_k, b \in \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ ($\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$) be a correct input for $\mathrm{SMP}_{\mathsf{d}\text{-coh}}(\mathcal{K})$ (i.e., a d-coherent input for $\mathrm{SMP}(\mathcal{K})$). Then conditions (1)–(5) in Definition 6.2 hold. By condition (3), the algebras $\mathbf{A}_j$ ($j \in [n]$)

Algorithm 8 For $\mathrm{SMP}_{\mathsf{d\text{-}coh}}(\mathcal{K})$ if $\mathcal{K}$ is in a residually small variety

Require: d-coherent $a_1, \dots, a_k, b \in \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ ($\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$) where $a_i = (a_{i1}, \dots, a_{in})$ for all $i \in [k]$

Ensure: Is b in the subalgebra of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ generated by $a_1, \dots, a_k$?

1: let μ_j be the monolith of $\mathbf{A}_j$, $\rho_j := (0 : \mu_j)$ for every $j \in [n]$ and $\rho := \rho_1 \times \cdots \times \rho_n$
2: reindex $a_1, \dots, a_k$ so that $a_{11}/\rho_1, \dots, a_{r1}/\rho_1$ are pairwise distinct and $\{a_{11}/\rho_1, \dots, a_{r1}/\rho_1\} = \{a_{11}/\rho_1, \dots, a_{k1}/\rho_1\}$
let $\mathcal{O} := \{a_1, \dots, a_r\}$
3: generate $\mathbf{A}_1/\rho_1$ by $a_{11}/\rho_1, \dots, a_{r1}/\rho_1$, and simultaneously,
4: **for** each new $a/\rho_1 = t(a_{11}/\rho_1, \dots, a_{r1}/\rho_1) \in \mathbf{A}_1/\rho_1$ (t a term) **do**
5: $\quad \mathcal{O} := \mathcal{O} \cup \{t(a_1, \dots, a_r)\}$
6: **end for**
7: find the equivalence relation $\equiv$ on $[n]$ defined by

$$s \equiv t \quad \Leftrightarrow \quad \mathbf{A}_s = \mathbf{A}_t \text{ and } o|_s = o|_t \text{ for all } o \in \mathcal{O} \qquad (s, t \in [n])$$

let T be a transversal for the blocks of $\equiv$, and let $\mathbf{A}_T := \prod_{j \in T} \mathbf{A}_j$
8: enumerate the elements of the subalgebra P of $\mathbf{A}_T^{A_T}$ generated by the identity function $A_T \to A_T$ and by the constant functions with value $o|_T$ ($o \in \mathcal{O}$) (so every $p \in P$ is of the form $A_T \to A_T, (x_t)_{t \in T} \mapsto (p_t(x_t))_{t \in T}$ for some function $p_t \colon A_t \to A_t$)
9: find $o \in \mathcal{O}$ such that $b \in o/\rho$
10: $H := \emptyset$
11: **for** $p \in P$ **do**
12: $\quad$ **for** $c \in \{a_1, \dots, a_k\}$ **do**
13: $\quad\quad$ **for** $j \in [n]$ **do**
14: $\quad\quad\quad$ find $t \in T$ with $t \equiv j$
15: $\quad\quad\quad$ let $d_j := p_t(c|_j)$
16: $\quad\quad$ **end for**
17: $\quad\quad$ $d := (d_1, \dots, d_n)$
18: $\quad\quad$ **if** $d \in o/\rho$ **then**
19: $\quad\quad\quad$ $H := H \cup \{d\}$
20: $\quad\quad$ **end if**
21: $\quad$ **end for**
22: **end for**
23: run Sims' algorithm for $\mathrm{SMP}(\mathcal{G})$ with the input $H \cup \{b\} \subseteq \mathbf{G}_1 \times \cdots \times \mathbf{G}_n$ to check whether $b \in \langle H \rangle$ where $\mathbf{G}_j$ is the group $(o_j/\rho_j; +_{o_j}, -_{o_j}, o_j)$ for each $j \in [n]$ and $\mathcal{G}$ is the family of all induced abelian groups on blocks of abelian congruences of algebras in $\mathcal{K}$; get answer $\mathsf{A} \in \{\mathrm{YES}, \mathrm{NO}\}$
24: **return** A

are subdirectly irreducible with abelian monoliths, so in Step 1 of Algorithm 8 the monoliths μ_j and their centralizers ρ_j will be found. Moreover, since $\mathcal{K}$ is assumed to be in a residually small variety, we get from Theorem 2.1(2) and Corollary 2.4 that

$$\rho_j \text{ is an abelian congruence of } \mathbf{A}_j \text{ for every } j \in [n]. \tag{6.1}$$

Let $\mathbf{B}$ denote the subalgebra of $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ generated by the input tuples $a_1, \dots, a_k$, and let ρ denote the product congruence $\rho_1 \times \cdots \times \rho_n$ on $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$. The restriction of

ρ to $\mathbf{B}$ will be denoted by $\rho_{\mathbf{B}}$. Condition (5) in Definition 6.2 implies that

$$\text{the map } \mathbf{B}/\rho_{\mathbf{B}} \to \mathbf{A}_j/\rho_j, (x_1, \dots, x_n)/\rho_{\mathbf{B}} \mapsto x_j/\rho_j \text{ is a bijection for every } j \in [n]. \quad (6.2)$$

Finally, conditions (1) and (4) together imply that for the input tuple b we have $b|_I \in \mathbf{B}|_I$ for all sets $I \in \binom{[n]}{2}$. Hence, we have $b|_{i,j}/(\rho_i \times \rho_j) \in \mathbf{B}|_{i,j}/(\rho_i \times \rho_j)$ for all $i, j \in [n]$. Since, by condition (5), $\mathbf{B}|_{i,j}/(\rho_i \times \rho_j)$ is the graph of an isomorphism $\mathbf{A}_i/\rho_i \to \mathbf{A}_j/\rho_j$ for every pair $i, j \in [n]$ $(i \neq j)$, it follows that the tuple $b/\rho = (b|_1/\rho_1, \dots, b|_n/\rho_n)$ belongs to $\mathbf{B}[\rho]/\rho$. Hence,

$$b \text{ is an element of the algebra } \mathbf{B}[\rho], \text{ the } \rho\text{-saturation of } \mathbf{B}. \quad (6.3)$$

We return to the analysis of Algorithm 8. After appropriately reindexing $a_1, \dots, a_k$, Steps 2–6 produce a subset $\mathcal{O}$ of $\mathbf{B}$ such that the first coordinates of the tuples in $\mathcal{O}$ form a transversal for the ρ_1-classes of $\mathbf{A}_1$. Thus, it follows from (6.2) that the tuples in $\mathcal{O}$ form a transversal for the $\rho_{\mathbf{B}}$-classes of $\mathbf{B}$, and hence also for the ρ-classes of $\mathbf{B}[\rho]$. Let $|\mathcal{O}| = \ell$. Since $|\mathcal{O}| = \left|\mathcal{O}|_1\right|$, we have $\ell = |\mathbf{A}_1/\rho_1| < |A_1|$.

Now let $\equiv$, T, and $\mathbf{A}_T$ be as defined (and computed) in Step 7, and let $\mathbf{A} := \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$. During our discussion of Steps 8–24 we will use computations where the same term is used in different algebras. To make it easier for the reader to keep track of where each computation takes place, we will use superscripts to indicate the relevant algebras.

It is easy to see that the set P of functions $A_T \to A_T$ computed in Step 8 is

$$P = \{\mathsf{t}^{\mathbf{A}_T}(x, \mathcal{O}|_T) : \mathsf{t} \text{ is a } (1+\ell)\text{-ary term}\} \quad (6.4)$$

where we assume that an ordering of $\mathcal{O}$ has been fixed to ensure that its elements are always substituted into terms in that fixed order. Our observation (6.3) shows that in Step 9 Algorithm 8 will find an element $o \in \mathcal{O}$ such that $b \in o/\rho$.

Claim 6.6. The set H obtained by Algorithm 8 after completing Steps 10–22 is

$$H = \{\mathsf{t}^{\mathbf{A}}(a_i, \mathcal{O}) \in o/\rho : i \in [k],\ \mathsf{t} \text{ is a } (1+\ell)\text{-ary term}\}. \quad (6.5)$$

Proof of Claim 6.6. We will use the notation from Steps 10–22. Let $p \in P$ and $c \in \{a_1, \dots, a_k\}$, say $c = (c_1, \dots, c_n)$. By (6.4), $p \in P$ if and only if p is a unary polynomial operation of $\mathbf{A}_T$ of the form $\mathsf{t}^{\mathbf{A}_T}(x, \mathcal{O}|_T)$ for some $(1+\ell)$-ary term t. Our goal is to show that for every choice of p and c the tuple d computed in Steps 13–17 is

$$d = \mathsf{t}^{\mathbf{A}}(c, \mathcal{O}). \quad (6.6)$$

This will prove that the tuples d computed in Steps 13–17 are exactly the elements of $\mathbf{A}$ of the form $\mathsf{t}^{\mathbf{A}}(c, \mathcal{O})$ where $c \in \{a_1, \dots, a_k\}$ and t is a $(1+\ell)$-ary term. Since such a tuple d is added to H in Steps 18–20 if and only if d also satisfies $d \in o/\rho$, the equality (6.5) will follow.

To verify (6.6) let $j \in [n]$ and let $t \in T$ be the unique transversal element such that $t \equiv j$. By the definition of $\equiv$ we have that $o'|_t = o'|_j$ for all $o' \in \mathcal{O}$. The latter condition may be written as $\mathcal{O}|_t = \mathcal{O}|_j$ (with the fixed ordering of $\mathcal{O}$ in mind, these are tuples of elements in $\mathbf{A}_t = \mathbf{A}_j$). The function $p_t \colon \mathbf{A}_t \to \mathbf{A}_t$ computed in Step 8 is the polynomial function $\mathsf{t}^{\mathbf{A}_t}(x, \mathcal{O}|_t)$ of $\mathbf{A}_t$. Thus, using the equalities $\mathbf{A}_t = \mathbf{A}_j$ and $\mathcal{O}|_t = \mathcal{O}|_j$ we get that

$$d_j = p_t(c_j) = \mathsf{t}^{\mathbf{A}_t}(c_j, \mathcal{O}|_t) = \mathsf{t}^{\mathbf{A}_j}(c_j, \mathcal{O}|_j) = \mathsf{t}^{\mathbf{A}}(c, \mathcal{O})|_j.$$

This holds for every $j \in [n]$, so the proof of (6.6), and hence the proof of Claim 6.6, is complete. ◇

To establish the correctness of the last two steps of Algorithm 8 recall from (6.1) that ρ_j is an abelian congruence of $\mathbf{A}_j$ for every $j \in [n]$. Therefore, $\rho = \rho_1 \times \cdots \times \rho_n$ is an abelian congruence of $\mathbf{A} = \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$. Hence, by Theorem 2.2 (1), there is an induced abelian group $\mathbf{G} := (o/\rho; +_0, -_o, o)$ on the ρ-class o/ρ. Moreover, since ρ is the product congruence $\rho_1 \times \cdots \times \rho_n$ of $\mathbf{A}$, we get that $\mathbf{G} = \mathbf{G}_1 \times \cdots \times \mathbf{G}_n$ where $\mathbf{G}_j$ is the group $(o_j/\rho_j; +_{o_j}, -_{o_j}, o_j)$ for every $j \in [n]$. Recall also that $b \in o/\rho$, that is, b is contained in $\mathbf{G} = \mathbf{G}_1 \times \cdots \times \mathbf{G}_n$.

Claim 6.7. The following conditions on b are equivalent:
(1) b is in the subalgebra $\mathbf{B}$ of $\mathbf{A} = \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ generated by $\{a_1, \dots, a_k\}$;
(2) b is in the subgroup of $\mathbf{G} = \mathbf{G}_1 \times \cdots \times \mathbf{G}_n$ generated by the set H.

Proof of Claim 6.7. To prove the implication (1) $\Rightarrow$ (2) assume that $b \in \mathbf{B}$, that is, $b = g(a_1, \dots, a_k)$ for some k-ary term g. For each $i \in [k]$, let $o^{(i)}$ denote the unique element of $\mathcal{O}$ in the $\rho_{\mathbf{B}}$-class of a_i. Since $g(a_1, \dots, a_k) = b \in o/\rho$, it follows from Theorem 2.2 (2) that

$$g(a_1, \dots, a_k) = g(a_1, o^{(2)} \dots, o^{(k)}) +_o g(o^{(1)}, a_2, o^{(3)} \dots, o^{(k)}) +_o \dots \\ +_o g(o^{(1)}, \dots, o^{(k-1)}, a_k) -_o (k-1) g(o^{(1)}, \dots, o^{(k-1)}, o^{(k)}).$$

All $+_o$-summands on the right hand side belong to H, therefore b is in the subgroup of $\mathbf{G} = \mathbf{G}_1 \times \cdots \times \mathbf{G}_n$ generated by H.

For the reverse implication (2) $\Rightarrow$ (1) notice first that $H \subseteq B$, because $\mathbf{B}$ is a subalgebra of $\mathbf{A} = \mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ and the elements of H are obtained from $a_1, \dots, a_k \in B$ by unary polynomial operations of $\mathbf{A}$ that are constructed from term operations by using parameters from $\mathcal{O} \subseteq B$ only. Since the group operations $+_o$, $-_o$, o of $\mathbf{G}$ are also polynomial operations of $\mathbf{A}$ obtained from term operations using parameters from $\mathcal{O}$ only, we get that the subgroup of $\mathbf{G}$ generated by H is contained in $\mathbf{B}$. ◇

As in Step 23 of Algorithm 8, let $\mathcal{G}$ denote the set of all induced abelian groups on blocks of abelian congruences of algebras in $\mathcal{K}$. Then, clearly, $\mathbf{G}_1, \dots, \mathbf{G}_n \in \mathcal{G}$. Therefore, Claim 6.7 shows that $\mathrm{SMP}_{\mathsf{d\text{-}coh}}(\mathcal{K})$ run with the input $a_1, \dots, a_k, b$ in $\mathbf{A}_1 \times \cdots \times \mathbf{A}_n$ ($\mathbf{A}_1, \dots, \mathbf{A}_n \in \mathcal{K}$) has the same answer as $\mathrm{SMP}(\mathcal{G})$ run with the input H and b in $\mathbf{G}_1 \times \cdots \times \mathbf{G}_n$ ($\mathbf{G}_1, \dots, \mathbf{G}_n \in \mathcal{G}$). Hence, Algorithm 8 finds the correct answer in Steps 23, 24. The proof of the correctness of Algorithm 8 is complete.

To prove that Algorithm 8 runs in polynomial time, we will estimate the time complexity of each step separately. Recall that $\mathsf{a}_{\mathcal{K}}$ denotes the maximum size of an algebra in $\mathcal{K}$.

Steps 1 and 3–6 run in $O(n)$ time, while Step 2 runs in $O(kn)$ time. In Step 7, $\equiv$ and a transversal T for the blocks of $\equiv$ can be found in $O(n^2)$ time, since $\ell = |\mathcal{O}|$ is bounded above by a constant ($< |A_1| \leq \mathsf{a}_{\mathcal{K}}$) which is independent of the size of the input. Since $\equiv$ is the kernel of the map $[n] \to \mathcal{K} \times \bigcup_{i \in [n]} A_i^\ell$, $j \mapsto (\mathbf{A}_j, \mathcal{O}|_j)$, the number $|T|$ of the $\equiv$-blocks is at most $|\mathcal{K}| \mathsf{a}_{\mathcal{K}}^\ell$, independent of the input. Therefore, $\mathbf{A}_T$ can be computed in constant time, and Step 7 altogether requires $O(n^2)$ time. For the same reason, $|\mathbf{A}_T^{A_T}|$ is also bounded above by a constant, independent of the input, therefore Step 8 runs in constant time. Step 9 also runs in constant time, because, in view of (6.2), $b \in o/\rho$ is equivalent to $b|_1 \in o|_1/\rho_1$. Clearly, Step 10 also runs in constant time.

Using the previous estimates on $|P|$ ($\leq |\mathbf{A}_T^{A_T}|$) and $|T|$ we see that the number of iterations of the outer **for** loop (line 11) is bounded above by a constant, while the inner **for** loops starting in Step 12 and Step 13 iterate k and n times, respectively. In each iteration, Steps 14–15 and 18–20 require constant time. Thus Steps 11–22 run in $O(kn)$ time.

In Steps 11–22 at most one element is added to H for each choice of $p \in P$ and $c \in \{a_1, \dots, a_k\}$. Since $|P|$ is bounded above by a constant independent of the input, we get that $|H|$ has size $O(k)$. Thus, in Step 23 the size of the input $H \cup \{b\}$ for $\mathrm{SMP}(\mathcal{G})$ is $O(kn)$. Moreover, the size of each group in $\mathcal{G}$ is $\leq \mathtt{a}_{\mathcal{K}}$. Since Sims' algorithm for $\mathrm{SMP}(\mathcal{G})$ runs in $O(kn^3)$ time [28, Corollary 3.7] on an input $H \cup \{b\} \subseteq \mathbf{G}_1 \times \cdots \times \mathbf{G}_n$ with $|H| = O(k)$, we get that Step 23 of Algorithm 8 requires $O(kn^3)$ time. Clearly, Step 24 runs in constant time.

Combining the time complexities of Steps 1–24 we get that Algorithm 8 runs in $O(kn^3)$ time. This completes the proof of Theorem 6.5. □

Acknowledgement. The authors are thankful to the anonymous referees for their careful reading and suggestions.

References

[1] K. A. Baker and A. F. Pixley. Polynomial interpolation and the Chinese remainder theorem for algebraic systems. *Math. Z.*, 143(2):165–174, 1975.

[2] G. Baumslag, C. F. Miller, III, and H. Short. Unsolvable problems about small cancellation and word hyperbolic groups. *Bull. London Math. Soc.*, 26(1):97–101, 1994.

[3] J. Berman, P. Idziak, P. Marković, R. McKenzie, M. Valeriote, and R. Willard. Varieties with few subalgebras of powers. *Trans. Amer. Math. Soc.*, 362(3):1445–1473, 2010.

[4] A. Bulatov, H. Chen, and V. Dalmau. Learning intersection-closed classes with signatures. *Theoret. Comput. Sci.*, 382(3):209–220, 2007.

[5] A. Bulatov and V. Dalmau. A simple algorithm for Mal′tsev constraints. *SIAM J. Comput.*, 36(1):16–27 (electronic), 2006.

[6] A. Bulatov, M. Kozik, P. Mayr, and M. Steindl. The subpower membership problem for semigroups. *Internat. J. Algebra Comput.*, 26(7):1435–1451, 2016.

[7] S. Burris and H. P. Sankappanavar. *A course in universal algebra*. Springer New York Heidelberg Berlin, 1981.
Available from `https://www.math.uwaterloo.ca/~snburris/htdocs/UALG/univ-algebra2012.pdf`.

[8] V. Dalmau and P. Jeavons. Learnability of quantified formulas. *Theoret. Comput. Sci.*, 306(1-3):485–511, 2003.

[9] T. Dent, K. Kearnes, and Á. Szendrei. An easy test for congruence modularity. *Algebra Universalis*, 67(4):375–392, 2012.

[10] M. Dyer and D. Richerby. An effective dichotomy for the counting constraint satisfaction problem. *SIAM J. Comput.*, 42(3):1245–1274, 2013.

[11] R. Freese and R. McKenzie. *Commutator Theory for Congruence Modular Varieties*, volume 125 of *London Math. Soc. Lecture Note Ser.* Cambridge University Press, 1987. Available from `http://math.hawaii.edu/~ralph/Commutator/comm.pdf`.

[12] R. Freese and R. McKenzie. Residually small varieties with modular congruence lattices. *Trans. Amer. Math. Soc.*, 264(2):419–430, 1981.

[13] M. Furst, J. Hopcroft, and E. Luks. Polynomial-time algorithms for permutation groups. In *21st Annual Symposium on Foundations of Computer Science (Syracuse, N.Y., 1980)*, pages 36–41. IEEE, New York, 1980.

[14] R. I. Grigorchuk and J. S. Wilson. A structural property concerning abstract commensurability of subgroups. *J. London Math. Soc. (2)*, 68(3):671–682, 2003.

[15] P. Idziak, P. Marković, R. McKenzie, M. Valeriote, and R. Willard. Tractability and learnability arising from algebras with few subpowers. *SIAM J. Comput.*, 39(7):3023–3037, 2010.

[16] K. A. Kearnes and Á. Szendrei. Clones of algebras with parallelogram terms. *Internat. J. Algebra Comput.*, 22(1), 2012.

[17] D. Knuth. Efficient representation of perm groups. *Combinatorica*, 11(1):33–43, 1991.

[18] D. Kozen. Complexity of finitely presented algebras. In *Ninth Annual ACM Symposium on Theory of Computing (Boulder, Colo., 1977)*, pages 164–177, 1977.
[19] D. Kozen. Lower bounds for natural proof systems. In *18th Annual Symposium on Foundations of Computer Science (Providence, R.I., 1977)*, pages 254–266. IEEE Comput. Sci., Long Beach, Calif., 1977.
[20] M. Kozik. A finite set of functions with an EXPTIME-complete composition problem. *Theoret. Comput. Sci.*, 407(1-3):330–341, 2008.
[21] J. S. Leon. On an algorithm for finding a base and a strong generating set for a group given by generating permutations. *Math. Comp.*, 35(151):941–974, 1980.
[22] P. Mayr. The subpower membership problem for Mal'cev algebras. *International Journal of Algebra and Computation (IJAC)*, 22(7), 2012.
[23] V. Shpilrain and A. Ushakov. *Thompson's group and public key cryptography*, volume 3531 of *LNCS*. 2005.
[24] V. Shpilrain and G. Zapata. Using the subgroup membership search problem in public key cryptography. In *Algebraic methods in cryptography*, volume 418 of *Contemp. Math.*, pages 169–178. Amer. Math. Soc., Providence, RI, 2006.
[25] C. Sims. Computational methods in the study of permutation groups. In J. Leech, editor, *Computational problems in abstract algebra, Conf. Oxford*, pages 169–183, 1970.
[26] M. Steindl. The subpower membership problem for bands. *J. Algebra*, 489:529–551, 2017.
[27] R. Willard. Four unsolved problems in congruence permutable varieties. Talk at the "Conference on Order, Algebra, and Logics", Nashville, 2007.
[28] S. Zweckinger. Computing in direct powers of expanded groups, Master's Thesis, JKU Linz, Austria, 2015.