Growth Functions of Finite Algebras

K. A. Kearnes

University of Colorado

with E. W. Kiss and Á. Szendrei

Growth functions

Growth functions

Question: For an algebra $\mathbf{A}$, what is the least size of a generating set for $\mathbf{A}^n$?

Growth functions

Question: For an algebra $\mathbf{A}$, what is the least size of a generating set for $\mathbf{A}^n$?

Answer: $d_{\mathbf{A}}(n) := \min\{g : \mathbf{A}^n \text{ has a generating set of size } g\}$.

Examples

Examples

1. If **A** is a 2-element set, then $d_{\mathbf{A}}(n) = 2^n$.

Examples

1. If $\mathbf{A}$ is a 2-element set, then $d_{\mathbf{A}}(n) = 2^n$.
2. If $\mathbf{B}$ is a 2-element Boolean algebra, then $d_{\mathbf{B}}(n) = \lceil \log_2(n) \rceil$.

Examples

1. If $\mathbf{A}$ is a 2-element set, then $d_{\mathbf{A}}(n) = 2^n$.
2. If $\mathbf{B}$ is a 2-element Boolean algebra, then $d_{\mathbf{B}}(n) = \lceil \log_2(n) \rceil$.
3. Finite implication algebras: $\mathbf{C} \in \mathcal{V}_{\text{fin}}(\langle \{0, 1\}; \to \rangle)$

Examples

1. If $\mathbf{A}$ is a 2-element set, then $d_{\mathbf{A}}(n) = 2^n$.
2. If $\mathbf{B}$ is a 2-element Boolean algebra, then $d_{\mathbf{B}}(n) = \lceil \log_2(n) \rceil$.
3. Finite implication algebras: $\mathbf{C} \in \mathcal{V}_{\text{fin}}(\langle \{0, 1\}; \to \rangle)$
    1. If $\mathbf{C}$ has a unique minimal element, then $d_{\mathbf{C}}(n) \in \Theta(\log(n))$.

Examples

1. If $\mathbf{A}$ is a 2-element set, then $d_{\mathbf{A}}(n) = 2^n$.
2. If $\mathbf{B}$ is a 2-element Boolean algebra, then $d_{\mathbf{B}}(n) = \lceil \log_2(n) \rceil$.
3. Finite implication algebras: $\mathbf{C} \in \mathcal{V}_{\text{fin}}(\langle \{0, 1\}; \rightarrow \rangle)$
    1. If $\mathbf{C}$ has a unique minimal element, then $d_{\mathbf{C}}(n) \in \Theta(\log(n))$.
    2. Otherwise $d_{\mathbf{C}}(n) \in 2^{\Theta(n)}$.

Examples

1. If **A** is a 2-element set, then $d_{\mathbf{A}}(n) = 2^n$.
2. If **B** is a 2-element Boolean algebra, then $d_{\mathbf{B}}(n) = \lceil \log_2(n) \rceil$.
3. Finite implication algebras: $\mathbf{C} \in \mathcal{V}_{\text{fin}}(\langle \{0, 1\}; \to \rangle)$
   1. If **C** has a unique minimal element, then $d_{\mathbf{C}}(n) \in \Theta(\log(n))$.
   2. Otherwise $d_{\mathbf{C}}(n) \in 2^{\Theta(n)}$.
4. If M is an R-module, then $d_M(n) \in \Theta(n)$.

Examples

1. If **A** is a 2-element set, then $d_{\mathbf{A}}(n) = 2^n$.
2. If **B** is a 2-element Boolean algebra, then $d_{\mathbf{B}}(n) = \lceil \log_2(n) \rceil$.
3. Finite implication algebras: $\mathbf{C} \in \mathcal{V}_{\text{fin}}(\langle \{0, 1\}; \rightarrow \rangle)$
   1. If **C** has a unique minimal element, then $d_{\mathbf{C}}(n) \in \Theta(\log(n))$.
   2. Otherwise $d_{\mathbf{C}}(n) \in 2^{\Theta(n)}$.
4. If M is an R-module, then $d_M(n) \in \Theta(n)$.
   1. (Linear upper bound) $M^n = \oplus^n M$,

Examples

1. If $\mathbf{A}$ is a 2-element set, then $d_{\mathbf{A}}(n) = 2^n$.
2. If $\mathbf{B}$ is a 2-element Boolean algebra, then $d_{\mathbf{B}}(n) = \lceil \log_2(n) \rceil$.
3. Finite implication algebras: $\mathbf{C} \in \mathcal{V}_{\text{fin}}(\langle \{0, 1\}; \rightarrow \rangle)$
    1. If $\mathbf{C}$ has a unique minimal element, then $d_{\mathbf{C}}(n) \in \Theta(\log(n))$.
    2. Otherwise $d_{\mathbf{C}}(n) \in 2^{\Theta(n)}$.
4. If M is an R-module, then $d_M(n) \in \Theta(n)$.
    1. (Linear upper bound) $M^n = \oplus^n M$,

Examples

1. If **A** is a 2-element set, then $d_{\mathbf{A}}(n) = 2^n$.
2. If **B** is a 2-element Boolean algebra, then $d_{\mathbf{B}}(n) = \lceil \log_2(n) \rceil$.
3. Finite implication algebras: $\mathbf{C} \in \mathcal{V}_{\text{fin}}(\langle \{0, 1\}; \to \rangle)$
   1. If **C** has a unique minimal element, then $d_{\mathbf{C}}(n) \in \Theta(\log(n))$.
   2. Otherwise $d_{\mathbf{C}}(n) \in 2^{\Theta(n)}$.
4. If M is an R-module, then $d_M(n) \in \Theta(n)$.
   1. (Linear upper bound) $M^n = \oplus^n M$, so $d_M(n) \leq d_M(1) \cdot n$.

Examples

1. If $\mathbf{A}$ is a 2-element set, then $d_{\mathbf{A}}(n) = 2^n$.
2. If $\mathbf{B}$ is a 2-element Boolean algebra, then $d_{\mathbf{B}}(n) = \lceil \log_2(n) \rceil$.
3. Finite implication algebras: $\mathbf{C} \in \mathcal{V}_{\text{fin}}(\langle \{0,1\}; \to \rangle)$
    1. If $\mathbf{C}$ has a unique minimal element, then $d_{\mathbf{C}}(n) \in \Theta(\log(n))$.
    2. Otherwise $d_{\mathbf{C}}(n) \in 2^{\Theta(n)}$.
4. If M is an R-module, then $d_M(n) \in \Theta(n)$.
    1. (Linear upper bound) $M^n = \oplus^n M$, so $d_M(n) \leq d_M(1) \cdot n$.
    2. (Linear lower bound) $|M^n| \leq |R^{d_M(n)}|$,

Examples

1. If $\mathbf{A}$ is a 2-element set, then $d_{\mathbf{A}}(n) = 2^n$.
2. If $\mathbf{B}$ is a 2-element Boolean algebra, then $d_{\mathbf{B}}(n) = \lceil \log_2(n) \rceil$.
3. Finite implication algebras: $\mathbf{C} \in \mathcal{V}_{\text{fin}}(\langle \{0, 1\}; \to \rangle)$
   1. If $\mathbf{C}$ has a unique minimal element, then $d_{\mathbf{C}}(n) \in \Theta(\log(n))$.
   2. Otherwise $d_{\mathbf{C}}(n) \in 2^{\Theta(n)}$.
4. If M is an R-module, then $d_M(n) \in \Theta(n)$.
   1. (Linear upper bound) $M^n = \oplus^n M$, so $d_M(n) \leq d_M(1) \cdot n$.
   2. (Linear lower bound) $|M^n| \leq |R^{d_M(n)}|$,

Examples

1. If $\mathbf{A}$ is a 2-element set, then $d_{\mathbf{A}}(n) = 2^n$.
2. If $\mathbf{B}$ is a 2-element Boolean algebra, then $d_{\mathbf{B}}(n) = \lceil \log_2(n) \rceil$.
3. Finite implication algebras: $\mathbf{C} \in \mathcal{V}_{\text{fin}}(\langle \{0, 1\}; \to \rangle)$
    1. If $\mathbf{C}$ has a unique minimal element, then $d_{\mathbf{C}}(n) \in \Theta(\log(n))$.
    2. Otherwise $d_{\mathbf{C}}(n) \in 2^{\Theta(n)}$.
4. If M is an R-module, then $d_M(n) \in \Theta(n)$.
    1. (Linear upper bound) $M^n = \oplus^n M$, so $d_M(n) \leq d_M(1) \cdot n$.
    2. (Linear lower bound) $|M^n| \leq |R^{d_M(n)}|$, so $\log_{|R|}(|M|) \cdot n \leq d_M(n)$.

CSP Background, I

CSP Background, I

The Constraint Satisfaction Problem is the problem of testing whether a relational structure satisfies a p.p. sentence ($\exists \bigwedge$ atomic).

CSP Background, I

The Constraint Satisfaction Problem is the problem of testing whether a relational structure satisfies a p.p. sentence ($\exists \bigwedge$ atomic).

The "algebraic approach" to CSP studies the polymorphism algebra, **A**, of the test structure.

CSP Background, I

The Constraint Satisfaction Problem is the problem of testing whether a relational structure satisfies a p.p. sentence ($\exists \bigwedge$ atomic).

The "algebraic approach" to CSP studies the polymorphism algebra, $\mathbf{A}$, of the test structure.

The "Tractability and Learnability" paper (IMMVW) proves that the CSP is tractable if there is a polynomial p such that every subalgebra of $\mathbf{A}^n$ is generated by at most $p(n)$ elements.

CSP Background, I

The Constraint Satisfaction Problem is the problem of testing whether a relational structure satisfies a p.p. sentence ($\exists \bigwedge$ atomic).

The "algebraic approach" to CSP studies the polymorphism algebra, $\mathbf{A}$, of the test structure.

The "Tractability and Learnability" paper (IMMVW) proves that the CSP is tractable if there is a polynomial p such that every subalgebra of $\mathbf{A}^n$ is generated by at most $p(n)$ elements. ($\mathbf{A}$ has polynomially generated subpowers.)

CSP Background, I

The Constraint Satisfaction Problem is the problem of testing whether a relational structure satisfies a p.p. sentence ($\exists \bigwedge$ atomic).

The "algebraic approach" to CSP studies the polymorphism algebra, **A**, of the test structure.

The "Tractability and Learnability" paper (IMMVW) proves that the CSP is tractable if there is a polynomial p such that every subalgebra of $\mathbf{A}^n$ is generated by at most $p(n)$ elements. (**A** has polynomially generated subpowers.)

The "Few Subpowers" paper (BIMMVW) proves that **A** has polynomially generated subpowers iff **A** has a cube term.

CSP Background, II

CSP Background, II

The Quantified Constraint Satisfaction Problem is the problem of testing whether a relational structure satisfies a sentence $(Q_1 \ldots Q_k \bigwedge \text{atomic})$.

CSP Background, II

The Quantified Constraint Satisfaction Problem is the problem of testing whether a relational structure satisfies a sentence $(Q_1 \ldots Q_k \bigwedge \text{atomic})$.

Hubie Chen showed that if there is a polynomial q such that $\mathbf{A}^n$ can be generated by at most $q(n)$ elements (and there is a pol. time algorithm for finding the generators), then there is a pol. time reduction of the QCSP to the ordinary CSP.

CSP Background, II

The Quantified Constraint Satisfaction Problem is the problem of testing whether a relational structure satisfies a sentence $(Q_1 \ldots Q_k \bigwedge \text{atomic})$.

Hubie Chen showed that if there is a polynomial q such that $\mathbf{A}^n$ can be generated by at most $q(n)$ elements (and there is a pol. time algorithm for finding the generators), then there is a pol. time reduction of the QCSP to the ordinary CSP. (**A** has polynomially generated powers.)

Classical Algebra Background

Classical Algebra Background

1. (P. Hall, 1936) If G is a finite simple nonabelian group, then $d_G(n) \leq g$ iff
$$n \leq \frac{1}{|\mathrm{Aut}(G)|} \sum_{H \leq G} \mu(H)|H|^g.$$

Classical Algebra Background

1. (P. Hall, 1936) If G is a finite simple nonabelian group, then $d_G(n) \leq g$ iff
$$n \leq \frac{1}{|\mathrm{Aut}(G)|} \sum_{H \leq G} \mu(H)|H|^g.$$
2. (J. Wiegold, 1974) If G is a finite simple nonabelian group, then
$$d_G(n) \approx \log_{|G|}(n) + \log_{|G|}(|\mathrm{Aut}(G)|).$$

Classical Algebra Background

1. (P. Hall, 1936) If G is a finite simple nonabelian group, then $d_G(n) \leq g$ iff

$$n \leq \frac{1}{|\mathrm{Aut}(G)|} \sum_{H \leq G} \mu(H)|H|^g.$$

2. (J. Wiegold, 1974) If G is a finite simple nonabelian group, then

$$d_G(n) \approx \log_{|G|}(n) + \log_{|G|}(|\mathrm{Aut}(G)|).$$

3. (J. Wiegold, 1987) The d-function of a finite semigroup S is

Classical Algebra Background

1. (P. Hall, 1936) If G is a finite simple nonabelian group, then $d_G(n) \leq g$ iff
$$n \leq \frac{1}{|\mathrm{Aut}(G)|} \sum_{H \leq G} \mu(H)|H|^g.$$
2. (J. Wiegold, 1974) If G is a finite simple nonabelian group, then
$$d_G(n) \approx \log_{|G|}(n) + \log_{|G|}(|\mathrm{Aut}(G)|).$$
3. (J. Wiegold, 1987) The d-function of a finite semigroup S is
    1. logarithmic (=$\Theta(\log(n))$) if S is a perfect group ($[S,S] = S$),

Classical Algebra Background

1. (P. Hall, 1936) If G is a finite simple nonabelian group, then $d_G(n) \leq g$ iff
$$n \leq \frac{1}{|\mathrm{Aut}(G)|} \sum_{H \leq G} \mu(H)|H|^g.$$
2. (J. Wiegold, 1974) If G is a finite simple nonabelian group, then
$$d_G(n) \approx \log_{|G|}(n) + \log_{|G|}(|\mathrm{Aut}(G)|).$$
3. (J. Wiegold, 1987) The d-function of a finite semigroup S is
    1. logarithmic ($=\Theta(\log(n))$) if S is a perfect group ($[S, S] = S$),
    2. linear ($=\Theta(n)$) if S has a unit but is not a perfect group, and

Classical Algebra Background

1. (P. Hall, 1936) If G is a finite simple nonabelian group, then $d_G(n) \leq g$ iff
$$n \leq \frac{1}{|\mathrm{Aut}(G)|} \sum_{H \leq G} \mu(H)|H|^g.$$
2. (J. Wiegold, 1974) If G is a finite simple nonabelian group, then
$$d_G(n) \approx \log_{|G|}(n) + \log_{|G|}(|\mathrm{Aut}(G)|).$$
3. (J. Wiegold, 1987) The d-function of a finite semigroup S is
    1. logarithmic ($=\Theta(\log(n))$) if S is a perfect group ($[S,S] = S$),
    2. linear ($=\Theta(n)$) if S has a unit but is not a perfect group, and
    3. exponential ($=2^{\Theta(n)}$) if S does not have a unit.

Classical Algebra Background

1. (P. Hall, 1936) If G is a finite simple nonabelian group, then $d_G(n) \leq g$ iff
$$n \leq \frac{1}{|\mathrm{Aut}(G)|} \sum_{H \leq G} \mu(H)|H|^g.$$
2. (J. Wiegold, 1974) If G is a finite simple nonabelian group, then
$$d_G(n) \approx \log_{|G|}(n) + \log_{|G|}(|\mathrm{Aut}(G)|).$$
3. (J. Wiegold, 1987) The d-function of a finite semigroup S is
    1. logarithmic (=$\Theta(\log(n))$) if S is a perfect group ($[S,S] = S$),
    2. linear (=$\Theta(n)$) if S has a unit but is not a perfect group, and
    3. exponential (=$2^{\Theta(n)}$) if S does not have a unit.
4. (H. Riedel, 1985) A finite perfect algebra in a congruence uniform variety has logarithmic growth (=d-function).

Classical Algebra Background

1. (P. Hall, 1936) If G is a finite simple nonabelian group, then $d_G(n) \leq g$ iff
$$n \leq \frac{1}{|\mathrm{Aut}(G)|} \sum_{H \leq G} \mu(H)|H|^g.$$
2. (J. Wiegold, 1974) If G is a finite simple nonabelian group, then
$$d_G(n) \approx \log_{|G|}(n) + \log_{|G|}(|\mathrm{Aut}(G)|).$$
3. (J. Wiegold, 1987) The d-function of a finite semigroup S is
    1. logarithmic (=$\Theta(\log(n))$) if S is a perfect group ($[S,S] = S$),
    2. linear (=$\Theta(n)$) if S has a unit but is not a perfect group, and
    3. exponential (=$2^{\Theta(n)}$) if S does not have a unit.
4. (H. Riedel, 1985) A finite perfect algebra in a congruence uniform variety has logarithmic growth (=d-function).
5. (M. Quick, N. Ruškuc, 2010) If $\mathbf{A}$ is a finite ring, module, k-algebra or Lie algebra, then $d_{\mathbf{A}}$ is logarithmic if $\mathbf{A}$ is perfect and linear otherwise.

Motivating Questions

Motivating Questions

Question

(Riedel) Is it true that a finite algebra in a congruence permutable variety has logarithmic growth if it is perfect and linear growth if it is imperfect?

Motivating Questions

Question

(Riedel) Is it true that a finite algebra in a congruence permutable variety has logarithmic growth if it is perfect and linear growth if it is imperfect?

Motivating Questions

Question

(Riedel) Is it true that a finite algebra in a congruence permutable variety has logarithmic growth if it is perfect and linear growth if it is imperfect?

Answer: Yes.

Motivating Questions

Question

(Riedel) Is it true that a finite algebra in a congruence permutable variety has logarithmic growth if it is perfect and linear growth if it is imperfect?

Answer: Yes.

Question

(Quick, Ruškuc) Must the d-function of a finite algebra be logarithmic, linear or exponential?

Motivating Questions

Question

(Riedel) Is it true that a finite algebra in a congruence permutable variety has logarithmic growth if it is perfect and linear growth if it is imperfect?

Answer: Yes.

Question

(Quick, Ruškuc) Must the d-function of a finite algebra be logarithmic, linear or exponential?

Motivating Questions

Question

(Riedel) Is it true that a finite algebra in a congruence permutable variety has logarithmic growth if it is perfect and linear growth if it is imperfect?

Answer: Yes.

Question

(Quick, Ruškuc) Must the d-function of a finite algebra be logarithmic, linear or exponential?

Answer: No.

Basic Facts

Basic Facts

1. Any monotone increasing function $f\colon \mathbb{Z}^+ \to \mathbb{Z}^+$ can be the d-function of a countably infinite algebra.

Basic Facts

1. Any monotone increasing function $f\colon \mathbb{Z}^+ \to \mathbb{Z}^+$ can be the d-function of a countably infinite algebra.
2. If $\mathbf{B}$ is an expansion of $\mathbf{A}$, then $d_{\mathbf{B}}(n) \leq d_{\mathbf{A}}(n)$;

Basic Facts

1. Any monotone increasing function $f\colon \mathbb{Z}^+ \to \mathbb{Z}^+$ can be the d-function of a countably infinite algebra.
2. If $\mathbf{B}$ is an expansion of $\mathbf{A}$, then $d_{\mathbf{B}}(n) \leq d_{\mathbf{A}}(n)$;

Basic Facts

1. Any monotone increasing function $f\colon \mathbb{Z}^+ \to \mathbb{Z}^+$ can be the d-function of a countably infinite algebra.
2. If $\mathbf{B}$ is an expansion of $\mathbf{A}$, then $d_{\mathbf{B}}(n) \leq d_{\mathbf{A}}(n)$; if $\mathbf{B}$ is the polynomial expansion of $\mathbf{A}$, then $d_{\mathbf{A}}(n) - d_{\mathbf{A}}(1) \leq d_{\mathbf{B}}(n) \leq d_{\mathbf{A}}(n)$.

Basic Facts

1. Any monotone increasing function $f\colon \mathbb{Z}^+ \to \mathbb{Z}^+$ can be the d-function of a countably infinite algebra.
2. If $\mathbf{B}$ is an expansion of $\mathbf{A}$, then $d_{\mathbf{B}}(n) \leq d_{\mathbf{A}}(n)$; if $\mathbf{B}$ is the polynomial expansion of $\mathbf{A}$, then $d_{\mathbf{A}}(n) - d_{\mathbf{A}}(1) \leq d_{\mathbf{B}}(n) \leq d_{\mathbf{A}}(n)$.
3. If $\mathbf{A}$ is finite, then $\lceil \log_{|A|}(n) \rceil \leq d_{\mathbf{A}}(n) \leq |A|^n$.

Classifying Algebras By Maltsev Conditions

Definition

Classifying Algebras By Maltsev Conditions

Definition

1. A set Σ of identities is *realizable* in $\mathbf{A}$ if there is a way to interpret the function symbols occurring in Σ as terms in such a way that Σ holds in $\mathbf{A}$.

Classifying Algebras By Maltsev Conditions

Definition

1. A set Σ of identities is *realizable* in $\mathbf{A}$ if there is a way to interpret the function symbols occurring in Σ as terms in such a way that Σ holds in $\mathbf{A}$.
2. A term is basic if it has at most one nonnullary function symbol.

Classifying Algebras By Maltsev Conditions

Definition

1. A set Σ of identities is *realizable* in $\mathbf{A}$ if there is a way to interpret the function symbols occurring in Σ as terms in such a way that Σ holds in $\mathbf{A}$.
2. A term is basic if it has at most one nonnullary function symbol.
3. An identity is basic if the terms on both sides are basic.

Classifying Algebras By Maltsev Conditions

Definition

1. A set Σ of identities is *realizable* in $\mathbf{A}$ if there is a way to interpret the function symbols occurring in Σ as terms in such a way that Σ holds in $\mathbf{A}$.
2. A term is basic if it has at most one nonnullary function symbol.
3. An identity is basic if the terms on both sides are basic.

Classifying Algebras By Maltsev Conditions

Definition

1. A set Σ of identities is *realizable* in $\mathbf{A}$ if there is a way to interpret the function symbols occurring in Σ as terms in such a way that Σ holds in $\mathbf{A}$.
2. A term is basic if it has at most one nonnullary function symbol.
3. An identity is basic if the terms on both sides are basic.

Example

A group realizes

Classifying Algebras By Maltsev Conditions

Definition

1. A set Σ of identities is *realizable* in $\mathbf{A}$ if there is a way to interpret the function symbols occurring in Σ as terms in such a way that Σ holds in $\mathbf{A}$.
2. A term is basic if it has at most one nonnullary function symbol.
3. An identity is basic if the terms on both sides are basic.

Example

A group realizes

1. $\Sigma_1 = \{p(x,y,y) \approx x, p(y,y,x) \approx x\}$.

Classifying Algebras By Maltsev Conditions

Definition

1. A set Σ of identities is *realizable* in $\mathbf{A}$ if there is a way to interpret the function symbols occurring in Σ as terms in such a way that Σ holds in $\mathbf{A}$.
2. A term is basic if it has at most one nonnullary function symbol.
3. An identity is basic if the terms on both sides are basic.

Example

A group realizes

1. $\Sigma_1 = \{p(x,y,y) \approx x, p(y,y,x) \approx x\}$.
2. $\Sigma_2 = \{1 \cdot x \approx x, x \cdot 1 \approx x\}$.

Pointed Cube Terms

Pointed Cube Terms

Example

Write $\mathbf{A} \models F(1,x,2) \approx x$ *and* $\mathbf{A} \models F(x,y,3) \approx x$ *as a matrix equation:*

$$\mathbf{A} \models F\begin{pmatrix} 1 & x & 2 \\ x & y & 3 \end{pmatrix} \approx \begin{pmatrix} x \\ x \end{pmatrix}$$

F *is a* 3*-pointed,* 3*-ary,* 2*-cube term for* $\mathbf{A}$.

Pointed Cube Terms

Example

Write $\mathbf{A} \models F(1, x, 2) \approx x$ *and* $\mathbf{A} \models F(x, y, 3) \approx x$ *as a matrix equation:*

$$\mathbf{A} \models F \begin{pmatrix} 1 & x & 2 \\ x & y & 3 \end{pmatrix} \approx \begin{pmatrix} x \\ x \end{pmatrix}$$

F is a 3*-pointed,* 3*-ary,* 2*-cube term for* $\mathbf{A}$.

Definition

F is a p-pointed, m-ary, k-cube term for $\mathbf{A}$ if $\mathbf{A} \models F(M) \approx \begin{pmatrix} x \\ x \end{pmatrix}$ where M is a $k \times m$-matrix consisting of p constants and some variables, and every column of M contains a symbol different from x.

Some Results

Theorem

Let Σ be a set of basic identities involving only finitely many constant symbols. If Σ does not entail the existence a pointed cube term, then realizability of Σ imposes no restriction on growth functions.

Some Results

Theorem

Let Σ be a set of basic identities involving only finitely many constant symbols. If Σ does not entail the existence a pointed cube term, then realizability of Σ imposes no restriction on growth functions.

Some Results

Theorem

Let Σ be a set of basic identities involving only finitely many constant symbols. If Σ does not entail the existence a pointed cube term, then realizability of Σ imposes no restriction on growth functions. (Given any finite $\mathbf{B}$, there is a finite $\mathbf{A}$ realizing Σ such that $d_{\mathbf{A}} = d_{\mathbf{B}}$.)

Some Results

Theorem

Let Σ be a set of basic identities involving only finitely many constant symbols. If Σ does not entail the existence a pointed cube term, then realizability of Σ imposes no restriction on growth functions. (Given any finite $\mathbf{B}$, there is a finite $\mathbf{A}$ realizing Σ such that $d_{\mathbf{A}} = d_{\mathbf{B}}$.)

Theorem

If Σ is a set of basic identities that entails the existence a pointed cube term, then any finite algebra realizing Σ has growth function bounded by a polynomial.

Σ Restrictive $\Rightarrow$ Pointed Cube Term

Σ Restrictive $\Rightarrow$ Pointed Cube Term

Assumption: Σ (involving finitely many constants) entails no pointed cube term.

Σ Restrictive $\Rightarrow$ Pointed Cube Term

Assumption: Σ (involving finitely many constants) entails no pointed cube term.

Goal: Given finite $\mathbf{B}$, $|B| > 1$, construct finite $\mathbf{A}$ realizing Σ such that $d_{\mathbf{A}} = d_{\mathbf{B}}$.)

Σ Restrictive $\Rightarrow$ Pointed Cube Term

Assumption: Σ (involving finitely many constants) entails no pointed cube term.

Goal: Given finite **B**, $|B| > 1$, construct finite **A** realizing Σ such that $d_{\mathbf{A}} = d_{\mathbf{B}}$.)

Stage 1: Show how to enlarge **B** without changing $d_{\mathbf{B}}$.

Σ Restrictive $\Rightarrow$ Pointed Cube Term

Assumption: Σ (involving finitely many constants) entails no pointed cube term.

Goal: Given finite $\mathbf{B}$, $|B| > 1$, construct finite $\mathbf{A}$ realizing Σ such that $d_{\mathbf{A}} = d_{\mathbf{B}}$.)

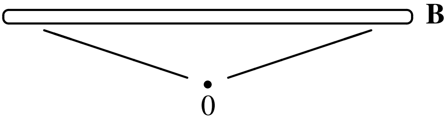
Stage 1: Show how to enlarge $\mathbf{B}$ without changing $d_{\mathbf{B}}$. This uses the 1-point extension, $\mathbf{B}_0 = \langle \mathbf{B} \cup \{0\}; \wedge \rangle$:

Σ Restrictive $\Rightarrow$ Pointed Cube Term

Assumption: Σ (involving finitely many constants) entails no pointed cube term.

Goal: Given finite $\mathbf{B}$, $|B| > 1$, construct finite $\mathbf{A}$ realizing Σ such that $d_{\mathbf{A}} = d_{\mathbf{B}}$.)

Stage 1: Show how to enlarge $\mathbf{B}$ without changing $d_{\mathbf{B}}$. This uses the 1-point extension, $\mathbf{B}_0 = \langle \mathbf{B} \cup \{0\}; \wedge \rangle$:

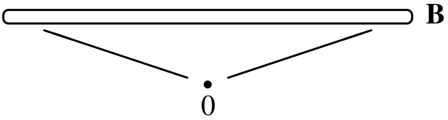


Σ Restrictive $\Rightarrow$ Pointed Cube Term

Assumption: Σ (involving finitely many constants) entails no pointed cube term.

Goal: Given finite $\mathbf{B}$, $|B| > 1$, construct finite $\mathbf{A}$ realizing Σ such that $d_{\mathbf{A}} = d_{\mathbf{B}}$.)

Stage 1: Show how to enlarge $\mathbf{B}$ without changing $d_{\mathbf{B}}$. This uses the 1-point extension, $\mathbf{B}_0 = \langle \mathbf{B} \cup \{0\}; \wedge \rangle$:



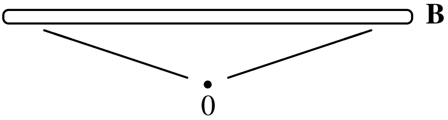
Stage 2: Show how to construct models of Σ of every cardinal larger than the number of constants in Σ.

Σ Restrictive $\Rightarrow$ Pointed Cube Term

Assumption: Σ (involving finitely many constants) entails no pointed cube term.

Goal: Given finite $\mathbf{B}$, $|B| > 1$, construct finite $\mathbf{A}$ realizing Σ such that $d_{\mathbf{A}} = d_{\mathbf{B}}$.)

Stage 1: Show how to enlarge $\mathbf{B}$ without changing $d_{\mathbf{B}}$. This uses the 1-point extension, $\mathbf{B}_0 = \langle \mathbf{B} \cup \{0\}; \wedge \rangle$:



Stage 2: Show how to construct models of Σ of every cardinal larger than the number of constants in Σ.

Stage 3: Show how to merge a sufficiently large model from Stage 1 with a sufficently large model from Stage 2 to form $\mathbf{A}$ with $d_{\mathbf{A}} = d_{\mathbf{B}}$.

Pointed Cube Term $\Rightarrow \Sigma$ Restrictive

Pointed Cube Term $\Rightarrow \Sigma$ Restrictive

Assumption: Σ entails a pointed cube term.

Pointed Cube Term $\Rightarrow$ Σ Restrictive

Assumption: Σ entails a pointed cube term.

Goal: Show that any finite model of Σ has polynomially generated powers.

Pointed Cube Term $\Rightarrow$ Σ Restrictive

Assumption: Σ entails a pointed cube term.

Goal: Show that any finite model of Σ has polynomially generated powers.

Example $(F(1, x, 2) \approx x, F(x, 2, 3) \approx x)$

Idea: Split coordinates of $[\mathbf{a}] \in \mathbf{A}^n$ *into* k *equal size segments,* $\begin{bmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \end{bmatrix}$. *Then*

$$F\left(\begin{bmatrix} \mathbf{1} \\ \mathbf{a}_2 \end{bmatrix}, \begin{bmatrix} \mathbf{a}_1 \\ \mathbf{2} \end{bmatrix}, \begin{bmatrix} \mathbf{2} \\ \mathbf{3} \end{bmatrix}\right) = \begin{bmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \end{bmatrix}$$

shows how to generate $[\mathbf{a}]$ *from more "highly processed" tuples.*

Pointed Cube Term $\Rightarrow \Sigma$ Restrictive

Assumption: Σ entails a pointed cube term.

Goal: Show that any finite model of Σ has polynomially generated powers.

Example ($F(1,x,2) \approx x, F(x,2,3) \approx x$)

Idea: Split coordinates of $[\mathbf{a}] \in \mathbf{A}^n$ *into k equal size segments,* $\begin{bmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \end{bmatrix}$. *Then*

$$F\left(\begin{bmatrix} \mathbf{1} \\ \mathbf{a}_2 \end{bmatrix}, \begin{bmatrix} \mathbf{a}_1 \\ \mathbf{2} \end{bmatrix}, \begin{bmatrix} \mathbf{2} \\ \mathbf{3} \end{bmatrix}\right) = \begin{bmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \end{bmatrix}$$

shows how to generate $[\mathbf{a}]$ *from more "highly processed" tuples.*

Pointed Cube Term $\Rightarrow \Sigma$ Restrictive

Assumption: Σ entails a pointed cube term.

Goal: Show that any finite model of Σ has polynomially generated powers.

Example ($F(1,x,2) \approx x, F(x,2,3) \approx x$)

Idea: Split coordinates of $[\mathbf{a}] \in \mathbf{A}^n$ *into k equal size segments,* $\begin{bmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \end{bmatrix}$. *Then*

$$F\left(\begin{bmatrix} \mathbf{1} \\ \mathbf{a}_2 \end{bmatrix}, \begin{bmatrix} \mathbf{a}_1 \\ \mathbf{2} \end{bmatrix}, \begin{bmatrix} \mathbf{2} \\ \mathbf{3} \end{bmatrix} \right) = \begin{bmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \end{bmatrix}$$

shows how to generate $[\mathbf{a}]$ *from more "highly processed" tuples.*
Must show: there are only polynomially many fully processed tuples.

Polynomial Growth of Any Degree

Example

Polynomial Growth of Any Degree

Example

Polynomial Growth of Any Degree

Example

Universe: $A = \{a_1, \ldots, a_q, 1\}$

Polynomial Growth of Any Degree

Example

Universe: $A = \{a_1, \ldots, a_q, 1\}$
k-ary Partial Operation:
$F(1, x, x, \ldots, x) = F(x, 1, x, \ldots, x) = \cdots = F(x, x, x, \ldots, 1) = x$

Polynomial Growth of Any Degree

Example

Universe: $A = \{a_1, \ldots, a_q, 1\}$
k-ary Partial Operation:
$F(1, x, x, \ldots, x) = F(x, 1, x, \ldots, x) = \cdots = F(x, x, x, \ldots, 1) = x$
$\mathbf{P} = \langle A; F, 1\rangle$

Polynomial Growth of Any Degree

Example

Universe: $A = \{a_1, \ldots, a_q, 1\}$
k-ary Partial Operation:
$F(1, x, x, \ldots, x) = F(x, 1, x, \ldots, x) = \cdots = F(x, x, x, \ldots, 1) = x$
$\mathbf{P} = \langle A; F, 1\rangle$
$\mathbf{A} = \mathbf{P}_0 = \langle \mathbf{P} \cup \{0\}; \wedge\rangle$.

Polynomial Growth of Any Degree

Example

Universe: $A = \{a_1, \ldots, a_q, 1\}$
k-ary Partial Operation:
$F(1, x, x, \ldots, x) = F(x, 1, x, \ldots, x) = \cdots = F(x, x, x, \ldots, 1) = x$
$\mathbf{P} = \langle A; F, 1\rangle$
$\mathbf{A} = \mathbf{P}_0 = \langle \mathbf{P} \cup \{0\}; \wedge\rangle.$
$d_{\mathbf{A}}(n) = \binom{n}{0} + q\binom{n}{1} + \cdots + q^{k-1}\binom{n}{k-1}$

Polynomial Growth of Any Degree

Example

Universe: $A = \{a_1, \ldots, a_q, 1\}$
k-ary Partial Operation:
$F(1, x, x, \ldots, x) = F(x, 1, x, \ldots, x) = \cdots = F(x, x, x, \ldots, 1) = x$
$\mathbf{P} = \langle A; F, 1\rangle$
$\mathbf{A} = \mathbf{P}_0 = \langle \mathbf{P} \cup \{0\}; \wedge\rangle$.
$d_{\mathbf{A}}(n) = \binom{n}{0} + q\binom{n}{1} + \cdots + q^{k-1}\binom{n}{k-1}$

At present, the only growth rates we know are logarithmic, polynomial or exponential.

Ordinary (0-Pointed) Cube Terms

Ordinary (0-Pointed) Cube Terms

Theorem

If $\mathbf{A}$ is a finite algebra with a cube term, then $d_{\mathbf{A}}(n)$ is logarithmic if $\mathbf{A}$ is perfect and linear if $\mathbf{A}$ is imperfect.

Ordinary (0-Pointed) Cube Terms

Theorem

If $\mathbf{A}$ *is a finite algebra with a cube term, then* $d_{\mathbf{A}}(n)$ *is logarithmic if* $\mathbf{A}$ *is perfect and linear if* $\mathbf{A}$ *is imperfect.*

Ordinary (0-Pointed) Cube Terms

Theorem

If $\mathbf{A}$ *is a finite algebra with a cube term, then* $d_{\mathbf{A}}(n)$ *is logarithmic if* $\mathbf{A}$ *is perfect and linear if* $\mathbf{A}$ *is imperfect.*

Proof is based on an analysis of maximal subuniverses.

Ordinary (0-Pointed) Cube Terms

Theorem

If $\mathbf{A}$ *is a finite algebra with a cube term, then* $d_{\mathbf{A}}(n)$ *is logarithmic if* $\mathbf{A}$ *is perfect and linear if* $\mathbf{A}$ *is imperfect.*

Proof is based on an analysis of maximal subuniverses.

A maximal subuniverse M of $\mathbf{B}$ is induced by surjective $\varphi\colon \mathbf{B} \to \mathbf{C}$ if $\varphi(M) \neq C$.

Ordinary (0-Pointed) Cube Terms

Theorem

If $\mathbf{A}$ is a finite algebra with a cube term, then $d_{\mathbf{A}}(n)$ is logarithmic if $\mathbf{A}$ is perfect and linear if $\mathbf{A}$ is imperfect.

Proof is based on an analysis of maximal subuniverses.

A maximal subuniverse M of $\mathbf{B}$ is induced by surjective $\varphi\colon \mathbf{B} \to \mathbf{C}$ if $\varphi(M) \neq C$. (Then $M = \varphi^{-1}(\varphi(M))$.)

Ordinary Cube Terms ⇒ Wiegold Dichotomy: Proof

Ordinary Cube Terms ⇒ Wiegold Dichotomy: Proof

Lemma

If $\mathbf{A}$ *has a k-cube term, then any maximal subuniverse* M *of* $\mathbf{A}^n$ *is induced by either*

Ordinary Cube Terms ⇒ Wiegold Dichotomy: Proof

Lemma

If $\mathbf{A}$ *has a k-cube term, then any maximal subuniverse* M *of* $\mathbf{A}^n$ *is induced by either*

1. *a projection* $\pi_U \colon \mathbf{A}^n \to \mathbf{A}^U$ *for some* $U \subseteq \{1, \ldots, n\}$ *satisfying* $|U| < \max(3, k)$, *or*

Ordinary Cube Terms $\Rightarrow$ Wiegold Dichotomy: Proof

Lemma

If $\mathbf{A}$ has a k-cube term, then any maximal subuniverse M of $\mathbf{A}^n$ is induced by either

1. *a projection $\pi_U \colon \mathbf{A}^n \to \mathbf{A}^U$ for some $U \subseteq \{1, \ldots, n\}$ satisfying $|U| < \max(3, k)$, or*
2. *the natural map $\nu \colon \mathbf{A}^n \to (\mathbf{A}/[1,1])^n$.*

Ordinary Cube Terms ⇒ Wiegold Dichotomy: Proof

Lemma

If $\mathbf{A}$ *has a k-cube term, then any maximal subuniverse* M *of* $\mathbf{A}^n$ *is induced by either*

1. *a projection* $\pi_U \colon \mathbf{A}^n \to \mathbf{A}^U$ *for some* $U \subseteq \{1, \ldots, n\}$ *satisfying* $|U| < \max(3, k)$, *or*
2. *the natural map* $\nu \colon \mathbf{A}^n \to (\mathbf{A}/[1, 1])^n$.

Lemma

Ordinary Cube Terms $\Rightarrow$ Wiegold Dichotomy: Proof

Lemma

If $\mathbf{A}$ has a k-cube term, then any maximal subuniverse M of $\mathbf{A}^n$ is induced by either

1. *a projection $\pi_U \colon \mathbf{A}^n \to \mathbf{A}^U$ for some $U \subseteq \{1, \ldots, n\}$ satisfying $|U| < \max(3, k)$, or*
2. *the natural map $\nu \colon \mathbf{A}^n \to (\mathbf{A}/[1,1])^n$.*

Lemma

1. *$\mathbf{A}^n$ has a subset of size $\Theta(\log(n))$ that is not contained in any maximal subuniverse induced by a projection $\pi_U \colon \mathbf{A}^n \to \mathbf{A}^U$ for $U \subseteq \{1, \ldots, n\}$ satisfying $|U| < \max(3, k)$.*

Ordinary Cube Terms $\Rightarrow$ Wiegold Dichotomy: Proof

Lemma

If $\mathbf{A}$ has a k-cube term, then any maximal subuniverse M of $\mathbf{A}^n$ is induced by either

1. *a projection $\pi_U\colon \mathbf{A}^n \to \mathbf{A}^U$ for some $U \subseteq \{1, \ldots, n\}$ satisfying $|U| < \max(3, k)$, or*
2. *the natural map $\nu\colon \mathbf{A}^n \to (\mathbf{A}/[1,1])^n$.*

Lemma

1. *$\mathbf{A}^n$ has a subset of size $\Theta(\log(n))$ that is not contained in any maximal subuniverse induced by a projection $\pi_U\colon \mathbf{A}^n \to \mathbf{A}^U$ for $U \subseteq \{1, \ldots, n\}$ satisfying $|U| < \max(3, k)$.*
2. *$\mathbf{A}^n$ has a subset of size $\Theta(n)$ that is not contained in any maximal subuniverse induced by the natural map $\nu\colon \mathbf{A}^n \to (\mathbf{A}/[1,1])^n$.*

Things We Don't Know

Things We Don't Know

1. Is there a finite algebra with sublinear growth rate that is not logarithmic?

Things We Don't Know

1. Is there a finite algebra with sublinear growth rate that is not logarithmic?

Things We Don't Know

1. Is there a finite algebra with sublinear growth rate that is not logarithmic?
2. If finite **A** has a polynomial $B(x, y)$ with a 2-sided unit ($B(1, x) \approx x \approx B(x, 1)$), must the growth rate of **A** be linear or logarithmic?

Things We Don't Know

1. Is there a finite algebra with sublinear growth rate that is not logarithmic?
2. If finite **A** has a polynomial $B(x, y)$ with a 2-sided unit ($B(1, x) \approx x \approx B(x, 1)$), must the growth rate of **A** be linear or logarithmic?

Things We Don't Know

1. Is there a finite algebra with sublinear growth rate that is not logarithmic?
2. If finite **A** has a polynomial $B(x, y)$ with a 2-sided unit ($B(1, x) \approx x \approx B(x, 1)$), must the growth rate of **A** be linear or logarithmic?
3. Is there a nice description of finite algebras with polynomially generated powers?