Berkeley, February 8, 2019

Computation with Atoms

Szymon Toruńczyk, University of Warsaw

includes work with Bojańczyk, Klin, Kopczyński, Lasota, Ochremiak,...

Atoms

a fixed *underlying* logical structure

Atoms

a fixed *underlying* logical structure

Examples:

$(\mathrm{N}, =)$ – *pure set*

$(\mathrm{Q}, \leq)$ – *dense order*

$(\mathrm{R}, +, \times, 0, 1)$ – *field of reals*

$(\mathrm{N}, +, \leq)$ – *Presburger arithmetic*

Atoms

a fixed *underlying* logical structure

Examples:

$(\mathbb{N}, =)$ – *pure set* ← main example in this talk

$(\mathbb{Q}, \leq)$ – *dense order*

$(\mathbb{R}, +, \times, 0, 1)$ – *field of reals*

$(\mathbb{N}, +, \leq)$ – *Presburger arithmetic*

Hereditarily definable set

Hereditarily definable set

Examples

5 if $5 \in \text{Atoms}$

$\{a : a \in \text{Atoms}\}$

$\{a : a \in \text{Atoms},\ a \neq 7 \wedge a \neq 5\}$ if $5,7 \in \text{Atoms}$

$\{\{a : a \in \text{Atoms},\ a \neq b\} : b \in \text{Atoms}\}$

$\{\{b : b \in \text{Atoms},\ a < b \wedge b < c\} : a,c \in \text{Atoms},\ a < c\}$ if $\text{Atoms} = (\mathbf{Q}, \leq)$

Hereditarily definable set

Examples

5 if $5 \in \text{Atoms}$

$\{a : a \in \text{Atoms}\}$

$\{a : a \in \text{Atoms},\ a \neq 7 \wedge a \neq 5\}$ if $5,7 \in \text{Atoms}$

$\{\{a : a \in \text{Atoms},\ a \neq b\} : b \in \text{Atoms}\}$

$\{\{b : b \in \text{Atoms},\ a < b \wedge b < c\} : a,c \in \text{Atoms},\ a < c\}$ if $\text{Atoms} = (\mathbf{Q}, \leq)$

Syntax

hdef ::= variable | parameter from Atoms

| { hdef : variable,...,variable ∈ **Atoms,** first order formula }

in language of Atoms,
with parameters

| hdef ∪ hdef

Hereditarily definable set

Examples

5 if $5 \in \text{Atoms}$

$\{a \colon a \in \text{Atoms}\}$

$\{a \colon a \in \text{Atoms},\ a \neq 7 \wedge a \neq 5\}$ if $5,7 \in \text{Atoms}$

$\{\{a \colon a \in \text{Atoms},\ a \neq b\} \colon b \in \text{Atoms}\}$

$\{\{b \colon b \in \text{Atoms},\ a < b \wedge b < c\} \colon a,c \in \text{Atoms},\ a < c\}$ if $\text{Atoms} = (\mathbf{Q}, \leq)$

$\{x,y\} \overset{\text{def}}{=} \{x\} \cup \{y\}$ $(x,y) \overset{\text{def}}{=} \{x, \{x,y\}\}$

Syntax

hdef ::= variable | parameter from Atoms

| { hdef **:** variable,...,variable ∈ **Atoms,** first order formula }
in language of Atoms,
with parameters

| hdef ∪ hdef

Hereditarily definable sets have finite descriptions

e.g. $\{\{a : a \in \text{Atoms}, a \neq b\} : b \in \text{Atoms}\}$

→ can be input and processed by algorithms

Hereditarily definable sets have finite descriptions

e.g. $\{\{a : a \in \text{Atoms}, a \neq b\} : b \in \text{Atoms}\}$

→ can be input and processed by algorithms

equality of sets is decidable ⇔ the theory of Atoms is decidable

Hereditarily definable *X*

graphs *automata* *Turing machines*

graphs

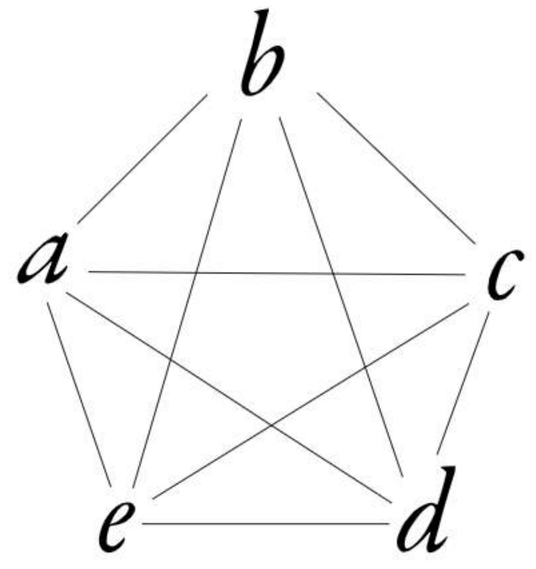
graphs

A pair (V,E) of hereditarily definable sets with E$\subseteq \binom{V}{2}$

infinite clique

vertices: $\{ a : a \in \text{Atoms} \}$

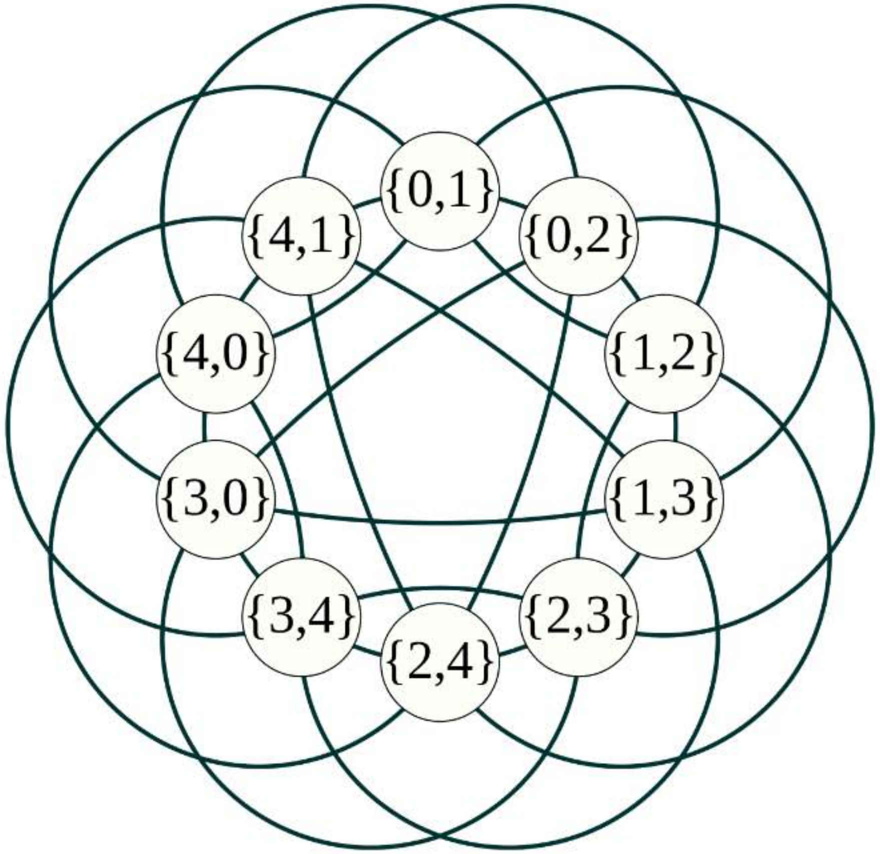
edges: $\{\{a,b\} : a,b \in \text{Atoms},\ a \neq b\}$



Johnson graph

vertices: $\{\{a,b\} : a,b \in \text{Atoms},\ a \neq b\ \}$

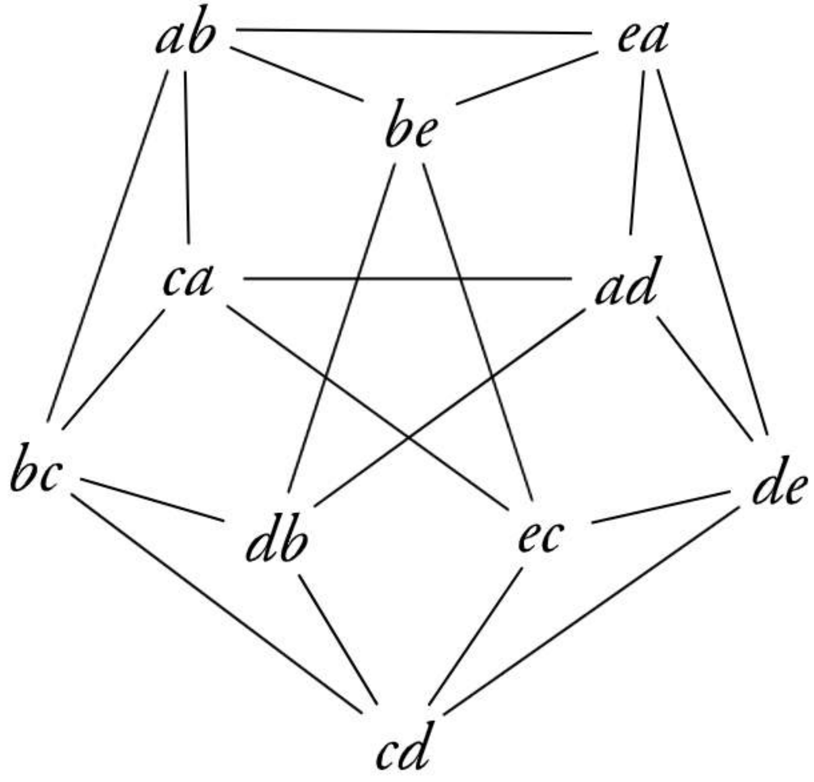
edges: $\{\{\{a,b\},\{b,c\}\} : a,b,c \in \text{Atoms},\ a \neq b \wedge b \neq c \wedge a \neq c\}$



some other graph

vertices: $\{(a,b) : a,b \in \text{Atoms},\ a \neq b\}$

edges: $\{\{(a,b),(b,c)\} : a,b,c \in \text{Atoms},\ a \neq b \wedge b \neq c \wedge a \neq c\}$

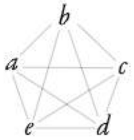


graphs

A pair (V,E) of hereditarily definable sets with $E \subseteq \binom{V}{2}$

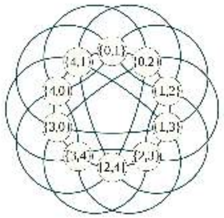
infinite clique

vertices: $\{\ a : a \in \text{Atoms}\ \}$
edges: $\{\{a,b\} : a,b \in \text{Atoms},\ a \neq b\}$



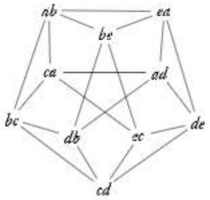
Johnson graph

vertices: $\{\{a,b\} : a,b \in \text{Atoms},\ a \neq b\ \}$
edges: $\{\{\{a,b\},\{b,c\}\} : a,b,c \in \text{Atoms},\ a \neq b \wedge b \neq c \wedge a \neq c\}$



some other graph

vertices: $\{(a,b) : a,b \in \text{Atoms},\ a \neq b\ \}$
edges: $\{\{(a,b),(b,c)\} : a,b,c \in \text{Atoms},\ a \neq b \wedge b \neq c \wedge a \neq c\}$

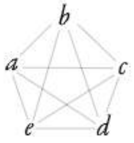


graphs

A pair (V,E) of hereditarily definable sets with $E \subseteq \binom{V}{2}$

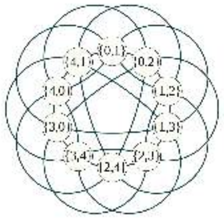
infinite clique

vertices: $\{\, a : a \in \text{Atoms} \,\}$
edges: $\{\{a,b\} : a,b \in \text{Atoms},\ a \neq b\}$



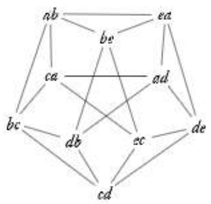
Johnson graph

vertices: $\{\{a,b\} : a,b \in \text{Atoms},\ a \neq b\,\}$
edges: $\{\{\{a,b\},\{b,c\}\} : a,b,c \in \text{Atoms},\ a \neq b \wedge b \neq c \wedge a \neq c\}$



some other graph

vertices: $\{(a,b) : a,b \in \text{Atoms},\ a \neq b\,\}$
edges: $\{\{(a,b),(b,c)\} : a,b,c \in \text{Atoms},\ a \neq b \wedge b \neq c \wedge a \neq c\}$



decision problems:

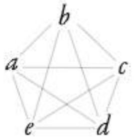
- connectedness
- 3-colorability
- homomorphism
- isomorphism
- ...

graphs

A pair (V,E) of hereditarily definable sets with $E \subseteq \binom{V}{2}$

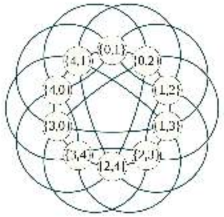
infinite clique

vertices: $\{ a : a \in \text{Atoms} \}$
edges: $\{\{a,b\} : a,b \in \text{Atoms}, a \neq b\}$



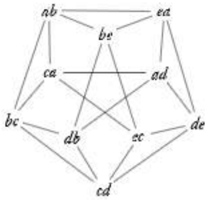
Johnson graph

vertices: $\{\{a,b\} : a,b \in \text{Atoms}, a \neq b \}$
edges: $\{\{\{a,b\},\{b,c\}\} : a,b,c \in \text{Atoms}, a \neq b \wedge b \neq c \wedge a \neq c\}$



some other graph

vertices: $\{(a,b) : a,b \in \text{Atoms}, a \neq b \}$
edges: $\{\{(a,b),(b,c)\} : a,b,c \in \text{Atoms}, a \neq b \wedge b \neq c \wedge a \neq c\}$



decision problems:

- connectedness
- 3-colorability
- homomorphism
- isomorphism
- …

discussed later

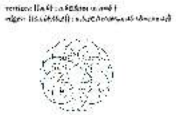
Hereditarily definable X

graphs

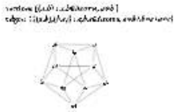
pair (V,E) of hereditarily definable sets with E$\subseteq \binom{V}{2}$

infinite clique

Johnson graph



some other graph



decision problems:

- connectedness
- 3-colorability
- homomorphism
- isomorphism
- ...

$\Bigg\}$ *discussed later*

automata

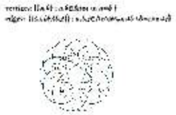
Turing machines

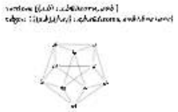
Hereditarily definable *X*

graphs

pair (V,E) of hereditarily definable sets with $E \subseteq \binom{V}{2}$


infinite clique


Johnson graph


some other graph

decision problems:

- connectedness
- 3-colorability
- homomorphism
- isomorphism
- ...

discussed later

automata

Turing machines

automata

automata

A tuple of hereditarily definable sets

$$(\text{States}, \text{Alphabet}, \text{Initial}, \text{Accepting}, \delta)$$

where $\text{Initial}, \text{Accepting} \subseteq \text{States}$ and $\delta \subseteq \text{States} \times \text{Alphabet} \times \text{States}$

Atoms are $(\mathbf{Q},\leq)$.

An automaton can accept sequences

$$q_1\ q_2\ q_3\ q_4\ \dots\ q_n$$

such that $q_1<q_2<q_3<q_4<\dots<q_n$

Atoms are $(\mathbf{N},=)$.

An automaton can accept sequences

$$a_1\ a_2\ a_3\ a_4\ \dots\ a_n$$

such that $a_i=a_j$ for some $i\neq j$

Atoms are $(\mathbf{N},=)$.

An automaton can accept sequences

$$a_1\ a_2\ a_3\ a_4\ \dots\ a_n$$

such that $a_i = a_j$ for some $i \neq j$

deterministic automata ≠ nondeterministic automata

Atoms are (**N**,=).

An automaton can accept sequences

$$\{a_1, a_2\} \quad \{a_3, a_4\} \quad \{a_5, a_6\} \quad \ldots\ldots\ldots\ldots\ldots \quad \{a_{2n\text{-}1}, a_{2n}\}$$

automata

A tuple of hereditarily definable sets

$$(\text{States}, \text{Alphabet}, \text{Initial}, \text{Accepting}, \delta)$$

where Initial, Accepting $\subseteq$ States and $\delta \subseteq \text{States} \times \text{Alphabet} \times \text{States}$

Atoms are $(\mathbf{Q}, \leq)$.

An automaton can accept sequences

$$q_1 \; q_2 \; q_3 \; q_4 \; \dots \; q_n$$

such that $q_1 < q_2 < q_3 < q_4 < \dots < q_n$

Atoms are $(\mathbf{N}, =)$.

An automaton can accept sequences

$$a_1 \; a_2 \; a_3 \; a_4 \; \dots \; a_n$$

such that $a_i = a_j$ for some $i \neq j$

Atoms are $(\mathbf{N}, =)$.

An automaton can accept sequences

$$\{a_1, a_2\} \; \{a_3, a_4\} \; \{a_5, a_6\} \; \dots\dots\dots\dots \; \{a_{2n-1}, a_{2n}\}$$

deterministic automata ≠ nondeterministic automata

automata

A tuple of hereditarily definable sets

$$(\text{States}, \text{Alphabet}, \text{Initial}, \text{Accepting}, \delta)$$

where Initial, Accepting $\subseteq$ States and $\delta \subseteq \text{States} \times \text{Alphabet} \times \text{States}$

Atoms are $(\mathbf{Q}, \leq)$.

An automaton can accept sequences

$$q_1\ q_2\ q_3\ q_4 \dots q_n$$

such that $q_1 < q_2 < q_3 < q_4 < \dots < q_n$

Atoms are $(\mathbf{N}, =)$.

An automaton can accept sequences

$$a_1\ a_2\ a_3\ a_4 \dots a_n$$

such that $a_i = a_j$ for some $i \neq j$

Atoms are $(\mathbf{N}, =)$.

An automaton can accept sequences

$$\{a_1, a_2\}\ \{a_3, a_4\}\ \{a_5, a_6\}\ \dots\dots\dots\dots\ \{a_{2n-1}, a_{2n}\}$$

deterministic automata ≠ nondeterministic automata

Can model some infinite-state systems with restricted data access e.g. register automata (Kaminsky–Francez) etc.

automata

A tuple of hereditarily definable sets

$$(\text{States}, \text{Alphabet}, \text{Initial}, \text{Accepting}, \delta)$$

where Initial, Accepting ⊆ States and $\delta \subseteq \text{States} \times \text{Alphabet} \times \text{States}$

Atoms are $(\mathbf{Q}, \leq)$.

An automaton can accept sequences

$$q_1\ q_2\ q_3\ q_4 \ldots q_n$$

such that $q_1 < q_2 < q_3 < q_4 < \ldots < q_n$

Atoms are $(\mathbf{N}, =)$.

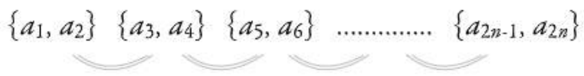
An automaton can accept sequences

$$a_1\ a_2\ a_3\ a_4 \ldots a_n$$

such that $a_i = a_j$ for some $i \neq j$

Atoms are $(\mathbf{N}, =)$.

An automaton can accept sequences



$$\{a_1, a_2\}\ \{a_3, a_4\}\ \{a_5, a_6\}\ \ldots\ldots\ldots\ldots\ \{a_{2n-1}, a_{2n}\}$$

deterministic automata ≠ nondeterministic automata

Can model some infinite-state systems with restricted data access e.g. register automata (Kaminsky–Francez) etc.

computational problems:

- emptiness
- language equality
- minimization
- …

Turing machines

Turing machines

A tuple of hereditarily definable sets

$$(\text{States}, \text{Alphabet}, \delta)$$

where $\delta \subseteq \text{States} \times \text{Alphabet} \times \text{States} \times \text{Alphabet} \times \{\leftarrow, \rightarrow\}$

when Atoms = $(\mathbf{Q}, \leq)$:

deterministic = nondeterministic

when Atoms $=((\mathbf{Z}/2\mathbf{Z})^{\omega},+)$:

$$\mathrm{P} \neq \mathrm{NP}$$

separating language:

sequences of vectors $v_1\ v_2\ v_3\ v_4\ \ \ldots\ldots\ \ v_n$

which are linearly dependent

when Atoms = ($\mathbf{N}$,=):

$$P \neq NP$$

deterministic ≠ nondeterministic

when Atoms = $(\mathbf{N},=)$:

$$P \neq NP$$

deterministic ≠ nondeterministic

separating language:

sequences $a_1 a_2 a_3 \ldots a_n b_1 b_2 b_3 \ldots b_n$

of elements of $A = \{\{\{a,b,c\},\{d,e,f\}\} : a,b,c,d,e,f \in \mathbf{N}\}$

when Atoms = $(\mathbf{N},=)$:

$$\mathrm{P} \neq \mathrm{NP}$$

deterministic ≠ nondeterministic

separating language:

sequences $a_1\, a_2\, a_3 \ldots\, a_n\, b_1\, b_2\, b_3 \ldots\, b_n$

of elements of $A = \{\{\{a,b,c\},\{d,e,f\}\} \colon a,b,c,d,e,f \in \mathbf{N}\}$

such that for some bijection $\pi\colon \mathbf{N} \to \mathbf{N}$

$$(b_1\, b_2 \ldots\, b_n) = (\pi(\mathrm{a}_1)\pi(\mathrm{a}_2)\ldots\pi(\mathrm{a}_n))$$

when Atoms = $(\mathbf{N},=)$:

P ≠ NP

deterministic ≠ nondeterministic

separating language:

sequences $a_1\, a_2\, a_3 \dots\, a_n\, b_1\, b_2\, b_3 \dots\, b_n$

of elements of $A = \{\{\{a,b,c\},\{d,e,f\}\} : a,b,c,d,e,f \in \mathbf{N}\}$

such that for some bijection $\pi\colon \mathbf{N} \to \mathbf{N}$

$$(b_1\, b_2 \dots\, b_n) = (\pi(\mathrm{a}_1)\pi(\mathrm{a}_2)\dots\pi(\mathrm{a}_n))$$

related to:

- Cai-Furer-Immermann graphs
- universal algebra
- A is is not homogeneous in a finite relational/functional language

Turing machines

A tuple of hereditarily definable sets

(States, Alphabet, δ)

where $\delta \subseteq \text{States} \times \text{Alphabet} \times \text{States} \times \text{Alphabet} \times \{\leftarrow,\rightarrow\}$

when Atoms = $(\mathbf{Q},\leq)$:
deterministic = nondeterministic

when Atoms = $((\mathbf{Z}/2\mathbf{Z})^{\omega},+)$:

P ≠ NP

separating language:

sequences of vectors $v_1\ v_2\ v_3\ v_4\ \ldots.\ v_n$
which are linearly dependent

when Atoms = $(\mathbf{N},=)$:

P ≠ NP

deterministic ≠ nondeterministic

separating language:

sequences $a_1 a_2 a_3 \ldots a_n b_1 b_2 b_3 \ldots b_n$

of elements of $A = \{\{\{a,b,c\},\{d,e,f\}\} : a,b,c,d,e,f \in \mathbf{N}\}$

such that for some bijection $\pi: \mathbf{N} \to \mathbf{N}$
$(b_1 b_2 \ldots b_n) = (\pi(a_1)\pi(a_2)\ldots\pi(a_n))$

related to:
- Cai-Furer-Immermann graphs
- universal algebra
- A is is not homogeneous in a finite relational/functional language

Hereditarily definable *X*

graphs

pair (V,E) of hereditarily definable sets with $E \subseteq \binom{V}{2}$

infinite clique

Johnson graph

some other graph

decision problems:

- connectedness
- 3-colorability
- homomorphism
- isomorphism
- ...

} *discussed later*

automata

A tuple of hereditarily definable sets

$$(\text{States}, \text{Alphabet}, \text{Initial}, \text{Accepting}, \delta)$$

where Initial, Accepting ⊆ States and $\delta \subseteq \text{States} \times \text{Alphabet} \times \text{States}$

Can model some infinite-state systems with restricted data access
e.g. register automata (Kaminsky–Francez) etc.

computational problems:

- emptiness
- language equality
- minimization
- ...

Turing machines

A tuple of hereditarily definable sets

$$(\text{States}, \text{Alphabet}, \delta)$$

where $\delta \subseteq \text{States} \times \text{Alphabet} \times \text{States} \times \text{Alphabet} \times \{\leftarrow, \rightarrow\}$

P ≠ NP

P ≠ NP

deterministic ≠ nondeterministic

History

1. In set theory, Fraenkel and Mostowski studied sets constructed on top of an underlying set of *urelementa* or *atoms*.

   "Hereditarily definable sets" are a special case of these, and have finite syntax.

2. Gabbay and Pitts (2002) rediscovered Fraenkel-Mostowski sets in the case of atoms ($\mathbf{N}$,=), in the context of *name binding in semantics,* and called them *nominal sets.*

3. Bojańczyk et al. (2011) rediscovered these sets in the case of *homogeneous atoms,* in the context of *automata theory* and called them *orbit-finite sets with atoms.*

4. Up to isomorphism, a structure is *hereditarily definable* ⇔ it *interprets* in Atoms

Computation with Atoms

Szymon Toruńczyk, University of Warsaw

includes work with Bojańczyk, Klin, Kopczyński, Lasota, Ochremiak,...

Atoms

a fixed *underlying* logical structure

Examples:

- $(\mathbb{N}, =)$ – *pure set* ← main example in this talk
- $(\mathbb{Q}, \leq)$ – *dense order*
- $(\mathbb{R}, +, \times, 0, 1)$ – *field of reals*
- $(\mathbb{N}, +, \leq)$ – *Presburger arithmetic*

Hereditarily definable set

Examples

5 if $5 \in$ Atoms

$\{a : a \in \text{Atoms}\}$

$\{a : a \in \text{Atoms}, a \neq 7 \wedge a \neq 5\}$ if $5,7 \in$ Atoms

$\{\{a : a \in \text{Atoms}, a \neq b\} : b \in \text{Atoms}\}$

$\{\{b : b \in \text{Atoms}, a < b \wedge b < c\} : a, c \in \text{Atoms}, a < c\}$ if Atoms = $(\mathbb{Q}, \leq)$

$\{x, y\} \overset{\text{def}}{=} \{x\} \cup \{y\}$ $\quad (x, y) \overset{\text{def}}{=} \{x, \{x, y\}\}$

Syntax

hdef ::= variable | parameter from Atoms

| { hdef : variable,...,variable ∈ **Atoms**, first order formula } (in language of Atoms, with parameters)

| hdef ∪ hdef

Hereditarily definable sets have finite descriptions

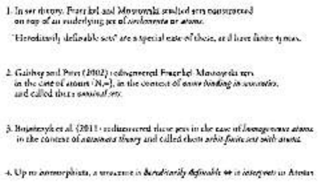
e.g. $\{\{a : a \in \text{Atoms}, a \neq b\} : b \in \text{Atoms}\}$

→ can be input and processed by algorithms

equality of sets is decidable ⇔ the theory of Atoms is decidable

Hereditarily definable X

History



Berkeley, February 8, 2019

Computation with Atoms

Szymon Toruńczyk, University of Warsaw

includes work with Bojańczyk, Klin, Kopczyński, Lasota, Ochremiak,...

Atoms

a fixed *underlying* logical structure

Examples:

$(\mathbb{N}, =)$ – *pure set* ← main example in this talk

$(\mathbb{Q}, \leq)$ – *dense order*

$(\mathbb{R}, +, \times, 0, 1)$ – *field of reals*

$(\mathbb{N}, +, \leq)$ – *Presburger arithmetic*

Hereditarily definable set

Examples

5 if $5 \in$ Atoms

$\{a : a \in \text{Atoms}\}$

$\{a : a \in \text{Atoms}, a \neq 7 \wedge a \neq 5\}$ if $5,7 \in$ Atoms

$\{\{a : a \in \text{Atoms}, a \neq b\} : b \in \text{Atoms}\}$

$\{\{b : b \in \text{Atoms}, a < b \wedge b < c\} : a, c \in \text{Atoms}, a < c\}$ if Atoms = $(\mathbb{Q}, \leq)$

$\{x,y\} \overset{\text{def}}{=} \{x\} \cup \{y\}$ $(x,y) \overset{\text{def}}{=} \{x, \{x,y\}\}$

Syntax

hdef ::= variable | parameter from Atoms

| { hdef : variable,...,variable ∈ **Atoms**, first order formula } (first order formula in language of Atoms, with parameters)

| hdef ∪ hdef

Hereditarily definable sets have finite descriptions

e.g. $\{\{a : a \in \text{Atoms}, a \neq b\} : b \in \text{Atoms}\}$

→ can be input and processed by algorithms

equality of sets is decidable ⇔ the theory of Atoms is decidable

Hereditarily definable *X*

graphs *automata* *Turing machines*

(States, Alphabet, Initial, Accepting, δ)

(States, Alphabet, δ)

History

1. In set theory, Fraenkel and Mostowski studied sets constructed on top of an underlying set of *urelements* or *atoms*.

   "Hereditarily definable sets" are a special case of these, and have finite names.

2. Gabbay and Pitts (2002) rediscovered Fraenkel-Mostowski sets in the case of atoms (N,=), in the context of *name binding in semantics*, and called them *nominal sets*.

3. Bojańczyk et al. (2011) rediscovered these sets in the case of *homogeneous atoms* in the context of *automata theory* and called them *orbit-finite sets with atoms*.

4. Up to isomorphism, a structure is *hereditarily definable* ⇔ it *interprets* in Atoms

Computation with Atoms

Szymon Toruńczyk, University of Warsaw

includes work with Bojańczyk, Klin, Kopczyński, Lasota, Ochremiak,...

Atoms

a fixed *underlying* logical structure

Examples:

(N , =) – *pure set* ← main example in this talk

(Q , ≤) – *dense order*

(R , +, ×, 0, 1) – *field of reals*

(N , +, ≤) – *Presburger arithmetic*

Hereditarily definable set

Examples

5 if $5 \in$ Atoms

$\{a: a \in \text{Atoms}\}$

$\{a: a \in \text{Atoms}, a \neq 7 \wedge a \neq 5\}$ if $5,7 \in$ Atoms

$\{\{a: a \in \text{Atoms}, a \neq b\}: b \in \text{Atoms}\}$

$\{\{b: b \in \text{Atoms}, a<b \wedge b<c\}: a,c \in \text{Atoms}, a<c\}$ if Atoms = $(\mathbf{Q}, \leq)$

$\{x,y\} \overset{\text{def}}{=} \{x\} \cup \{y\}$ $(x,y) \overset{\text{def}}{=} \{x, \{x,y\}\}$

Syntax

hdef ::= variable | parameter from Atoms

| { hdef : variable,...,variable ∈ **Atoms**, first order formula } (in language of Atoms, with parameters)

| hdef ∪ hdef

Hereditarily definable sets have finite descriptions

e.g. $\{\{a: a \in \text{Atoms}, a \neq b\}: b \in \text{Atoms}\}$

→ can be input and processed by algorithms

equality of sets is decidable ⇔ the theory of Atoms is decidable

Hereditarily definable *X*

graphs *automata* *Turing machines*

(States, Alphabet, Initial, Accepting, δ)

(States, Alphabet, δ)

History

1. In set theory, Fraenkel and Mostowski studied sets constructed on top of an underlying set of *urelements* or *atoms*.

   "Hereditarily definable sets" are a special case of these, and have finite supports.

2. Gabbay and Pitts (2002) rediscovered Fraenkel-Mostowski sets in the case of atoms (N,=), in the context of *name binding in semantics*, and called them *nominal sets*.

3. Bojańczyk et al. (2011) rediscovered these sets in the case of *homogeneous atoms* in the context of *automata theory* and called them *orbit-finite sets with atoms*.

4. Up to isomorphism, a structure is *hereditarily definable* ⇔ it *interprets* in Atoms

Computational Problems

definable sets can be presented as input to algorithms

Computational Problems

definable sets can be presented as input to algorithms

- Graph reachability
- Deterministic automata minimisation
- Context-free grammar emptiness
- Tree/pushdown automata emptiness
- Graph planarity
- Graph isomorphism
- Graph 3–colorability
- Solvability of systems of equations over finite field
- Satisfiability of sets of clauses
- Constraint Satisfaction Problems over finite template
- Homomorphism problem

Reachability

Input: a hereditarily definable graph $G=(V,E)$, vertices $s,t \in V$

Decide: is t reachable from s?

decidable when atoms are $(\mathbf{N},=)$ or $(\mathbf{Q},\leq)$

Reachability

Input: a hereditarily definable graph $G=(V,E)$, vertices $s,t \in V$

Decide: is t reachable from s?

decidable when atoms are $(\mathbf{N},=)$ or $(\mathbf{Q},\leq)$

undecidable for Atoms = $(\mathbf{N},+1)$ (~Minsky machines)

Algorithm pseudocode

```
function reachability(V,E,s,t)

R_0 := {s};
n=0;
repeat
    $R_{n+1}$ := $R_n$ ∪ {w | (v,w)∈E, v∈$R_n$, w∈V};
    n := n+1;
until ($R_n$=$R_{n-1}$);
return (t∈$R_n$)
```

Algorithm pseudocode

```
function reachability(V,E,s,t)

$R_0$ := {s};
n=0;
repeat
   $R_{n+1}$ := $R_n$ ∪ {w | (v,w)∈E, v∈$R_n$, w∈V};
   n := n+1;
until ($R_n$=$R_{n-1}$);
return (t∈$R_n$)
```

Terminates for *ω-categorical* Atoms

tes for ω-*categorical* A

Examples: (**N**,=),(**Q**,≤), Rado graph

Theorem [Ryll-Nardzewski, Engeler, Svenonius]

a structure Atoms is ω–*categorical* if and only if for all $n \in \mathbf{N}$

tes for *ω-categorical* A

Examples: (**N**,=),(**Q**,≤), Rado graph

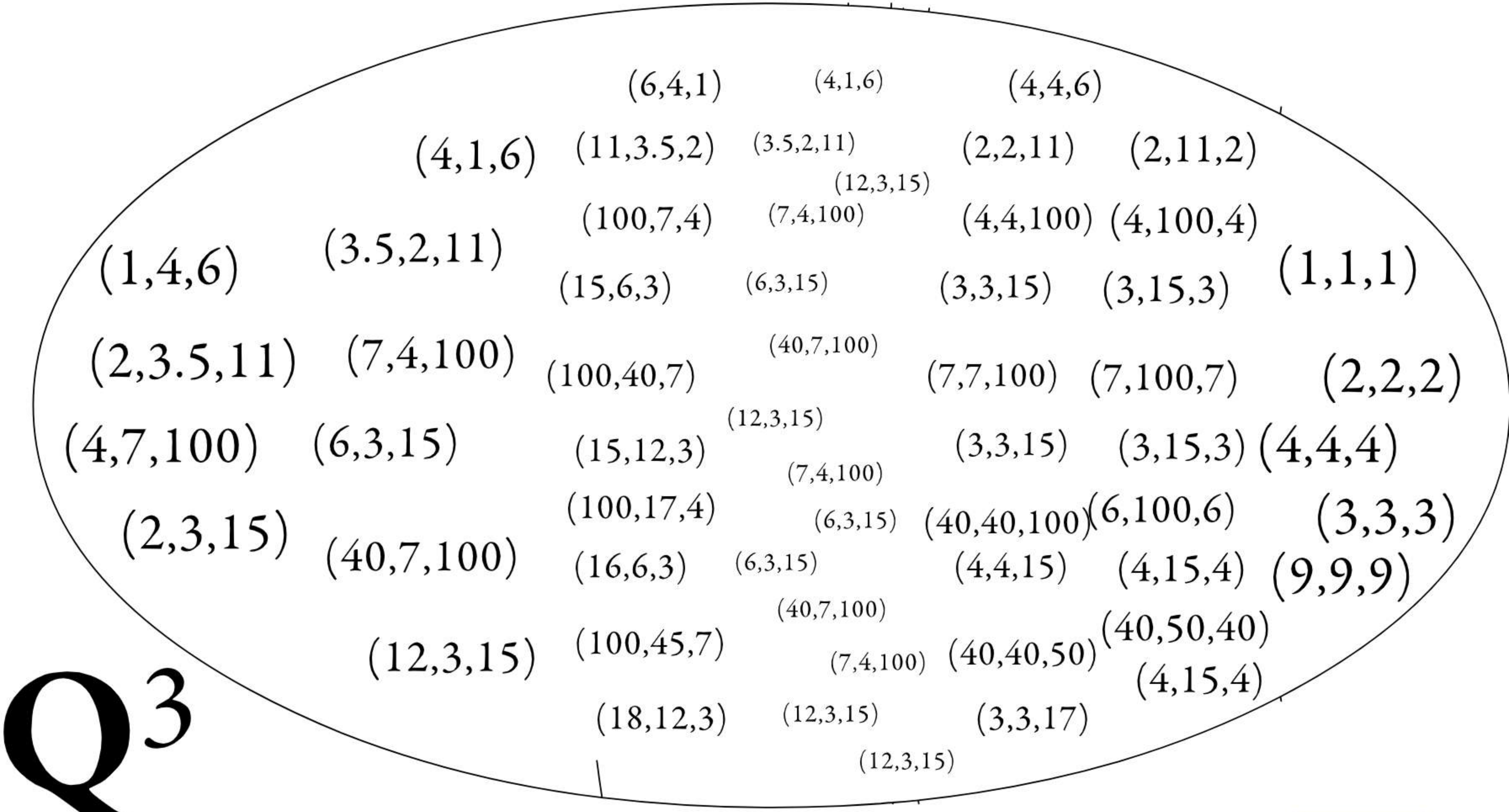
Theorem [Ryll-Nardzewski, Engeler, Svenonius]

a structure Atoms is *ω–categorical* if and only if for all $n \in \mathbf{N}$

Atoms^n *has finitely many orbits under the action of Aut(*Atoms*)*

nitely many orbits under the action of

Atoms = $(\mathbf{Q},\leq)$



(6,4,1) (4,1,6) (4,4,6)

(4,1,6) (11,3.5,2) (3.5,2,11) (2,2,11) (2,11,2)

(12,3,15)

(100,7,4) (7,4,100) (4,4,100) (4,100,4)

(1,4,6) (3.5,2,11) (1,1,1)

(15,6,3) (6,3,15) (3,3,15) (3,15,3)

(40,7,100)

(2,3.5,11) (7,4,100) (100,40,7) (7,7,100) (7,100,7) (2,2,2)

(12,3,15)

(4,7,100) (6,3,15) (15,12,3) (3,3,15) (3,15,3) (4,4,4)

(7,4,100)

(100,17,4) (6,3,15) (40,40,100) (6,100,6) (3,3,3)

(2,3,15) (40,7,100) (16,6,3) (6,3,15) (4,4,15) (4,15,4) (9,9,9)

(40,7,100)

(100,45,7) (40,50,40)

(12,3,15) (7,4,100) (40,40,50)

(4,15,4)

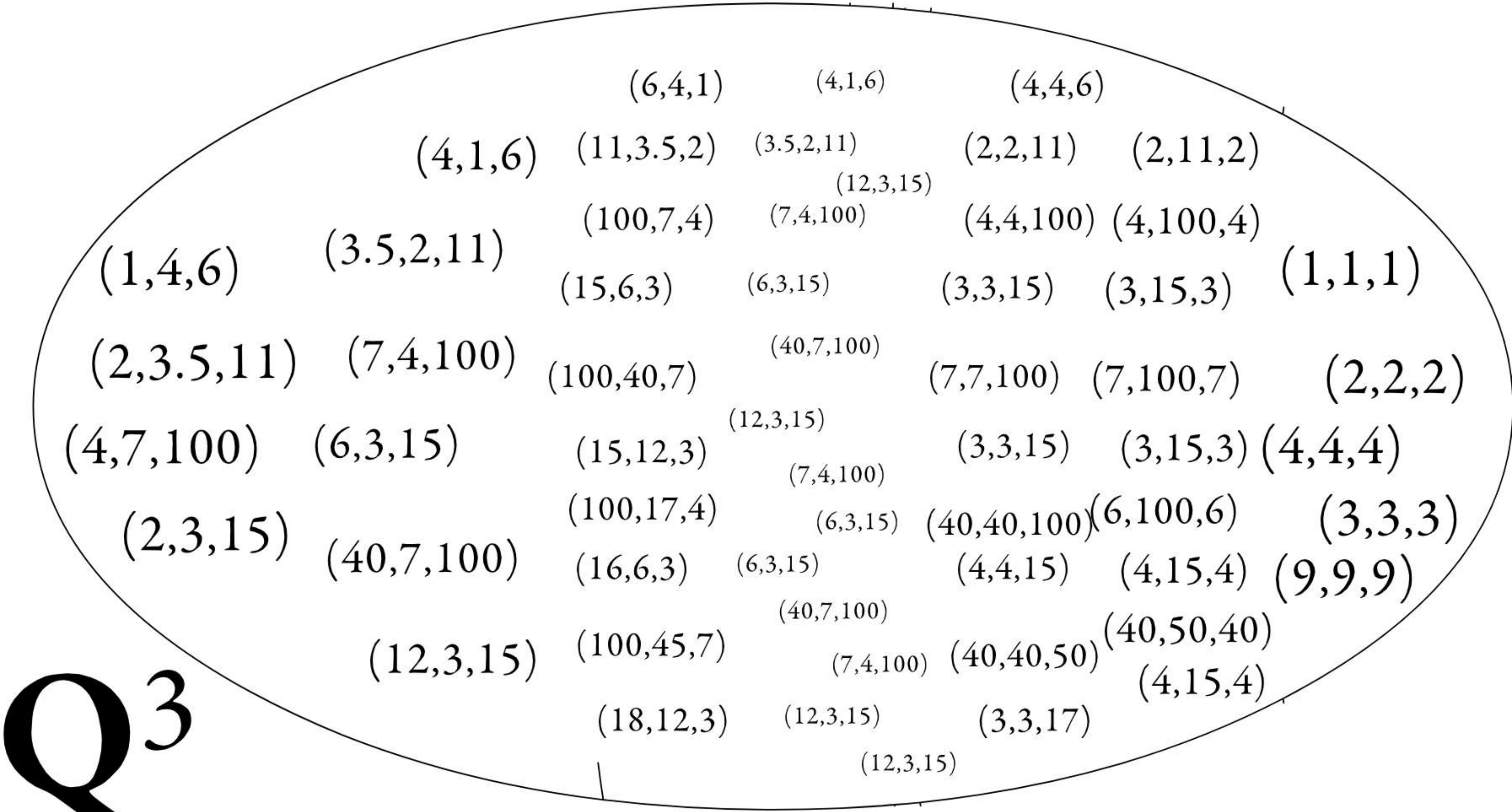
(18,12,3) (12,3,15) (3,3,17)

(12,3,15)

$\mathbf{Q}^3$

nitely many orbits under the action of

Atoms = $(\mathbf{Q}, \leq)$



(6,4,1) (4,1,6) (4,4,6)

(4,1,6) (11,3.5,2) (3.5,2,11) (2,2,11) (2,11,2)

(12,3,15)

(100,7,4) (7,4,100) (4,4,100) (4,100,4)

(3.5,2,11)

(1,4,6) (15,6,3) (6,3,15) (3,3,15) (3,15,3) (1,1,1)

(40,7,100)

(2,3.5,11) (7,4,100) (100,40,7) (7,7,100) (7,100,7) (2,2,2)

(12,3,15)

(4,7,100) (6,3,15) (15,12,3) (3,3,15) (3,15,3) (4,4,4)

(7,4,100)

(100,17,4) (6,100,6) (3,3,3)

(2,3,15) (6,3,15) (40,40,100)

(40,7,100) (16,6,3) (6,3,15) (4,4,15) (4,15,4) (9,9,9)

(40,7,100)

(40,50,40)

(12,3,15) (100,45,7) (7,4,100) (40,40,50)

(4,15,4)

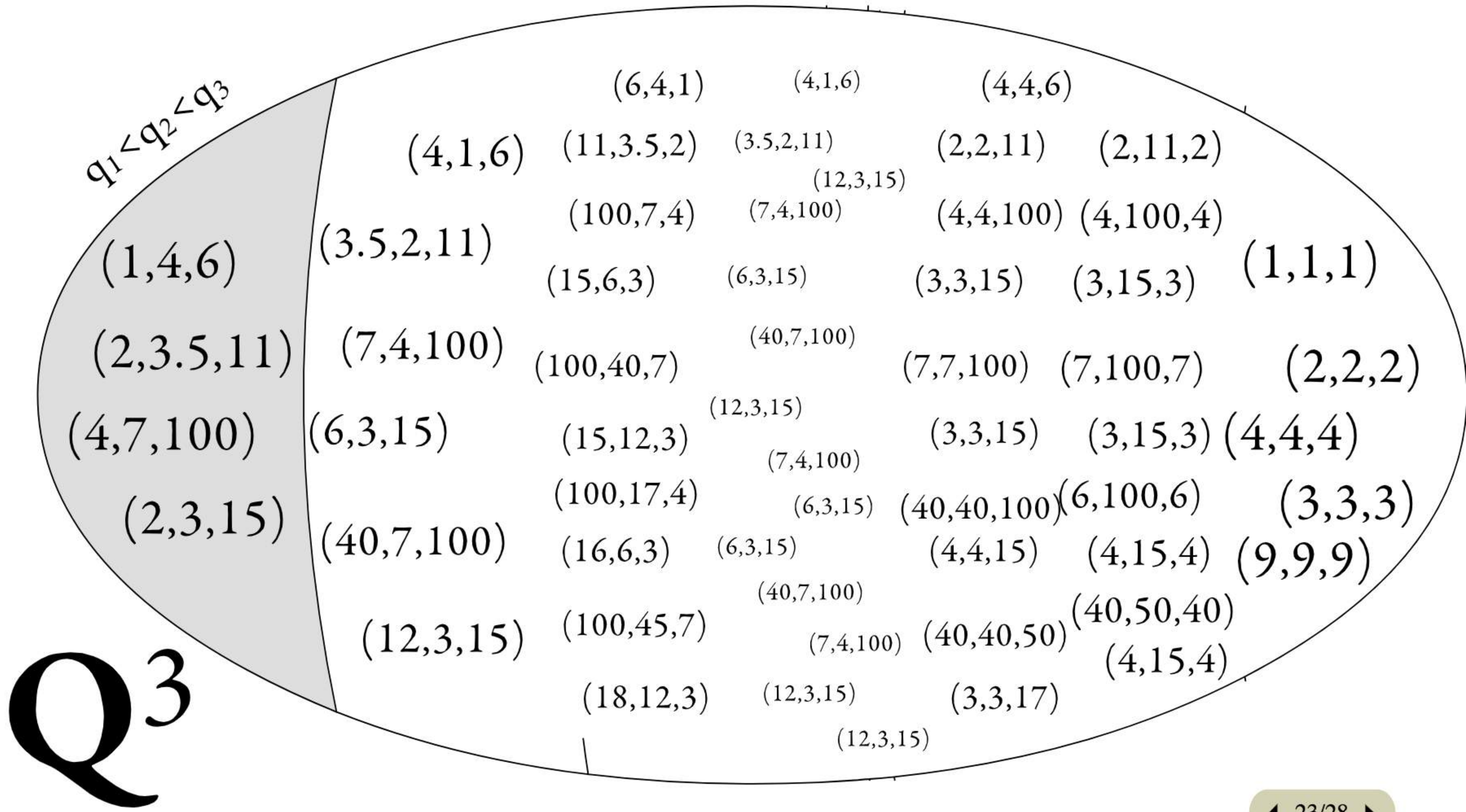
(18,12,3) (12,3,15) (3,3,17)

(12,3,15)

$\mathbf{Q}^3$

nitely many orbits under the action of

Atoms = $(\mathbf{Q},\leq)$



$q_1<q_2<q_3$

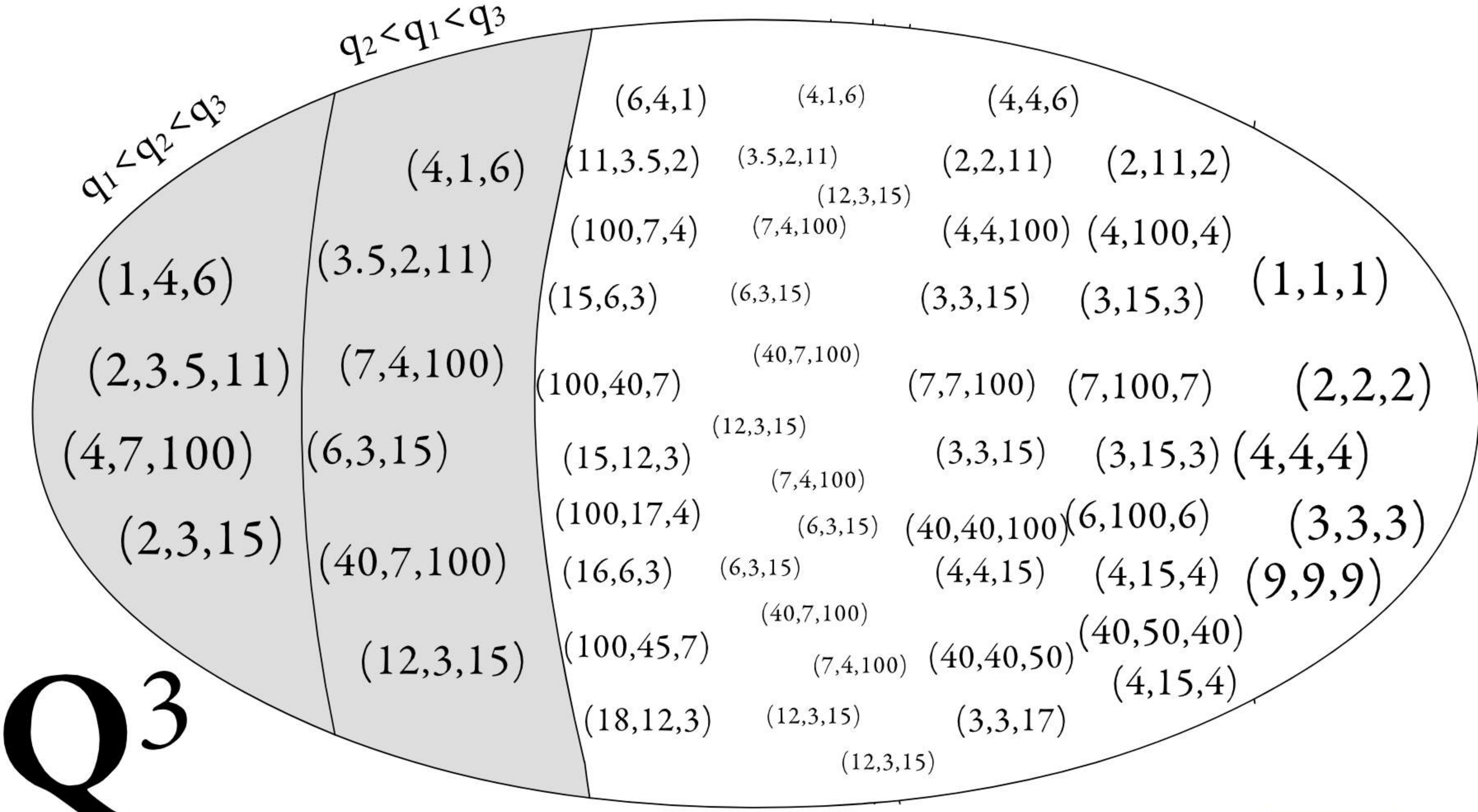
(1,4,6)
(2,3.5,11)
(4,7,100)
(2,3,15)

(6,4,1) (4,1,6) (4,4,6)
(4,1,6) (11,3.5,2) (3.5,2,11) (2,2,11) (2,11,2)
(12,3,15)
(100,7,4) (7,4,100) (4,4,100) (4,100,4)
(3.5,2,11)
(15,6,3) (6,3,15) (3,3,15) (3,15,3) (1,1,1)
(40,7,100)
(7,4,100) (100,40,7) (7,7,100) (7,100,7) (2,2,2)
(12,3,15)
(6,3,15) (15,12,3) (3,3,15) (3,15,3) (4,4,4)
(7,4,100)
(100,17,4) (6,3,15) (40,40,100) (6,100,6) (3,3,3)
(40,7,100) (16,6,3) (6,3,15) (4,4,15) (4,15,4) (9,9,9)
(40,7,100)
(40,50,40)
(12,3,15) (100,45,7) (7,4,100) (40,40,50)
(4,15,4)
(18,12,3) (12,3,15) (3,3,17)
(12,3,15)

$\mathbf{Q}^3$

nitely many orbits under the action of

Atoms = $(\mathbf{Q},\leq)$



$q_1<q_2<q_3$

(1,4,6)
(2,3.5,11)
(4,7,100)
(2,3,15)

$q_2<q_1<q_3$

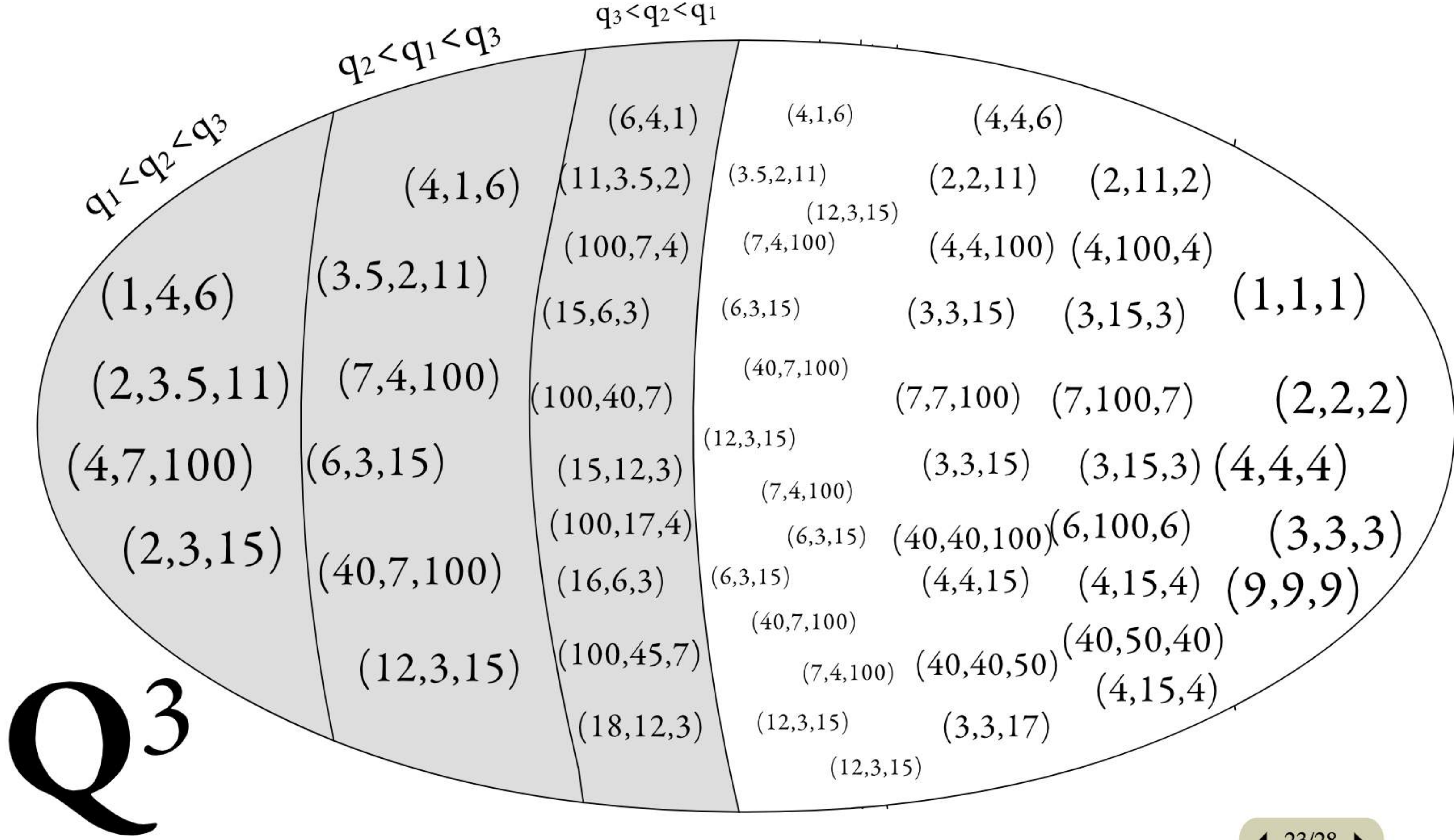
(4,1,6)
(3.5,2,11)
(7,4,100)
(6,3,15)
(40,7,100)
(12,3,15)

(6,4,1) (4,1,6) (4,4,6)
(11,3.5,2) (3.5,2,11) (2,2,11) (2,11,2)
(12,3,15)
(100,7,4) (7,4,100) (4,4,100) (4,100,4)
(1,1,1)
(15,6,3) (6,3,15) (3,3,15) (3,15,3)
(40,7,100)
(100,40,7) (7,7,100) (7,100,7) (2,2,2)
(12,3,15)
(15,12,3) (3,3,15) (3,15,3) (4,4,4)
(7,4,100)
(100,17,4) (6,3,15) (40,40,100) (6,100,6) (3,3,3)
(16,6,3) (6,3,15) (4,4,15) (4,15,4) (9,9,9)
(40,7,100)
(40,50,40)
(100,45,7) (7,4,100) (40,40,50)
(4,15,4)
(18,12,3) (12,3,15) (3,3,17)
(12,3,15)

$\mathbf{Q}^3$

nitely many orbits under the action of

Atoms = ($\mathbf{Q}$,≤)



$q_3 < q_2 < q_1$

$q_2 < q_1 < q_3$

$q_1 < q_2 < q_3$

(1,4,6)
(2,3.5,11)
(4,7,100)
(2,3,15)

(4,1,6)
(3.5,2,11)
(7,4,100)
(6,3,15)
(40,7,100)
(12,3,15)

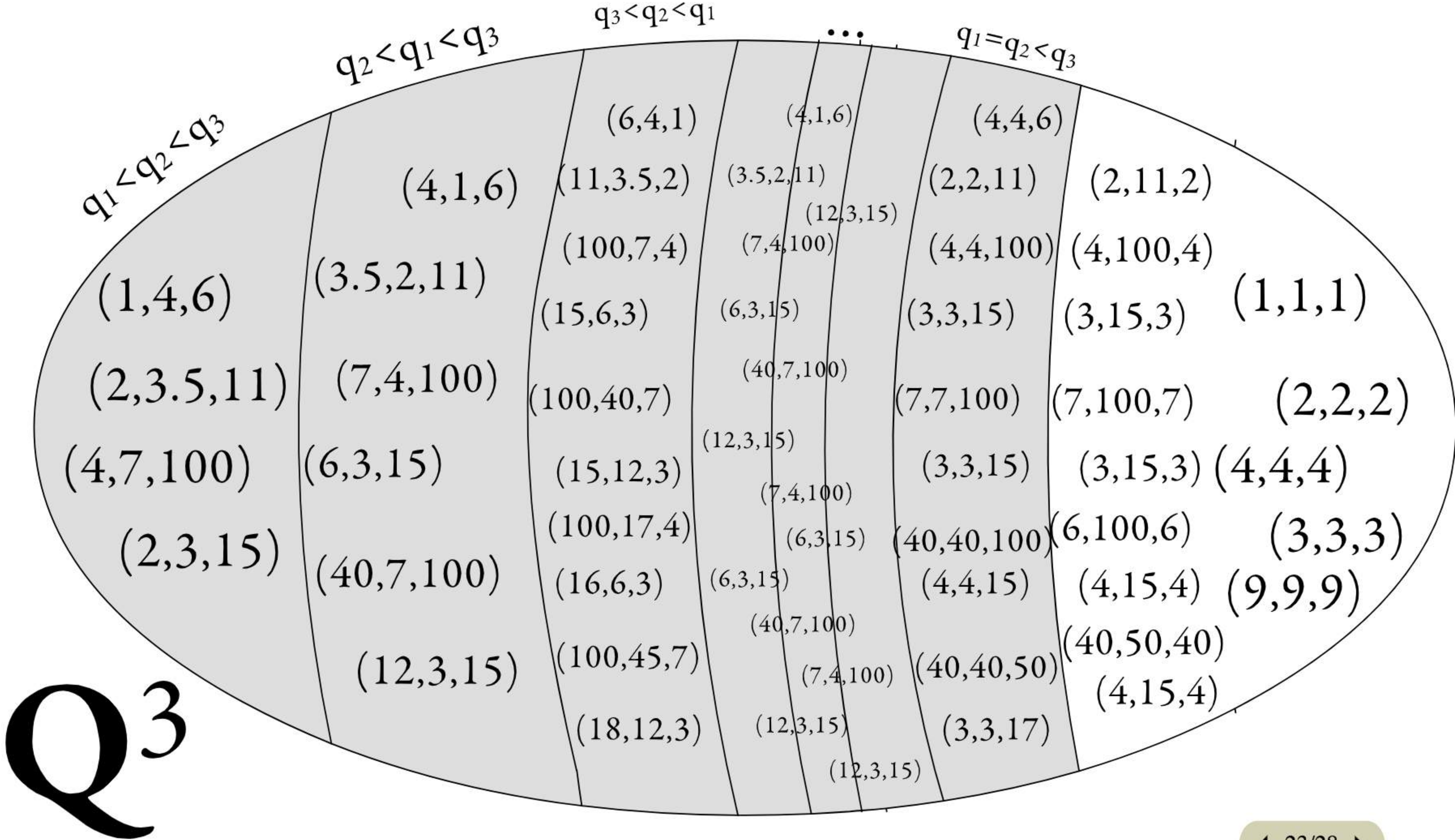
(6,4,1)
(11,3.5,2)
(100,7,4)
(15,6,3)
(100,40,7)
(15,12,3)
(100,17,4)
(16,6,3)
(100,45,7)
(18,12,3)

(4,1,6) (4,4,6)
(3.5,2,11) (2,2,11) (2,11,2)
(12,3,15)
(7,4,100) (4,4,100) (4,100,4)
(1,1,1)
(6,3,15) (3,3,15) (3,15,3)
(40,7,100)
(7,7,100) (7,100,7) (2,2,2)
(12,3,15)
(3,3,15) (3,15,3) (4,4,4)
(7,4,100)
(6,3,15) (40,40,100) (6,100,6) (3,3,3)
(6,3,15) (4,4,15) (4,15,4) (9,9,9)
(40,7,100)
(40,50,40)
(7,4,100) (40,40,50)
(4,15,4)
(12,3,15) (3,3,17)
(12,3,15)

$\mathbf{Q}^3$

nitely many orbits under the action of

Atoms = $(\mathbf{Q},\leq)$



$q_1<q_2<q_3$

(1,4,6)
(2,3.5,11)
(4,7,100)
(2,3,15)

$q_2<q_1<q_3$

(4,1,6)
(3.5,2,11)
(7,4,100)
(6,3,15)
(40,7,100)
(12,3,15)

$q_3<q_2<q_1$

(6,4,1)
(11,3.5,2)
(100,7,4)
(15,6,3)
(100,40,7)
(15,12,3)
(100,17,4)
(16,6,3)
(100,45,7)
(18,12,3)

…

(4,1,6)
(3.5,2,11)
(12,3,15)
(7,4,100)
(6,3,15)
(40,7,100)
(12,3,15)
(7,4,100)
(6,3,15)
(6,3,15)
(40,7,100)
(7,4,100)
(12,3,15)
(12,3,15)

$q_1=q_2<q_3$

(4,4,6)
(2,2,11)
(4,4,100)
(3,3,15)
(7,7,100)
(3,3,15)
(40,40,100)
(4,4,15)
(40,40,50)
(3,3,17)

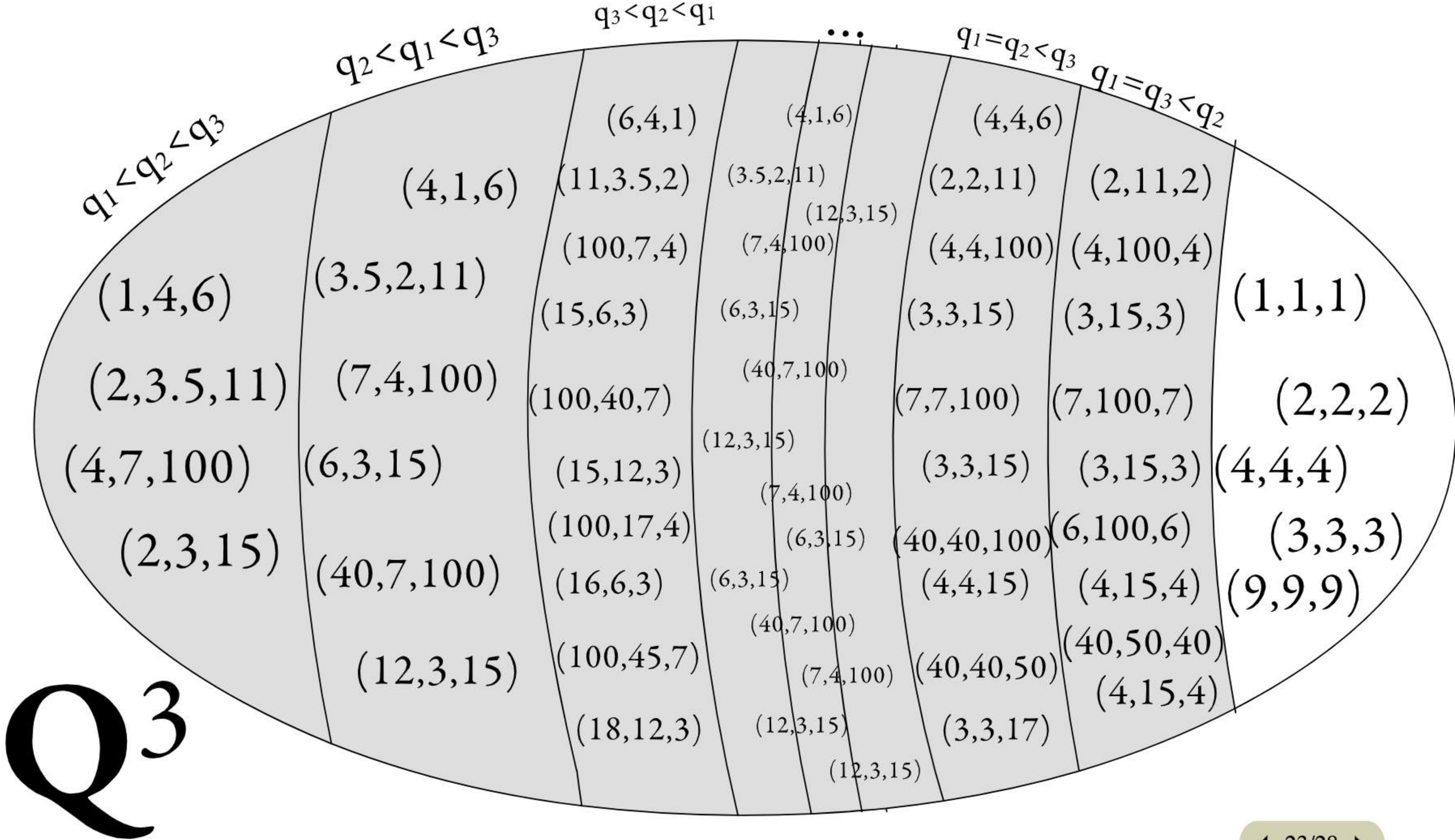
(2,11,2)
(4,100,4)
(3,15,3)
(7,100,7)
(3,15,3)
(6,100,6)
(4,15,4)
(40,50,40)
(4,15,4)

(1,1,1)
(2,2,2)
(4,4,4)
(3,3,3)
(9,9,9)

$\mathbf{Q}^3$

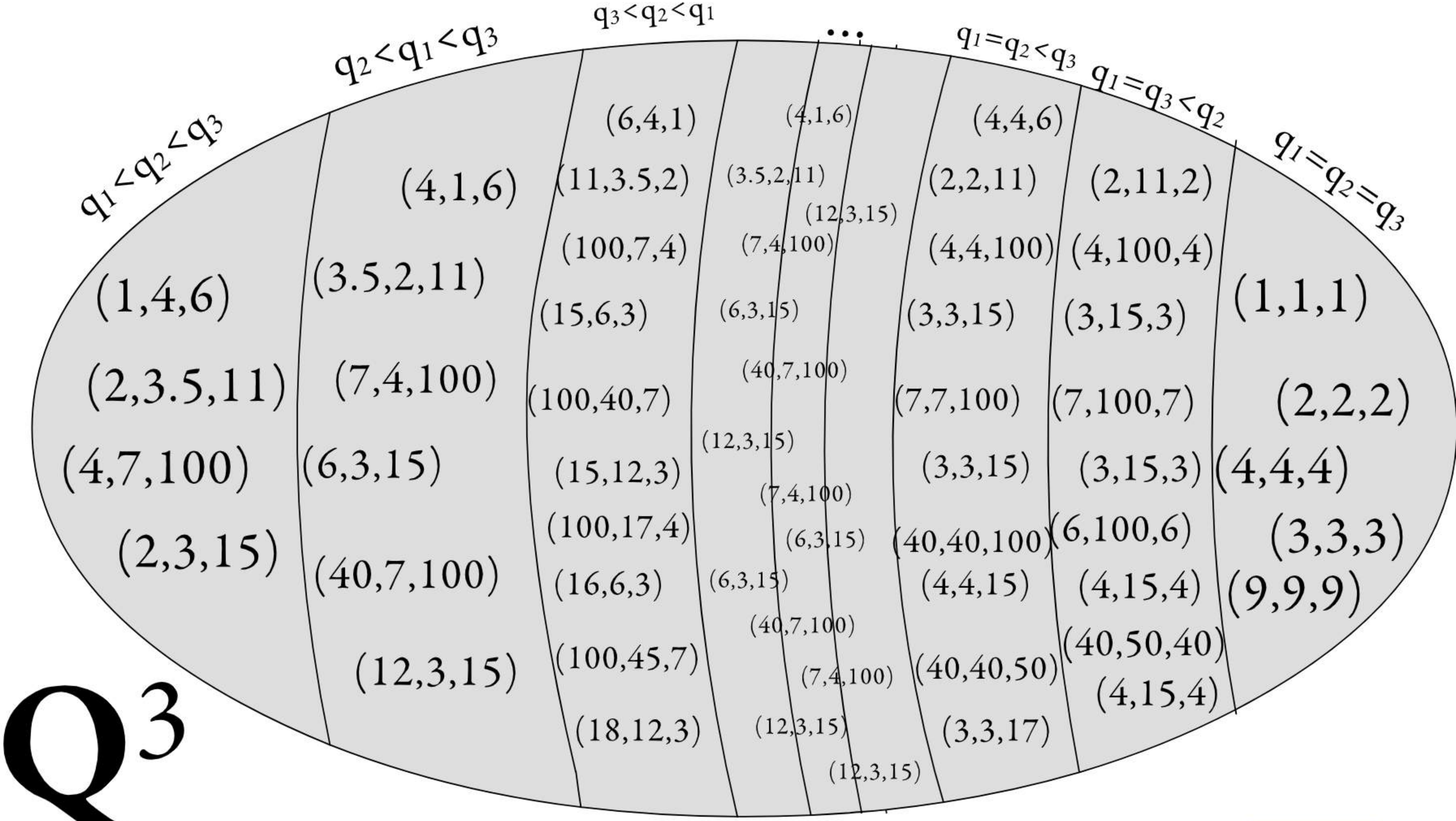
nitely many orbits under the action of

$\text{Atoms} = (\mathbf{Q}, \leq)$



nitely many orbits under the action of

Atoms = (**Q**,≤)



q₁<q₂<q₃: (1,4,6) (2,3.5,11) (4,7,100) (2,3,15)

q₂<q₁<q₃: (4,1,6) (3.5,2,11) (7,4,100) (6,3,15) (40,7,100) (12,3,15)

q₃<q₂<q₁: (6,4,1) (11,3.5,2) (100,7,4) (15,6,3) (100,40,7) (15,12,3) (100,17,4) (16,6,3) (100,45,7) (18,12,3)

…: (4,1,6) (3.5,2,11) (12,3,15) (7,4,100) (6,3,15) (40,7,100) (12,3,15) (7,4,100) (6,3,15) (6,3,15) (40,7,100) (7,4,100) (12,3,15) (12,3,15)

q₁=q₂<q₃: (4,4,6) (2,2,11) (4,4,100) (3,3,15) (7,7,100) (3,3,15) (40,40,100) (4,4,15) (40,40,50) (3,3,17)

q₁=q₃<q₂: (2,11,2) (4,100,4) (3,15,3) (7,100,7) (3,15,3) (6,100,6) (4,15,4) (40,50,40) (4,15,4)

q₁=q₂=q₃: (1,1,1) (2,2,2) (4,4,4) (3,3,3) (9,9,9)

$\mathbf{Q}^3$

tes for *ω-categorical* A

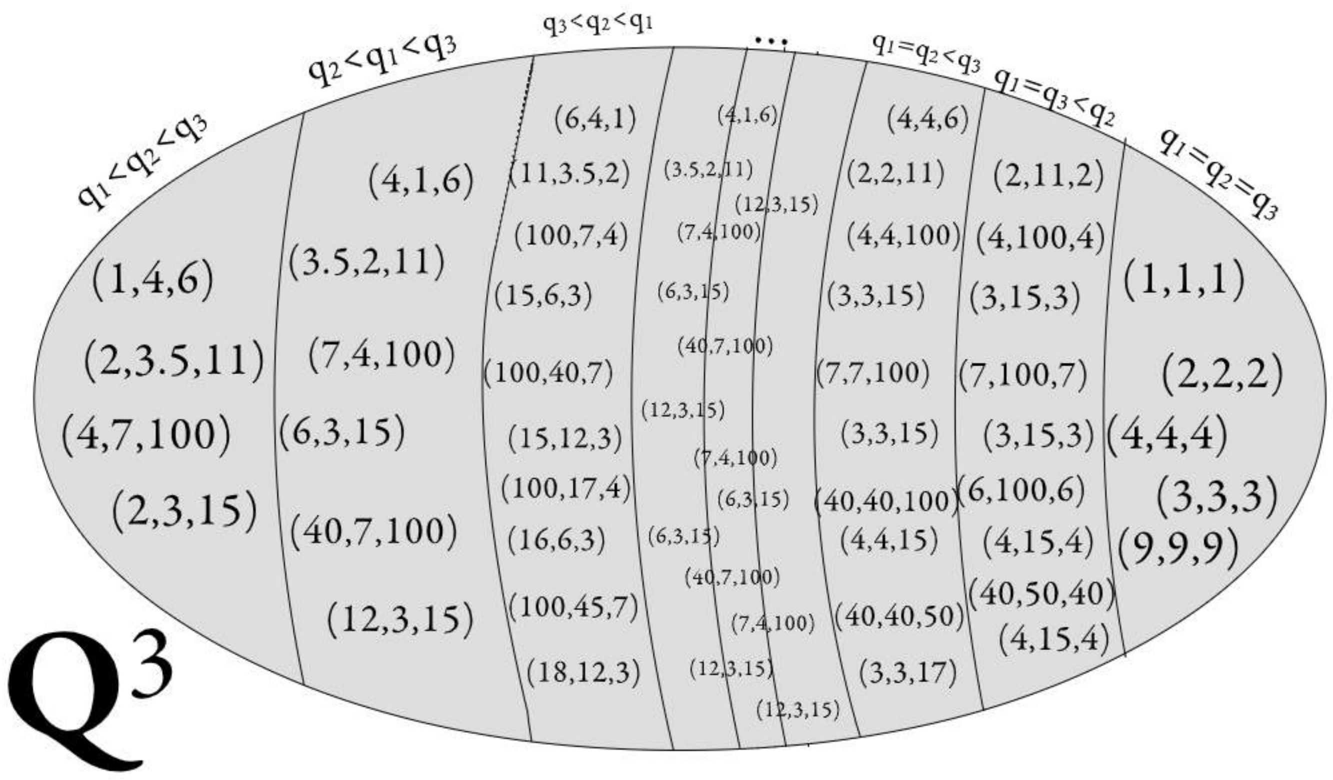
Examples: (**N**,=),(**Q**,≤), Rado graph

Theorem [Ryll-Nardzewski, Engeler, Svenonius]

a structure Atoms is *ω–categorical* if and only if for all $n \in \mathbf{N}$

Atoms^n *has finitely many orbits under the action of Aut(*Atoms*)*

Atoms = (**Q**,≤)



orithm pseudocode

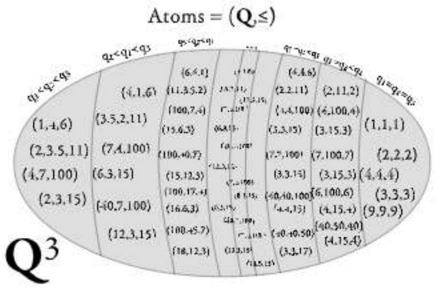
```
function reachability(V,E,s,t)

$R_0$ := {s};
n=0;
repeat
    $R_{n+1}$ := $R_n$ ∪ {w | (v,w)∈E, v∈$R_n$, w∈V};
    n := n+1;
until ($R_n$=$R_{n-1}$);
return (t∈$R_n$)
```

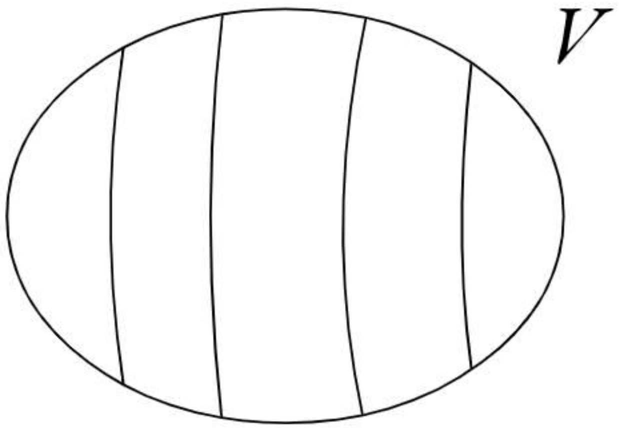
minates for *ω-categorical* Atoms: $R_0 \subseteq R_1 \subseteq R_2 \subseteq R_3 \subseteq \ldots \subseteq V$ are invariant subsets of V, which has finitely many orbits.

Examples: (**N**,=),(**Q**,≤), Rado graph

Theorem [Ryll-Nardzewski, Engeler, Svenonius]
a structure Atoms is *ω–categorical* if and only if for all $n \in \mathbf{N}$

Atoms^n *has finitely many orbits under the action of Aut(*Atoms*)*





orithm pseudocode

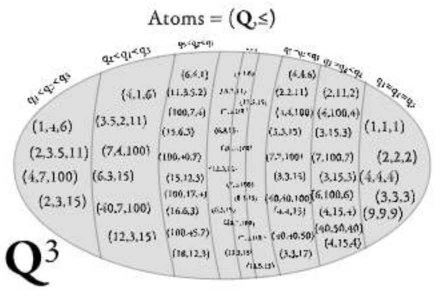
```
function reachability(V,E,s,t)

$R_0$ := {s};
n=0;
repeat
   $R_{n+1}$ := $R_n$ ∪ {w | (v,w)∈E, v∈$R_n$, w∈V};
   n := n+1;
until ($R_n$=$R_{n-1}$);
return (t∈$R_n$)
```

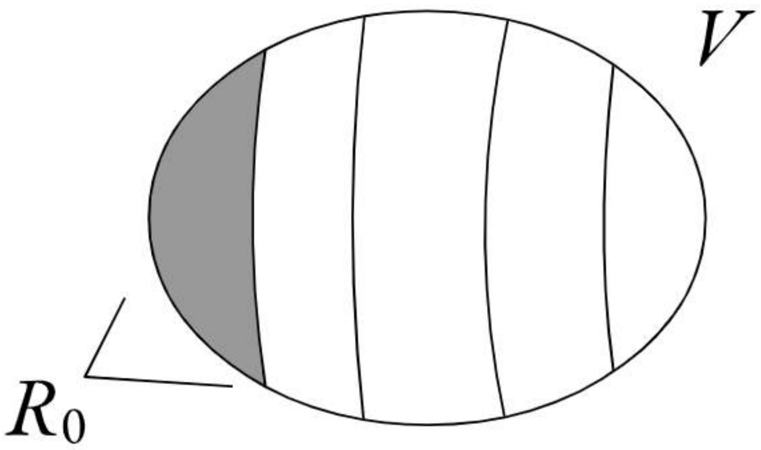
minates for *ω-categorical* Atoms: $R_0 \subseteq R_1 \subseteq R_2 \subseteq R_3 \subseteq \ldots \subseteq V$ are invariant subsets of V, which has finitely many orbits.

Examples: $(\mathbf{N},=)$,$(\mathbf{Q},\leq)$, Rado graph

Theorem [Ryll-Nardzewski, Engeler, Svenonius]
a structure Atoms is *ω–categorical* if and only if for all $n \in \mathbf{N}$

Atoms^n *has finitely many orbits under the action of Aut(*Atoms*)*





orithm pseudocode

```
function reachability(V,E,s,t)

$R_0$ := {s};
n=0;
repeat
    $R_{n+1}$ := $R_n$ ∪ {w | (v,w)∈E, v∈$R_n$, w∈V};
    n := n+1;
until ($R_n$=$R_{n-1}$);
return (t∈$R_n$)
```

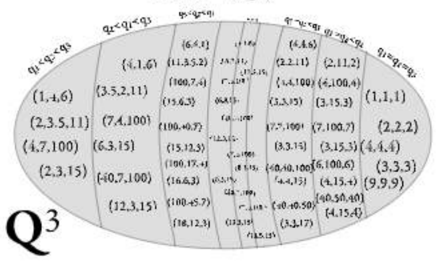
minates for *ω-categorical* Atoms: $R_0 \subseteq R_1 \subseteq R_2 \subseteq R_3 \subseteq \ldots \subseteq V$ are invariant subsets of V, which has finitely many orbits.

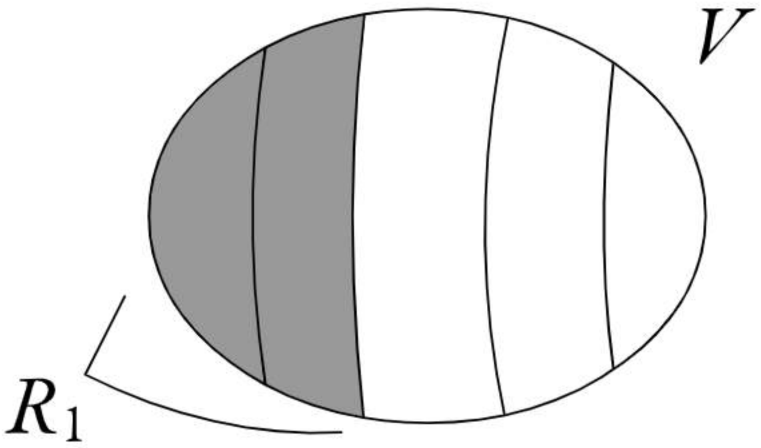
Examples: (**N**,=),(**Q**,≤), Rado graph

Theorem [Ryll-Nardzewski, Engeler, Svenonius]
a structure Atoms is *ω–categorical* if and only if for all $n \in \mathbf{N}$

Atoms^n *has finitely many orbits under the action of Aut(*Atoms*)*

Atoms = (**Q**,≤)





orithm pseudocode

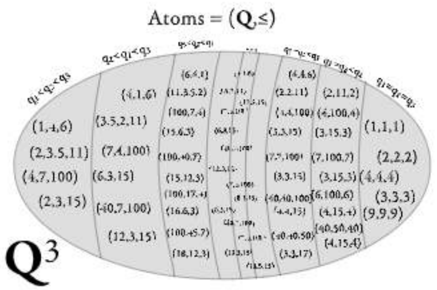
```
function reachability(V,E,s,t)

$R_0$ := {s};
n=0;
repeat
    $R_{n+1}$ := $R_n$ ∪ {w | (v,w)∈E, v∈$R_n$, w∈V};
    n := n+1;
until ($R_n$=$R_{n-1}$);
return (t∈$R_n$)
```

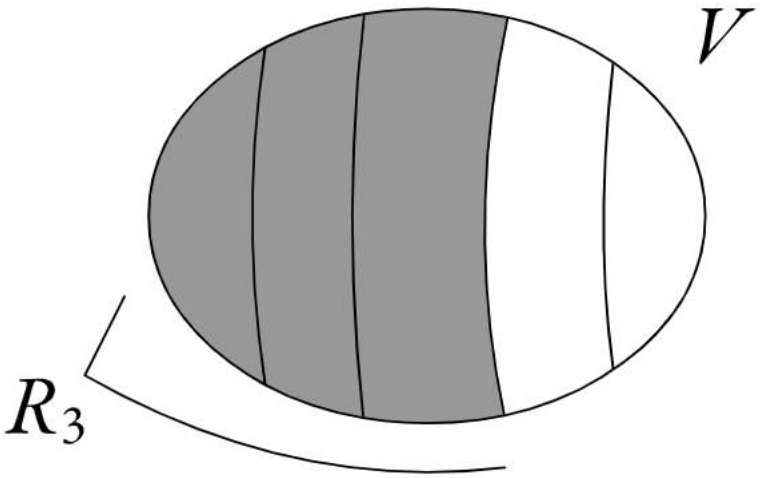
minates for *ω-categorical* Atoms: $R_0 \subseteq R_1 \subseteq R_2 \subseteq R_3 \subseteq \ldots \subseteq V$ are invariant subsets of V, which has finitely many orbits.

Examples: (**N**,=),(**Q**,≤), Rado graph

Theorem [Ryll-Nardzewski, Engeler, Svenonius]
a structure Atoms is *ω–categorical* if and only if for all $n \in \mathbf{N}$

Atoms^n *has finitely many orbits under the action of Aut(*Atoms*)*





orithm pseudocode

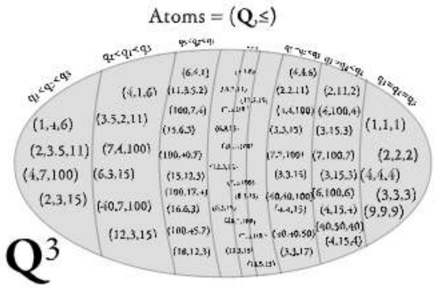
```
function reachability(V,E,s,t)

$R_0$ := {s};
n=0;
repeat
    $R_{n+1}$ := $R_n$ ∪ {w | (v,w)∈E, v∈$R_n$, w∈V};
    n := n+1;
until ($R_n$=$R_{n-1}$);
return (t∈$R_n$)
```

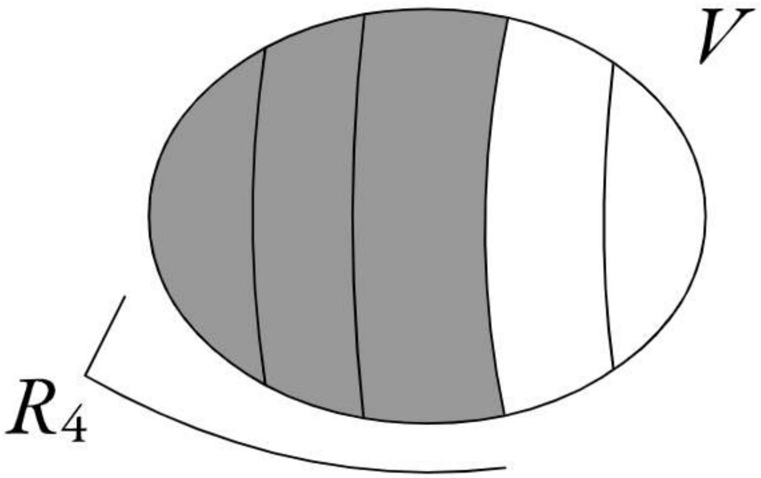
minates for *ω-categorical* Atoms: $R_0 \subseteq R_1 \subseteq R_2 \subseteq R_3 \subseteq \ldots \subseteq V$ are invariant subsets of V, which has finitely many orbits.

Examples: $(\mathbf{N},=)$, $(\mathbf{Q},\leq)$, Rado graph

Theorem [Ryll-Nardzewski, Engeler, Svenonius]
a structure Atoms is *ω–categorical* if and only if for all $n \in \mathbf{N}$

Atoms^n *has finitely many orbits under the action of Aut(*Atoms*)*





Algorithm pseudocode

```
function reachability(V,E,s,t)

$R_0$ := {s};
n=0;
repeat
    $R_{n+1}$ := $R_n$ ∪ {w | (v,w)∈E, v∈$R_n$, w∈V};
    n := n+1;
until ($R_n$=$R_{n-1}$);
return (t∈$R_n$)
```

Terminates for *ω-categorical* Atoms: $R_0 \subseteq R_1 \subseteq R_2 \subseteq R_3 \subseteq \ldots \subseteq V$ are inva

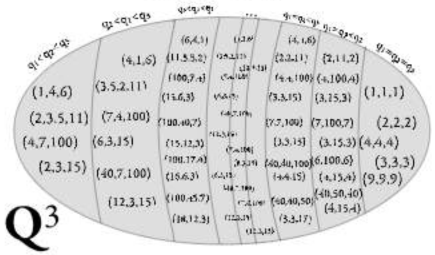
subsets of V, which has finitely many

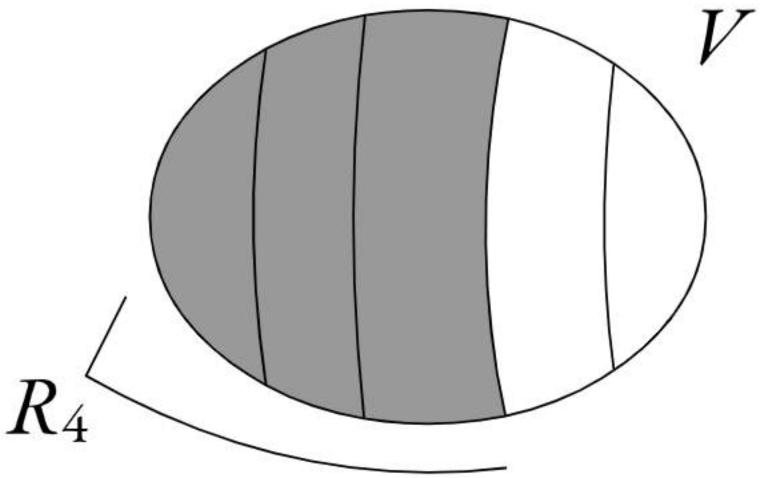
Examples: ($\mathbf{N}$,=),($\mathbf{Q}$,≤), Rado graph

Theorem [Ryll-Nardzewski, Engeler, Sveronius]
a structure Atoms is *ω–categorical* if and only if for all $n \in \mathbf{N}$

Atoms^n *has finitely many orbits under the action of Aut(*Atoms*)*

Atoms = ($\mathbf{Q}$,≤)





Reachability

Input: a hereditarily definable graph $G=(V,E)$, vertices $s,t \in V$
Decide: is t reachable from s?

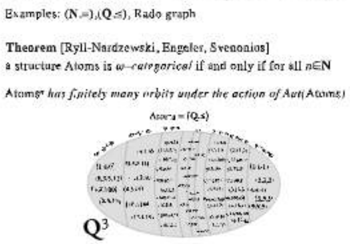
decidable when atoms are $(\mathbf{N},=)$ or $(\mathbf{Q},\leq)$

undecidable for Atoms = $(\mathbf{N},+1)$ (~Minsky machines)

Algorithm pseudocode

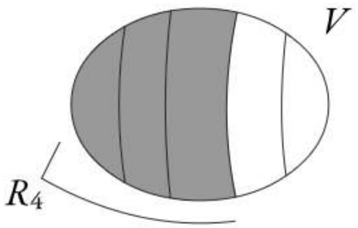
```
function reachability(V,E,s,t)

$R_0$ := {s};
n=0;
repeat
   $R_{n+1}$ := $R_n$ ∪ {w | (v,w)∈E, v∈$R_n$, w∈V};
   n := n+1;
until ($R_n$=$R_{n-1}$);
return (t∈$R_n$)
```

Terminates for *ω-categorical* Atoms: $R_0 \subseteq R_1 \subseteq R_2 \subseteq R_3 \subseteq \ldots \subseteq V$ are invariant subsets of V, which has finitely many orbits.



Examples: $(\mathbf{N},=)$, $(\mathbf{Q},\leq)$, Rado graph

Theorem [Ryll-Nardzewski, Engeler, Svenonios]
a structure Atoms is *ω–categorical* if and only if for all $n \in \mathbf{N}$
Atoms^n *has finitely many orbits under the action of* Aut(Atoms)



Reachability

Input: a hereditarily definable graph $G=(V,E)$, vertices $s,t \in V$

Decide: is t reachable from s?

decidable when atoms are $(\mathbf{N},=)$ or $(\mathbf{Q},\leq)$

undecidable for Atoms = $(\mathbf{N},+1)$ (~Minsky machines)

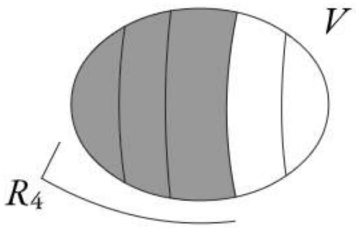
Algorithm pseudocode

```
function reachability(V,E,s,t)

$R_0$ := {s};
n=0;
repeat
   $R_{n+1}$ := $R_n \cup \{w \mid (v,w) \in E,\ v \in R_n,\ w \in V\}$;
   n := n+1;
until ($R_n = R_{n-1}$);
return ($t \in R_n$)
```

Terminates for *ω-categorical* Atoms: $R_0 \subseteq R_1 \subseteq R_2 \subseteq R_3 \subseteq \ldots \subseteq V$ are invariant subsets of V, which has finitely many orbits.

Examples: $(\mathbf{N},=)$, $(\mathbf{Q},\leq)$, Rado graph

Theorem [Ryll-Nardzewski, Engeler, Svenonius]
a structure Atoms is *ω–categorical* if and only if for all $n \in \mathbf{N}$
Atoms^n has finitely many orbits under the action of Aut(Atoms)



Computational Problems

definable sets can be presented as input to algorithms

- Graph reachability

Reachability

- Deterministic automata minimisation
- Context-free grammar emptiness
- Tree/pushdown automata emptiness
- Graph planarity
- Graph isomorphism
- Graph 3–colorability
- Solvability of systems of equations over finite field
- Satisfiability of sets of clauses
- Constraint Satisfaction Problems over finite template
- Homomorphism problem

Computational Problems

definable sets can be presented as input to algorithms

✓ Graph reachability 

Reachability

– Deterministic automata minimisation

– Context-free grammar emptiness

– Tree/pushdown automata emptiness

– Graph planarity

– Graph isomorphism

– Graph 3–colorability

– Solvability of systems of equations over finite field

– Satisfiability of sets of clauses

– Constraint Satisfaction Problems over finite template

– Homomorphism problem

Computational Problems

definable sets can be presented as input to algorithms

- ✓ Graph reachability

Reachability

- – Deterministic automata minimisation
- – Context-free grammar emptiness
- – Tree/pushdown automata emptiness
- – Graph planarity
- – Graph isomorphism
- – Graph 3–colorability
- – Solvability of systems of equations over finite field
- – Satisfiability of sets of clauses
- – Constraint Satisfaction Problems over finite template
- – Homomorphism problem

Computational Problems

definable sets can be presented as input to algorithms

(ω-categoricity)

- ✓ Graph reachability

Reachability

- ✓ Deterministic automata minimisation
- ✓ Context-free grammar emptiness
- ✓ Tree/pushdown automata emptiness
- Graph planarity
- Graph isomorphism
- Graph 3–colorability
- Solvability of systems of equations over finite field
- Satisfiability of sets of clauses
- Constraint Satisfaction Problems over finite template
- Homomorphism problem

Computational Problems

definable sets can be presented as input to algorithms

(ω-categoricity)

- ✓ Graph reachability


Reachability

- ✓ Deterministic automata minimisation
- ✓ Context-free grammar emptiness
- ✓ Tree/pushdown automata emptiness
- Graph planarity
- Graph isomorphism
- Graph 3–colorability
- Solvability of systems of equations over finite field
- Satisfiability of sets of clauses
- Constraint Satisfaction Problems over finite template
- Homomorphism problem

Input: A graph G definable over Atoms = $(\mathbf{N},=)$

Decide: Is G 3-colorable?

Is this problem decidable?

Example

Is the following graph 3-colorable?

$$V = \{\ (a,b) : a,b \in \text{Atoms},\ a \neq b\ \}$$

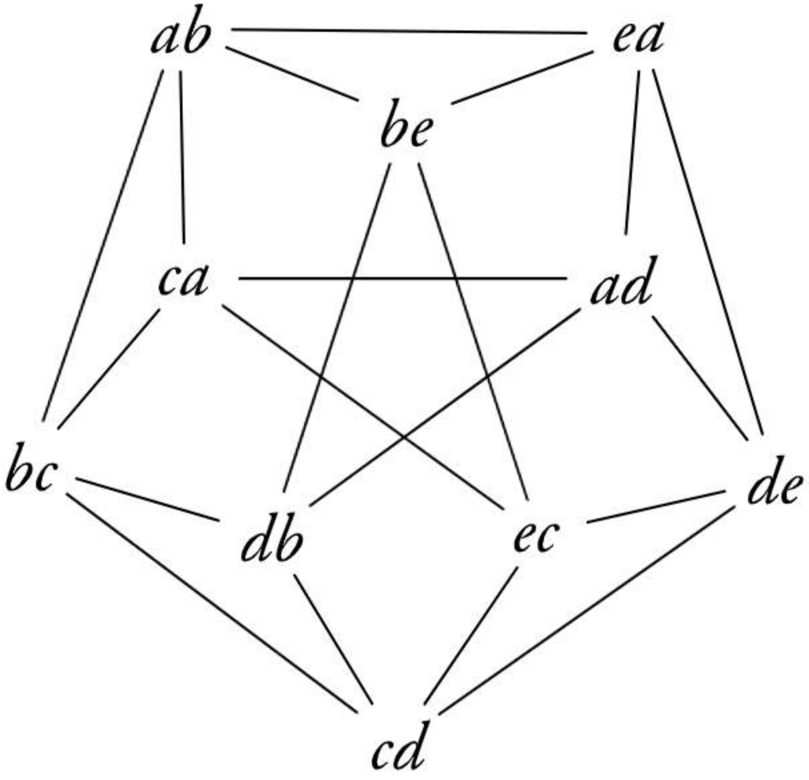
$$E = \{\ \{(a,b),(b,c)\} : a,b,c \in \text{Atoms},\ a \neq b \wedge b \neq c \wedge a \neq c\ \}$$

Example

Is the following graph 3-colorable?

$V = \{ (a,b) : a,b \in \text{Atoms},\ a \neq b \}$

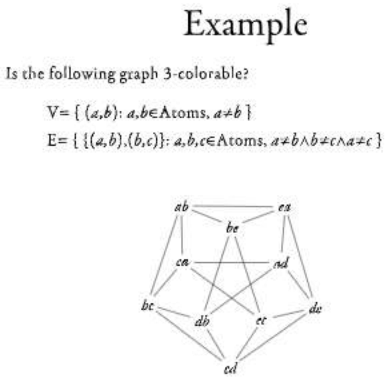
$E = \{ \{(a,b),(b,c)\} : a,b,c \in \text{Atoms},\ a \neq b \wedge b \neq c \wedge a \neq c \}$



Input: A graph *G* definable over Atoms = ($\mathbf{N}$,=)

Decide: Is *G* 3-colorable?

Is this problem decidable?



Example

Is the following graph 3-colorable?

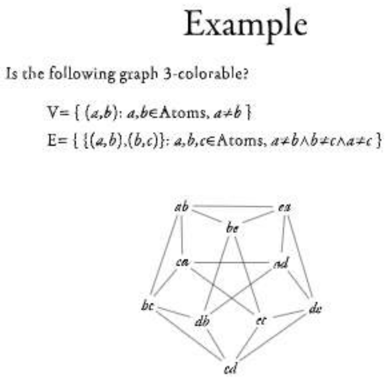
V= { (a,b): $a,b \in$ Atoms, $a \neq b$ }

E= { $\{(a,b),(b,c)\}$: $a,b,c \in$ Atoms, $a \neq b \wedge b \neq c \wedge a \neq c$ }

Input: A graph G definable over Atoms = $(\mathbf{N},=)$

Decide: Is G 3-colorable?

Is this problem decidable?



Example

Is the following graph 3-colorable?

V= $\{ (a,b) : a,b \in \text{Atoms}, a \neq b \}$

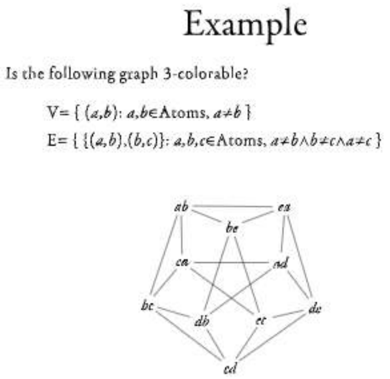
E= $\{ \{(a,b),(b,c)\} : a,b,c \in \text{Atoms}, a \neq b \wedge b \neq c \wedge a \neq c \}$

Proof of decidability uses Ramsey theory [Bodirsky, Pinsker, Tsankov'13]
[Klin, Kopczyński, Ochremiak, T.15]

Input: A graph G definable over Atoms = $(\mathbf{N},=)$

Decide: Is G 3-colorable?

Is this problem decidable?



Example

Is the following graph 3-colorable?

$V = \{ (a,b) : a,b \in \text{Atoms}, a \neq b \}$

$E = \{ \{(a,b),(b,c)\} : a,b,c \in \text{Atoms}, a \neq b \wedge b \neq c \wedge a \neq c \}$

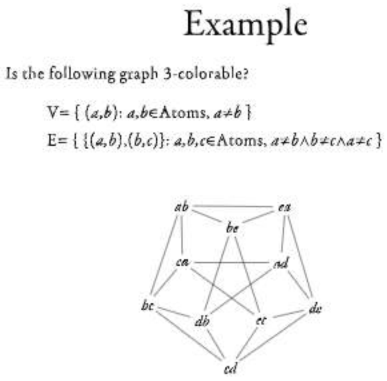
Proof of decidability uses Ramsey theory [Bodirsky, Pinsker, Tsankov'13]
[Klin, Kopczyński, Ochremiak, T.15]

Proof. Instead of $(\mathbf{N},=)$ use $(\mathbf{Q},\leq)$ as Atoms.

Input: A graph G definable over Atoms = $(\mathbf{N},=)$

Decide: Is G 3-colorable?

Is this problem decidable?



Example

Is the following graph 3-colorable?

$V = \{ (a,b) : a,b \in \text{Atoms}, a \neq b \}$

$E = \{ \{(a,b),(b,c)\} : a,b,c \in \text{Atoms}, a \neq b \wedge b \neq c \wedge a \neq c \}$

Proof of decidability uses Ramsey theory [Bodirsky, Pinsker, Tsankov'13] [Klin, Kopczyński, Ochremiak, T.15]

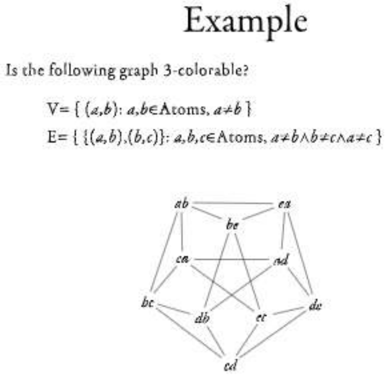
Proof. Instead of $(\mathbf{N},=)$ use $(\mathbf{Q},\leq)$ as Atoms.

Theorem (Pestov). Any continuous action of $\text{Aut}(\mathbf{Q},\leq)$ on a nonempty compact space has a fixpoint.

Input: A graph G definable over Atoms = $(\mathbf{N},=)$

Decide: Is G 3-colorable?

Is this problem decidable?


Example

Is the following graph 3-colorable?

$V = \{ (a,b) : a,b \in \text{Atoms}, a \neq b \}$

$E = \{ \{(a,b),(b,c)\} : a,b,c \in \text{Atoms}, a \neq b \wedge b \neq c \wedge a \neq c \}$

Proof of decidability uses Ramsey theory [Bodirsky, Pinsker, Tsankov'13]
[Klin, Kopczyński, Ochremiak, T.15]

Proof. Instead of $(\mathbf{N},=)$ use $(\mathbf{Q},\leq)$ as Atoms.

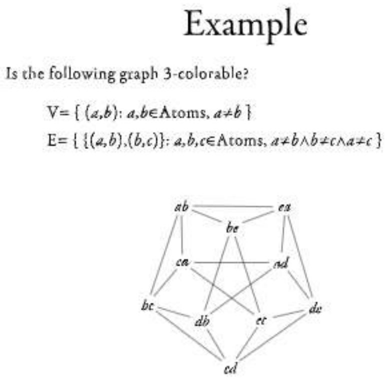
Theorem (Pestov). Any continuous action of $\mathrm{Aut}(\mathbf{Q},\leq)$ on a nonempty compact space has a fixpoint.

Let $\quad Col = \{\text{proper colorings of } G\} \subseteq \{1,2,3\}^{V(G)}$

Input: A graph G definable over Atoms = $(\mathbf{N},=)$

Decide: Is G 3-colorable?

Is this problem decidable?



Example

Is the following graph 3-colorable?

$V = \{ (a,b) : a,b \in \text{Atoms}, a \neq b \}$

$E = \{ \{(a,b),(b,c)\} : a,b,c \in \text{Atoms}, a \neq b \wedge b \neq c \wedge a \neq c \}$

Proof of decidability uses Ramsey theory [Bodirsky, Pinsker, Tsankov'13] [Klin, Kopczyński, Ochremiak, T.15]

Proof. Instead of $(\mathbf{N},=)$ use $(\mathbf{Q},\leq)$ as Atoms.

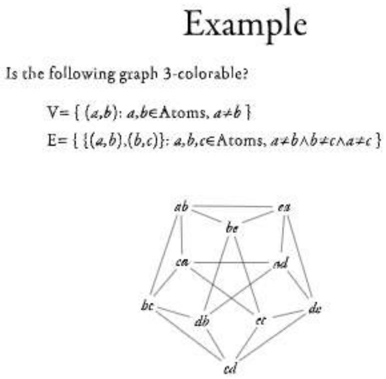
Theorem (Pestov). Any continuous action of $\text{Aut}(\mathbf{Q},\leq)$ on a nonempty compact space has a fixpoint.

Let $$Col = \{\text{proper colorings of } G\} \subseteq \underset{\text{compact}}{\{1,2,3\}^{V(G)}}$$

Input: A graph G definable over Atoms = $(\mathbf{N},=)$

Decide: Is G 3-colorable?

Is this problem decidable?



Example

Is the following graph 3-colorable?

$V = \{ (a,b) : a,b \in \text{Atoms}, a \neq b \}$

$E = \{ \{(a,b),(b,c)\} : a,b,c \in \text{Atoms}, a \neq b \wedge b \neq c \wedge a \neq c \}$

Proof of decidability uses Ramsey theory [Bodirsky, Pinsker, Tsankov'13]
[Klin, Kopczyński, Ochremiak, T.15]

Proof. Instead of $(\mathbf{N},=)$ use $(\mathbf{Q},\leq)$ as Atoms.

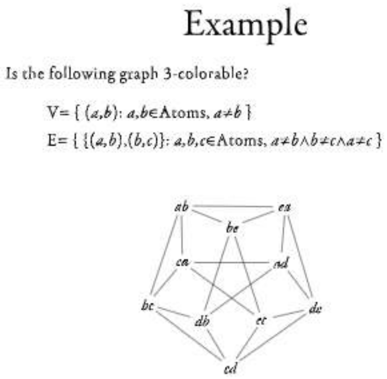
Theorem (Pestov). Any continuous action of $\mathrm{Aut}(\mathbf{Q},\leq)$ on a nonempty compact space has a fixpoint.

$$\text{Let} \qquad Col = \{\text{proper colorings of } G\} \underset{\text{closed}}{\subseteq} \underset{\text{compact}}{\{1,2,3\}^{V(G)}}$$

Input: A graph G definable over Atoms = $(\mathbf{N},=)$

Decide: Is G 3-colorable?

Is this problem decidable?



Example

Is the following graph 3-colorable?

$V = \{ (a,b) : a,b \in \text{Atoms}, a \neq b \}$

$E = \{ \{(a,b),(b,c)\} : a,b,c \in \text{Atoms}, a \neq b \wedge b \neq c \wedge a \neq c \}$

Proof of decidability uses Ramsey theory [Bodirsky, Pinsker, Tsankov'13]
[Klin, Kopczyński, Ochremiak, T.15]

Proof. Instead of $(\mathbf{N},=)$ use $(\mathbf{Q},\leq)$ as Atoms.

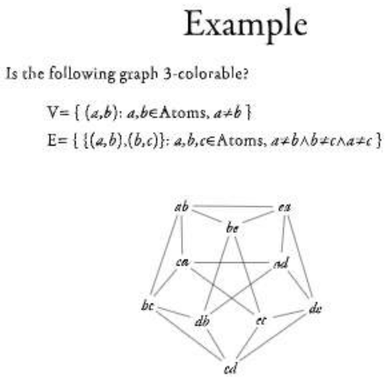
Theorem (Pestov). Any continuous action of $\text{Aut}(\mathbf{Q},\leq)$ on a nonempty compact space has a fixpoint.

$$\text{Let} \qquad \mathit{Col} = \{\text{proper colorings of } G\} \underset{\text{closed}}{\subseteq} \underset{\substack{\text{compact}\\ \text{Aut}(\mathbf{Q},\leq)\text{ action}}}{\{1,2,3\}^{V(G)}}$$

Input: A graph G definable over Atoms = $(\mathbf{N},=)$

Decide: Is G 3-colorable?

Is this problem decidable?



Example

Is the following graph 3-colorable?

$V = \{ (a,b) : a,b \in \text{Atoms}, a \neq b \}$

$E = \{ \{(a,b),(b,c)\} : a,b,c \in \text{Atoms}, a \neq b \wedge b \neq c \wedge a \neq c \}$

Proof of decidability uses Ramsey theory [Bodirsky, Pinsker, Tsankov'13]
[Klin, Kopczyński, Ochremiak, T.15]

Proof. Instead of $(\mathbf{N},=)$ use $(\mathbf{Q},\leq)$ as Atoms.

Theorem (Pestov). Any continuous action of $\mathrm{Aut}(\mathbf{Q},\leq)$ on a nonempty compact space has a fixpoint.

Let $\quad Col = \{\text{proper colorings of } G\} \subseteq \{1,2,3\}^{V(G)}$

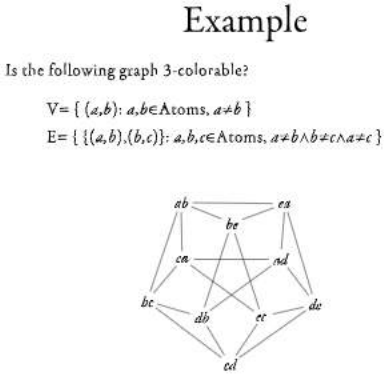
closed
invariant

compact
$\mathrm{Aut}(\mathbf{Q},\leq)$ action

Input: A graph G definable over Atoms = ($\mathbf{N}$,=)

Decide: Is G 3-colorable?

Is this problem decidable?



Example

Is the following graph 3-colorable?

V= { (a,b): $a,b\in$Atoms, $a\neq b$ }

E= { $\{(a,b),(b,c)\}$: $a,b,c\in$Atoms, $a\neq b\wedge b\neq c\wedge a\neq c$ }

Proof of decidability uses Ramsey theory [Bodirsky, Pinsker, Tsankov'13]
[Klin, Kopczyński, Ochremiak, T.15]

Proof. Instead of ($\mathbf{N}$,=) use ($\mathbf{Q}$,≤) as Atoms.

Theorem (Pestov). Any continuous action of Aut($\mathbf{Q}$,≤) on a nonempty compact space has a fixpoint.

Let Col = {proper colorings of G} $\subseteq \{1,2,3\}^{V(G)}$

closed invariant

compact Aut($\mathbf{Q}$,≤) action

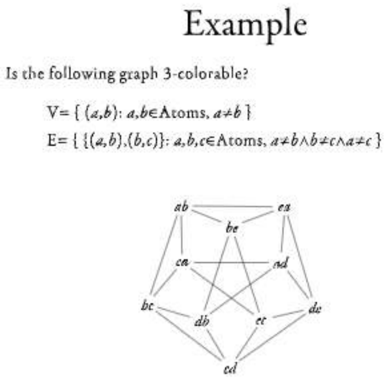
By Pestov:

if there is a 3-coloring of G then there is an invariant one.

Input: A graph G definable over Atoms = $(\mathbf{N},=)$

Decide: Is G 3-colorable?

Is this problem decidable?



Example

Is the following graph 3-colorable?

$V = \{ (a,b) : a,b \in \text{Atoms}, a \neq b \}$

$E = \{ \{(a,b),(b,c)\} : a,b,c \in \text{Atoms}, a \neq b \wedge b \neq c \wedge a \neq c \}$

Proof of decidability uses Ramsey theory [Bodirsky, Pinsker, Tsankov'13]
[Klin, Kopczyński, Ochremiak, T.15]

Proof. Instead of $(\mathbf{N},=)$ use $(\mathbf{Q},\leq)$ as Atoms.

Theorem (Pestov). Any continuous action of $\mathrm{Aut}(\mathbf{Q},\leq)$ on a nonempty compact space has a fixpoint.

$$\text{Let} \qquad Col = \{\text{proper colorings of } G\} \underset{\substack{\text{closed} \\ \text{invariant}}}{\subseteq} \underbrace{\{1,2,3\}^{V(G)}}_{\substack{\text{compact} \\ \mathrm{Aut}(\mathbf{Q},\leq) \text{ action}}}$$

By Pestov:

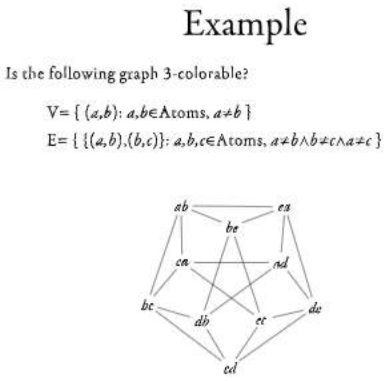
if there is a 3-coloring of G then there is an invariant one.

Check 3-colorolability of the finite graph G/Aut$(\mathbf{Q},\leq)$ of orbits of G. ■

Input: A graph G definable over Atoms = $(\mathbf{N},=)$

Decide: Is G 3-colorable?

Is this problem decidable?



Example

Is the following graph 3-colorable?

V= { (a,b): $a,b\in$Atoms, $a\neq b$ }

E= { $\{(a,b),(b,c)\}$: $a,b,c\in$Atoms, $a\neq b\wedge b\neq c\wedge a\neq c$ }

Proof of decidability uses Ramsey theory [Bodirsky, Pinsker, Tsankov'13] [Klin, Kopczyński, Ochremiak, T.15]

Proof. Instead of $(\mathbf{N},=)$ use $(\mathbf{Q},\leq)$ as Atoms.

Theorem (Pestov). Any continuous action of $\mathrm{Aut}(\mathbf{Q},\leq)$ on a nonempty compact space has a fixpoint.

$$\text{Let}\qquad Col = \{\text{proper colorings of } G\} \underset{\substack{\text{closed}\\ \text{invariant}}}{\subseteq} \underbrace{\{1,2,3\}^{V(G)}}_{\substack{\text{compact}\\ \mathrm{Aut}(\mathbf{Q},\leq)\text{ action}}}$$

By Pestov:

if there is a 3-coloring of G then there is an invariant one.

Check 3-colorolability of the finite graph $G/\mathrm{Aut}(\mathbf{Q},\leq)$ of orbits of G. ■

Also works when Atoms are a Ramsey structure (KPT theorem).

Computational Problems

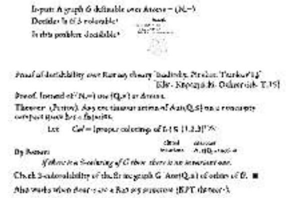
definable sets can be presented as input to algorithms

(ω-categoricity)

- ✓ Graph reachability

Reachability

- ✓ Deterministic automata minimisation
- ✓ Context-free grammar emptiness
- ✓ Tree/pushdown automata emptiness
- Graph planarity
- Graph isomorphism
- Graph 3–colorability



- Solvability of systems of equations over finite field
- Satisfiability of sets of clauses
- Constraint Satisfaction Problems over finite template
- Homomorphism problem

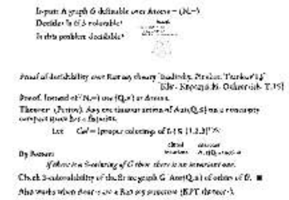
Computational Problems

definable sets can be presented as input to algorithms

(ω-categoricity)

- ✓ Graph reachability

- ✓ Deterministic automata minimisation
- ✓ Context-free grammar emptiness
- ✓ Tree/pushdown automata emptiness
- Graph planarity
- Graph isomorphism
- Graph 3–colorability



- Solvability of systems of equations over finite field
- Satisfiability of sets of clauses
- Constraint Satisfaction Problems over finite template
- Homomorphism problem

Computational Problems

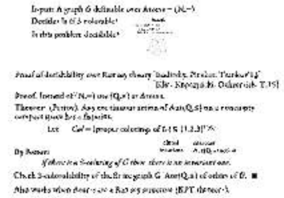
definable sets can be presented as input to algorithms

(ω-categoricity)

- ✓ Graph reachability

Reachability

- ✓ Deterministic automata minimisation
- ✓ Context-free grammar emptiness
- ✓ Tree/pushdown automata emptiness
- – Graph planarity
- – Graph isomorphism
- ✓ Graph 3–colorability



(Ramsey)

- ✓ Solvability of systems of equations over finite field
- ✓ Satisfiability of sets of clauses
- ✓ Constraint Satisfaction Problems over finite template
- – Homomorphism problem

Computational Problems

definable sets can be presented as input to algorithms

(ω-categoricity)

✓ Graph reachability


Reachability

✓ Deterministic automata minimisation

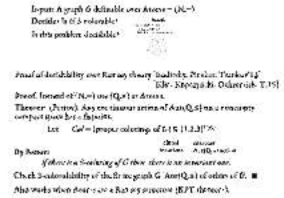
✓ Context-free grammar emptiness

✓ Tree/pushdown automata emptiness

– Graph planarity

– Graph isomorphism

✓ Graph 3–colorability



(Ramsey)

✓ Solvability of systems of equations over finite field

✓ Satisfiability of sets of clauses

✓ Constraint Satisfaction Problems over finite template

✓ Homomorphism problem [Bodirsky, Pinsker, Tsankov'13]

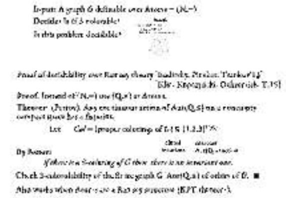
Computational Problems

definable sets can be presented as input to algorithms

(ω-categoricity)

- ✓ Graph reachability

- ✓ Deterministic automata minimisation
- ✓ Context-free grammar emptiness
- ✓ Tree/pushdown automata emptiness
- ✓ Graph planarity
- Graph isomorphism
- ✓ Graph 3–colorability



(Ramsey)

- ✓ Solvability of systems of equations over finite field
- ✓ Satisfiability of sets of clauses
- ✓ Constraint Satisfaction Problems over finite template
- ✓ Homomorphism problem [Bodirsky, Pinsker, Tsankov'13]

Computational Problems

definable sets can be presented as input to algorithms

(ω-categoricity)

✓ – Graph reachability

Reachability

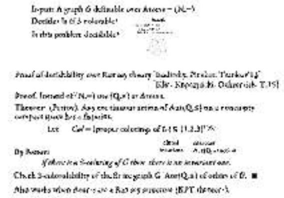
✓ – Deterministic automata minimisation

✓ – Context-free grammar emptiness

✓ – Tree/pushdown automata emptiness

✓ – Graph planarity

? – Graph isomorphism



✓ – Graph 3–colorability

✓ – Solvability of systems of equations over finite field

(Ramsey)

✓ – Satisfiability of sets of clauses

✓ – Constraint Satisfaction Problems over finite template

✓ – Homomorphism problem [Bodirsky, Pinsker, Tsankov'13]

Summary

1. Definable sets are a data structure for representing certain infinite sets.

Summary

1. Definable sets are a data structure for representing certain infinite sets.

2. Some classical algorithms can be lifted to definable structures using the *same code* as normally, e.g.,

   –automata reachability,
   –automata minimisation,
   –pushdown/tree automata emptiness,
   –automata–expression–grammar conversions,...

Summary

1. Definable sets are a data structure for representing certain infinite sets.

2. Some classical algorithms can be lifted to definable structures using the *same code* as normally, e.g.,

   –automata reachability,
   –automata minimisation,
   –pushdown/tree automata emptiness,
   –automata–expression–grammar conversions,...

3. Model theory helps prove termination/correctness.

Summary

1. Definable sets are a data structure for representing certain infinite sets.

2. Some classical algorithms can be lifted to definable structures using the *same code* as normally, e.g.,

   –automata reachability,
   –automata minimisation,
   –pushdown/tree automata emptiness,
   –automata–expression–grammar conversions,...

3. Model theory helps prove termination/correctness.

Summary

1. Definable sets are a data structure for representing certain infinite sets.

2. Some classical algorithms can be lifted to definable structures using the *same code* as normally, e.g.,

    –automata reachability,
    –automata minimisation,
    –pushdown/tree automata emptiness,
    –automata–expression–grammar conversions,...

3. Model theory helps prove termination/correctness.

Not in this talk:

- Non-ω-categorical Atoms, application to timed automata
- A programming language with loops over infinite sets

```
X = ∅;

for a in Reals
  for b in Reals
     for c in Reals
        for x in Reals
           if (a*x*x+b*x+c*x=0)
              X=X∪{(a,b,c)};
return X;
```

```
X = ∅;

for a in Reals
  for b in Reals
     for c in Reals
        for x in Reals
           if (a*x*x+b*x+c*x=0)
              X=X∪{(a,b,c)};
return X;
```

output: $\{(a,b,c): a,b,c \in Reals,\ b*b-4*a*c \geq 0\}$

```
X = ∅;

for a in Reals
   for b in Reals
      for c in Reals
         for x in Reals
            if (a*x*x+b*x+c*x=0)
               X=X∪{(a,b,c)};
return X;
```

output: $\{(a,b,c)\colon a,b,c \in Reals,\ b^*b\text{-}4^*a^*c \geq 0\}$

```
function reachable(V,E,s,t)

R = {s}
P = ∅

while (R≠P)
   P=R;
   for v in P
      for w in V
         if {v,w}∈E
            R=RU{w}

return (t∈R);
```

Summary

1. Definable sets are a data structure for representing certain infinite sets.

2. Some classical algorithms can be lifted to definable structures using the *same code* as normally, e.g.,

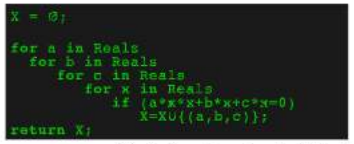
   –automata reachability,
   –automata minimisation,
   –pushdown/tree automata emptiness,
   –automata–expression–grammar conversions,...

3. Model theory helps prove termination/correctness.

Not in this talk:

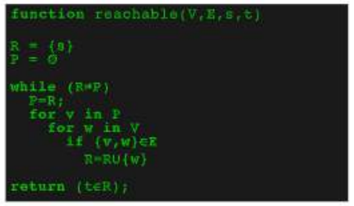
– Non-ω-categorical Atoms, application to timed automata

– A programming language with loops over infinite sets



```
X = ∅;

for a in Reals
  for b in Reals
     for c in Reals
        for x in Reals
           if (a*x*x+b*x+c*x=0)
              X=X∪{(a,b,c)};
return X;
```

output: $\{(a,b,c): a,b,c \in Reals,\ b*b-4*a*c \geq 0\}$



```
function reachable(V,E,s,t)

R = {s}
P = ∅

while (R≠P)
  P=R;
  for v in P
    for w in V
      if (v,w)∈E
        R=R∪{w}

return (t∈R);
```

Summary

1. Definable sets are a data structure for representing certain infinite sets.

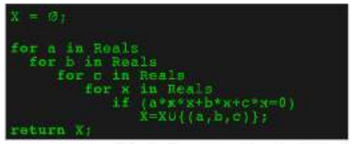
2. Some classical algorithms can be lifted to definable structures using the *same code* as normally, e.g.,

    - –automata reachability,
    - –automata minimisation,
    - –pushdown/tree automata emptiness,
    - –automata–expression–grammar conversions,...

3. Model theory helps prove termination/correctness.

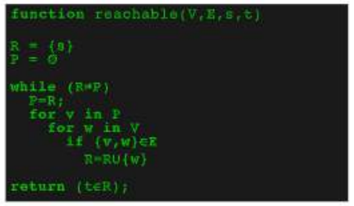
Not in this talk:

- – Non-ω-categorical Atoms, application to timed automata
- – A programming language with loops over infinite sets
- – notions of computational tractability ("PTime"), relation to CPT of Blass, Gurevich, Shelah.



```
X = ∅;

for a in Reals
  for b in Reals
     for c in Reals
        for x in Reals
           if (a*x*x+b*x+c*x=0)
              X=X∪{(a,b,c)};
return X;
```

output: $\{(a,b,c): a,b,c \in Reals, b*b-4*a*c \geq 0\}$



```
function reachable(V,E,s,t)

R = {s}
P = ∅

while (R≠P)
  P=R;
  for v in P
    for w in V
      if (v,w)∈E
        R=R∪{w}

return (t∈R);
```

Berkeley, February 8, 2019

Computation with Atoms

Szymon Toruńczyk, University of Warsaw

includes work with Bojańczyk, Klin, Kopczyński, Lasota, Ochremiak,...

Atoms

a fixed *underlying* logical structure

Examples:

$(\mathbb{N}, =)$ – *pure set* ← main example in this talk

$(\mathbb{Q}, \leq)$ – *dense order*

$(\mathbb{R}, +, \times, 0, 1)$ – *field of reals*

$(\mathbb{N}, +, \leq)$ – *Presburger arithmetic*

Hereditarily definable set

Examples

5 if $5 \in$ Atoms

$\{a : a \in \text{Atoms}\}$

$\{a : a \in \text{Atoms}, a \neq 7 \wedge a \neq 5\}$ if $5,7 \in$ Atoms

$\{\{a : a \in \text{Atoms}, a \neq b\} : b \in \text{Atoms}\}$

$\{\{b : b \in \text{Atoms}, a < b \wedge b < c\} : a, c \in \text{Atoms}, a < c\}$ if Atoms $= (\mathbb{Q}, \leq)$

$\{x, y\} \overset{\text{def}}{=} \{x\} \cup \{y\}$ $\quad (x, y) \overset{\text{def}}{=} \{x, \{x, y\}\}$

Syntax

hdef ::= variable | parameter from Atoms

| { hdef : variable,...,variable ∈ **Atoms**, first order formula }

in language of Atoms, with parameters

| hdef ∪ hdef

Hereditarily definable sets have finite descriptions

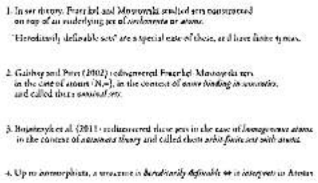
e.g. $\{\{a : a \in \text{Atoms}, a \neq b\} : b \in \text{Atoms}\}$

→ can be input and processed by algorithms

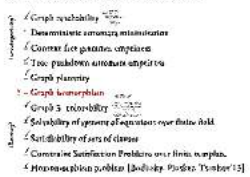
equality of sets is decidable ⇔ the theory of Atoms is decidable

Hereditarily definable *X*

History



Computational Problems



Summary

1. Definable sets are a data structure for representing certain infinite sets.
2. Some classical algorithms can be lifted to definable structures using the *same code* as normally, e.g.,
   - –automata reachability,
   - –automata minimisation,
   - –pushdown/tree automata emptiness,
   - –automata–expression–grammar conversions,...
3. Model theory helps prove termination/correctness.

Not in this talk:

- – Non-ω-categorical Atoms, application to timed automata
- – A programming language with loops over infinite sets
- – notions of computational tractability ("PTime"), relation to CPT of Blass, Gurevich, Shelah.

Berkeley, February 8, 2019

Computation with Atoms

Szymon Toruńczyk, University of Warsaw

includes work with Bojańczyk, Klin, Kopczyński, Lasota, Ochremiak,...

Atoms

a fixed *underlying* logical structure

Examples:

$(\mathbb{N}, =)$ – *pure set* ← main example in this talk

$(\mathbb{Q}, \leq)$ – *dense order*

$(\mathbb{R}, +, \times, 0, 1)$ – *field of reals*

$(\mathbb{N}, +, \leq)$ – *Presburger arithmetic*

Hereditarily definable set

Examples

5 if $5 \in$ Atoms

$\{a : a \in \text{Atoms}\}$

$\{a : a \in \text{Atoms}, a \neq 7 \wedge a \neq 5\}$ if $5,7 \in$ Atoms

$\{\{a : a \in \text{Atoms}, a \neq b\} : b \in \text{Atoms}\}$

$\{\{b : b \in \text{Atoms}, a < b \wedge b < c\} : a, c \in \text{Atoms}, a < c\}$ if Atoms $= (\mathbb{Q}, \leq)$

$\{x, y\} \overset{\text{def}}{=} \{x\} \cup \{y\}$ $\qquad (x, y) \overset{\text{def}}{=} \{x, \{x, y\}\}$

Syntax

hdef ::= variable | parameter from Atoms

| { hdef : variable,...,variable ∈ **Atoms**, first order formula } (in language of Atoms, with parameters)

| hdef ∪ hdef

Thank you for your attention!

Hereditarily definable sets have finite descriptions

e.g. $\{\{a : a \in \text{Atoms}, a \neq b\} : b \in \text{Atoms}\}$

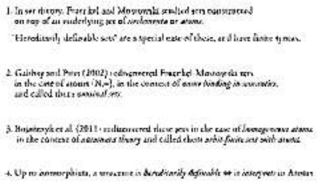
→ can be input and processed by algorithms

equality of sets is decidable ⇔ the theory of Atoms is decidable

Hereditarily definable *X*

graphs *automata* *Turing machines*

History



Computational Problems

Summary

1. Definable sets are a data structure for representing certain infinite sets.
2. Some classical algorithms can be lifted to definable structures using the *same code* as normally, e.g.,
   - –automata reachability,
   - –automata minimisation,
   - –pushdown/tree automata emptiness,
   - –automata–expression–grammar conversions,...
3. Model theory helps prove termination/correctness.

Not in this talk:

- – Non-ω-categorical Atoms, application to timed automata
- – A programming language with loops over infinite sets
- – notions of computational tractability ("PTime"), relation to CPT of Blass, Gurevich, Shelah.

Berkeley, February 8, 2019

Computation with Atoms

Szymon Toruńczyk, University of Warsaw

includes work with Bojańczyk, Klin, Kopczyński, Lasota, Ochremiak,...

Atoms

a fixed *underlying* logical structure

Examples:

- $(\mathbb{N}, =)$ – *pure set* ← main example in this talk
- $(\mathbb{Q}, \leq)$ – *dense order*
- $(\mathbb{R}, +, \times, 0, 1)$ – *field of reals*
- $(\mathbb{N}, +, \leq)$ – *Presburger arithmetic*

Hereditarily definable set

Examples

5 if $5 \in$ Atoms

$\{a : a \in \text{Atoms}\}$

$\{a : a \in \text{Atoms}, a \neq 7 \wedge a \neq 5\}$ if $5,7 \in$ Atoms

$\{\{a : a \in \text{Atoms}, a \neq b\} : b \in \text{Atoms}\}$

$\{\{b : b \in \text{Atoms}, a < b \wedge b < c\} : a, c \in \text{Atoms}, a < c\}$ if Atoms $= (\mathbb{Q}, \leq)$

$\{x,y\} \overset{\text{def}}{=} \{x\} \cup \{y\}$ $\quad (x,y) \overset{\text{def}}{=} \{x, \{x,y\}\}$

Syntax

hdef ::= variable | parameter from Atoms

| { hdef : variable,...,variable ∈ **Atoms**, first order formula }
in language of Atoms, with parameters

| hdef ∪ hdef

Thank you for your attention!

Hereditarily definable sets have finite descriptions

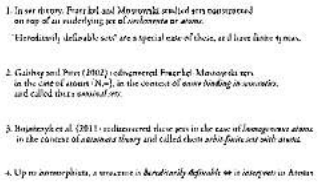
e.g. $\{\{a : a \in \text{Atoms}, a \neq b\} : b \in \text{Atoms}\}$

→ can be input and processed by algorithms

equality of sets is decidable ⇔ the theory of Atoms is decidable

Hereditarily definable *X*

History



Computational Problems

Summary

1. Definable sets are a data structure for representing certain infinite sets.
2. Some classical algorithms can be lifted to definable structures using the *same code* as normally, e.g.,
   - –automata reachability,
   - –automata minimisation,
   - –pushdown/tree automata emptiness,
   - –automata–expression–grammar conversions,...
3. Model theory helps prove termination/correctness.

Not in this talk:

- – Non-ω-categorical Atoms, application to timed automata
- – A programming language with loops over infinite sets
- – notions of computational tractability ("PTime"), relation to CPT of Blass, Gurevich, Shelah.

Berkeley, February 8, 2019

Computation with Atoms

Szymon Toruńczyk, University of Warsaw

includes work with Bojańczyk, Klin, Kopczyński, Lasota, Ochremiak,...

Thank you for your attention!

Atoms

a fixed *underlying* logical structure

Examples:

$(\mathbb{N}, =)$ – *pure set* ← main example in this talk

$(\mathbb{Q}, \leq)$ – *dense order*

$(\mathbb{R}, +, \times, 0, 1)$ – *field of reals*

$(\mathbb{N}, +, \leq)$ – *Presburger arithmetic*

Hereditarily definable set

Examples

5 if 5∈Atoms

$\{a : a \in \text{Atoms}\}$

$\{a : a \in \text{Atoms}, a \neq 7 \wedge a \neq 5\}$ if 5,7∈Atoms

$\{\{a : a \in \text{Atoms}, a \neq b\} : b \in \text{Atoms}\}$

$\{\{b : b \in \text{Atoms}, a < b \wedge b < c\} : a, c \in \text{Atoms}, a < c\}$ if Atoms = (Q,≤)

$\{x,y\} \overset{\text{def}}{=} \{x\} \cup \{y\}$ $(x,y) \overset{\text{def}}{=} \{x, \{x,y\}\}$

Syntax

hdef ::= variable | parameter from Atoms

| { hdef : variable,...,variable ∈ **Atoms**, first order formula } in language of Atoms, with parameters

| hdef ∪ hdef

Hereditarily definable sets have finite descriptions

e.g. $\{\{a : a \in \text{Atoms}, a \neq b\} : b \in \text{Atoms}\}$

→ can be input and processed by algorithms

equality of sets is decidable ⇔ the theory of Atoms is decidable

Hereditarily definable *X*

graphs *automata* *Turing machines*

(States, Alphabet, Initial, Accepting, δ)

(States, Alphabet, δ)

History

1. In set theory, Fraenkel and Mostowski studied sets constructed on top of an underlying set of *urelements* or *atoms*.

   "Hereditarily definable sets" are a special case of these, and have finite types.

2. Gabbay and Pitts (2002) rediscovered Fraenkel-Mostowski sets in the case of atoms (N,=), in the context of *name binding in semantics*, and called them *nominal sets*.

3. Bojańczyk et al. (2011) rediscovered these sets in the case of *homogeneous atoms* in the context of *automata theory* and called them *orbit-finite sets with atoms*.

4. Up to *isomorphism*, a structure is *hereditarily definable* ⇔ it *interprets* in Atoms

Computational Problems

definable sets can be presented as input to algorithms

- ✓ Graph reachability
- ✓ Deterministic automata minimisation
- ✓ Context free grammar emptiness
- ✓ Tree/pushdown automata emptiness
- ✓ Graph planarity
- ? – Graph isomorphism
- ✓ Graph 3-colorability
- ✓ Solvability of systems of equations over finite field
- ✓ Satisfiability of sets of clauses
- ✓ Constraint Satisfaction Problems over finite templates
- ✓ Homomorphism problem [Bodirsky, Pinsker, Toruńczyk'13]

Summary

1. Definable sets are a data structure for representing certain infinite sets.
2. Some classical algorithms can be lifted to definable structures using the *same code* as normally, e.g.,
   - –automata reachability,
   - –automata minimisation,
   - –pushdown/tree automata emptiness,
   - –automata–expression–grammar conversions,...
3. Model theory helps prove termination/correctness.

Not in this talk:

- – Non-ω-categorical Atoms, application to timed automata
- – A programming language with loops over infinite sets
- – notions of computational tractability ("PTime"), relation to CPT of Blass, Gurevich, Shelah.