

Satisfiability Checking

Decidability and Decision Procedures

–Some Historical Notes–

Prof. Dr. Erika Ábrahám

RWTH Aachen University
Informatik 2
LuFG Theory of Hybrid Systems

WS 19/20

FO theories and their decidability

Some first-order theories:

Logic	decidability	algorithm

FO theories and their decidability

Some first-order theories:

Logic	decidability	algorithm
Propositional logic		

FO theories and their decidability

Some first-order theories:

Logic	decidability	algorithm
Propositional logic	decidable	

FO theories and their decidability

Some first-order theories:

Logic	decidability	algorithm
Propositional logic	decidable	SAT-solving

FO theories and their decidability

Some first-order theories:

Logic	decidability	algorithm
Propositional logic	decidable	SAT-solving
Equational logic	decidable	SAT-encoding
Equational logic with uninterpr. functions	decidable	SAT-encoding

FO theories and their decidability

Some first-order theories:

Logic	decidability	algorithm
Propositional logic	decidable	SAT-solving
Equational logic	decidable	SAT-encoding
Equational logic with uninterpr. functions	decidable	SAT-encoding
Linear real algebra (\mathbb{R} with $+$)		

FO theories and their decidability

Some first-order theories:

Logic	decidability	algorithm
Propositional logic	decidable	SAT-solving
Equational logic	decidable	SAT-encoding
Equational logic with uninterpr. functions	decidable	SAT-encoding
Linear real algebra (\mathbb{R} with $+$)	decidable	

FO theories and their decidability

Some first-order theories:

Logic	decidability	algorithm
Propositional logic	decidable	SAT-solving
Equational logic	decidable	SAT-encoding
Equational logic with uninterpr. functions	decidable	SAT-encoding
Linear real algebra (\mathbb{R} with $+$)	decidable	Simplex

FO theories and their decidability

Some first-order theories:

Logic	decidability	algorithm
Propositional logic	decidable	SAT-solving
Equational logic	decidable	SAT-encoding
Equational logic with uninterpr. functions	decidable	SAT-encoding
Linear real algebra (\mathbb{R} with $+$)	decidable	Simplex
Real algebra (\mathbb{R} with $+$ and $*$)		

FO theories and their decidability

Some first-order theories:

Logic	decidability	algorithm
Propositional logic	decidable	SAT-solving
Equational logic	decidable	SAT-encoding
Equational logic with uninterpr. functions	decidable	SAT-encoding
Linear real algebra (\mathbb{R} with $+$)	decidable	Simplex
Real algebra (\mathbb{R} with $+$ and $*$)	decidable	

FO theories and their decidability

Some first-order theories:

Logic	decidability	algorithm
Propositional logic	decidable	SAT-solving
Equational logic	decidable	SAT-encoding
Equational logic with uninterpr. functions	decidable	SAT-encoding
Linear real algebra (\mathbb{R} with $+$)	decidable	Simplex
Real algebra (\mathbb{R} with $+$ and $*$)	decidable	CAD virtual substitution

FO theories and their decidability

Some first-order theories:

Logic	decidability	algorithm
Propositional logic	decidable	SAT-solving
Equational logic	decidable	SAT-encoding
Equational logic with uninterpr. functions	decidable	SAT-encoding
Linear real algebra (\mathbb{R} with $+$)	decidable	Simplex
Real algebra (\mathbb{R} with $+$ and $*$)	decidable	CAD virtual substitution
Presburger arithmetic (\mathbb{N} with $+$)		

FO theories and their decidability

Some first-order theories:

Logic	decidability	algorithm
Propositional logic	decidable	SAT-solving
Equational logic	decidable	SAT-encoding
Equational logic with uninterpr. functions	decidable	SAT-encoding
Linear real algebra (\mathbb{R} with $+$)	decidable	Simplex
Real algebra (\mathbb{R} with $+$ and $*$)	decidable	CAD virtual substitution
Presburger arithmetic (\mathbb{N} with $+$)	decidable	

FO theories and their decidability

Some first-order theories:

Logic	decidability	algorithm
Propositional logic	decidable	SAT-solving
Equational logic	decidable	SAT-encoding
Equational logic with uninterpr. functions	decidable	SAT-encoding
Linear real algebra (\mathbb{R} with $+$)	decidable	Simplex
Real algebra (\mathbb{R} with $+$ and $*$)	decidable	CAD virtual substitution
Presburger arithmetic (\mathbb{N} with $+$)	decidable	branch and bound, Omega test

FO theories and their decidability

Some first-order theories:

Logic	decidability	algorithm
Propositional logic	decidable	SAT-solving
Equational logic	decidable	SAT-encoding
Equational logic with uninterpr. functions	decidable	SAT-encoding
Linear real algebra (\mathbb{R} with $+$)	decidable	Simplex
Real algebra (\mathbb{R} with $+$ and $*$)	decidable	CAD virtual substitution
Presburger arithmetic (\mathbb{N} with $+$)	decidable	branch and bound, Omega test
Peano arithmetic (\mathbb{N} with $+$ and $*$)		

FO theories and their decidability

Some first-order theories:

Logic	decidability	algorithm
Propositional logic	decidable	SAT-solving
Equational logic	decidable	SAT-encoding
Equational logic with uninterpr. functions	decidable	SAT-encoding
Linear real algebra (\mathbb{R} with $+$)	decidable	Simplex
Real algebra (\mathbb{R} with $+$ and $*$)	decidable	CAD virtual substitution
Presburger arithmetic (\mathbb{N} with $+$)	decidable	branch and bound, Omega test
Peano arithmetic (\mathbb{N} with $+$ and $*$)	undecidable	

FO theories and their decidability

Some first-order theories:

Logic	decidability	algorithm
Propositional logic	decidable	SAT-solving
Equational logic	decidable	SAT-encoding
Equational logic with uninterpr. functions	decidable	SAT-encoding
Linear real algebra (\mathbb{R} with $+$)	decidable	Simplex
Real algebra (\mathbb{R} with $+$ and $*$)	decidable	CAD virtual substitution
Presburger arithmetic (\mathbb{N} with $+$)	decidable	branch and bound, Omega test
Peano arithmetic (\mathbb{N} with $+$ and $*$)	undecidable	-

FO theories and their decidability

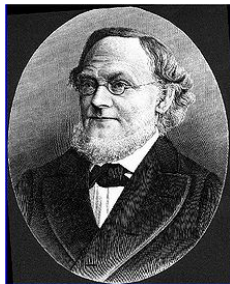
Some first-order theories:

Logic	decidability	algorithm
Propositional logic	decidable	SAT-solving
Equational logic	decidable	SAT-encoding
Equational logic with uninterpr. functions	decidable	SAT-encoding
Linear real algebra (\mathbb{R} with $+$)	decidable	Simplex
Real algebra (\mathbb{R} with $+$ and $*$)	decidable	CAD virtual substitution
Presburger arithmetic (\mathbb{N} with $+$)	decidable	branch and bound, Omega test
Peano arithmetic (\mathbb{N} with $+$ and $*$)	undecidable	-

But actually what does it mean “decidable” or “undecidable”?

Milestones for Peano arithmetic

- Peano arithmetic goes all the way back to ancient Greek mathematics.
- But the modern theory of arithmetic was developed only in the second half of the 19th century.
- Hermann Graßmann (1809-1877)
- Richard Dedekind (1833-1916)
- Gottlob Frege (1848-1925)
- Giuseppe Peano (1858-1932)
- ...



- “Die lineare Ausdehnungslehre, ein neuer Zweig der Mathematik” [The Theory of Linear Extension, a New Branch of Mathematics] (1844)
- Basics of **linear algebra** and **vector spaces**.
- Grassmann showed that once geometry is put into the algebraic form, then the number 3 has no privileged role as the number of spatial dimensions; the number of possible dimensions is in fact unbounded.



“Stetigkeit und irrationale Zahlen” [Continuity and irrational numbers]
(1912)

- **Dedekind cut:** An irrational number divides the rational numbers into two sets, with all the members of one set (upper) being strictly greater than all the members of the other (lower) set.
- Every location on the number line continuum contains either a rational or an irrational number. Thus there are **no empty locations, gaps, or discontinuities.**

Richard Dedekind (1833-1916)

- If there existed a one-to-one correspondence between two sets, Dedekind said them to be “similar”. He invoked similarity to give the first precise definition of an **infinite set**:

Definition (Dedekind's theorem)

A set is infinite when it is “similar to a proper part of itself”.

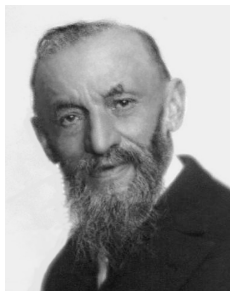
E.g., the set \mathbb{N} of natural numbers can be shown to be similar to the subset of square numbers.

- He also proposed an **axiomatic** foundation for the natural numbers, one year before Peano formulated an equivalent but simpler set of axioms, now the standard ones.



- “Begriffsschrift” (1879)
- Rigorous treatment of **functions** and **variables**.
- Frege invented axiomatic predicate logic, especially **quantified variables**. Previous logic had dealt with the logical operators “and”, “or”, “some”, “all” etc., but iterations of these operations, especially “some” and “all”, were little understood.

- One of Frege's stated purposes was to isolate the logical principles of inference, so that **in the proper representation of mathematical proof, one would at no point appeal to "intuition"**. If there was an intuitive element, it was to be isolated and represented separately as an axiom; from there on, the proof was to be purely logical and without gaps.
- Having exhibited this possibility, Frege's larger purpose was to defend **the view that arithmetic is a branch of logic**: unlike geometry, arithmetic was to be shown to have no basis in "intuition", and no need for non-logical axioms.
- The analysis of logical concepts and the machinery of formalization that is essential to the **incompleteness results** of Gödel and Turing is ultimately due to Frege.



“The principles of arithmetic, presented by a new method” (1889)

- Influence of Dedekind
- **Axiomatic theory of arithmetic**

Axiomatize Peano arithmetic, i.e.,

- the set \mathbb{N} of natural numbers,
- the operator $+$ (successor function S), and
- the operator $*$.

Peano axioms (first-order form)

Peano axioms (first-order form)

A1 $0 \in \mathbb{N}$

A2 $\forall n \in \mathbb{N}. n + 1 \in \mathbb{N}$

Peano axioms (first-order form)

A1 $0 \in \mathbb{N}$

A2 $\forall n \in \mathbb{N}. n + 1 \in \mathbb{N}$

A3 $\forall n \in \mathbb{N}. n + 1 \neq 0$

A4 $\forall n, m \in \mathbb{N}. n + 1 = m + 1 \rightarrow n = m$

Peano axioms (first-order form)

$$\text{A1 } 0 \in \mathbb{N}$$

$$\text{A2 } \forall n \in \mathbb{N}. n + 1 \in \mathbb{N}$$

$$\text{A3 } \forall n \in \mathbb{N}. n + 1 \neq 0$$

$$\text{A4 } \forall n, m \in \mathbb{N}. n + 1 = m + 1 \rightarrow n = m$$

$$\text{A5 } \forall n \in \mathbb{N}. n + 0 = n$$

$$\text{A6 } \forall n, m \in \mathbb{N}. n + (m + 1) = (n + m) + 1$$

Peano axioms (first-order form)

$$\text{A1 } 0 \in \mathbb{N}$$

$$\text{A2 } \forall n \in \mathbb{N}. n + 1 \in \mathbb{N}$$

$$\text{A3 } \forall n \in \mathbb{N}. n + 1 \neq 0$$

$$\text{A4 } \forall n, m \in \mathbb{N}. n + 1 = m + 1 \rightarrow n = m$$

$$\text{A5 } \forall n \in \mathbb{N}. n + 0 = n$$

$$\text{A6 } \forall n, m \in \mathbb{N}. n + (m + 1) = (n + m) + 1$$

$$\text{A7 } \forall n \in \mathbb{N}. n * 0 = 0$$

$$\text{A8 } \forall n, m \in \mathbb{N}. n * (m + 1) = n * m + n$$

Peano axioms (first-order form)

A1 $0 \in \mathbb{N}$

A2 $\forall n \in \mathbb{N}. n + 1 \in \mathbb{N}$

A3 $\forall n \in \mathbb{N}. n + 1 \neq 0$

A4 $\forall n, m \in \mathbb{N}. n + 1 = m + 1 \rightarrow n = m$

A5 $\forall n \in \mathbb{N}. n + 0 = n$

A6 $\forall n, m \in \mathbb{N}. n + (m + 1) = (n + m) + 1$

A7 $\forall n \in \mathbb{N}. n * 0 = 0$

A8 $\forall n, m \in \mathbb{N}. n * (m + 1) = n * m + n$

A9 For every $k \in \mathbb{N}$ and every Peano formula $\varphi(x_0, \dots, x_k)$ an instance of the following first-order induction schema:

$$\forall \vec{m} \in \mathbb{N}^k. [(\varphi(0, \vec{m}) \wedge \\ \forall n \in \mathbb{N}. (\varphi(n, \vec{m}) \rightarrow \varphi(n + 1, \vec{m}))) \rightarrow (\forall n. \varphi(n, \vec{m}))]$$

Peano axioms (first-order form)

$$A1 \quad 0 \in \mathbb{N}$$

$$A2 \quad \forall n \in \mathbb{N}. n + 1 \in \mathbb{N}$$

$$A3 \quad \forall n \in \mathbb{N}. n + 1 \neq 0$$

$$A4 \quad \forall n, m \in \mathbb{N}. n + 1 = m + 1 \rightarrow n = m$$

$$A5 \quad \forall n \in \mathbb{N}. n + 0 = n$$

$$A6 \quad \forall n, m \in \mathbb{N}. n + (m + 1) = (n + m) + 1$$

$$A7 \quad \forall n \in \mathbb{N}. n * 0 = 0$$

$$A8 \quad \forall n, m \in \mathbb{N}. n * (m + 1) = n * m + n$$

A9 For every $k \in \mathbb{N}$ and every Peano formula $\varphi(x_0, \dots, x_k)$ an instance of the following first-order induction schema:

$$\forall \vec{m} \in \mathbb{N}^k. [(\varphi(0, \vec{m}) \wedge \\ \forall n \in \mathbb{N}. (\varphi(n, \vec{m}) \rightarrow \varphi(n + 1, \vec{m}))) \rightarrow (\forall n. \varphi(n, \vec{m}))]$$

Note: there are infinitely many axioms in the second-order form!

Peano axioms (first-order form)

$$A1 \quad 0 \in \mathbb{N}$$

$$A2 \quad \forall n \in \mathbb{N}. n + 1 \in \mathbb{N}$$

$$A3 \quad \forall n \in \mathbb{N}. n + 1 \neq 0$$

$$A4 \quad \forall n, m \in \mathbb{N}. n + 1 = m + 1 \rightarrow n = m$$

$$A5 \quad \forall n \in \mathbb{N}. n + 0 = n$$

$$A6 \quad \forall n, m \in \mathbb{N}. n + (m + 1) = (n + m) + 1$$

$$A7 \quad \forall n \in \mathbb{N}. n * 0 = 0$$

$$A8 \quad \forall n, m \in \mathbb{N}. n * (m + 1) = n * m + n$$

A9 For every $k \in \mathbb{N}$ and every Peano formula $\varphi(x_0, \dots, x_k)$ an instance of the following first-order induction schema:

$$\forall \vec{m} \in \mathbb{N}^k. [(\varphi(0, \vec{m}) \wedge \\ \forall n \in \mathbb{N}. (\varphi(n, \vec{m}) \rightarrow \varphi(n + 1, \vec{m}))) \rightarrow (\forall n. \varphi(n, \vec{m}))]$$

Note: there are infinitely many axioms in the second-order form!

Q: Prove $5 \in \mathbb{N}$!

Peano axioms (first-order form)

$$A1 \quad 0 \in \mathbb{N}$$

$$A2 \quad \forall n \in \mathbb{N}. n + 1 \in \mathbb{N}$$

$$A3 \quad \forall n \in \mathbb{N}. n + 1 \neq 0$$

$$A4 \quad \forall n, m \in \mathbb{N}. n + 1 = m + 1 \rightarrow n = m$$

$$A5 \quad \forall n \in \mathbb{N}. n + 0 = n$$

$$A6 \quad \forall n, m \in \mathbb{N}. n + (m + 1) = (n + m) + 1$$

$$A7 \quad \forall n \in \mathbb{N}. n * 0 = 0$$

$$A8 \quad \forall n, m \in \mathbb{N}. n * (m + 1) = n * m + n$$

A9 For every $k \in \mathbb{N}$ and every Peano formula $\varphi(x_0, \dots, x_k)$ an instance of the following first-order induction schema:

$$\forall \vec{m} \in \mathbb{N}^k. [(\varphi(0, \vec{m}) \wedge \\ \forall n \in \mathbb{N}. (\varphi(n, \vec{m}) \rightarrow \varphi(n + 1, \vec{m}))) \rightarrow (\forall n. \varphi(n, \vec{m}))]$$

Note: there are infinitely many axioms in the second-order form!

Q: Prove $5 \in \mathbb{N}$! Prove $\forall n \in \mathbb{N}. n \neq n + 1$!

- When the Peano axioms were first proposed, **Bertrand Russell** and others agreed that these axioms implicitly defined what we mean by a “natural number”.
- **Henri Poincaré** was more cautious, saying they only defined natural numbers if they were **consistent**; if there is a proof that starts from just these axioms and derives a contradiction such as $0 = 1$, then the axioms are inconsistent, and don't define anything.
- International Congress of Mathematicians at Paris in 1900: David Hilbert posed the problem of proving their consistency using only finitistic methods as the second of his 23 problems.



- Researchers' primary aim should be to establish mathematics on a solid and **provably consistent foundation of axioms**, from which, in principle, **all mathematical truths could be deduced** (by the standard methods of predicate logic).
- **Entscheidungsproblem (decision problem)**: Could an **effective procedure** be devised which would demonstrate—in a finite time—whether any given mathematical proposition was, or was not, provable from a given set of axioms?

Here we can see three distinguishable concerns.

- **Consistency:** The set of axioms should be consistent, and provably so.
- **Completeness:** All mathematical truths should be deducible from those axioms.
- **Decidability:** There should be a clearly formulated procedure which is such that, given any statement of mathematics, it can definitively establish within a finite time whether or not that statement follows from the given axioms.

Consistency vs completeness

- A **consistent** system is one in which it is never possible to prove **both** a proposition P and its negation $\neg P$.
- A **complete** system is one in which it is always possible to prove **either** P **or** $\neg P$, for any proposition P that is expressible within the system.

Consistency vs completeness

- A **consistent** system is one in which it is never possible to prove **both** a proposition P and its negation $\neg P$.
- A **complete** system is one in which it is always possible to prove **either** P **or** $\neg P$, for any proposition P that is expressible within the system.

If we were able to achieve a **consistent and complete system of arithmetic**, with true axioms and valid rules, then any arithmetical proposition would be provable if, and only if, it is true. A major part of Hilbert's dream would thus be realised.

- Around 1920: Presburger proved that Presburger arithmetic is complete (using quantifier elimination).

Completeness of Presburger arithmetic

- Around 1920: Presburger proved that **Presburger arithmetic is complete** (using quantifier elimination).
- Would multiplication make the difference?

- Around 1920: Presburger proved that **Presburger arithmetic is complete** (using quantifier elimination).
- Would multiplication make the difference?
- 1929: **Thoralf Skolem** showed that the theory of \mathbb{N} with $*$ but without the successor function and $+$ is **complete**.

- Around 1920: Presburger proved that **Presburger arithmetic is complete** (using quantifier elimination).
- Would multiplication make the difference?
- 1929: **Thoralf Skolem** showed that the theory of \mathbb{N} with $*$ but without the successor function and $+$ is **complete**.
- Around 1930: **Alfred Tarski** showed **completeness of real algebra**.

- Around 1920: Presburger proved that **Presburger arithmetic is complete** (using quantifier elimination).
- Would multiplication make the difference?
- 1929: **Thoralf Skolem** showed that the theory of \mathbb{N} with $*$ but without the successor function and $+$ is **complete**.
- Around 1930: **Alfred Tarski** showed **completeness of real algebra**.
- So Peano arithmetic was expected to be complete, too.

- Around 1920: Presburger proved that **Presburger arithmetic is complete** (using quantifier elimination).
- Would multiplication make the difference?
- 1929: **Thoralf Skolem** showed that the theory of \mathbb{N} with $*$ but without the successor function and $+$ is **complete**.
- Around 1930: **Alfred Tarski** showed **completeness of real algebra**.
- So Peano arithmetic was expected to be complete, too.
- (In 1950 Raphael M. Robinson will show that Peano arithmetic without induction, called **Robinson arithmetic**, is **complete**.)

Kurt Gödel (1906-1978), first incompleteness theorem



In a famous paper published in 1931, Gödel proved his

Theorem (First incompleteness theorem)

- *In any true (and hence **consistent**) axiomatic theory*
- ***sufficiently rich** to enable the expression and proof of basic arithmetic propositions,*
- *it will be possible to construct an arithmetical proposition G such that **neither G , nor its negation, is provable** from the given axioms.*

*Hence the system must be **incomplete**.*

*Moreover **G must be a true statement** of arithmetic.*

Proof of Gödel's first incompleteness theorem

Proof.

- Gödel's proof ingeniously shows how statements about mathematical relationships (e.g. that a particular sequence of propositions provides a **proof** of some proposition P) can be **encoded** as statements within arithmetic.
- This encoding, moreover, is **truth-preserving**, so that the encoded "meta-mathematical" statement will be true if, and only if, the encoding statement of arithmetic is true.



Proof of Gödel's first incompleteness theorem (cont.)

Proof.

- Gödel derived an arithmetical proposition G which encodes the statement that G itself is unprovable within the system.
- Assume that the system would be complete.
 - Completeness $\rightarrow G$ is false
 - G is false $\rightarrow G$ is provable
 - G is false $\rightarrow \neg G$ is true $\xrightarrow{\text{completeness}}$ $\neg G$ is provable.
- Thus if the system is complete, it cannot be consistent, since both G and $\neg G$ would then be provable within it.



Gödel's second incompleteness theorem

Gödel's second incompleteness theorem, also published in the same article, follows from the first.

Theorem

No *consistent* axiomatic theory *sufficiently rich* to enable the expression and proof of basic arithmetic propositions *can prove its own consistency*.

Proof.

Suppose that a system was able to prove its own consistency. Then by the above argumentation G is provable within the system. But since G encodes the statement that G is unprovable within the system, we have a contradiction. It follows that the system cannot after all prove its own consistency. □

What remains open

- Gödel's incompleteness theorems left the Entscheidungsproblem as unfinished business.
- He had shown that any consistent axiomatic system of arithmetic would leave some arithmetical truths unprovable (without any computable function).

Definition

A function is **computable**, if there is an algorithm that can calculate its result, in a finite number of steps.

- However, this did not in itself rule out the existence of some “effectively computable” decision procedure which would infallibly, and in a finite time, reveal whether or not any given proposition was, or was not, provable.



“On Computable Numbers, with an Application to the Entscheidungsproblem” (1936)

- He devised a rigorous notion of **effective computability** based on the “Turing Machine”.

Definition

An **effective method** is one which reduces the solution of some class of problems to a series of routine steps which

- always gives some answer after a finite time (**termination**),
 - always gives the right answer (**soundness**), and
 - works for all problem instances of the class (**completeness**).
-
- An effective method for calculating the values of a function is an **algorithm**.
 - Functions with an effective method are sometimes called **effectively computable**.

“On Computable Numbers, with an Application to the Entscheidungsproblem” (1936):

- He devised a rigorous notion of **effective computability** based on the “Turing Machine”.
- He then showed that **there exist problems that cannot be effectively computed** by this means.
- He did so by proving the impossibility of devising a Turing Machine program that can determine infallibly (and within a finite time) whether or not a given Turing Machine will eventually halt given some arbitrary input (**Halting Problem**).

Hence Turing proved that **Hilbert's Entscheidungsproblem was unsolvable**.



Theorem (Church's Thesis)

If some algorithm exists to carry out a calculation, then the same calculation can also be carried out by a Turing machine (as well as by a recursively definable function).

- Church's thesis is not a mathematical statement and **cannot be proven** by a mathematical proof.
- Despite this fact, the Church–Turing thesis now has near-universal acceptance.

Hilbert's dream was shattered...

- Any consistent axiomatic theory sufficiently rich to enable the expression and proof of basic arithmetic propositions can be neither complete (as Gödel had shown) nor effectively decidable (by Turing).
- Paris and Harrington (1977) gave the first “natural” example of a statement which is true for the integers but unprovable in Peano arithmetic.