Peano Arithmetic

MATH230

Te Kura Pāngarau
Te Whare Wānanga o Waitaha

Outline

1. Foundations of Mathematics

2. Identity

3. Peano Arithmetic

4. Predicates on PA

Axioms of Mathematics

> *"The investigations that follow concern the domain of predicate logic. It comprises the types of inference that are continually used in all parts of mathematics. What remains to be added to these are axioms and forms of inference that may be considered as being proper to the particular branches of mathematics."*
>
> — *Gerhard Gentzen (1935[1])*

[1] "Investigations into Logical Deduction."

Hilbert's Program

David Hilbert asked for mathematics to be formalised into a language so that proofs could be easily checked. He even hoped there would be a finite procedure that could decide whether a given proof was correct, or a finite procedure that could *generate* a proof of any given statement.

Hilbert knew that this would require stating axioms for mathematics, at least for each specific part of mathematics. Previous foundations provided by Euclid (some two millenia prior!) were no longer sufficient.

- Mathematics is now too rich, and
- there were known problems with Euclid's axioms and proofs.

First-order logic is a language in which these axioms and proofs can be written.

Direction

We now have the precise language required to express mathematics!

Formalising a particular area of mathematics in first-order logic requires us to:

- Pick a first-order language $\mathcal{L}$ i.e. fix a signature,
- Pick axioms that define the structure of the signature.

Throughout the entire course we have been carrying around a set Σ of hypotheses. Now we think of that set as our axioms and the conclusions in our arguments as the theorems we hope to deduce from those axioms.

First-Order Theories

Mathematics abounds with first-order theories. This idea was adopted for much of mathematics in the early 1900s, following the Hilbert's annoucement of his program. To the point that now you would be unlikely to find an advanced undergraduate course in mathemtaics that does not introduce the objects of interest (e.g. groups, rings, metric spaces) by specifying them as objects which obey some first-order axioms.

We are going to focus on a first-order theory of *arithmetic*. By that we will mean the natural (non-negative) numbers under addition and multiplication.

Language of Arithmetic

There are a number of different first-order theories of arithmetic, the signatures of which are typically made up of some number of these symbols:

$$\mathcal{L}_A : \{0, 1, +, \times, s, =, <\}$$

In order to define a *first order theory* one must make a choice of signature together with a collection of axioms that pin down the arithmetic structure.

$$\mathcal{T}_A = (\mathcal{L}_A, \Sigma_A)$$

The word theory may also be used to refer to the collection of all theorems deducible from the axioms, rather than just the pair consisting of signature and axioms.

Signature of Natural Numbers

Peano arithmetic, the first order theory of the natural numbers that we are to study, consists of terms and propositions generated by the signature $\mathcal{L} : \{0, s, +, \times, =\}$.

0	:	Constant
s	:	Unary function
$+$	:	Binary function
$\times$	:	Binary function
$=$	:	Binary predicate

Equality/identity is a predicate relevant to many first order theories. We will start by formalising what we mean by identity.

Identity: Introduction

Identity is an essential predicate for first-order theories of mathematics. We will now state introduction and elimination rules for the identity binary predicate $=$

$$\frac{}{t = t} = I$$

The introduction rule for the identity predicate is simply the claim that identity is reflexive — all terms are identical to themselves.

No two terms are equal - in this sense - unless they are equal symbol for symbol. If we want to impose further identities on terms, say in the definition of addition, then we are going to have to do that with axioms i.e. hypotheses.

Identity: Elimination

If ${}^{\Sigma_1}_{\alpha}\mathcal{D}_1$ is a deduction of α from Σ_1 and ${}^{\Sigma_2}_{s=t}\mathcal{D}_2$ is a deduction of $s = t$ from Σ_2, then

$$\frac{\begin{array}{cc}\Sigma_1 & \Sigma_2 \\ \mathcal{D}_1 & \mathcal{D}_2 \\ \alpha & s = t\end{array}}{\alpha[s/t]} = E$$

is a deduction of $\alpha[s/t]$ from the hypotheses $\Sigma_1 \cup \Sigma_2$. This states that identical terms may be substituted for identical terms. One can be selective about the instances of s which are replaced by t.

Example: Symmetric

$$s = t \dashv\vdash t = s$$

Example: Substitutivity

$$\vdash \ \forall x \, \forall y \, (Px \to (x = y \to Py))$$

Example: Substitutivity

$$\vdash \ \forall x \forall y \forall z \, (Rxy \to (x = z \to Rzy))$$

Example: Well Defined Functions

$$\vdash \ \forall x \forall y \ (x = y \rightarrow f(x) = f(y))$$

Towards PA: Signature

Peano arithmetic consists of terms and wff generated by the signature $\mathcal{L} : \{0, s, +, \times, =\}$.

0	:	Constant
s	:	Unary function
$+$	:	Binary function
$\times$	:	Binary function
$=$	:	Binary predicate.

Towards PA: Terms

Terms are generated by the constant(s) and function symbols. By a slighlt abuse of notation, we denote the collection of such terms as $\mathbb{N}$. These are some examples of such terms.

$$\begin{array}{rcl@{\qquad\qquad}rcl}
0 & : & \mathbb{N} & 0+0 & : & \mathbb{N} \\
s\ 0 & : & \mathbb{N} & s\ (0+0) & : & \mathbb{N} \\
s\ (s\ 0) & : & \mathbb{N} & (s\ 0 \times s\ 0) & : & \mathbb{N} \\
s\ (s\ (s\ 0)) & : & \mathbb{N} & s\ 0\ + (s\ (s\ (s\ 0))) & : & \mathbb{N} \\
s\ (s\ (s\ (s\ 0))) & : & \mathbb{N} & (s\ (0+0)) \times (0 + (s\ 0)) & : & \mathbb{N} \\
\vdots & : & \mathbb{N} & \vdots & : & \mathbb{N}
\end{array}$$

Of course terms involving addition and multiplication should always be equal to a (unique) term consisting of some (possibly zero) number of applications of the successor function to the constant 0. However, the signature alone does not enforce this.

Towards PA: Well Formed Formulae

Since the only predicate in the signature is the identity, all well-formed formulae of Peano arithmetic consist of possibily quantified assertions of equality.

$$\begin{gathered}
0 = 0 \\
s\ 0 = (s\ 0) + 0 \\
\neg(0 = 0 + 0) \\
\forall x : \mathbb{N},\ x + y = y + x \\
\forall x : \mathbb{N}\ \exists y : \mathbb{N},\ y = x + (s\ 0) \\
\exists x : \mathbb{N},\ x = x
\end{gathered}$$

In order to *prove* or *refute* any of these we need some grounds on which to do so. There is nothing for it, we must impose axioms to allow for the proof or refutation of such identities. In order to be trustworthy and amenable to meta-analysis, an economy of axioms is necessary.

Towards PA: Addition

How should one define addition of the natural numbers when represented in this way?

$$x + y = ?$$

If we focus on the second argument, then there are only two cases to consider. Either $y = 0$, or there exists some other term z such that $y = s\ z$. In the case that $y = 0$ the computation is clear:

$$x + 0 = x$$

What do we do in the other case?

Towards PA: Addition

Given that we know how to sum zero, on the right, we might expect to define the (right) sum of a successor by reducing it to the sum of a term with one less successor.

$$x + (s\ z) = \ ?$$

This is intended to represent $x + (z + 1)$

This strategy of defining a function by the possible forms of an argument is known as *pattern matching*. It has lead to a *recursive* definition of addition.

Example

Compute $3 + 2$ using this algorithm.

Towards PA: Multiplication

We can use the same pattern matching strategy to define multiplication recursively. Again, the case of right multiplication by zero is clear

$$x \times 0 = 0$$

While the recursive step may not be immediately obvious.

$$x \times (s\ z) = ?$$

It might be helpful to recall this represents $x \times (z + 1)$

Example

Compute 3×2 using this algorithm.

Axioms of PA

Peano Arithmetic has signature PA: $\{0, s, +, \times, =\}$ and axioms

1. $\forall x \neg(s(x) = 0)$
2. $\forall x\ \forall y((s(x) = s(y)) \rightarrow (x = y))$
3. $\forall x\ (x + 0 = x)$
4. $\forall x\ \forall y\ (x + s(y) = s(x + y))$
5. $\forall x\ (x \times 0 = 0)$
6. $\forall x\ \forall y\ (x \times s(y) = (x \times y) + x)$
7. $[P(0) \wedge \forall x\ (P(x) \rightarrow P(s(x)))] \rightarrow \forall y(P(y))$

Syntactic Sugar

We make the following abuse of notation

$$\begin{aligned} 1 &= s(0) \\ 2 &= s(1) = s(s(0)) \\ 3 &= s(2) = s(s(1)) = s(s(s(0))) \\ &\vdots \end{aligned}$$

Example

$$\mathsf{PA} \vdash 2 + 1 = 3$$

Example

$$\mathsf{PA} \vdash 2 \neq 1$$

Example

$$\mathsf{PA} \vdash 0 + 1 = 1$$

Example

$$\mathsf{PA} \vdash 7 \times 1 = 7$$

Computation à la Peano

$$\mathsf{PA} \vdash 3 + 2 = 5$$

$$\begin{array}{rcll}
3 + 2 & = & \mathsf{s\ s\ s\ 0 + s\ s\ 0} & \text{by [DESUGAR]} \\
 & = & \mathsf{s\ (s\ s\ s\ 0 + s\ 0)} & \text{by [PA4]} \\
 & = & \mathsf{s\ s\ (s\ s\ s\ 0 + 0)} & \text{by [PA4]} \\
 & = & \mathsf{s\ s\ (s\ s\ s\ 0)} & \text{by [PA3]} \\
 & = & \mathsf{s\ s\ s\ s\ s\ 0} & \text{by [REFL]} \\
 & = & 5 & \text{by [SUGAR]}
\end{array}$$

This abbreviation is looking ahead to the final topic.

Example

$$\mathsf{PA} \vdash \ \forall x\ (s(x) = x + 1)$$

Induction

How could one possibly prove a proposition of the form:

$$\forall x : \mathbb{N},\ \ldots$$

This signature generates infinitely many terms; how can we possibly check a predicate is provable at each of them?

Induction: Informal Example

Induction: Informal Example

Euclidea: Interactive Geometric Construction Puzzles.

Induction Axiom Schema

For each unary predicate P in the first-order theory PA we have the following axiom:

$$[P(0) \wedge \forall x\ (P(x) \rightarrow P(s(x)))] \rightarrow \forall y(P(y))$$

How do we make use of this in a proof?

Induction Proofs

Induction Proofs: Steps

Induction Proofs: Induction Step

Structure of Induction Proofs

$$\dfrac{\dfrac{\begin{matrix}\vdots \\ P(0)\end{matrix} \qquad \begin{matrix}\vdots \\ \forall x\ (P(x) \to P(s(x)))\end{matrix}}{(P(0) \land \forall x\ (P(x) \to P(s(x))))}\,\land I \qquad (P(0) \land \forall x\ (P(x) \to P(s(x)))) \to \forall y(P(y))}{\forall y\ P(y)}\ \text{MP}$$

Induction: Rule of Inference

This suggests that we can think of induction as a rule of inference.

$$\dfrac{\begin{array}{c}\vdots\\ \mathcal{D}_{BC}\\ \hline P(0)\end{array} \qquad \dfrac{\dfrac{\begin{array}{c}[P(a)]\\ \vdots\\ \mathcal{D}_{IS}\\ \hline P(s(a))\end{array}}{P(a)\rightarrow P(s(a))}\rightarrow I}{\forall x\ (P(x)\rightarrow P(s(x)))}\,\forall I}{\forall y\ P(y)}\ \mathrm{IND}$$

This presents the inductive mode of reasoning in the same manner as the modes we have formalised. It will also be helpful to think in these terms when we come to type theory.

Induction Summary

In order to prove $\forall n : \mathbb{N}, P(n)$ for a unary predicate $P : \mathbb{N} \to \mathrm{Prop}$ it is sufficient to prove the following two sequents:

$$\begin{array}{rrcr} \text{Base case:} & \mathrm{PA} & \vdash & P(0) \\ \text{Induction Step:} & \mathrm{PA}, P(n) & \vdash & P(s(n)) \end{array}$$

In the induction step, we refer to the hypothesis $P(n)$ as the induction hypothesis. In this step, we assume the predicate is true at some arbitrary $n : \mathbb{N}$ and aim to show it holds at the successor of that natural number.

Induction Example

$$\mathsf{PA} \vdash \forall x\ (0 + x = x)$$

Induction Example

Induction Example

Induction Example

$$\mathsf{PA} \vdash \forall x\ \forall y\ \ s(y) + x = s(y + x)$$

Induction Example

Induction Example

Number Theory

Mathematicians use other pieces of notation that aren't explicitly present in the signature of Peano arithmetic. For example we speak of the order $<$, $>$, $\leq, \geq$ on the natural numbers. As well as divisibility, primality, and much much more.

Although not explicitly present, many terms can be defined as abbreviations for well-formed formulae using this "restricted" signature of Peano Arithmetic. For example:

$$a \leq b :\equiv \ (\exists x : \mathbb{N},\ b = a + x)$$

$$a|b :\equiv \ (\exists x : \mathbb{N},\ (b = a \times x) \wedge \neg(a = 0))$$

Example: Reflexive

$$\mathrm{PA} \ \vdash \ x \leq x$$

Example: Transitive

$$\mathrm{PA} \ \vdash \ (a \leq b \wedge b \leq c) \rightarrow a \leq c$$

Example: Transitive

$$\mathrm{PA} \ \vdash \ (a \leq b \wedge b \leq c) \to a \leq c$$

Better Way

Adding further syntax abbreviations (so called, syntactic sugar) and proving theorems about those structures becomes unwieldly. In reaction to this, one can return to the less formal natural language reasoning typical of mathematics. However, there are now software tools that allow for nicer syntax than the trees we have been writing, without any compromise in the rigour that they provide.

Over the next two topics we will see where these tools have come from, before learning how to use them in the final topic of the course. We will return to Peano arithmetic at the end of the course to prove further theorems about the natural numbers, their order, and divisibility properties.

Comments on Metatheory

Hilbert wanted to know:

Q1: PA $\vdash \bot$?

Q2: If α wff, can we know whether PA $\vdash \alpha$ or PA $\vdash \neg\alpha$?

Kurt Gödel provided answers to the first two questions. We can't determine the consistency of Peano arithmetic. There are wff which are neither provable nor refutable from the axioms of Peano arithmetic. Any other theory at least as strong as PA *still has these properties* - they are unavoidable.

Comments on Metatheory

Hilbert wanted to know:

Q3: Is there a "finite procedure" to determine theoremhood?
Q4: Is there a "finite procedure" for checking a proof?

Brouwer, Heyting, and Kolmogorov defined the notion of proof in terms of the word "algorithm".

These terms "finite procedure" and "algorithm" were treated on a *you know one when you see one* basis. However, to possibly answer these questions of Hilbert in the negative precise definitions were needed.

Q: What would it mean for it to be impossible that such an algorthim could exist?

Further Reading

Dirk van Dalen, Logic and Structure.

Jeremy Avigad et al, Logic and Proof[2] Chapter 17.

The Natural Number Game gives a gamified introduction to using Lean to prove many of the theorems from this topic on Peano Arithmetic. Your assignment will be very similar to the “levels” of this “game”.

[2] Note: this uses an outdated version of Lean, but is still a good resource for the theory of this course.