Proof and Algorithms

Xiaoyi Cui

This is just a brief summary of the contents of the lecture. Please note: most of the calculations and demonstrations are neglected.

Axiomatic mathematics — the components of a theory

Mathematics are commonly put into axiomatic systems, which is any set of axioms from which some or all axioms can be used in conjunction to logically derive theorems.

Undefined terms: objects or manipulations that are explicit or intuitive in its own.

Axioms: properties (which are assumed to be true) that undefined terms need to satisfy are called axioms/postulates (from the Greek word meaning "worthy").

Definition: the means for binding a concept and a set of associated properties that describe the concept.

Any statement that is not an axiom or definition needs to be proved.

Theorems: Important statements that have been proved are called theorems.

Propositions: Less important (true) statements.

Lemma: subtheorems that will help prove part of a more important theorem or proposition.

Corollary: a statement whose truth is an immediate consequence of some other theorem or proposition.

Example 1 (Euclid's Postulates) *Undefined terms: straight line, point, circle, angle, etc.*

Axioms: "Let the following be postulated":

- *"To draw a straight line from any point to any point."*
- *"To produce [extend] a finite straight line continuously in a straight line."*
- *"To describe a circle with any centre and distance (radius)."*
- *"That all right angles are equal to one another."*
- *"That, if a straight line falling on two straight lines make the interior angles on the same side less than two right angles, the two straight lines, if produced indefinitely, meet on that side on which are the angles less than the two right angles."*

Example 2 (Boolean Algebra)

Further properties of axioms: they need to be independent and consistent. I.e., let I labels the axioms of a system $\{A_i\}_{i\in I}$, then we have that

$$(\forall i)((\forall j \in I\backslash\{i\})A_j \to A_i) \iff \mathbb{F},$$

and

$$(\forall i)(\forall j \in I\backslash\{i\})(\wedge_j A_j \to \neg A_i) \iff \mathbb{F}.$$

Proof techniques

Assume that we would like to show that $P \to Q$ is true.

Trivial proof

From truth table of $P \to Q$ and that P is false, deduce that the statement is true.

Direct proof

$$P \to Q \Leftarrow (P \to S_1) \wedge (S_1 \to S_2) \wedge \cdots \wedge (S_n \to Q)$$

Proof by contraposition

Consider $(\neg Q \to \neg P) \Leftrightarrow (P \to Q)$.

Proof by contradiction

Assume that $P \to Q$ is false (i.e., $P \wedge \neg Q$ is true), conclude that it is logically equivalent to a contradiction, or $\neg P$, or Q. Consider $\neg(P \to Q) \Leftrightarrow \mathbb{F}$, or $\neg(P \to Q) \Leftrightarrow \neg P$, or $\neg(P \to Q) \Leftrightarrow Q$.

Proof by case

If $P \Rightarrow P_1 \vee P_2 \vee P_3 \vee \cdots$, then $\wedge_i(P_i \to Q) \Rightarrow P \to Q$.

Example 3 *If $n \in \mathbb{Z}$, then $n^3 - n$ is even.*

Proof by construction

Use the implications with existential quantifiers. $(P(a) \to Q) \Rightarrow (\exists x)(P(x) \to Q)$.

Proof of biconditionals

Example 4 *The following statements are equivalent:*

- A_1

- ...
- A_n.

There are essentially two approaches using graphic representation.

Further examples

Infinite numbers of primes

Theorem 1 *There is an infinite number of distinct primes.*

Proof. Suppose instead that there is only a finite number of primes. Denote them as $\{P_1, \cdots, p_n\}$ where n is a positive integer.

Consider the positive integer $q = (p_1 p_2 \cdots P_n) + 1$. Since $q \notin \{p_1, P_2, \cdots P_n\}$, q must be composite.

But none of the primes P_k divide q and $\{P_1, P_2, \cdots p_n\}$ are all the primes, q cannot be composite. This leads to a contradiction. Therefore, there must be an infinite number of primes. Q.E.D.

P ="S is the set of all primes", Q ="$|S|$ is finite". Now assume that $P \wedge \neg Q$ is true.
W ="$(p_1 p_2 \cdots P_n) + 1$ is composite", and we have that $(P \wedge \neg Q) \Rightarrow W$.

Now we have that $(P \wedge \neg Q) \Rightarrow \neg W$.

$(P \wedge \neg Q) \Rightarrow (W \wedge \neg W)$, hence $\mathbb{T} \Rightarrow \neg(P \wedge \neg Q)$, i.e., $P \Rightarrow Q$.

Prime Divisibility Property

Theorem 2 *Let p be a prime. If p divides the product $a_1 a_2 \cdots a_n$, then p divides at least one of the factors a_i.*

We shall look into different cases. First, if $n = 2$, and then, if $n > 2$.

Exercise 1 *Use the above result to prove the following proposition: every integer $n > 2$ can be uniquely written as a product of primes in ascending order.*

Mathematical Induction

Theorem 3 *If $\{P(i)\}$ is a set of statements such that*

- *$P(1)$ is true,*
- *$P(i) \rightarrow P(i+1)$ is true for $i > 1$,*

then P_k is true for all positive integers k. This can be stated more succinctly as $[P(1) \wedge (\forall i)(P(i) \rightarrow P(i+1))] \Rightarrow [(\forall k)P(k)]$.

Remark 1 *In the above theorem, the second condition can be replaced by "$\wedge_{j \leq i} P(j) \rightarrow P(i+1)$ is true for $i > 1$". (Why?) This is called the theorem of complete induction.*

Exercise 2 *Prove that "every integer $n > 2$ can be written as a product of primes".*

Exercise 3 *Let $n > 1$. Suppose we have a $2^n \times 2^n$ chess board, with one square missing, and a box full of L-shaped tiles. Each tile can cover 3 squares on the chess board. No matter which square on the chess board is missing, we can entirely cover the remaining squares with the tiles.*

The well-ordering principal is an axiom which says "every nonempty set of natural numbers has a smallest element".

Theorem 4 *The following are equivalent[1]: 1) the well-ordering principle (WOP) 2) the theorem of mathematical induction (MI) and 3) the theorem of complete induction (CI).*

[1] If you are interested in the proof, or more aspects, try looking for "Piano axioms" in Wikipedia.

Algorithms and how to measure the efficiency

Definition 1 *An algorithm is a finite sequence of unambiguous steps for solving a problem or completing a task in a finite amount of time.*

Example 5

Algorithms are like functions — they take input, and produce output. It is important to know about the domain. On the other hand, we are interested in “multi-variable" things.

If the size of a problem increases, will the algorithms become complicated? And how to decide the relation between the size and the computability?

Question 1: is the problem solvable?

Question 2: for how long can one solve a problem?

Let us discuss the second question first. Assume a problem is solvable, the complexicity of an algorithm

Definition 2

- *A function f is said to be in big-O of g, if there exists N and c constant, such that for all $n > N$, $|f(n)| \leq c \cdot |g(n)|$.*
- *A function f is said to be in big-Ω of g, if there exists N and c constant, such that for all $n > N$, $|f(n)| \geq c \cdot |g(n)|$.*
- *A function f is said to be in big-Θ of g if it is in $O(g) \cap \Omega(g)$.*

Exercise 4 *Show that if $f_1 \in \Theta(g_1)$ and $f_2 \in \Theta(g_2)$, then $f_1 + f_2 \in \Theta(max\{g_1, g_2\})$, and $f_1 \cdot f_2 \in \Theta(g_1 \cdot g_2)$.*

Exercise 5 *Try to give sufficient conditions for each of those above definitions using limit.*