Foundations of Mathematics

Branches of Logic
1. Theory of Computations (i.e. Recursion Theory).
2. Proof Theory.
3. Model Theory.
4. Set Theory.

Informal Statement Calculus

Statements and Connectives

Example
If A then B, or $(A \to B)$, is "conditional" or "implication".

Definition: Statement/Boolean Variable
A statement variable (or Boolean variable) is a variable that can assume two values, "true" and "false" (denoted T, F or 1, 0).

Definition: Boolean Function
A Boolean function is a function of one or of several Boolean variables that can assume two values: "true" and "false".

If F is a Boolean function of n Boolean variables, then $F : \underbrace{\{T, F\} \times \cdots \times \{T, F\}}_{n} = \{T, F\}^n \to \{T, F\}$.

Truth Functions and Truth Tables

Definition: Truth Table
A table of all values of Boolean funtions is called a truth table.

Example
How many Boolean functions of n Boolean variables are there? 2^{2^n}.

Example
For $n = 2$, there are 16 Boolean functions. Some important Boolean fuctions of 2 variables are:
1. Conjunction ("AND"), denoted $A \wedge B$, $A \& B$, or AB.
2. Disjuction ("OR"), denoted $A \vee B$.
3. Equivalence, denoted $A \leftrightarrow B$.

Example

x	0	1	0	1	
y	0	0	1	1	Comments
0	0	0	0	0	Contradiction
$x \wedge y$	0	0	0	1	AND, conjunction

$\sim(y\to x)$	0	0	1	0	
y	0	0	1	1	
$\sim(x\to y)$	0	1	0	0	
x	0	1	0	1	
$x\oplus y$	0	1	1	0	addition modulo 2
$x\vee y$	0	1	1	1	
$x\downarrow y$	1	0	0	0	NOR, Pierce arrow
$x\leftrightarrow y$	1	0	0	1	equivalence, biconditional
$\sim x$	1	0	1	0	not x
$x\to y$	1	0	1	1	conditional implication
$\sim y$	1	1	0	0	not y
$y\to x$	1	1	0	1	conditional implication
$x\mid y$	1	1	1	0	NAND, Scheffer stroke
1	1	1	1	1	tautology

Normal Forms

Theorem

Any Boolean function that is not a contradiction can be represented by an expression involving only the connectors $\sim$, $\wedge$, $\vee$.

Proof:

1. Identify all n-tuples where the Boolean function is 1.
2. For each such n-tuples construct a monomial as follows: For every $i=1,\ldots,n$ take x_i if $x_i=1$ and $(\sim x_i)$ if $x_i=0$. Form the conjunction of these variables and their negations.
3. Form the disjunction of the constructed monomials.

The result is called a disjunctive normal form (DNF) of the considered Boolean function.

Example

$f(x,y,z)=(x\to y)\to z$.

x	y	z	$x\to y$	$(x\to y)\to z$	
F	F	F	T	F	
F	F	T	T	T	$(\sim x)\wedge(\sim y)\wedge z$
F	T	F	T	F	
F	T	T	T	T	$(\sim x)\wedge y\wedge z$
T	F	F	F	T	$x\wedge(\sim y)\wedge(\sim z)$
T	F	T	F	T	$x\wedge(\sim y)\wedge z$
T	T	F	T	F	
T	T	T	T	T	$x\wedge y\wedge z$

So $f(x,y,z)=(x\to y)\to z=((\sim x)\wedge(\sim y)\wedge z)\vee((\sim x)\wedge y\wedge z)\vee(x\wedge(\sim y)\wedge(\sim z))\vee(x\wedge(\sim y)\wedge z)\vee(x\wedge y\wedge z)$, a

disjunctive normal form.

De Morgan's Laws
1. $(\sim(A\wedge B))=((\sim A)\vee(\sim B))$.
2. $(\sim(A\vee B))=((\sim A)\wedge(\sim B))$.

Proof: Verify the truth table.

Theorem: De Morgan's Laws
1. $\sim(x_1\wedge\cdots\wedge x_n)=(\sim x_1)\vee\cdots\vee(\sim x_n)$.
2. $\sim(x_1\vee\cdots\vee x_n)=(\sim x_1)\wedge\cdots\wedge(\sim x_n)$.

Proof: Use induction.

Example
Let $f(x_{1,}\ldots,x_n)$ be not a tautology. Consider $(\sim f)(x_{1,}\ldots,x_n)$ (not a contradiction) and its DNF $(\sim f)(x_{1,}\ldots,x_n)=\bigvee_{i=1}^{m}\left(\bigwedge_{j=1}^{k_i}a_{ij}\right)$, where a_{ij} are Boolean variables or their negations.

$f=\sim(\sim f)=\sim\left(\bigvee_{i=1}^{m}\left(\bigwedge_{j=1}^{k_i}a_{ij}\right)\right)=\bigwedge_{i=1}^{m}\left(\bigvee_{j=1}^{k_i}(\sim a_{ij})\right)=\bigwedge_{i=1}^{m}\left(\bigvee_{j=1}^{k_i}b_{ij}\right)$. This is a conjunctive normal form.

Theorem
Each Boolean function that is not a tautology can be represented in a conjunctive normal form (CNF).

Example
$f(x,y,z)=x\rightarrow(y\rightarrow z)$.

x	y	z	$y\rightarrow z$	$x\rightarrow(y\rightarrow z)$	$\sim(x\rightarrow(y\rightarrow z))$
F	F	F	T	T	F
F	F	T	T	T	F
F	T	F	F	T	F
F	T	T	T	T	F
T	F	F	T	T	F
T	F	T	T	T	F
T	T	F	F	F	T
T	T	T	T	T	F

DNF of $(\sim f)(x,y,z)=x\wedge y\wedge(\sim z)$.
CNF of $f(x,y,z)=(\sim x)\vee(\sim y)\vee z$.

Adequate Sets of Connectives

Definition: Adequate
A set of connectives is called adequate if every Boolean function can be written down using this set of connectors.

Proposition
The set of connectors $\{\sim, \wedge, \vee\}$ is adequate.

Theorem
$\{\sim, \wedge\}$ and $\{\sim, \vee\}$ are both adequate sets of connectors.

Proof: By De Morgan's Laws, $\bigvee_{i=1}^{m}\left(\bigwedge_{j=1}^{k_i} a_{ij}\right)=\sim\bigwedge_{i=1}^{m}\left(\sim\bigwedge_{j=1}^{k_i} a_{ij}\right)$. Also $\bigvee_{i=1}^{m}\left(\bigwedge_{j=1}^{k_i} a_{ij}\right)=\bigvee_{i=1}^{m}\left(\sim\bigvee_{j=1}^{k_i}(\sim a_{ij})\right)$ since $\bigwedge_{j=1}^{k_i} a_{ij}=\sim\left(\bigvee_{j=1}^{k_i}(\sim a_{ij})\right)$.

Theorem
$\{\sim, \rightarrow\}$ is an adequate set of connectors.

Proof: $(\sim x)\rightarrow y=x\vee y$.

Theorem
$\{\downarrow\}$ and $\{|\}$ are both adequate sets of connectors.

Proof: $x\downarrow x=\sim x$, $(x\downarrow x)\downarrow(y\downarrow y)=x\wedge y$. Also $x|x=\sim x$, $(x|x)|(y|y)=x\vee y$.

Arguments and Validity

Definition: Statement Form
A statement form is a particular expression for a Boolean function.

Definition: Statement Form
A statement form is an expression involving variables which can be formed by the following rules:

1. Each statement variable is a statement form.
2. If A, B are statement forms, then $(\sim A)$, $(A\vee B)$, $(A\wedge B)$, $(A\rightarrow B)$ are statement forms.

Two statement forms are logically equivalent if they determine the same Boolean function.
A logically implies B if $(A\rightarrow B)$ is a tautology.

Definition: Argument Form
An argument form is a finite set of statement forms $A_1,\ldots,A_n$. Here $A_1,\ldots,A_{n-1}$ are called premises and A_n is called the conclusion.

Definition: Valid
An argument $A_1,\ldots,A_n$ is valid if for every set of values of statment variables such that $A_1,\ldots,A_{n-1}$ are all true, A_n is also true. Equivalently, $A_1\wedge\cdots\wedge A_{n-1}$ logically implies A_n. Equivalently, $A_1\wedge\cdots\wedge A_{n-1}\rightarrow A_n$ is a tautology.

Examples
p, $p\rightarrow q$, therefore q is a valid argument.
q, $p\rightarrow q$, therefore p is an invalid argument.

Checking Validity

There are two ways to check if an argument is valid.

1. Consturct the truth table for $A_1 \wedge \cdots \wedge A_{n-1} \rightarrow A_n$ and check if it is a tautology.
2. Verify that there are no ways to choose values of statement variables so that A_n is false but $A_1, \ldots, A_{n-1}$ are all true by a direct argument.

Example

$x \rightarrow y$, $y \rightarrow z$, therefore $x \rightarrow z$.

Assume $x \rightarrow z = F$. Then $x = T$, $z = F$. If $x \rightarrow y = T$ then $y = T$. But then $y \rightarrow z = T$. We see that there is no way to choose x, y, z such that $x \rightarrow z = F$ but $x \rightarrow y = T$ and $y \rightarrow z = T$.

Formal Statement Calculus

The Formal System *L*

Set of Symbols

P_1, P_2, etc., (,), $\rightarrow$, $\sim$.

Well-Formed Formula (WF)

1. Any statement variable is welled formed.
2. A, B are well-formed, $(\sim A)$, $(A \rightarrow B)$ are well-formed.
3. All well-formed fornulae can be obtained by applying (1) and (2) finitely many times.

Axioms

Let A, B be well-formed. There are three axiom schemes:

- (L1) $(A \rightarrow (B \rightarrow A))$.
- (L2) $(A \rightarrow (B \rightarrow C)) \rightarrow ((A \rightarrow B) \rightarrow (A \rightarrow C))$.
- (L3) $((\sim A) \rightarrow (\sim B)) \rightarrow (B \rightarrow A)$.

Rule of Deduction

There is one rule of deduction: Molus Ponens (MP). From A and $A \rightarrow B$ we can conclude B.

Definition: Proof

A proof is a finite sequence $A_1, \ldots, A_n$ of well-formed formulas such that for every i, A_i is either an axiom or the result of application of the rule of deduction to two previous formulas.

The proof is regarded as the proof of A_n.

A_n is regarded as a theorem.

Definition: Theorem

A theorem is something that can be proved.

Remarks

1. Every axiom is a theorem.
2. If $A_1, \ldots, A_n$ is a proof and $m < n$, then $A_1, \ldots, A_m$ is also a proof (of A_m).
3. If A_i is obtained from A_j and A_k by an application of MP, then for some well-formed A and B, $A_j = B$ and A_i

and A_j are A and $A \to B$.

Definition: Deduction
Let Γ be a set of wf's. A well-formed A can be deduced from Γ if there exists a finite set of wf's $A_1, \ldots, A_n$ such that $A_n = A$, and for every i A_i is either an axiom or a formula from Γ or the result of application of MP to previous formulas.

Example
If $\Gamma = \varnothing$, A can be deduced from Γ iff A is a theorem from L.

Notation
$\vdash_L A$ means A is a theorem in L.
$\Gamma \vdash_L A$ means A can be deduced from Γ.

Theorem: Deduction Theorem
If $\Gamma \cup \{A\} \vdash_L B$, then $\Gamma \vdash_L A \to B$.
In particular if $\Gamma = \varnothing$, then if $A \vdash_L B$, then $\vdash_L A \to B$ is a theorem.

Theorem: Hypothetical Syllogism
If $\Gamma \vdash_L A \to B$ and $\Gamma \vdash_L B \to C$, then $\Gamma \vdash_L A \to C$.
In particular, if $\vdash_L A \to B$ and $\vdash_L B \to C$, then $\vdash_L A \to C$.

Lemma
$\vdash_L (\sim A \to A) \to A$ for every wf A.

The Adequacy Theorem for *L*

Definition: Valuation
A valuation is a function v on the set of all wf's in L with values in $\langle T, F \rangle$ that has the following properties:

1. $v(A) \neq v(\tilde{A})$.
2. $v(A \to B) = F$ if and only if $v(A) = T$ and $v(B) = F$.

In particular, very valuation assigns values to all Boolean variables $P_1, P_2, \ldots$ in an arbitrary way. Once values are assigned to $P_1, P_2, \ldots$, there is no further freedom to assigning values to more complicated wf's.
Informally, $v(A)$ can be determined by substituting vales of all Boolean variables in A given by v into the Boolean function described by A.

Definition: Tautology
A wf A is a tautology if $v(A) = T$ for all valuations.

Theorem: Soundness Theorem
All theorems in L are tautologies.

Definition: Extension
A formal system L^* is an extension of a formal system L (or more generally L_0) if L^* and L have the same language and rules of deduction, but the set of axioms of L (or L_0) is altered in some way so that all theorems of L (or L_0)

remains theorems in L^*.
Note: L^* can have new theorems that are not theorems of L (or L_0).
Note: Altered means we are allowed to add axioms or replace sets of axioms by other sets of wf's which will be new axioms.

Definition: Consistency
An extension L^* of L is called consistent if there is no wf A such that both A and $(\sim A)$ are theorems of L^*. If there exists A such that $\vdash_{L^*} A$ and $\vdash_{L^*} (\sim A)$, then L^* is inconsistent.

Theorem
L is consistent.

Observation
If L^* is consistent, then there exists a wf which is not a theorem of L^*. In other words, either one of A or $(\sim A)$ is not a theorem.

Theorem
If L^* is inconsistent, then every wf of L^* is a theorem.

Theorem
L is consistent if and only if there exists a wf A which is not a theorem of L.

Theorem
Let L^* be a consistent extension of L. Assume A is a wf such that $(\sim A)$ is not a theorem in L^*. then if we add A to the set of axioms of L^*, then the resulting extension L^{**} of L will be consistent.

Lemma
Assume that A is not a theorem of an extension L^* of L. Then we can add $(\sim A)$ to the set of axioms of L^* and the result L^{**} will be a consistent extension of L^* (and L).

Definition: Complete Extension
An extension L^* of L is complete if for every wf A either A or $(\sim A)$ is a theorem.

Theorem
There exists a complete consistent extension of L.

Lemma
If L^* is a consistent extension of L, then there exists a valuation v such that $v(A)=T$ for all theorems A of L^*.

Theorem: Adequacy Theorem for *L*
Every tautology is a theorem of L.

Theorem
L is decidable. That is, there exists an effective procedure that verifies whether or not any given wf A is a theorem of L.

Informal Predicate Calculus

Predicates and Quantifiers

Set of Symbols

- $\rightarrow$, $\sim$, (,), “,”
- Constants: $a_1, a_2, \ldots$.
- Variables: $x_1, x_2, \ldots$.
- Functions: f_i^k where $k = 1, 2, \ldots$ is the number of variables, and $i = 1, 2, \ldots$ is the index.
- Predicates: A_i^k where $k = 1, 2, \ldots$ is the number of variables, and $i = 1, 2, \ldots$ is the index.
- Quantifiers: $\forall$ (universal quantifier), $\exists$ (existential quantifier).

First Order Languages

Definition: Term

1. Variables and constants are terms.
2. If $t_1, \ldots, t_n$ are terms and f_i^n is a functional letter, then $f_i^n(t_1, \ldots, t_n)$ is a term.
3. All terms are obtained by application of 1 and 2.

Definition: Well-Formed Formulas

Let an atomic formula be $A_i^n(t_1, \ldots, t_n)$.

1. Atomic formulas are well-formed formulas.
2. If A and B are well-formed formulas, then $(\sim A)$, $(A \rightarrow B)$, $(\forall x_i)A$ are well-formed formulas.

Definition: Interpretation

An interpretation I can be obtained by:

1. Choosing a set D_I such that $x_i \in D_I$, $a_i \in D_I$.
2. Choosing specific functions $f_i^k : {D_I}^n \rightarrow D_I$, $k = 1, \ldots, n$.
3. Choosing a predicate $A_i^n : {D_I}^n \rightarrow \langle T, F \rangle$.

Note: Let F be a wf with no quantifier and let $x_1, \ldots, x_n$ enter F . When is F valid in an interpretation I ? It is if F is valid for all possible values of x_i .

Definition: Scope

Consider $(\forall x_i)A$. A is called the scope of a quantifier $(\forall x_i)$.

Definition: Bound, Free

An appearance of a variable x_i is called bound if it is either in $(\forall x_i)$ or in the scope of $(\forall x_i)$. If an appearance of x_i is not bound, it is called free.

Definition: Free Term

A term t is free for x_i if x_i does not occur free within the scope of any quantifier $(\forall x_j)$ where x_j is a variable that enters t .

Satisfaction, Truth

Definition: Valuation
A valuation v is a function on the set of all terms with values in D_I such that
1. $v(a_i)=\bar{a}_i$.
2. $v(f_i^n(t_1,\ldots,t_n))=\bar{f}_i^n(v(t_1),\ldots,v(t_n))$

Informally, a valuation assigns values in D_I to all variables.

Definition: *i*-Equivalent
Two valuations v_1 and v_2 are i-equivalent if $v_1(x_n)=v_2(x_n)$ for every $n\neq i$.

Definition: Satisfaction
A valuation v satisfies a well-formed A if
1. If A is an atomic formula $A_i^n(t_1,\ldots,t_n)$, v satisfies A if and only if $\bar{A}_i^n(v(t_1),\ldots,v(t_n))$ is true.
2. v satisfies $(\sim A)$ if an only if v does not satisfy A.
3. v satisfies $(A\rightarrow B)$ if an only if v satisfies B or v does not satisfy A.
4. v satisfies $(\forall x_i)A$ if an only if every i-equivalent valuation v_i satisfies A.

Definition: True, False
If every valuation in I satisfies A, we say A is true in I, denoted $I\models A$. If no valuation in I satisfies A, we say A is false, denoted $I\models(\sim A)$.

Definition: Closure
A formula A is closed iff no variable occurs free in A.
Note: If A is a formula where $x_{i_1},\ldots,x_{i_k}$ occur free, then $(\forall x_{i_1})\cdots(\forall x_{i_k})A$ is a closure of A.

Theorem
If A is a well-formed and I is an interpretation, then $I\models A$ if and only if $I\models(\forall x_i)A$.

Corollary
1. If $I\models A$, then the closure of A is true in I.
2. If A is not true in I, then the closure of A is false in I.

Remark
Every closed formula is either true or false in I.

Theorem
A valuation v satisfies $(\exists x_i)A$ if and only if there exists a valuation v^i i-equivalent to v that satisfies A.

Definition: Tautology
Let A be a well-formed formula of L (the formal statement calculus) that involves $p_{i_1},\ldots,p_{i_k}$. Let $A_{i_1},\ldots,A_{i_k}$ be well-formed formulas in the predicate calculus. Substitute A_{i_1} for p_{i_1}, A_{i_2} for p_{i_2}, etc. The resulting formula F will be a well-formed formula in the predicate calculus; it will be called a substitution instance of A. If A is a tautology in L, F will be also called a tautology.

Definition: Logically Valid
A formula is logically valid if it is true in any interpretation.

Theorem
Tautologies are logically valid.

Remark
There are logically valid formulas that are not tautologies.

Formal Predicate Calculus

The Formal System *K*

Axioms of Formal Predicate Calculus

(K1): $A \to (B \to A)$.
(K2): $(A \to (B \to C)) \to ((A \to B) \to (A \to C))$.
(K3): $(\sim B \to \sim A) \to (A \to B)$.
(K4): $(\forall x_i) A \to A$ if x_i does not occur free in A.
(K5): $(\forall x_i) A(x_i) \to A(t)$ if t is a term free for x_i in A.
(K6): $(\forall x_i)(A \to B) \to (A \to (\forall x_i) B)$ if A contains no free occurrences of x_i.

Rules of Deduction

1. MP: Molus Ponens.
2. Gen: Generalization. For A, we can conclude $(\forall x_i) A$.

Definition: Proof

A proof is a finite sequence of wf's of K where each formula is either an axiom, the result of an application of MP to two previous formulas, or the result of an application of Gen to a previous formula.

Definition: Deduction

A deduction from a set Γ is a finite sequence of formulas such that each of them is either an axiom, a formula from Γ, or the result of an application of MP or Gen.

Theorem

All axioms are logically valid.

Theorem: Soundness Theorem

All theorems of K are logically valid.

Theorem: Deduction Theorem

Assume that $\Gamma \cup \{A\} \vdash_K B$ and the deduction does not involve using Gen with respect to variables that occur free in A. Then $\Gamma \vdash_K A \to B$.

Corollary

If A is closed (all variables bound), then $\Gamma \cup \{A\} \vdash_K B \Rightarrow \Gamma \vdash_K A \to B$.

Theorem

K is consistent. That is, there is no wf A such that $\vdash_K A$ and $\vdash_K \sim A$.

Equivalence, Substitution

Proposition
$\vdash_K A \leftrightarrow B$ if and only if $\vdash_K A \rightarrow B$ and $\vdash_K B \rightarrow A$. Here $A \leftrightarrow B = \sim((A \rightarrow B) \rightarrow \sim(B \rightarrow A))$.

Definition: Provably Equivalent
A and B are provably equivalent if $\vdash_K A \leftrightarrow B$.

Proposition
Let A be a wf formula whose free variables are $y_1, \ldots, y_n$. Then $\vdash_K A$ if and only if $\vdash_K (\forall y_1) \cdots (\forall y_n) A$. Here $A' = (\forall y_1) \cdots (\forall y_n) A$ is the universal closure of A.

Proposition
Let A and B be wf's. Suppose that B_0 from a wf A_0 by substituting B for one or more occurrences of A in A_0. Then $\vdash_K (A \leftrightarrow B)' \rightarrow (A_0 \leftrightarrow B_0)$.

Corollary
If $\vdash_K A \leftrightarrow B$, then $\vdash_K (A_0 \leftrightarrow B_0)$.

Corollary
Assume that x_j does not appear free or bound in $A(x_i)$. Let B_0 arise from A_0 by replacing one or more occurrences of $(\forall x_i) A(x_i)$ by $(\forall x_j) A(x_j)$. Then $\vdash_K (A_0 \leftrightarrow B_0)$.

Prenex Form

Definition: Prenex Form
Let A be a wf formula. A formula A_0 is called a prenex form of A if:

1. A_0 is provably equivalent to A.
2. $A_0 = (Q_{i_1} x_{i_1}) \cdots (Q_{i_n} x_{i_n}) B$ where every Q_i is either a $\forall$ or $\exists$ and B is a wf that does not involve any quantifiers.

Theorem
If x_i does not occur free in A, then

- $\vdash_K (\forall x_i)(A \rightarrow B) \leftrightarrow (A \rightarrow (\forall x_i) B)$ and
- $\vdash_K (\exists x_i)(A \rightarrow B) \leftrightarrow (A \rightarrow (\exists x_i) B)$.

If x_i does not occur free in B, then

- $\vdash_K (\forall x_i)(A \rightarrow B) \leftrightarrow ((\exists x_i) A \rightarrow B)$ and
- $\vdash_K (\exists x_i)(A \rightarrow B) \leftrightarrow ((\forall x_i) A \rightarrow B)$.

Remark
$\sim (Q_1 x_{i_1}) \cdots (Q_n x_{i_n}) B$ is provably equivalent to $(Q_1^* x_{i_1}) \cdots (Q_n^* x_{i_n})(\sim B)$ where $\forall^* = \exists$ and $\exists^* = \forall$.

Adequacy Theorem for *K*

Theorem
If A is a closed formula and $(\sim A)$ is not a theorem of K^+ (a consistent extension of K), then one can add A to the list of axioms of K^+ and the result K^{++} will be a consistent extension of K^+.

Theorem
Let K^+ be a consistent extension of K. Then there exists a complete extension.

Theorem
Let K^+ be a complete consistent extension of K. Then there exists an interpretation where all theorems of K^+ are true and every true wf in this interpretation is a theorem of K^+.

Theorem: Adequacy Theorem for *K*
All logically valid formulas are theorems of K.

Definition: First Order System
Any consistent extension of K is called a first order system.

Definition
Let A be a wf in prenex form. Let n denote the number of blocks of quantifiers.

- If the first quantifier is $\forall$, the formula is in Π_n.
- If the first quantifier is $\exists$, the formula is in Σ_n.

Models

Definition: Model
1. Let Γ be a set of formulas. An interpretation such that all formulas of Γ are true is called a model for Γ.
2. If K^* is a first order system, then an interpretation where all theorems of K^* are true is called a model of K^*.

Remark
Every first order system has a model.

Theorem: Löwenheim-Skolem Theorem
Every first order system has a countable model.

Theorem
If all axioms of a first order system K^* are true in its interpretation I, then I is a model of K^*.

Theorem
If a first order system K^* is not complete, then it has more than one model.

Theorem
If K^* is inconsistent, then it does not have a model.

Theorem: Compactness Theorem
If every finite subset of a set of axioms of K^* (an extension of K) has a model, then K^* has a model.

Remark
If K^* is a complete first order system, then for every model I of K^*, all true formulas are theorems of K^*.

Theorem
If K^* is a first order system and a formula A is true in every model I of K^*, then A is a theorem

First Order System With Equality

Definition: Normal Models
Models where A_1^2 is interpreted as "=" are called normal models.

Axioms of Normal Models
In addition to (K1) to (K6), normals modes have three additional axioms:

- (E7): $A_1^2(x_1, x_1)$.
- (E8): $A_1^2(t_i, u) \to A_1^2(f_k^j(t_1, \ldots, t_i, \ldots, t_k), f_k^j(t_1, \ldots, u, \ldots, t_k))$.
- (E9): $A_1^2(t_i, u) \to \left(A_k^j(t_1, \ldots, t_i, \ldots, t_k) \to A_k^j(t_1, \ldots, u, \ldots, t_k)\right)$.

(K1) to (K6) and (E7) to (E9) gives first order predicate calculus with equality.

Theorem
The following are theorems in first order predicate calculus with equality:

1. $(\forall x_1) A_1^2(x_1, x_1)$.
2. $(\forall x_1)(\forall x_2)(A_1^2(x_1, x_2) \to A_1^2(x_2, x_1))$.
3. $(\forall x_1)(\forall x_2)(\forall x_3)(A_1^2(x_1, x_2) \to (A_1^2(x_2, x_3) \to A_1^2(x_1, x_3)))$.

Remark
In every model of first order predicate calculus with equality, A_1^2 is interpreted by a relation $\overline{A_1^2}$ that is reflexive (1), symmetric (2), and transitive (3).
Assume we have a set S and a binary relation that is reflexive, symmetric, and transitive. Then the set can be partitioned in to equivalence classes with respect to this relation $\sim$. Let $E(x_i) = \{y \in S | x_i \ y\}$ be the equivalence class of x_i. Then for every x_1 and x_2, either $E(x_1) \cap E(x_2) = \emptyset$ or $E(x_1) = E(x_2)$. Also $S =$ union of disjoint equivalence classes.

Claim
One can replace I with a new interpretation $\overline{I}$ where $D_{\overline{I}} = \overline{D_I}$ (the set of equivalence classes of elements of D_I). Then $\overline{A_1^2}$ on $D_{\overline{I}}$ will be just =.

Theorem
A first order system with equality has a normal model (i.e. A_1^2 interpreted as "=").

Mathematical Systems

Definition
$(\exists_1 x_i)A(x_1)$ means $(\exists x_i)(A(x_1)\wedge(\forall x_j)(A(x_j)\rightarrow(x_i=x_j)))$, i.e. there exists unique x_i such that $A(x_i)$ is true).

Peano System of Axioms for Arithmetic

The Peano System
- $D_I=\{0,1,\ldots\}$.
- $a_1=0$.
- The successor function $\bar{f}_1^1(x)=x+1=x'$.
- $\bar{f}_1^2(x_1,x_2)=x_1+x_2$.
- $\bar{f}_2^2(x_1,x_2)=x_1\times x_2$

Axioms
We have (K1) to (K6) and (E7) to (E9), and additionally,
(N1): $(\forall x_1)\sim(f_1^1(x_1)=a_1)$.
(N2): $(\forall x_1)(\forall x_2)(f_1^1(x_1)=f_1^1(x_2)\rightarrow x_1=x_2)$.
(N3): $(\forall x_1)(f_1^2(x_1,a_1)=x_1)$.
(N4): $(\forall x_1)(\forall x_2)(f_1^2(x_1,f_1^1(x_2))=f_1^1(f_1^2(x_1,x_2)))$.
(N5): $(\forall x_1)(f_2^2(x_1,a_1)=a_1)$.
(N6): $(\forall x_1)(\forall x_2)(f_1^2(x_1,f_1^1(x_2))=f_1^2(f_2^2(x_1,x_2),x_1))$.
(N7): For every wf $A(x_1)$ where x_1 enters free, $A(a_1)\rightarrow((\forall x_1)(A(x_1)\rightarrow A(f_1^1(x_1)))\rightarrow(\forall x_1)A(x_1))$.

Remark
Completeness of Peano Arithmetic is equivalent to:
- All closed wf that are true in the arithmetic interpretation are theorems in Peano Arithmetic.
- $\mathbb{N}$ to be the only interpretation of the Peano Arithmetic.

Theorem: Gödel Incompleteness Theorem
Peano Arithmetic is not complete.

Peano Induction Principle
Let A be a set of natural numbers. If A contains 0, and for every k if A contains k then A contains $k+1$, then $A=\{0,1,2,\ldots\}=\mathbb{N}$.
Note: This is applicable to much more than (N7).
Note: If one accepts this, then can prove any model of Peano Arithmetic coincides with $\mathbb{N}$. Contraction to Gödel Incompleteness Theorem.

Set Theory

Zermelo-Fraenkel System of Axioms
$A_2^2(x_1,x_2)$ is interpreted as $x_1\in x_2$.
We have (K1) to (K6) and (E7) to (E9), and additionally,
- (ZF1): Axiom of Extensionality. Sets are equal if they have the same elements.
  $(\forall x_1)(\forall x_2)(x_1=x_2\leftrightarrow(\forall x_3)(x_3\in x_1\leftrightarrow x_3\in x_2))$.
- (ZF2): Nul Set Axiom. There exists the empty set. $(\exists x_1)(\forall x_2)\sim(x_2\in x_1)$.

- (ZF3): Axiom of Paring. Given sets x and y, we can form a set $z=\{x, y\}$. $(\forall x_1)(\forall x_2)(\exists x_3)(\forall x_4)(x_4\in x_3\leftrightarrow(x_4=x_1\vee x_4=x_2))$.
- (ZF4): Axiom of Unions. Given a set x whose elements are sets, there exists a set y which is a union of all elements of x. $(\forall x_1)(\exists x_2)(\forall x_3)(x_3\in x_2\leftrightarrow(\exists x_4)(x_4\in x_1\wedge x_3\in x_4))$.
- (ZF5): Power Set Axiom. If x is a set, there exists the set of all subsets of x. $(\forall x_1)(\exists x_2)(\forall x_3)(x_3\in x_1\leftrightarrow x_3\subset x_1)$.
- (ZF6): Power Scheme of Replacement. If A is a set and $f:A\to B$ is a function, then $f(A)\subset B$. Let $A(x_1, x_2)$ be a wf where x_1 and x_2 occur free, then $(\forall x_1)(\exists_1 x_2)\,A(x_1, x_2)\to(\forall x_3)(\exists x_4)(\forall x_5)(x_5\in x_4\leftrightarrow(\exists x_6)(x_6\in x_3\wedge A(x_6, x_5)))$.
- (ZF7): Axiom of Infinity. There are infinite sets. $(\exists x_1)(\varnothing\in x_1\wedge(\forall x_2)(x_2\in x_1\to x_2\cup\{x_2\}\in x_1))$.
- (ZF8): Axiom of Foundation. Every nonempty set has an element disjoint from itself (no set can be its own element). $(\forall x_1)(\sim(x_1=\varnothing)\to(\exists x_2)(x_2\in x_1\wedge\sim(\exists x_3)(x_3\in x_2\wedge x_3\in x_1)))$.

Axiom of Choice

(AC): Let x be a set of non-empty sets. There exists a set y such that

1. every element of y is an element of one of the elements of x
2. y has only one element in common with every element of x.

That is, if $x=\{x_\alpha\}$ where each $x_\alpha\neq\varnothing$, then y contains exactly one element from every x_α.
Note: ZF + AC = ZFC

Continuum Hypothesis

Is there a set C such that $\mathbb{N}\subset C\subset\mathbb{R}$ and no 1-1, onto functions $f:\mathbb{N}\to C$, $g:C\to\mathbb{R}$?
(CH): No.

Theorem: Gödel-Cohen

Neither (CH) nor ~(CH) are theorems in ZFC.

Computability, Unsolvability, Undecidability

Algorithms and Computability

Definition: Dominance

Consider functions $f:\mathbb{N}\to\mathbb{N}$ and $g:\mathbb{N}\to\mathbb{N}$. We say that f dominates g if for all sufficiently large n, $f(n)>g(n)$.

Definition: Computability (informal)

A function is computable if it can be described by a computer program.

Remark

The set of computable functions $\mathbb{N}\to\mathbb{N}$ is countable. One can enumerate all functions computed by a program of length 1, 2, etc.

The set of all functions $\mathbb{N}\to\mathbb{N}$ form an uncountable set. Consider $\mathbb{N}\to\{0,1\}$. $0.f(1)f(2)f(3)\ldots=\sum f\frac{(n)}{2^n}$ is a real number in $[0,1]$ written in binary expansion.

Example: Busy Beaver Function

Consider all valid programs of length $\leq N$ that have the following property: once the program is run, it works for a certain amount of time, and then stops (not all programs have this property). If P is such a program, define $t(P)$ to be its running time, measured in the number of operations. $t(P)$ is a non-negative integer. Let $B(N)=\max\{t(P)|P \text{ a valid program that stops}\}$ be the busy beaver function.

Theorem

$B(N)$ is not computable. Moreover, $B(N)$ dominates every computable function.

Theorem: Turing

There is no computer program that takes any given computer program P and decides whether or not P eventually stops (halts). That is, there is no algorithm solving the halting problem.

Definition: Partial, Total

A function $f: X \to Y$ is called partial if it is actually defined on a subset of X, i.e. $\text{dom}(f) \subset X$. If f is defined on all X, it is called total.

Note: Every partial function is total.

Example

Consider $f_1(\text{a text})=\begin{cases}1 & \text{if text is a valid computer program that stops}\\ \text{undefined} & \text{othewise}\end{cases}$, and

$f_2(\text{a text})=\begin{cases}1 & \text{if text is a valid computer program that stops}\\ 0 & \text{othewise}\end{cases}$.

There is no way to compute f_2. However, there is an algorithm to computer f_1.

f_1 is a computable partial function. f_2 is a non-computable total function.

Definition: Computable Set

A set $A \subseteq \mathbb{N}$ is a computable set if its characteristic function $\chi_A(n)=\begin{cases}1 & \text{if } n \in A\\ 0 & \text{if } n \notin A\end{cases}$ is computable.

Definition: Computable Relation

A relation $A=A(x_1,\ldots,x_n)$ is computable iff $\varphi_A(x_1,\ldots,x_n)=\begin{cases}1 & \text{if } A(x_1,\ldots,x_n)=\text{true}\\ 0 & \text{if } A(x_1,\ldots,x_n)=\text{false}\end{cases}$ is computable.

Turing Machines

Definition: Turing Machine

A Turing machine:

- Has an infinite type with cells such that at any time only finitely many cells are empty.
- Has an alphabet (ex: $\{B, 1\}$),
- Has a scanning head that scans a cell of a tape at any time,
- Can be in a finite number of different states $q_0, q_1, \ldots$.
- Has a set of finite quadruples (ex: $q_0 B R q_1$) that can be regarded as commands.

Remark

Turing machines with just two states but a very large alphabet can compute all functions computable by computers.

Recursive Functions

Definition: Recursive Function

Recursive function is a class of functions $\mathbb{N}^k \to \mathbb{N}$ that includes:

- zero function $z(n)=0$,
- successor function $s(n)=n+1$,
- projection function $p_i^n(x_1,\ldots,x_n)=x_i$,

and is closed with respect to the following three operations:

- Composition: If $h_1(n_1,\ldots,n_k),\ldots,h_m(n_1,\ldots,n_k)$ and $g(n_1,\ldots,n_m)$ are recursive, then $g(h_1(n_1,\ldots,n_k),\ldots,h_m(n_1,\ldots,n_k))$ is recursive.
- Recursion: Let $f(n_1,\ldots,n_k,0)=h(n_1,\ldots,n_k)$ be recursive. For all $n\in\mathbb{N}$, if $f(n_1,\ldots,n_k,n+1)=g(n_1,\ldots,n_k,n,f(n_1,\ldots,n_k))$ where g is a recursive function, then f is recursive.
- Minimalization (Least Number Operation): Let $g(n_1,\ldots,n_k,n)$ be recursive such that for every $n_1,\ldots,n_k$ there exists $n\in\mathbb{N}$ such that $g(n_1,\ldots,n_k,n)=0$. Let $\mu g(n_1,\ldots,n_k)=\min\{n|g(n_1,\ldots,n_k,n)=0\}$. Then μg is recursive.

A function is recursive iff it can be obtained from the three basic recursive functions by means of a finite sequence of operations.

Definition: Primitive Recursive

Functions that can be deduced from basic recursive functions using only composition and recursion are called primitive recursive.

Definition: Recursive Set, Recursive Relation

A set $A\subseteq\mathbb{N}$ is recursive iff its characteristic function $\chi_A(n)=\begin{cases}1 & \text{if } n\in A\\ 0 & \text{if } n\notin A\end{cases}$ is recursive.

A relation $R(x_1,\ldots,x_n)$ is recursive iff $\varphi_A(x_1,\ldots,x_n)=\begin{cases}1 & \text{if } R(x_1,\ldots,x_n)=\text{true}\\ 0 & \text{if } A(x_1,\ldots,x_n)=\text{false}\end{cases}$ is recursive.

Theorem

The class of Turing computable functions coincides with the class of recursive functions.

Lemma

Let $0^{(n)}$ denote $f_1^1(\cdots f_1^1(f_1^1(a_1))\cdots)=0'\cdots'$. Then

1. If $n\neq m$, $\vdash_N\sim(0^{(n)}=0^{(m)})$.
2. If $n=m$, $\vdash_N(0^{(n)}=0^{(m)})$.

Expressibility and Representability

Definition: Expressible

A relation $R(x_1,\ldots,x_n)$ is expressible in Peano Arithmetic iff there exists a wf $A(x_1,\ldots,x_n)$ where $x_1,\ldots,x_n$ occur free such that if:

1. If $R(m_1,\ldots,m_n)$ holds, then $\vdash_N A(0^{(m_1)},\ldots,0^{(m_n)})$.
2. If $R(m_1,\ldots,m_n)$ does not hold, then $\vdash_N\sim A(0^{(m_1)},\ldots,0^{(m_n)})$.

Theorem
$R(x_{1,}x_2)=(x_1=x_2)$ is expressible in Peano Arithmetic.

Definition: Representable
A function $f(x_{1,}\ldots,x_n)$ is representable in Peano Arithmetic iff there exists a wf $A(x_{1,}\ldots,x_n,x_{n+1})$ where $x_{1,}\ldots,x_{n+1}$ occur free such that if:

1. If $k_{n+1}=f(k_{1,}\ldots,k_n)$ holds, then $\vdash_N A(0^{(k_1)},\ldots,0^{(k_n)},0^{(k_{n+1})})$.
2. If $k_{n+1}\neq f(k_{1,}\ldots,k_n)$ does not hold, then $\vdash_N \sim A(0^{(k_1)},\ldots,0^{(k_n)},0^{(k_{n+1})})$.
3. For every $k_{1,}\ldots,k_n$, $\vdash_N (\exists_1 x_{k+1})\,A(0^{(k_1)},\ldots,0^{(k_n)},x_{k+1})$.

Theorem
A relation is representable in Peano Arithmetic if and only if it is computable (= Turing computable, = recursive computable).

Theorem
A function is representable in Peano Arithmetic if and only if it is computable (= Turing computable, = recursive computable).

Gödel Incompleteness Theorem

Theorem
Assume that Peano Arithmetic N is consistent. Then it is not complete.

Corollary
There is a closed wf such that it is true in the standard arithmetic interpretation but it is not a theorem.

Remark
There exists a wf U of N that is interpreted as “I'm not provable”.

Theorem (informal)
Let N^* be a consistent extension of N. Assume that the set of axioms of N^* is recursive. Then N^* is not complete.

Corollary
ZF, ZFC are not complete.

Theorem: Gödel's Second Theorem
Consistency of arithmetic can be express by a wf in N but cannot be proven in N.

Theorem: Tarski's Inexpressibility of Truth
Let $T\subset\mathbb{N}$ be the set of numbers of all true formulas. T is not expressible.

Definition: ω-Consistency
N or its extension is ω-consistent if for every wf A, $\vdash_N A(0^{(n)})$ for every n implies that $\sim(\forall x_1)\,A(x_1)$ is not a

theorem.
Note: ω-consistency is stronger than consistency.

Theorem
If N is ω-consistent, then N is not complete.

Gödel Numbering
Number well-forms as follows:
- Symbols: $g(()=3$, $g())=5$, $g(,)=7$, $g(\sim)=9$, $g(\rightarrow)=11$, $g(\forall)=13$, $g(x_k)=8k+7$, $g(a_k)=8k+9$, $g(f_k^n)=11+8\cdot 2^n\cdot 3^k$, $g(A_k^n)=13+8\cdot 2^n\cdot 3^k$.
- A string with symbols $s_1,\ldots,s_k$: $2^{g(s_1)}3^{g(s_2)}\cdots p_k^{g(s_k)}$ where p_k is the k -th prime.
- A text (or proof) with strings $S_1,\ldots,S_k$: $2^{g(S_1)}3^{g(S_2)}\cdots p_k^{g(S_k)}$ where p_k is the k -th prime.

Define the following well-forms:
$Wf(n)$ holds if and only if n is a Gödel number of a wf.
$Ax(n)$ holds if and only if n is a Gödel number of an axiom
$Prf(n)$ holds if and only if n is a Gödel number of a proof.
$Pf(m,n)$ holds if and only if n is a Gödel number of a proof of the wf with Gödel number m .
$W(m,n)$ holds if and only if m is a Gödel number of a wf $A(x_1)$ in which x_1 occurs free and n is the Gödel number of a proof of $A(0^{(m)})$.

Turing Machines
One can use a modification of Gödel numbering to effectively enumerate all Turing machines.

Definition
A Turing machine T computes a partial function f if $f(n)$ is undefined if T doesn't stop, or stops but the scanning head is not a the beginning of a block of $m+1$ 1's on the otherwise empty tape, and $f(n)=m$ otherwise.

Theorem
The set of all computable partial functions of one variable can be effectively enumerated. In this enumeration, we encounter every function infinitely many times.

Proposition
It is not possible to enumerate all total computable functions.

Theorem
The algorithmic problem "Does the Turing machine halts on input n " is unsolvable.

Universal Turing Machine
$U(m,n)$ computes $T_m(n)$.

Recursive Enumerability
Definition: Recursive Enumerablility

A set $A \subset \mathbb{N}$ is called recursively enumerable if $\chi_A(n) = \begin{cases} 1 \text{ if } n \in A \\ \text{undefined if } n \notin A \end{cases}$ is a computable function.

Theorem
A set is recursively enumerable if and only if it is the domain of a partial computable function if and only if it is the range of a partial computable function.

Theorem
A recursively enumerable set $A \subset \mathbb{N}$ is recursive if and only if its complement $\mathbb{N} - A$ is recursively enumerable.

Theorem
A recursively enumerable set $A \subset \mathbb{N}$ is recursive if and only if it can be effectively enumerated in increasing order (i.e. the range of an increasing partial computable function).