INCOMPLETENESS IN ZFC

VICTOR ZHANG

Abstract. The statements of Gödel's incompleteness theorems are famous in mathematics, concerning the foundations of the field. Whilst most mathematicians have heard of the theorems, few outside of mathematical logic would have gone through the technically sophisticated proofs. Instead of proving the original theorem for the theory of natural numbers and utilizing the notion of consistency, we prove a version of the first incompleteness theorem for the theory of sets and utilize the notion of correctness, thus allowing for a simpler and technically less demanding proof whilst losing none of the philosophical significance of the original. We target an audience of general mathematicians not specializing in mathematical logic, and hope to give a rigorous proof of an important result with relatively less technical demand.

Contents

Date: August 29 2014.

1. Introduction

Even though almost every mathematician should have heard at least the statements of Gödel's incompleteness theorems, it is a less known fact, that there are many versions of incompleteness theorems. Perhaps one of the most well known versions of the the first incompleteness theorem is not Gödel's original theorem as it first appeared, but one modified by Rosser:

Theorem. *If Peano Arithmetic is consistent, then it is incomplete.*

Informally, Peano Arithmetic is a system of first-order logic that formalizes part of the theory of natural numbers. We say a theory is consistent if there are no sentences that are both provable and refutable, and we say that a theory is incomplete if there is a sentence that is undecidable, that is it is neither provable nor refutable. For now, we stick to an informal discussion of these concepts, but we will give precise formulations in the next section of this paper.

The philosophical significance of this theorem is that it establishes the mutual exclusivity of completeness and consistency. Peano Arithmetic is either inconsistent, or it can express a sentence that it cannot prove or disprove. Of the two, the former seems more disastrous to a mathematician, for an inconsistent theory is able to prove any sentence, including a contradiction, i.e. both a sentence and its negation, or equally worse, something that is not true. So if we want to assume that Peano Arithmetic is consistent, we must accept that at the very least it is incomplete, and hence the incompleteness theorem outlines some of the fundamental limitations of Peano Arithmetic.

The implications of this theorem on the limitations of Peano Arithmetic however, actually extend far beyond Peano Arithmetic, for the theorem also applies to any theory that is able to express a version of Peano Arithmetic, including Zermelo-Fraenkel set theory with the Axiom of Choice (ZFC) that essentially serves as the foundation of almost all branches of modern mathematics outside of mathematical logic. Gödel's first incompleteness theorem thus arguably demonstrates the limitations of not just the theory of natural numbers, but almost all of conventional mathematics.

Another version of the first incompleteness theorem more faithful to Gödel's original work is:

Theorem. *If Peano Arithmetic is $\omega-$consistent, then it is incomplete.*

The term $'\omega-$consistency$'$, meaning that if a formula $F(n)$ is provable for every natural number n then the universal formula $\forall n F(n)$ is also provable, is a technical condition specific to Peano Arithmetic not important to this paper, so we skip a detailed exposition of it. The point however, is to illustrate that there really are multiple versions of the first incompleteness theorem.

In this paper, we prove one more version of the first incompleteness theorem:

Theorem. *If ZFC is correct, then it is incomplete.*

Here we say a theory is correct if every provable sentence is true and every refutable sentence is false. There are two important points to make about this particular theorem compared to the earlier ones.

First is the requirement of correctness in the hypothesis instead of the requirement of consistency. Correctness is a stronger requirement because if every provable sentence is true and every refutable sentence is false, then no sentence is both provable and refutable. For if there was a sentence both provable and refutable, it would be both true and false and we define the set of false sentences to be the set of sentences that are not true.

Even though correctness is a stronger requirement, we demonstrate it allows for a significantly simpler proof than one based on consistency, because consistency is based solely on provability and refutability, and we think of a provable sentence as one that has a proof, and all proofs must be finite in length. Informally, needing to keep track of the finiteness induced by proofs leads to some technical details regarding the constructibility of sentences. For example, the Gödel-Rosser version of the proof based on consistency required the ability for the theory to sustain the development of a combinatorial enumeration of strings of length less than the length of the undecidable sentence being constructed.

Such a task was made even harder by the fact that it was only allowed to operate within the language of natural numbers. That brings us to the second point about our theorem, which is that we consider ZFC instead of Peano Arithmetic. Even though a proof of the theorem for Peano Arithmetic would already imply the result for ZFC, informally because set theory contains number theory as s subsystem, a direct proof in ZFC has fewer technical requirements because ZFC is able to express certain concepts more concisely and clearly than Peano Arithmetic.

For those with experience with Gödel's original proof, the most notable apparent technical simplifications would be due to be our "version" of Gödel numbering, a tool Gödel developed for his original proof. A Gödel numbering in a theory is an injection from the set of finite words, to the set of natural numbers. Gödel numbering was required to encode sentences as numbers, so that Peano Arithmetic, being a theory of numbers, may indirectly talk about its own sentences. These issues are sidestepped in a theory of sets such as ZFC, because virtually any mathematical object, including systems of sentences, can already be thought of as sets, so with proper definitions, ZFC can talk about its own sentences without such encoding.

These two modifications do not reduce the impact of the theorem. First correctness is no less important than consistency, for failure of correctness of a theory means that in the theory, it is possible to prove a false statement or refute a true statement. It still implies the theory does not captue the mathematical reality one would expect. Second, arguably the real impact of the original incompleteness theorems in Peano Arithmetic actually derive from their implications in ZFC, given the importance of ZFC and similar formal systems in the foundations of modern mathematics.

The aim of this paper is to give a clear and simple proof of a version of the first incompleteness theorem with as little technical difficulty as possible, so that it is most easily appreciable to those outside mathematical logic, while not diluting the significance of the theorem. Of course a proof of the first incompleteness theorem will still be lengthy and contain technical parts, but we strove to isolate the more technical details so the reader can identify them easily and enjoy the conceptual parts independently.

The same approach in our proof will carry over naturally to other similarly expressive formal systems, with the main differences being in the technical customization to cater to the particularities of those formal systems.

An example of this segregation is given in the structure of this paper, which shall be as follows. We first give a proof of an abstract version of our first incompleteness theorem that outlines the general idea of the theorem and conditions for a theory to be incomplete given it is correct, giving the reader an incompleteness theorem in around two pages. We then spend the bulk of the paper constructing the theory of ZFC rigorously and applying the abstract theorem to get the main result.

Finally, as a supplement at the end, we give a abstract version of the first incompleteness theorem based on consistency, i.e. the Gödel-Rosser version. The significance of this version of the first incompleteness theorem is that regardless of whether or not we believe ZFC to be correct, consistency alone already guarantees its incompletness.

We note that while the subject at hand is axiomatic set theory, working knowledge of naive set theory should suffice, and any further expertise would only help. Additionally, we assume working knowledge of propositional and first order logic.

2. Abstract First Incompleteness theorem

Definition 1. A **language system** $\mathcal{L}$ is an ordered tuple $(\mathcal{E},\mathcal{S},\mathcal{P},\mathcal{R},\mathcal{H},\mathcal{T},\phi)$ consisting of the following: the set of **expressions** $\mathcal{E}$, the set of **sentences** $\mathcal{S}$, the set of **provable sentences** $\mathcal{P}$, the set of **refutable sentences** $\mathcal{R}$, the set of **predicates** $\mathcal{H}$, the set of **true sentences** $\mathcal{T}$, and a **predication function** $\phi : \mathcal{H} \times \mathcal{E} \to \mathcal{E}$, all of which fulfill the following properties:

1. $\mathcal{S} \subset \mathcal{E}$, $\mathcal{H} \subset \mathcal{E}$ i.e. sentences and predicates are expressions.
2. $\mathcal{P} \subset \mathcal{S}$,$\mathcal{R} \subset \mathcal{S}$, $\mathcal{T} \subset \mathcal{S}$, i.e. provable sentences, refutable sentences, and true sentences are all sentences.
3. For every $H \in \mathcal{H}$ and every $E \in \mathcal{E}$, we have $\phi(H,E) \in \mathcal{S}$. We write $H(E)$ to denote $\phi(H,E)$.

As suggested by the notation in condition 3, we can think of each predicate H as a function defined on $\mathcal{E}$, so that for every $E \in \mathcal{E}$ the image $H(E) = \phi(H,E)$ must be a sentence. Informally, we can think of a predicate H as encoding a formal property, which takes as an input an expression E and outputs the sentence $H(E)$ that can be thought of as "E has the property described by H".

Instead of saying 'S is a provable sentence', we simply say 'S is provable'. We likewise employ the similar abbreviation'E is refutable. Additionally, we say a sentence S is **false** if it is not true, i.e. $S \in (\mathcal{S} - \mathcal{T}) = \{S \in \mathcal{S} : S \notin \mathcal{T}\}$.

Definition 2. Let $E \in \mathcal{H}$. The **diagonal** of E is the expression $E(E)$, and the **diagonal function** $d\colon \mathcal{H} \to \mathcal{S}$ is the function which takes every $E \in \mathcal{E}$ to $E(E)$.

Similarly, for a set of predicates $A \subset \mathcal{H}$, the diagonal of A is the image of A under d, i.e. $d(A) = \{d(E)\colon E \in A\}$

Additionally, for a set A of sentences, we define the **pre-image** of the diagonal of A to be $A^* = d^{-1}(A) = \{E \in \mathcal{H} : d(E) \in A\}$.

By definition, A^* is the set of predicates whose diagonals are in A. If we let A be the set of provable and refutable sentences respectively and let H be a predicate, we have by definition $H(H)$ is provable if and only if $H \in \mathcal{P}^*$, and $H(H)$ is refutable if and only if $H(H) \in R^*$.

Definition 3. 1. A set $A \subset E$ is **expressible** if there exists a predicate H such that $H(E)$ is true if and only if $E \in A$. We say H expresses A.

2. If E is an expression such that $H(E)$ is true, we say E **satisfies** H.

This is an important definition, which aims to capture formally the colloquial meaning of 'expressing' something in a language system. It is clear that every predicate expresses some set, for given a predicate H we can just take the set of those expressions that turn into true sentences when described by a predicate, i.e. $\{E \in \mathcal{E} : H(E) \in \mathcal{T}\}$. On the other hand, a set being expressible just means that the converse holds, we can find a predicate that becomes a true sentence exactly when describing members of that set.

Example 4. We wish to express the notion of provability in $\mathcal{L}$. To do so, we must find a predicate H such that $H(E)$ is true if and only if $E \in \mathcal{P}$, i.e. if and only if E is provable. Informally, this means the sentence $H(E)$ must be saying'E fulfills the property of provability, or'E is provable'. If it happens that E is not provable, then these sentences will be false, so they are true if and only if E is actually provable.

In the version of the first incompleteness theorem we prove, the notion of correctness is needed in the hypothesis and the notion of incompleteness is needed in the conclusion.

Definition 5. Let $\mathcal{L}$ be a language system.

1. $\mathcal{L}$ is **correct** if every provable sentence is true, and every refutable sentence is false, i.e $\mathcal{P} \subset \mathcal{T}$ and $\mathcal{R} \subset (\mathcal{S} - \mathcal{T})$.
2. $\mathcal{L}$ is **incomplete** if there exists a sentence that is neither provable nor refutable, i.e. $\mathcal{P} \cup \mathcal{R} \neq \mathcal{S}$.
3. If a sentence G that is neither provable nor refutable, we say G is **undecidable**.

Hence in proving incompleteness in $\mathcal{L}$, it suffices to show that there is an undecidable sentence. Additionally, if $\mathcal{L}$ is correct, it is sufficient to show that there exists a true sentence that is not provable. We will prove this small fact and use it in our first incompleteness theorem.

Lemma 6. *If $\mathcal{L}$ is correct and there is a sentence G in $\mathcal{L}$ that is true but not provable, then G is also undecidable (and hence $\mathcal{L}$ is incomplete).*

Proof. Suppose $\mathcal{L}$ is correct and G is true but not provable. Suppose moreover G is refutable. Then by the assumption that $\mathcal{L}$ is correct, G is false, a contradiction because G is also true, and by definition the set of false sentences is the complement of the set of true sentences in $\mathcal{S}$. Hence G is not refutable. So G is neither provable nor refutable, hence it is undecidable. □

Now we state and prove a version of the first incompleteness theorem. Remarkably, so far we have said nothing specific about the language system $\mathcal{L}$. All that was required for us to develop a first incompleteness theorem, is a very limited and basic language system, which essentially is just a collection of sets and subsets and a predication function. For the statement of the proof, let $\mathcal{P}' = \{S \in \mathcal{S} : S \notin \mathcal{P}\}$ be the complement of $\mathcal{P}$ in $\mathcal{S}$. This means that if a sentence S is in $\mathcal{P}'$, it is by definition not provable, i.e. $\mathcal{P}'$ is the set of non-provable sentences.

Theorem 7. *In language system $\mathcal{L}$, if $(\mathcal{P}')^*$ is expressible and $\mathcal{L}$ is correct, then $\mathcal{L}$ is incomplete.*

Proof. Assume $\mathcal{L}$ is correct and let H be a predicate expressing $(\mathcal{P}')^*$. We show that the sentence $H(H)$ is undecidable. Since H expresses $(\mathcal{P}')^*$, $H(H)$ is true if and only if $H \in (\mathcal{P}')^*$. By the definition of the diagonal function, $H \in (\mathcal{P}')^*$ if and only if $H(H) \in \mathcal{P}'$. Combining the two equivalences, we have that $H(H)$ is true if and only if $H(H) \in \mathcal{P}'$, i.e. $H(H)$ is true if and only if it is not provable. We now just need to show $H(H)$ is true, then it would not be provable. Suppose $H(H)$ is false, then $H(H)$ is provable. By the assumption of correctness, this implies $H(H)$ is true, a contradiction. Hence $H(H)$ is true and not provable, hence is undecidable and $\mathcal{L}$ is incomplete. ☐

Now that we have the abstract first incompleteness theorem, all that needs to be done is to define the language system of ZFC, i.e. $(\mathcal{E}, \mathcal{S}, \mathcal{P}, \mathcal{H}, \mathcal{T}, \phi)$, and show that $(\mathcal{P}')^*$, the pre-image of the diagonal of the set of non-provable sentences, is expressible in ZFC. For the rest of the paper, we shall call the language system of ZFC the'Axiom system of ZFC' or just'ZFC' when abbreviated. The fact that $(\mathcal{P}')^*$ is expressible shall be proved in the following steps:

Proposition. *1. $\mathcal{P}$ is expressible in ZFC.*

2. For any set of sentences A that is expressible in ZFC, the complement A' is expressible in ZFC.

3. For any set of sentences B that is expressible in ZFC, the pre-image of the diagonal B^* is expressible in ZFC.

From these steps, letting $A = \mathcal{P}$ in the second part of this proposition and $B = \mathcal{P}'$ in the third part, the fact that $(\mathcal{P}')^*$is expressible follows immediately. In the next section, we construct the axiom system of ZFC, and in a further section, we prove the three parts of the proposition above.

Remark 8. The reader may find the construction of $(\mathcal{P}')^*$ abstract and lacking in intuition, so we briefly explain it. Informally, it gives us the set of predicates that are not provable when they describe themselves, i.e. the set of predicates K such that $K(K)$ is not provable. Now if

H expresses $(\mathcal{P}')^*$, for each predicate K, $H(K)$ is saying‘the sentence $K(K)$ is not provable’. What we wish to do is to take H and let it describe itself, so we have $H(H)$ that says‘ the sentence $H(H)$ is not provable, or just $'H(H)$ is not provable or‘I am not provable. That is the commonly used colloquial phrasing of the undecidable sentence, for is must be true and not provable under a correct language system.

3. Construction of The Axiom System of ZFC

3.1. **Preliminaries.** We fist give a set theoretic construction of the natural numbers and ordered tuples for use later.

Definition 9. A **natural number** n is a set defined as follows:

1. 0 is the empty set ϕ.
2. A number n is the set $(n-1) \cup \{(n-1)\}$.

So then $1 = \{0\}$, $2 = \{0, 1\}$, and so forth. Every natural number contains the previous numbers.

We now define a function, which is defined in terms of ordered pairs.

Definition 10. 1. An **ordered pair** (a, b) is the set $\{\{a\}, \{a, b\}\}$.

2. Let A and B bet sets. The set $A \times B$ or the product of A and B is the set $\{(a, b) : a \in A, b \in B\}$

3. A **function** f from a set A to a set B is a subset of $A \times B$ such that if (a, b) and (a, c) are in f then $b = c$, i.e. for every $a \in A$ there exists a unique element $b \in B$ such that $(a, b) \in f$.

This allows us to define an ordered tuple as a set.

Definition 11. An ordered tuple $(a_1, ..., a_n)$ is the function f from $\{1, ..., n\}$ to $\{a_1, ..., a_n\}$ such that $f(n) = a_n$.

3.2. **The Basic Language of ZFC.** In this subsection, we will define what are expressions, predicates and sentences, i,e, the sets $\mathcal{E}$, $\mathcal{H}$, $\mathcal{S}$, $\mathcal{T}$ in ZFC. In particular, we define expressions, predicates and sentences as strings of symbols with different properties. We also clarify the nature of the function ϕ in ZFC, which shall be thought of as the substitution of variables for instances of the variables. In the subsection after this one, we shall define the sets $\mathcal{P}$ and $\mathcal{R}$ and prove that $\mathcal{P}$ is expressible in ZFC.

3.2.1. *Symbols and Expressions:* The language of ZFC shall be constructed from the alphabet set β consisting of the following symbols:

$$\forall \in v \supset' =) \; (\sim$$

Additionally, we define each symbol to be a set according to the table below.

$\forall$	$\in$	v	$\supset$	$'$	$=$	$)$	$($	$\sim$
0	1	2	3	etc...	...	...	...	...

Definition 12. An **expression** E of ZFC is any finite ordered tuple of elements of β.

In writing down an expression $(a_1, ..., a_n)$ where each a_i is an alphabet symbol, we omit the parenthesis and commas, and simply write $a_1 a_2 ... a_{n-1} a_n$.

Every finite ordered tuple is a function, which is a set, and in particular some special subset of the product of the domain and the range. This means all expression are technically sets.

Definition 13. The expressions $(v'), (v''), (v'''), ...$ are called **variables**, and shall be abbreviated by $v_1, v_2, v_3, ...$ respectively in writing down larger expressions that contain them.

We now define the notion of a formula, a special type of expression, in ZFC.

3.2.2. *Formulas.*

Definition 14. An **atomic formula** is any expression of the form $x \in y$ or $x = y$ where x and y are variables. A **formula** is defined inductively as follows:

1. Every atomic formula is a formula.
2. For all formulas F and G, and for any variable x the expressions $\sim (F)$, $(F \supset G)$, $\forall x F$ are formulas.

Here, the symbols $\sim$, $\supset$ and $\forall$ mean‘the negation of’,‘implies’ and‘for all’ respectively.

Definition 15. If a variable x appears in an expression E, we say x **occurs** in E. The concept of occurrence need only be used informally. The notions of **free occurrences** and **bounded occurrences** of a variable in an expression and the notion of a **closed formula** are defined as follows.

1. If E is an atomic formula, and if x is a variable and occurs in E, then we say x has a free occurrence in E.
2. If F and G are formulas, the free occurrences of x in in $(F \supset G)$ are the free occurrences of x in F together with the free occurrences of x in G. The free occurrences of x in $\sim (F)$ is the same as the free occurrences of x in F. If y is a variable distinct from x, the free occurrences of x in $\forall y F$ is the same as the free occurrences of x in F.

3. The occurrences of x in $\forall x F$ are bounded. We do not say that the occurrences in x in $\forall x F$ are free.

4. A closed formula is a formula in which no variable has a free occurrence. An open formula is a formula in which at least one variable has a free occurance.

5. A **free variable** in a formula F is a variable that has a free occurance in F.

This allows us to define:

Definition 16. A **sentence** is a closed formula.

Recall that we defined expressions and alphabet symbols in such a way so that every expression, while written down as ordinary familiar symbols, is actually a pure set, so that we may use the language of ZFC to talk about expressions of ZFC. Now given a formula $F(x)$ that has free occurrences of a variable x and given a pure set M, we want to be able to generate a new expression $F(\overline{M})$ which is equivalent to substituting every occurrence of x with a token $\overline{M}$ that represents the set M. For example, if F is $\forall y(x \in y)$, we want a way of generating an expression $\forall y(\overline{M} \in y)$ that makes sense, since presently the symbol $\overline{M}$ and meaning of $\overline{M} \in y$ is undefined. To do this, we need a way of indirectly referring to a pure set that is an expression using formulas.

3.2.3. *Set-Expressions:* Given an expression E, we will generate a predicate $\overline{E}$ that expresses the singleton set containing E, i.e. the set $\{E\}$. That is, E is the only expression that satisfies $\overline{E}$.

Definition 17. Let $\overline{M}$ be a predicate that expresses the singleton set containing a set M. Let x be a vairable and let v be the free variable of $\overline{M}$.

1. Then the expression $x = \overline{M}$ is defined as $\forall v(v = x \supset \overline{M})$.
2. The expression $x \in \overline{M}$ is defined as $\forall v(x \in v \supset \overline{M})$

Even though $\overline{M}$ is a formula, we can think of it as the element of the singleton set it expresses. Hence $x = \overline{M}$ can be taken informally to just mean that the variable x is equal to the set M, and similarly so for $x \in \overline{M}$.

Definition 18. 1. Let E be an expression. If there exists a formula $\overline{E}$ that expresses the singleton set containing E, the we define $\overline{E}$ as the **set-expression** of E.

2. Let Γdenote the set of all set expressions.

There are multiple set expressions that express the singleton set containing $\overline{E}$, for example as we shall see later in discussing the notion of truth in ZFC, if K is one set expression that does this, then the double negation $\sim (\sim (K))$ also does it. So actually $\overline{E}$ is an equivalence class of formulas defined by the equivalence relation that two formulas are equivalent if they express the same singleton set. We omit a proof that such a relation is actually an equivalence relation. Outside of this paragraph in this the paper, when we write $\overline{E}$ for a set E, we may just take $\overline{E}$ to be a representative of the equivalence class whose elements express the singleton set containing E.

Remark 19. We use the term set-expression because the formula $\overline{E}$ acts as token to be used in writing expressions in place of the set E, such that for any variable x, writing down $x = \overline{E}$ makes sense. In an analogous way, the roman numeral III is a number-expression that acts as a token in place of the number three used to write expressions such as $n =$III. What we want to say is something like'n equals three', but we write III instead of the English language token'three'. In the same way, $\overline{E}$ is treated as a pseudo expression that is used as an element in forming other expressions.

Proposition 20. *For every expression E there exists a predicate $\overline{E}$ that expresses the singleton set containing E. Hence Γ is non empty and indexes the set of all expressions.*

Proof. In order for us to prove this result, we need to define what it means for a sentence to be true in ZFC. However, in order to do that, we need to make some more definitions regarding the substitution of variables for sets, and hence give the proof at the end of this section. ☐

Hence we may treat any expression E as having a corresponding token $\overline{E}$ that is a set-expression from Γthat can be used to represent E in formulas, such as in $x = \overline{E}$.

Remark 21. Our construction of set expressions is such that the only sets we care about are expressions. The reader should note that there are other sets that do not have a corresponding formula that expresses the singleton set containing it. This is because formulas are finite ordered tuples of alphabet symbols, so there are only countably many formulas. There are however, uncountably many sets. One example is the real numbers. For if every real number, taken as constructed as a set, has a corresponding formula, then there would be uncountably many formulas.

3.2.4. *Substitution of variables for sets.*

Definition 22. 1. When we write a formula as $F(v_{i_1}, .., v_{i_n})$ we mean that $v_{i_1}, ..., v_{i_n}$ are the only variables with free occurrences in F. Let

$F(v_{i_1}, ..., v_{i_n})$ be such a formula. If $\overline{m_1}, ..., \overline{m_n}$ are set-expressions, then $F(\overline{m_1}, ..., \overline{m_n})$ is the expression that results from replacing occurrences of v_{i_j} with $\overline{m_j}$ where j ranges from 1 to n. We say that $F(\overline{m_1}, ..., \overline{m_n})$ is an **instance** of $F(v_{i_1}, .., v_{i_n})$.

We will employ the following abbreviations to our language. The left hand side of each line below is defined by the right hand side. F, G, H, are formulas, x is a variable, and m and n are set-expressions or variables. Whenever we may omit parenthesis in expressions without giving rise to ambiguity, we may do so. For example, $(F \supset G)$ may be written as $F \supset G$, and $\sim F$ may be written instead of $\sim (F)$.

$$(F \vee G) \coloneqq (\sim F \supset G)$$
$$(F \wedge G) \coloneqq \sim (F \supset \sim G)$$
$$F \equiv G \coloneqq ((F \supset G) \wedge (F \supset G))$$
$$\exists x F \coloneqq \sim \forall x \sim F$$
$$m \neq n \coloneqq \sim m = n$$
$$(\forall x \in y) F \coloneqq \forall x ((x \in y) \wedge F)$$

We note that the above abbreviations are just formalizations of the standard meanings of $\wedge$, $\vee$, $\equiv$, $\exists$ for the working mathematician as conjunction, disjunction, equivalence and existence quantifier respectively.

Recall that in section 1, we required a language to have a predication function ϕ that associated every predicate H and every expression E with a sentence $H(E)$. ϕ is precisely the substitution of v_1 in $H(v_1)$ with the set-expression of E, so $\phi(H, E) = H(\overline{E})$.

3.2.5. *Substitution of Variables for Variables:*

Definition 23. Let $F(v_1, ..., v_n)$ be a formula of the n variables. For any tuple of variables $v_{k_1}, ..., v_{k_n}$, we define $F(v_{k_1}, ..., v_{k_n})$, or the substitution of the variables $v_{k_1}, ..., v_{k_n}$ for $v_1, ..., v_n$ in F to be the expression such that:

1. If none of the v_{k_1} occur in F as a bounded vairbale, then $F(v_{k_1}, ..., v_{k_n})$ is the result of replacing all free occurrences of v_i for w_i for each $i = 1, .., n$.

2. If there exists a i such that v_{k_i} occurs in F as a bounded variables, then we first take the smallest such i. We the replace every occurrence of v_{k_i} in F with the smallest j such that v_j does not occur in F, and then repeat the process for the next variable $v_{k_{i'}}$that occurs in F as an bounded variable, and so forth until we remain with a formula F' that has no v_{k_1}as a bounded variable. We then apply step 1 and substitute $v_{k_1}, ..., v_{k_n}$ for $v_1, ..., v_n$ in F' and define the result to be $F(v_{i_1}, ..., v_{i_n})$.

Example 24. Consider the formula $v_1 \neq v_2 \supset v_2 \in v_1$ of two free variables v_1 and v_2. The substitution of v_1,v_2 for v_2,v_3 by applying part 2 of the definition results in first in $v_1 \neq v_3 \supset v_3 \in v_1$, and then $v_2 \neq v_3 \supset v_3 \in v_2$ as the final result.

3.2.6. *True Sentences.*

Definition 25. What it means for a sentence to be **true** and for two sentences to be **equivalent** shall be defined inductively as follows.

1. An atomic sentence $m = n$, where m and n are set expressions, is true if the sets represented by m and n are equal. An atomic sentence $m \in n$ is true if the set represented by m is an element of the set represented by n.

2. If X is a sentence, the sentence $\sim X$ is true if X is not true.

3. If Y is another sentence, then $X \supset Y$ is true if either X is not true, or both X and Y are true.

4. If x is a variable, $\forall x F(x)$ is true if for every expression M, $F(\overline{M})$ is true.

5. X and Y are equivalent if they are both true or both false.

6. If $F(v_{i_1}, ..., v_{i_n})$ and $G(v_{i_1}, ..., v_{i_n})$ are open formulas with the same free variables, F and G are equivalent if for all sets $m_1, ...m_k$, the sentences $F(m_1, ..., m_n)$ and $G(m_1, ..., m_n)$ are equivalent.

We have thus defined the notion of an expression and true sentences. We now define the notion of a predicate.

Definition 26. A formula $F(v_{i_1}, ...v_{i_n})$ is **regular** if $i_j = j$. So $F(v_{i_1}, ..., v_{i_n}) = F(v_1, ..., v_n)$ if F is regular. A **predicate** is a regular formula with only one free variable, i.e. it can be written as $H(v_1)$.

We now prove the proposition that every expression E has a set-expression $\overline{E}$.

Proof. We prove this proposition in several steps.

1. The singleton set containing the empty set ϕ is expressible.

The empty set is just the set with no elements. Hence $\{\phi\}$ is expressed by $\forall x(\sim x \in y)$.

2. For every alphabet element K of β, K is expressible.

The first alphabet element is just ϕ, so $\{\phi\}$ is expressed by the same formula as the empty set. Now we construct formulas expressing the next seven alphabet elements inductively. Let $n = 2, ..., 8$. Each alphabet element apart from the first is nonempty and contains the previous. So $\exists x(x \in y) \wedge \forall x(x \in y \supset x = \overline{\phi_{n-1}})$ is the expression expressing the singleton set $\{\phi_n\}$.

3. For every expression $E = E_1 ... E_n$ consisting of of n symbols, where each E_i is an alphabet element in β, the singleton set containing $(E_1, ..., E_n)$ is expressible.

Let $funct(x)$ be an expression saying that the set x is a function. Let $dom_n(x)$ be an expression saying that the domain of x is the first n natural numbers. Let $n(x)$ be an expression saying that x is the first n natural numbers. Now let E be an alphabet element. Let $f_x(y,\, E)$ be an expression saying that the value of y in the function x is E. The requisite expression expressing the singleton set containing $(E_1, ..., E_n)$ is $funct(x) \wedge dom_n(x) \wedge (\forall z(n(z) \supset \forall y(y \in z \supset f_x(y, E_y)))$. The expressions above are expressible, and given below.

$n(x) \coloneqq \forall z(z \in x \supset (z = \overline{\phi} \wedge ... \wedge \overline{\phi_n}))$

$funct(x) \coloneqq \forall z((z \in x \supset pair(z)) \wedge \forall u \forall v(first(z,u) \wedge first(z,v) \wedge u = v \supset \forall l \forall k(second(z,k) \wedge second(z,l) \supset l = k)$

where $pair(z) \coloneqq \exists x \exists y(x \in z \wedge y \in z \wedge z \neq y \wedge (\forall z(z \in x \supset z \in y) \wedge x \neq \phi \wedge y \neq \phi \wedge Elt_1(x) \wedge Elt_2(y)) \wedge Elt_2(z)$, where for any $n \in \mathbb{N}$, $Elt_n(x)$ says that x has exactly n elements, and it is expressed by $Elt_n(x) \coloneqq \exists x_1 ... \exists x_n (\bigwedge_{1 \leq i < j \leq n} x_i \neq x_j) \wedge \forall z(z \in x \supset \bigvee_{1 \leq i \leq n} x = x_i)$, $first(z,y)$ and $second(z,y)$ say that the first or second elements of z is y respectively, expressed by $first(z,y) \coloneqq \forall x(Elt_1(x) \wedge y \in x \supset x \in z)$ and $second(z,y) \coloneqq \forall x(Elt_2(x) \wedge y \in x \supset x \in z)$.

$dom_n(x) \coloneqq Elt_n(x) \wedge \forall z(z \in x \supset \bigvee_{1 \leq i \leq n} first(z,i))$

$f_x(y,E) \coloneqq \forall z((z \in x \wedge first(z,y)) \supset second(z,E))$ □

3.3. The Axiom System of ZFC.

The aim of this subsection is to describe the sets $\mathcal{P}$ and $\mathcal{R}$ in ZFC and show that they are expressible. In doing so, it is necessary to define precisely what is a proof in as an expression, which involves notions of axioms, inference, and sequences of expressions.

3.3.1. *Axioms and Rules of Inference:* Informally, a proof of a theorem is a ordered finite list of expressions, starting from axioms and eventually getting to the statement of the theorem, with rules of inference that govern what is allowed in between. In order to formalize the notion of a proof in ZFC, we first need to define the axiom system of ZFC.

ZFC has two types of axioms. The first few axioms are the axioms that govern propositional and first order logic, the second group are those particular to ZFC. What the axioms precisely are is not particularly important for this paper, so we give a cursory presentation. What is important is that we can explicitly write the axioms down in the language of ZFC, and the very fact that we designate a special subset of expressions $\mathcal{A} \subset \mathcal{E}$ as axioms in the first place.

In order to write down some of the axioms of ZFC, we first make some preliminary abbreviations.

Definition 27. Let x, y, z and w be distinct variables.

1. The expression $z = x \cup y$ is defined as
$(\forall w \in z(w \in x) \vee (w \in y)) \wedge (\forall w \in x(x \in z)) \wedge (\forall w \in y(w \in z))$.
2. $y = x \cap z$ or "y is the intersection of x and z" is defined as
$\forall z(z \in y \equiv z \in x \wedge z \in w)$.
3. $\exists! yF$, or "there is a unique y such that F holds" is
$\exists y(F \wedge \forall z(F\underset{y}{z} \supset z = y))$
where $F\underset{y}{z}$ is the formula that results from substituting all free occurrences of y in F for z.
4. $w = \{z\}$ or "w is the singleton set containing z" is defined as
$z \in w \wedge \forall u(u \in w \supset q = z)$.

Now we present the axioms of ZFC:

Definition 28. Let F, G, H be formulas, and let u, v, w, x, y, z be distinct variables, and let m be a set-expression. Recall that ϕ is the empty set and $\overline{\phi}$ is its set expression. An **axiom** of ZFC is any expression of the following forms:

1. Axioms of Propositional and First Order Logic
L_1: $(F \supset (G \supset F))$
L_2: $(F \supset (G \supset H)) \supset ((F \supset G) \supset (F \supset H))$
L_3: $((\sim F \supset \sim G) \supset (G \supset F))$
L_4: $(\forall x(F \supset G) \supset (\forall xF \supset \forall xG))$
L_5: $(F \supset \forall xF)$ if x does not occur in F
L_6: $\exists x(x = m)$
L_7: $(x = m \supset (XxY \supset XmX))$ where X and Y are any expressions such that XxY is an atomic formula.
2. Axioms specific to ZFC
ZFC_1: Extensionality
$\forall x \forall y(\forall z(z \in x \equiv z \in y) \supset x = y)$
ZFC_2: Seperation
$\forall v_1 ... \forall v_n \forall x \exists y \forall z(z \in y \equiv z \in x \wedge F)$ where F has at most $n+1$ free variables $v_1, ..., v_n, z$.
ZFC_3: Small Union
$\forall x \forall y \exists w \forall z(z \in x \vee z \in y \supset z \in w)$
ZFC_4: Big Union
$\forall w \exists y \forall x \forall z(x \in w \wedge z \in x \supset z \in y)$

ZFC_5: Power Set
$\forall x \exists y \forall z (z \subset x \supset z \in y))$
ZFC_6: Infinity
$\exists x(\overline{\phi} \in x \wedge \forall z(z \in x \supset (\forall w(w = z \cup \{z\}) \supset w \in x))^1$
ZFC_7: Replacement
$\forall v_1, ..., \forall v_n(\forall x \exists ! y F \supset \forall u \exists v \forall x \forall y(x \in u \wedge F \supset y \in v))$ where F contains at most $n+2$ free variables $v_1, ..., v_n, x, y$.
ZFC_8: Foundation
$\forall w(w \neq \{\} \supset \exists x(x \in w \wedge x \cap w = \overline{\phi}))$
ZFC_9: Choice
$\forall w(\forall x \forall y(x \in w \wedge y \in w \supset x \neq \overline{\phi} \wedge (x \neq y \supset x \cap y = \overline{\phi})) \supset \exists v \forall x(x \in w \supset \exists z(v \cap x = \{z\})))$

In writing a proof, the list of expressions one writes down must follow specific rules. Generally at the beginning of a proof, one starts with some axioms, and the expressions further on in the list may be additional axioms or they were "inferred" from the preceding expressions by some rule.

Definition 29. The following are the **inference rules** of ZFC. For all formulas F and G and for all variables x,

1. We say G is inferred from F and $(F \supset G)$.
2. We say $\forall x F$ is inferred from F

3.3.2. *Provability and Refutability in ZFC:.* We now wish to define the notion of provability and express the set $\mathcal{P}$ of provable sentences and the set $\mathcal{R}$ of refutable sentences in ZFC. Informally, a sentence is provable if there exists a proof of it, and we define every proof to be a special type of finite tuple of expressions, of the form $(E_1, ..., E_n)$. Dually, a sentence is refutable if there is a refutation of it, and a refutation is just a proof of its negation.

Definition 30. We denote **the set of all finite tuples of expressions** by $\mathcal{E}^{<\omega} = \bigcup\limits_{n \in \mathbb{N}} \mathcal{E}^n$, where $\mathcal{E}^n = \{(E_1, ..., E_n) : E_i \in \mathcal{E}\}$ is the set of finite tuples of n members, each of which is an expression.

The notation $< \omega$ comes from the standard notation that ω refers to the first infinite cardinal; here it refers to the fact that we are taking the union of powers of $\mathcal{E}$ that are finite, and hence have cardinality less than ω.

Definition 31. Let K be an sentence of ZFC. We say that a tuple of expressions $P = (E_1, ..., E_n) \in \mathcal{E}^{<\omega}$ is a **proof** of K if there is a

---

[1] The expression $w = z \cup \{z\}$ is a shorthand for $w = z \cup u \wedge u = \{z\}$

member E_i such that $E_i = K$ and every member i satisfies one of the following:

1. E_i is an axiom of ZFC
2. There exists a $j < k < i$ such that E_i is inferred from E_j and E_k by the first inference rule.
3. There exists a $j < i$ such that E_i is inferred from E_j by the second inference rule.
4. We say a sentence X is **provable** in ZFC if there exist a proof of X.
5. A sentence Y is **refutable** if the sentence $\sim Y$ is provable. A **refutation** of Y is a proof of $\sim Y$.

Note that our definition of a proof of K did not require K to be the last member of its proof. This does not matter because the existence of a proof that contains K not as itd last member implies the existence of a shorter proof that has K as its last member by cutting off all the members after K.

4. Proof of the First Incompleteness Theorem in ZFC

Recall that the proof shall be done in three parts. First we shall prove that the set of provable sentences $\mathcal{P}$ is expressile. Then we shall prove that for any expressible set, the complement is expressible. Finally we prove that for any expressible set, the pre-image of the diagonal is also expressible. The result is that $(\mathcal{P}')^*$ is expressible, and by the abstract first incompleteness theorem, if ZFC is correct then it is incomplete.

4.1. Basic Definitions about Expressibility.

Here we make some new definitions that will be required in the proofs in the later subsections.

Recall that for a set A of expressions, we defined A to be expressible if there is a predicate H, i.e. formula with one free variable such that $H(\overline{m})$ that is true if and only if $m \in A$. Now we extend the notion of expression to sets of tuples of expressions, and formulas with more than one free variable.

Definition 32. A **relation** $R(k_1, ..., k_n)$ of n variables is a set of n tuples $(m_1, .., m_n)$ of expressions or finite tuples of expressions. For such an n tuple $(m_1, .., m_n)$, we say $R(m_1, .., m_n)$ **holds** if $(m_1, .., m_n) \in R(k_1, ..., k_n)$.

Example 33. The relation $R(x, y) = \{(x, y) : x \in y\}$ of two variables is the set of all tuples (x, y) of sets such that x is an element of y. For the tuple, $(x, \{x\})$, we say $R(x, \{x\})$ holds.

Definition 34. A **relation** $R(k_1, ..., k_n)$ of n variables is **expressible** if there is a regular formula $F(v_1, ..., v_n)$ with n free variables such that $F(\overline{m}_1, ..., \overline{m}_n)$ is true if and only if $(\overline{m}_1, ..., \overline{m}_n) \in R(k_1, ..., k_n)$.

Example 35. The relation $R(x, y) = \{(x, y) : x \in y\}$ is expressible, and the regular formula expressing it is $v_1 \in v_2$.

To avoid clunky vocabulary when we talk about relations or even sets of expressions, we shall rarely explicitly refer to them as sets of tuples. Instead, we may informally refer to them by a specific property that defines the relation or set. For example, instead of writing $R(x, y) = \{(x, y) : x \in y\}$, we may simply write "the relation $x \in y$". Whenever we talk about such properties, the context will be clear so that there will be no ambiguity as to what relation we are referring to.

Additionally, we may use the term "property" formally to mean a relation of one variable, i.e. just a set of single expressions. So $Var(x)$ could be the property that x is a variable, and it formally means the set of expressions that only have single letters v_1 or v_2 or v_3 etc.

Definition 36. A property is expressible if the relation or set defined by the property is expressible.

Example 37. The property of provability is expressible if the set $\mathcal{P}$ is expressible.

4.2. **Expressibility of $\mathcal{P}$.** In order to show that $\mathcal{P}$ is expressible, we need to find a formula with one free variable $H(v_1)$ that says'v_1 is provable'. When we take the set-expression representing x and substitute it for the free variable in H, $H(\overline{m})$ will be true if and only if $m \in \mathcal{P}$. Additionally, the only symbols allowed in constructing such an expression are the elements of β. Such a task would be impossible to do directly. So instead, we will build a vocabulary of auxiliary relations and formulas, such as formulas that say'v_1 is an axiom of ZFC', and then express $H(v_1)$ in terms of these auxiliary relations and formulas.

The most important auxiliary properties and relations will be:

$Var(x)$ and $Setexp(x)$ - informally stating that "x is a vairable" and "x is a set expression".

$Fm(x)$ - informally stating that "x is a formula".

$A(x)$ - informally stating that "x is an axiom".

$P(x)$ - informally stating that "x is provable in ZFC"

4.2.1. *Basic Auxiliary Relations:* Given two expressions $F = F_1, ..., F_n$, $G = G_1, ..., G_m$, the concatenation of F and G is $F{\star}G = F_1, ..., F_nG_1, ..., G_m$ that refers to the expression resulting from first writing the symbols of

F and then immediately after writing the symbols of G. One auxiliary relation we want to use is the relation $Z = F \star G$.

Proposition 38. *The relation $Z = F \star G$ is expressible.*

Proof. We express Z via auxiliary expressions.

1. The relation $Z = X \cup Y$, or "z is the **union** of x and y" is expressible, and expressed by the following predicate, as introduced before in the defintion of the axioms of ZFC:
$(\forall x \in Z(x \in X) \vee (x \in Y)) \wedge (\forall x \in X(x \in Z)) \wedge (\forall x \in Y(x \in Z))$.

2. The relation $X \subseteq Y$, or "X is a **subset** of Y" is expressible:
$\forall x \in \overline{X}(x \in \overline{Y})$.

3. The relation $Z = X - Y$, or "Z is the set X without elements of Y" is expressible:
$Z \subseteq X \wedge \forall x(x \in X \wedge x \notin Y \supset x \in Z)$

4. Note that for two natural numbers n, m, $n < m$ iff n is a proper subset of m. Define $x \subset y := x \neq y \wedge x \subseteq y$. We define the expression that "Z is the concatenation of F and G and follows. First we require F to be a subset of Z. Then we take the smallest element of the domain of $Z - F$ and require $Z - F$ to agree with G on the smallest element of G. Then we put an inductive statement saying that if $Z - F$ and G agree on one element, they agree on the next element. Thus inductively, it shows that Z agrees with F on the first few elements appropriate to the domain of F, and on the remaining elements it agrees with G.

Then $Z = F \star G := (F \subset Z) \wedge \forall K\{(K = Z - F \supset \wedge \forall k(smallest(k, K) \supset \forall E(f_G(1, E) \supset f_K(k, K)) \wedge [\forall E_1 \forall E_2 \forall u \forall v \forall w \forall x(u = v + 1 \wedge x = w + 1 \wedge f_k(u, E_1) \wedge f_G(w, E_1) \supset (F_k(w, E_2) \wedge F_G(v, E_2))]\}$

where $smallest(k, K) := \exists x \exists y(first(y, x) \wedge y \in K \wedge k = x) \wedge \forall x \forall y(first(y, x) \supset k \subseteq x)$ means that of all numbers in the domain of K, k is the smallest. ☐

Definition 39. 1. For any two expressions $F = F_1, ..., F_n$, $G = G_1, ..., G_m$, we say F **begins** G if $G = F_1, ..., F_n, G_{n+1}, ...G_m$. In shorthand, we write $F\mathbf{B}G$ to mean that F begins G. Similarly, we say that F **ends** G, or $F\mathbf{E}G$ if $G = G_1...G_{m-n}, F_1, ..., F_n$. Finally, we say that F is a **part** of G, or $F\mathbf{P}G$, if $G = G_1...G_kF_1...F_nG_{n+1}...G_m$.

Proposition 40. *The relations $F\mathbf{B}G$, $F\mathbf{E}G$ and $F\mathbf{P}G$ are are expressible.*

Proof. 1. Again keeping in mind that an expression $F = \{\{F_1\}, ..., \{F_1, ..., F_n\}\}$, $F\mathbf{B}G$ is expressed by:
$\forall x \in \overline{F}(x \in \overline{G})$

2. $F\mathbf{E}G$ is expressed by:
$(\overline{F} = \overline{G}) \vee (\exists \overline{H}(\overline{G} \star \overline{F} = \overline{H}))$.

3. If F is a part of G, then there is an expression H such that H begins G and F ends H. So $F\mathbf{P}G$ s expressed by:
$\exists H((H\mathbf{B}G) \wedge (F\mathbf{E}H))$ $\square$

Definition 41. 4. We say $F \sim \mathbf{P}G$ if F is not a part of G, and $F_1 ... F_n \mathbf{P} G$ if $F_1 \star ... \star F_n$ is a part of G.

Exercise 42. Show that $F \sim \mathbf{P}G$ and $F_1 ... F_n \mathbf{P} G$ are expressible. (Hint: $\sim F\mathbf{P}G$; and $F_1 \star ... \star F_n = H$ is expressible by induction on n)

Definition 43. Let $Z = (E_1, ..., E_n)$ be a sequence of expressions. We say $X \underset{Z}{\prec} Y$ or X **precedes** Y in Z if there exists i, j less than or equal to n such that $i < j$ and $E_i = x$ and $E_j = y$.

Proposition 44. *The relation* $X \underset{Z}{\prec} Y$ *is expressible.*

Proof. 1. First we say X is a member of Z if $X = E_i$ some i. Let $X \underset{\mathbf{m}}{\in} Z$ be the relation that X is a member of Z. $X \underset{\mathbf{m}}{\in} Z$ is expressible:
$\exists x \in \overline{Z}(\overline{X} \in x)$
2. Now $X \underset{Z}{\prec} Y$ is expressible:
$(\overline{X} \underset{\mathbf{m}}{\in} Z) \wedge (Y \underset{\mathbf{m}}{\in} Z) \wedge \exists W((W\mathbf{B}Z) \wedge (X \underset{\mathbf{m}}{\in} W) \wedge (\sim Y \underset{\mathbf{m}}{\in} W))$ $\square$

Using these auxiliary relations, we now employ the following abbreviations. Let F be a formula:

$$(\forall X \underset{\mathbf{m}}{\in} Y)F := \forall X(X \underset{\mathbf{m}}{\in} Y \supset F)$$

$$(\exists X, Y \underset{W}{\prec} Z)F := \exists X \exists Y(X \underset{W}{\prec} Z \wedge Y \underset{W}{\prec} Z \wedge F)$$

4.2.2. *Formation Sequences for Set-expressions and Formulas:* We have previously defined formulas and set-expressions inductively in terms of atomic formulas and the empty set, i.e. the atomic set, respectively. This means than an expression is a formula or a set-expression if and only if there is a sequence of formulas or set-expressions, starting from atomic formulas and atomic set-expressions and containing set-expression desired. In order to express the auxiliary properties $Setexp(x)$ and $Fm(x)$ that each say "x is a set-expression" or "x is a formula", we need to formalize the notion of a sequence of formulas or sequences of set-expressions that generate desired formulas and set-expressions.

For any expressions X, Y and Z define $R_s(X, Y, Z)$ to be the relation that Z is the set-expression$\{\overline{X}\}$ or $\{\overline{X}, \overline{Y}\}$. We say $R_s(X, Y, Z)$ is a formation relation of set-expressions.

Similarly, define $R_f(X,Y,Z)$ to be the relation that Z is one of the expressions $\sim X$ or $(X \supset Y)$ or $\forall v_i X$ for some variable v_i. We say $R_S(X,Y,Z)$ is a formation relation of formulas.

The reader may note that the definition of a formation sequence for set expressions and the definition of a proof in ZFC are very similar. Both proofs and formation sequences are sequences of expressions, with rules that each member of the sequence is some kind of atomic expression, such as an atomic formula or axiom of ZFC, with rules that each member must satisfy some relation with preceding members. Thus, the proofs for the expressibility of $\mathcal{P}$, $Fm(x)$ and $Setexp(x)$ are similar.

Definition 45. 1. A formation sequence of set-expressions is a finite sequence of expressions $(E_1, ..., E_n)$ such that each E_i is the empty set, or there exists preceding members E_j, E_k , $j \neq k$ such that $(E_j,E_kE_i) \in R_s(E_j,E_k,E_i)$.

2. Similarly, a formation sequence of formilas is a finite sequence of expressions $(E_1, ..., E_n)$ such that each E_i is an atomic formula, or there exists preceding E_j, E_k , $j \neq k$ such that $(E_j,E_kE_i) \in R_f(E_j,E_k,E_i)$.

4.2.3. *Auxiliary Properties $Var(x)$ and $Setexp(x)$:*

Definition 46. The property $Var(x)$ is the property that x is a variable.

Recall that a variable expression is of the form $(v'''...'')$ where there is an arbitrary finite string of commas after v. Let's call **a string of subscripts** a finite expression of the form $'''...'''$. An expression is a string of subscripts if and only if every part of that expression contains a comma.

Proposition 47. *$Var(x)$ is expressible*

Proof. 1. Let us first define the property $Sb(x)$ that x is a string of subscripts. First we show that $Sb(x)$ is expressible:

$\forall y(y\mathbf{P}x \supset 5\mathbf{P}y)$

2. Then we express $Var(x)$ using $Sb(x)$:

$\exists y(Sb(y) \wedge x = \overline{8 \star v \star y \star 7}$ □

Definition 48. The property $Setexp(x)$ is the property that x is a set expression.

Proposition 49. *$Setexp(x)$ is expressible. (i.e. the previously introduced set Γ, defined as the set of all set expressions, is expressible and it is expressed by $Setexp(x)$).*

Proof. Most of the work was done in the previous section in explaining the idea of $\overline{x}$ for every set expression x. $Setexp(x)$ is expressed by the formula.

$\exists y(y = \overline{x})$ □

4.2.4. *Auxiliary Property $Fm(x)$:*

Definition 50. The property $Fm(x)$ is the property that x is a formula.

Proposition 51. *$Fm(x)$ is expressible.*

Proof. 1. First define the property $atom(X)$ that X is an atomic formula. $atom(X)$ is expressible:
$\exists y \exists z((X = \overline{(y = z)}) \land (X = \overline{(y \in z)}))$

2. Next define the relation $Gen(X, Y)$ that X is an expression of the form $\forall v_i Y$ for some variable v_i. $Gen(X, Y)$ is expressible:
$\exists z(Var(z) \land y = \overline{(\forall \star z \star x)})$

3. $R_f(X, Y, Z)$ is expressible:
$Z = \overline{(X \supset Y)} \land Z = \overline{(\sim X)} \land Z = \overline{(Gen(X, Y))}$

4. Define $Seqf(x)$ that x is a formation sequence for formulas. $Seqf(x)$ is expressible:
$(\forall y \underset{m}{\in} x)(atom(y) \lor (\exists z, w \underset{x}{\prec} y)R_f(z, w, y))$

5. $Fm(x)$ is expressible:
$\exists y(Seqf(y) \land x \underset{m}{\in} y)$ □

4.2.5. *Auxiliary Property $A(x)$:*

Definition 52. $A(x)$ is a property that x is an axiom of ZFC.

Proposition 53. *$A(x)$ is expressible.*

Proof. Recall that we listed the axioms of ZFC earlier, split into two group: the axioms of first order and predicate logic L_1 to L_7, and the axioms specific to ZFC ZFC_1 to ZFC_2. Let $L_i(x)$ and $ZFC_i(x)$ be the property that x is an axiom of the form L_i and ZFC_i respectively. We show that $L_1(x), ..., L_7(x)$ and $ZFC_1(x)$,...,$ZFC_9(x)$ are all expressible, and the define $A(x) = L_1(x) \lor ... \lor L_7(x) \lor ZFC_1(x) \lor ... \lor ZFC_9(x)$, and hence $A(x)$ is expressible.

1. The properties $L_1(x), ..., L_7(x)$ can be routinely, and we give an example of a proof of the expressibility of $L_1(x)$. If x is an expression, then it is an axiom of the form L_1 if and only if there are formulas F and G such that $x = (F \supset (F \supset G))$. The expression expressing $L_1(x)$ is $(\exists y \exists z(Fm(y) \land Fm(z) \land x = \overline{(y \supset (z \supset y))}))$. Using the auxiliary properties such as $Fm(x)$, $Var(x)$ and $atom(x)$ previously defined and the over-line notation $\overline{x}$ for denoting the set-expression of x, the expressibility of $L_2, ..., L_6$ are very similarly proved.

2. The properties $ZFC_1(x), ..., ZFC_9(x)$ are all expressible, though not as routine as $L_1(x), .., L_7(x)$. Similarly, the expression expressing $ZFC_i(x)$ is still just $K \wedge x = \overline{zfc_i}$ where zfc_i is the corresponding expression we listed previously with axiom ZFC_i, the expression K is the concatenation of appropriate quantifiers $\exists u Var(u)$ and $\exists F(Fm(F))$ for each variable u and formula f used in the axiom and declared in the beginning of the definition of axioms of ZFC.

The only difficulty is that some axioms require a formula F to have at most a specific number m specific of free variables.

We thus first define a new relation $Free(v, F)$ that v is a free variable of F and prove it is expressible. Then, for axioms such as one of the form ZFC_2, that require a formula F to have at most $n+1$ free variables $v_1, ..., v_n, x$, we add the expression

$\exists x \exists v_1, ..., \exists v_n Var(x) \wedge Var(v_1) \wedge ... \wedge Var(v_n) \wedge \forall y(Free(y, F) \supset y = x \vee y = v_1 \vee ... \vee y = v_n)$

to K.

To prove the expressibility of $Free(v, F)$,we first define the relation $Bounded(v, F)$ that every occurrence of v in F is bounded.

A variable is not free in F if an only if every occurrence in F is bounded, which is to say that for every expression E of two characters long that is a part of F, if v is a part of E then $\forall$ precedes v in E or v is the first member of E. However, this is equivalent to saying it is true for expressions E of any length. In one direction proving this for an expression of any length proves it for expressions of length 2. In the other direction proving it only for expressions of length two proves it for expressions of any length, because an expression of length 1 with v in it will automatically have v as the first member, and an expression of length more than two has a an expression of length two inside it.

So $Bounded(v, F)$ is expressible and expressed by

$Var(v) \wedge Fm(F) \wedge (\forall E(E\mathbf{P}F \wedge v\mathbf{P}E \supset \overline{\forall} \underset{E}{\prec} v \vee \forall y \underset{m}{\in} E(y = x \vee v \underset{E}{\prec} y) \wedge E))$.

Then $Free(v, F)$ is expressible and is expressed by $\sim Bounded(v, F)$

□

4.2.6. *Expressibility of $\mathcal{P}$.*

Proposition 54. *The set of provable sentences $\mathcal{P}$ is expressible in ZFC.*

Proof. 1. Define the relation $Der(X, Y, Z)$ that Z is inferred from X and Y by the first inference rule (ie. Y is the expression $X \supset Z$), or Z is inferred from X by the second inference rule (i.e. Z is the expression $\forall v_i X$ for some variable v_i). $Der(X, Y, Z)$ is expressible:

$Y = \overline{(X \supset Z)} \wedge Gen(X, Z)$

2. Define the property $Pf(x)$ that x is a proof. $Pf(x)$ is expressible: $(\forall y \underset{m}{\in} x)(A(y) \vee (\exists z, w \underset{x}{\prec} y)Der(z, w, y))$

3. The set $\mathcal{P}$ and the property $Prox(x)$ that x is provable is expressed by:

$\exists y(Pf(y) \wedge x \underset{m}{\in} y)$ ☐

Even though it is not required for the proof, for the sake of completeness , we also have:

Proposition 55. *The set of refutable sentences $\mathcal{R}$ is expressible in ZFC.*

Proof. It is expressed by $Prov(\overline{\sim x})$. ☐

4.3. Expressibility of $\mathcal{P}'$and $(\mathcal{P}')^*$.

Proposition 56. *Let A be a set of sentences that is expressible in ZFC. Then $A' = \{S \in \mathcal{S} : S \notin A\}$ is expressible in ZFC.*

Proof. Let H be a predicate expressing A. Then $\sim H$ expresses A'. ☐

Corollary 57. *In ZFC, $\mathcal{P}'$ is expressible.*

For a set of expresions A, in order to prove the expressibility of the the pre-image of the diagonal, i.e. A^*, we shall define what it means for a function to be expressible.

Definition 58. A function of n variables $f(x_1, ..., x_n) : \mathcal{E}^n \to \mathcal{E}$ is expressible if the relation defined by $f(x_1, ..., x_n) = z$ is expressible.

We briefly re-define the predication function in order to express it in ZFC.

Definition 59. If $H(v_1)$ is a predicate and and E is an expression, the expression $H(E)$ is defined as the expression $\forall v_1(v_1 = E) \supset H(v_1)$.

Proposition 60. *The predication function defined by $\phi(H, E) = H(\overline{E})$ is expressible in ZFC.*

Proof. The expression $Z = \forall v_1(v_1 = E) \supset H$ expresses the relation $\{(H, E, Z) : Z = H(E)\}$ that is defined by the predication function. ☐

Proposition 61. *Let A be an expressible set in ZFC. Then $A^* = d^{-1}(A)$ is expressible in ZFC.*

Proof. The diagonal function $d(H) = H(H) = \phi(H, H)$ is expressible, since the predication function is expressible. Let $D(v_1, v_2)$ be a regular formula expressing the relation defined by $d(x) = y$. Let $F(v_1)$ be a formula expressing A. Then A^* is expressed by the formula $\exists v_2(D(v_1, v_2) \wedge F(v_2))$. $\square$

Corollary 62. *$(\mathcal{P}')^*$is expressible in ZFC.*

Corollary. *If ZFC is correct, then ZFC is incomplete.*

5. Abstract Incompleteness Based on Consistency

Now we return to the abstract language system $\mathcal{L}$ introduced in section 2.

Definition 63. A set of expressions A is **representable** if there exists a predicate H such that $A = \{E \in \mathcal{E} : H(E) \in \mathcal{P}\}$, i.e. A consists exactly of the expressions E such that $H(E)$ is provable. We also say H **represents** A.

Given any arbitrary predicate H, we can easily construct a set H' that is represented by H. We simply define $H' = \{E \in \mathcal{E}\colon H(E) \in \mathcal{P}\}$.

Additionally, we require the notion of separability in order to introduce some conditions, other than consistency, for language system to be incomplete.

Definition 64. A set of expressions A is separable from a set of expressions B if there exist a predicate H such that for all $E \in A$, $H(E)$ is provable, and for all $E \in B$, $H(E)$ is refutable. We say A is separable from B.

Now we are able to prove an abstract version of the first incompleteness theorem.

Theorem 65. *If $R^*$is separable from P^*in a language system $\mathcal{L}$, and $\mathcal{L}$ is consistent, then $\mathcal{L}$ is incomplete.*

This theorem follows from the two lemmas below, taking $A = R^*$ and $B = P^*$ in the second lemma.

Lemma 66. *If some superset of R^*, disjoint from P^*, is representable in $\mathcal{L}$, and $\mathcal{L}$ is consistent, then $\mathcal{L}$ is incomplete.*

Proof. Let K be a superset of R^* disjoint from P^* and let H be an expression representing K. Consider the sentence $H(H)$. By definition of the diagonal function, $H(H)$ is provable if and only if $H \in P^*$. Additionally since H represents K, $H(H)$ is provable if and only if $H \in K$. Combining there two equivalences, we have $H \in P^*$ if and only if $H \in K$. But P^* is disjoint from K and therefore $H \notin P^*$ and $H \notin K$. Since $K \supset R^*$, we have $h \notin R^*$. Then $H(H)$ is neither provable nor refutable in $\mathcal{L}$, so $\mathcal{L}$ is incomplete. $\square$

Lemma 67. *For any sets of expressions A and B, if A is separable from B, then some superset of A disjoint from B is representable in $\mathcal{L}$.*

Proof. Let H be a predicate that seperates A from B. Consider the set $H' = \{E \in \mathcal{E} : H(E) \in \mathcal{P}\}$. By definition, H' is represented by H. Since for all $E \in A$ the sentence $H(E)$ is provable, we have $A \subset H'$. Suppose there is some expression L that was in both H' and B. Then $H(L)$ would be both provable and refutable in $\mathcal{L}$, implying that $\mathcal{L}$ is inconsistent. Hence if $\mathcal{L}$ is consistent, no such L can exist and H' is disjoint from B, so H' is a representable superset of A disjoint from B. $\square$

6. Appendix - Tarski's Theorem

In this paper, we proved an incompleteness theorem for ZFC. We also defined a proof in ZFC as a set which is a finite sequence of expressions. One might think then, that perhaps the proof in this paper can be found in the language system of ZFC as a finite sequence of expressions. This however, is not possible. Indeed one theoretical disadvantage of our proof is that is a meta proof about ZFC but it cannot be formulated in the language of ZFC itself. The reason for this lies in our assumption of the correctness of ZFC, which we made for the sake of simplifying the proof, that required us to explicitly define what is a true statement in ZFC. It turns out that that section in our paper outlining the conditions for a sentence to be true cannot be realized as a predicate of one free variable that takes in a set expression and returns true or false. Here we prove Tarski's theorem, which states that no language can have a predicate H that expresses the set of all true sentences.

Theorem 68. *(Tarski's Theorem) The set $\mathcal{T}$ of true sentences is not expressible in ZFC.*

Proof. Suppose $\mathcal{T}$ is expressible. We proved earlier that in ZFC, for any expressible set A, the sets A' and A^* are also expressible. So then the set $(T')^*$ is expressible. Let H be an expression expressing it. Then consider the sentence $H(H)$. Since H expresses $(\mathcal{T}')^*$, $H(H)$ is true if and only if $H(H) \in T'$ if and only if $H(H)$ is false. So $H(H)$ is a sentence that is true if and only if it is false. So it is neither true nor false, because if it was either true or false it would be both true and false, which is a contradiction. So $\mathcal{T}$ is not expressible. $\square$

Although stated for ZFC, Tarski's theorem applies to any language system that satisfies the requirement that for any expressible set A, the sets A' and A^* are also expressible, which include many other first order theories such as Peano Arithmetic. The idea behind the theorem is that in such a language system, if $\mathcal{T}$ were expressible, then it would be possible to construct a sentence $H(H)$ that say'This sentence is false', which can be neither true nor false if we assume that every sentence cannot be both true and false. The contraction comes from the fact that every sentence must either be true or false.

Tarski's theorem has similar and perhaps even more significant implications than the first incompleteness theorem. It states that as long as a language system is capable of some simple conditions expected of a non trivial mathematical system, it is impossible to express the concept of truth in that system.

Acknowledgments. With pleasure I thank several individuals who have helped me in the process of writing this paper. First I thank Nir Gadish for his invaluable guidance and mentoring. I would also like to thank Jonathan Stephenson for his suggestions on literature for reading. Finally, I thank Peter May for his organization of the REU at the University of Chicago.

REFERENCES

[1] R.M. Smullyan. Gödel's Incompleteness theorems. Oxford University Press. 1992.
[2] M. Behrisch. Consistency and Incompleteness of ZFC. http://www.math.tu-dresden.de/~mbehri/documents/Incompleteness_ZFC.pdf
[3] K. Kunen. May. Set Theory. Elsevier Science Publishers B.V. 1980.