

Hierarchy of unsolvable problems

Jean-Simon Pacaud Lemay under the supervision of Professor Pieter Hofstra

Department of Mathematics, University of Ottawa

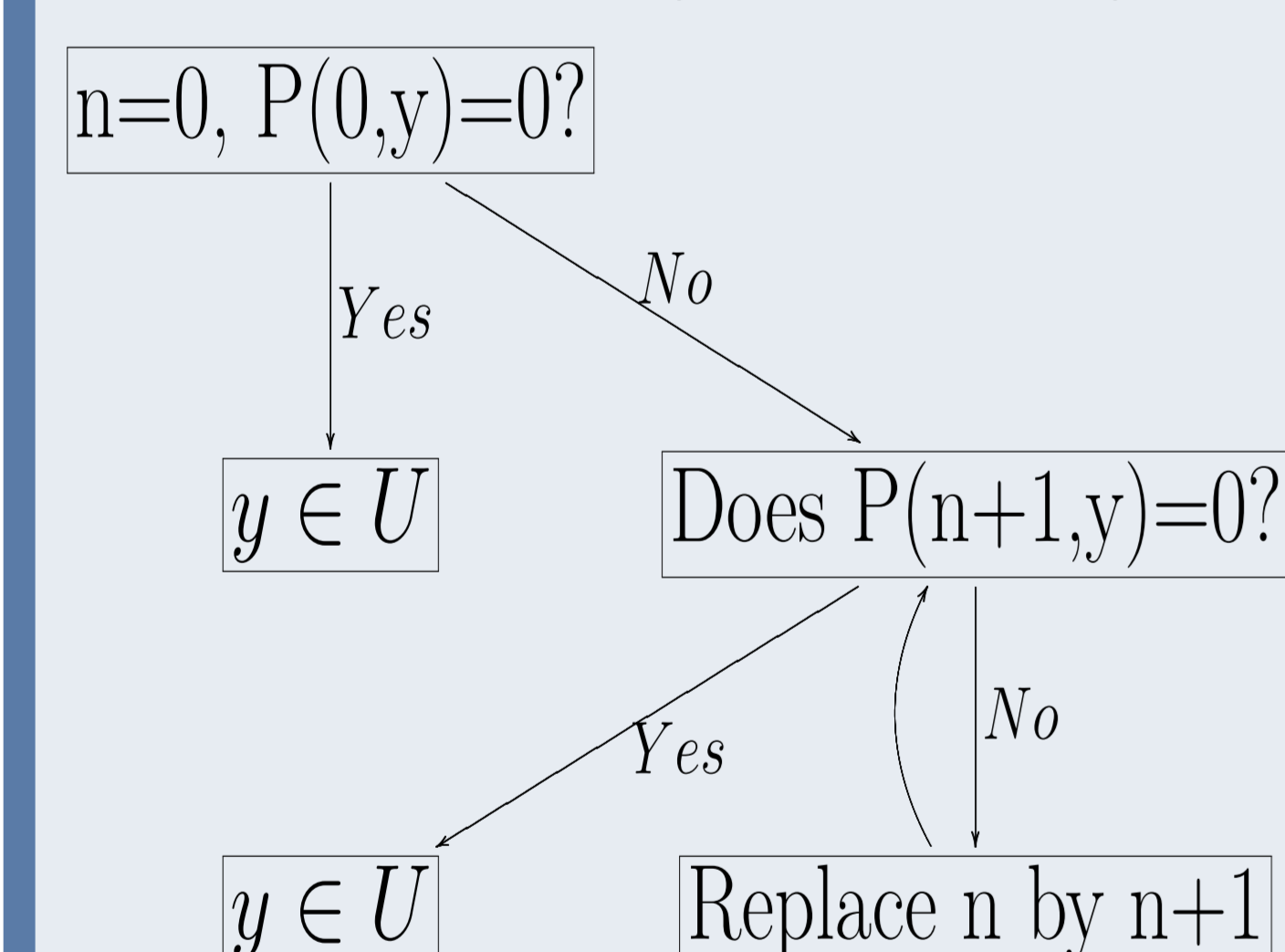
Solvable vs. unsolvable

Recursion theory is the study of functions over the natural numbers $\mathbb{N}=\{0, 1, 2, \dots\}$ and subsets of \mathbb{N} . Turing machines are idealized computation machines that input and output natural numbers. A function is called *computable* when there exists a Turing machine that will give the same output as the function for each input. Since computations need not halt on every input, computable functions are generally partial, in the sense that their domain of definition is a subset of \mathbb{N} .

A subset $A \subseteq \mathbb{N}$ can be thought of as a problem: given x , is $x \in A$. Such a problem is called *solvable* if there exists a Turing machine (algorithm) for solving it, and is called *unsolvable* otherwise. It turns out that many interesting problems can be coded into the natural numbers.

Example of an unsolvable problem

Let $P(x, y)$ be a two variable polynomial (with coefficients in \mathbb{N}). Given y , does there exist an $n \in \mathbb{N}$ such that $P(n, y) = 0$? This problem corresponds to the set $U = \{y \mid \exists x. P(x, y) = 0\}$. We attempt to solve it by trying the following:



This algorithm will either eventually halt at certain n or continue indefinitely. Since there is no way of knowing after long period of time which will be the case, we cannot solve every instance of the problem.

First level of unsolvable problems

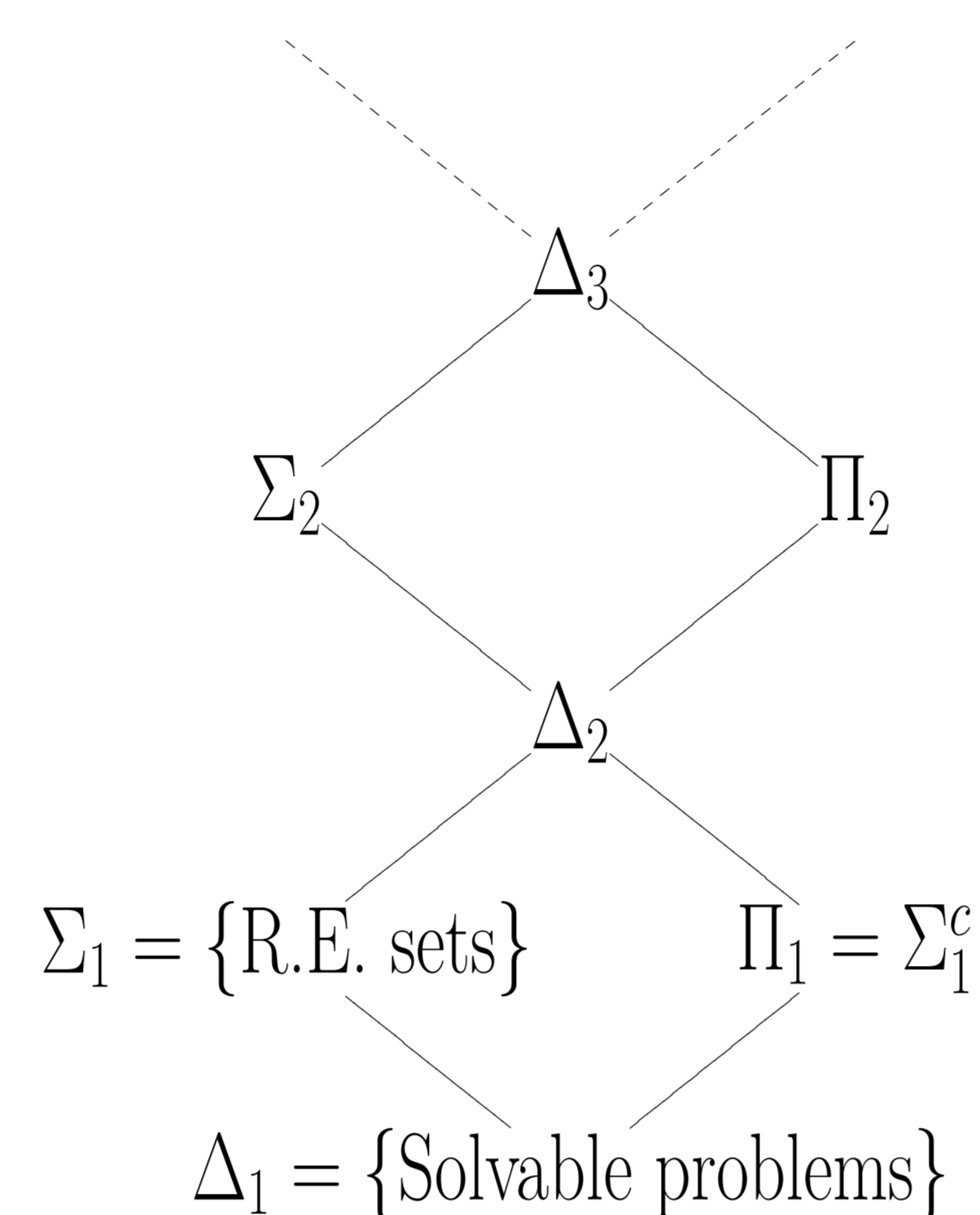
We define our first class of unsolvable problems as follows: A set A is called *recursively enumerable* or R.E. if there exists a computable function $f : \mathbb{N} \rightarrow \mathbb{N}$ for which:

$$\text{dom}(f) = A$$

In other words, there exists an algorithm which will terminate on an input which is in A , but which never terminates on inputs not in A . For this reason, R.E. sets are also called *semi-decidable*.

Arithmetical Hierarchy

Some unsolvable problems are more unsolvable than others. We can then classify unsolvable problems based on the complexity of their logical description.



Main Objective

In modern mathematics, we typically aren't interested in studying isolated objects but rather in the connections between objects. This is achieved by organizing a collection of objects and the connections or maps between them into what is called a *category*. We know how to do this at the base levels of the arithmetical hierarchy; in particular, it is very fruitful to study Σ_1 as a category (see below). Our goal is therefore to investigate the higher levels of the arithmetical hierarchy from a categorical point of view.

Σ_1 Category

We can form a category, D , from Σ_1 as follows:

- Objects: Σ_1 sets
- Maps: (Partial) computable functions $f : A \rightarrow B$ with $\text{dom}(f) \subseteq A$

Proof that the composition of functions with Σ_n graphs have a Σ_n graph

Let f and g have Σ_n graphs.

$$\begin{aligned}
 (x,y) \in Gr(g \circ f) &\Leftrightarrow \exists y. [(x,y) \in Gr(f) \wedge (y,z) \in Gr(g)] \\
 &\Leftrightarrow \exists y. [\exists v_1 \forall v_2 \dots Q_n v_n. P(x, y, v_1, v_2, \dots, v_n) \wedge \exists w_1 \forall w_2 \dots Q_n w_n. R(x, y, w_1, w_2, \dots, w_n)] \\
 &\Leftrightarrow \exists y \exists v_1 \exists w_1 [\forall v_2 \dots Q_n v_n. P(x, y, v_1, v_2, \dots, v_n) \wedge \forall w_2 \dots Q_n w_n. R(x, y, w_1, w_2, \dots, w_n)] \\
 &\Leftrightarrow \dots \\
 &\Leftrightarrow \exists y \exists v_1 \exists w_1 \forall v_2 \forall w_2 \dots Q_n v_n Q_n w_n [P(x, y, v_1, v_2, \dots, v_n) \wedge R(x, y, w_1, w_2, \dots, w_n)] \\
 &\Rightarrow g \circ f \text{ has a } \Sigma_n \text{ graph}
 \end{aligned}$$

Description of the levels

- Σ_n : $\exists y_1 \forall y_2 \dots Q_n y_n. P(\vec{x}, y_1, y_2, \dots, y_n)$, where Q_n is \exists if n is odd and \forall if n is even and P is a computable set.
- Π_n : $\forall y_1 \exists y_2 \dots Q_n y_n. P(\vec{x}, y_1, y_2, \dots, y_n)$, where Q_n is \forall if n is odd and \exists if n is even and P is a computable set.
- $\Delta_n = \Sigma_n \cap \Pi_n$
- $\Pi_n \subseteq \Sigma_{n+1}$ and $\Sigma_n \subseteq \Pi_{n+1}$

Can we make Σ_n into a category?

Our objective is to see if we can form a category with Σ_n sets as objects. The question is: what are the maps that connect the Σ_n sets? We propose the following:

Σ_n Category

- Objects: Σ_n sets
- Maps: functions $f : A \rightarrow B$ with Σ_n graphs, $Gr(f) \in \Sigma_n$ where we define the graph of f as $Gr(f) = \{(x, y) \mid f(x) = y\}$

It turns out, in fact that this does make a category. Below is the most interesting part of the proof, namely that the composition of two maps in the category again has a Σ_n graph.

Example of a Π_2 problem

A function $f : A \rightarrow B$ is *total* if for every $x \in A$ there is a $y \in B$ such that $f(x) = y$. Over the natural numbers we can ask the following question: Is a function $f : A \subseteq \mathbb{N} \rightarrow \mathbb{N}$ a total function?

We want to check whether f belongs to the set of all total functions:

$$f \in TOT = \{f \mid \forall x \exists y. f(x) = y\}$$

The logical description of TOT is $\forall \exists \Rightarrow TOT \in \Pi_2$ making this problem a Π_2 unsolvable problem.

Conclusion and further results

We have been able to make a category out of Σ_n . This means that we can now study the various connections between unsolvable problems of the same level. From here we can compare other known categorical properties of Σ_1 to Σ_n . Examples of these include: restriction structure (which captures the partiality of the maps), Turing structure (which describes the phenomenon in recursion theory that computable maps can be effectively enumerated and that there exists a universal Turing machine which can simulate all other machines), and further categorical structure such as products and coproducts. We have been able to prove that Σ_n is a restriction category and admits a *universal relation*, which can be regarded as a weak form of Turing structure. In the future, we hope to establish (or refute!) that Σ_n also admits a genuine Turing structure.

Acknowledgements

I would like to thank Professor Hofstra for his patience and guidance on this project. I would also like to thank the University of Ottawa and the Undergraduate Research Opportunity Program for providing me with this experience and to chance to work under an experienced mentor.

Contact information

Jean-Simon
Pacaud Lemay
jlema043@uottawa.ca



uOttawa