Excerpt from BOOK REVIEW COLUMN Sigact News Volume 40, No. 2, in 2010.

Review of[1]

The Annotated Turing

Author: Charles Petzoid

Publisher: Wiley, 2008

ISBN: 978-0-470-22905-7, $29.99

Reviewer: Kevin A. Wilson (kwilson@rti.org)

1 Overview

The Turing machine is a theoretical model of computation that is typically studied as part of a theoretical computer science class at the undergraduate level. Such a course typically introduces fundamental theoretical computer science concepts including several formal models of computation and their associated languages. Most often included are finite state automata and regular languages, push-down automata and context-free grammars. Such classes typically culminate with an introduction to Turing machines. While a significant amount of time is spent on the definition of a Turing machine, and a discussion of its computational power, most students rarely take the time to read Alan Turing's original 1936 paper that, in many ways, provides the theoretical underpinnings for modern Computer Science. Perhaps this can be attributed to the paper's complexity, questionable organization, and confusing terminology. While the paper is central to modern computer science, more modern, succinct treatments, such as those found in standard textbooks provide a significantly gentler introduction to the area.

Petzold's goal in this book is to bring the original Turing paper to the (computer science) masses by providing detailed annotation alongside the original paper to explain, inform, and clarify Turing's pivotal ideas. While clearly focused on the Turing machine, the book also provides a significant amount of context including discussion of the history of mathematics beginning with Diophantus, David Hilbert's famous 1900 address, Turing's life and education, and various mathematical foundations.

2 Review

The book is split into four discrete sections, each containing multiple chapters. The first section of the book covers the necessary mathematical foundations and provides a historical perspective for Turing's work (chapters 1-3). The second section discusses "computable numbers" and forms the major part of the book (chapters 4-11). This section discusses the nuts and bolts of Turing's computing machines. The third section describes "Das Entscheidungsproblem[2]" covering logic and computability, the notion of computable functions, and Turing's major proof (chapters 12-16). Sections two and three together provide a complete analysis of Turing's paper. Finally, section four discusses the future implications of Turing's work (chapters 17-18). Each of these sections and their individual chapters are described in detail below.

[2] "Das Entscheidungsproblem" is the problem of creating an algorithm to determine whether, given a description of a formal computing language, and a statement of that language, it is possible to decide conclusively if the statement is true or false.

Chapter 1 - This Tomb Holds Diophantus. Chapter 1 begins with an overview of ancient mathematics beginning with Diophantus and his famous *Arithmetica*, the mathematical tome that provided the beginnings of what we now refer to as algebra. Diophantus, in the third century of the Common Era, studied the question of whether, given an equation, does it have a solution in integers? This leads to the more general decision problem: given an equation, is it possible to determine if it has an integer solution? If this problem was decidable then other problems such as Fermat's last theorem may be able to be solved in the same way (although complexity issues might make this approach infeasible). We will return to this problem in Chapter 18.

Chapter 2 - The Irrational and Transcendental. This chapter provides some numerical background in preparation for some of the later discussion on countability. Rational numbers are discussed in the context of the algebraic equations defined in chapter 1. The concept that there exist real numbers that do not form solutions to such equations was discovered by Euler and these are referred to as Transcendental numbers. The existence of these numbers is of critical importance to Turing's work as it led to several foundations, particularly the definition of cardinality and the seminal, if controversial, work by Cantor.

Chapter 3 - Centuries of Progress. Chapter 3 describes the centuries of progress, beginning in the 18th century, that culminated in David Hilbert's famous turn-of-the-century address in Paris. In this address, Hilbert posed a series of problems that he expected to be solved in the next century. One such problem was the "Entscheidung" - to determine the solvability of a Diophantine equation. The key notion presented in this chapter is that Hilbert was looking for a *decision procedure* or *algorithm* for determining whether a given equation was solvable or not.

Chapter 4 - The Education of Alan Turing. With mathematical preliminaries covered, the author shifts focus onto the young Alan Turing, describing in detail Turing's educational background and how he came to write his seminal paper "On Computable Numbers, with an Application to the Entscheidungsproblem." This chapter starts the dissection, presentation, and analysis of Turing's paper, covering the preamble and basic definitions. The most important aspect of this chapter is the brief discussion of circular and circle-free machines. We now refer to Turing's circle-free machines as halting machines.

Chapter 5 - Machines at Work. This chapter provides a detailed introduction to the mechanics of what we now know as the Turning machine. The machines are described using Turing's original notation, with additional supporting text used to explain the many inconsistencies. The basic operation of the machine is discussed and the major parts described. Techniques such as reading from and writing to the tape, definition of configurations, and use of configuration tables are described in great detail. Specific examples include the calculation of rational numbers.

Chapter 6 - Addition and Multiplication. Chapter 6 continues with some more complex applications of Turing machines, including addition, multiplication, and calculation of a square root. Each of these operations is described in detail and in terms of the various configurations and moves of the Turing machine. Different steps of the operations are grouped together and labeled in order to illustrate the logical structure of these basic "computer programs". This collection of configuration tables can be reused, as is discussed in the next chapter. This chapter also describes Turing's initial contact with Alonzo Church, a partnership that would prove to be most productive.

Chapter 7 - Also Known as Subroutines. The natural extension of the configuration tables discussed in the previous chapter is the labeling of these tables and subsequent use of these labels in program flow. Chapter 7 discusses this scenario, presenting a shorthand mechanism for encapsulating groups of configurations that exhibit certain repeatable functionality. By using an "abbreviated table" references to external functions can be used to refer to predefined functionality. This represents the first use of the word function in the computing literature. Also in this chapter, the concepts of variables and supplied arguments to functions are introduced.

Chapter 8 - Everything is a Number. Chapter 8 provides the theoretical underpinnings for the Universal Turing Machine. In this chapter, the author discusses the notion of a standard description of a machine, reduced to a numeric representation known as a "description number". Here Turing's work at Bletchley Park and the decoding of the Enigma machine provides some historical insight into these developments. A concrete example of the Turing machine that calculates 1/3 is provided and the notion that any computer program can be represented numerically is discussed, with Microsoft Word being used as an example. Also in this chapter, Turing's description of circle-free machines comes to fruition with the assertion that there is no algorithmic method for determining whether a given machine is circle-free or not.

Chapter 9 - The Universal Machine. This chapter builds on the previous by defining the Universal Turing Machine - a machine that can take an input and simulate another machine when supplied with a standard description. Here Turing, and the author, provide a detailed exposition of the inner workings of this universal machine at the individual cell level. This level of detail is critical to the rest of the paper and so is explored and explained in great detail. It is here that the author identifies several mistakes in Turing's original paper (corrections that were subsequently made by Turing on the advice of Emil Post and Donald Davis). The creation of the Universal Turing Machine that can be "programmed" to carry out the operation of any computing machine forces the question: "Did Turing invent the computer?"

Chapter 10 - Computers and Computability. The transition from the theoretical to the practical is discussed in this chapter, including Turing's partnership with Von Neumann, development of the EDVAC computer, and development of Turing's Automatic Computing Engine (ACE), the first general purpose computer. Prior to its invention, computers had been envisioned as specialized devices built to perform a specific task, rather than general purpose "programmable" machines. The ACE was also the first incarnation of the first binary-based machine, justified by Turing's recognition that "it is so easy to produce mechanisms which have two positions of stability." The chapter concludes with discussion of Turing's 1951 "programmers manual" for the new Mark I computer, and the application of Cantor's diagonal argument to show that there is no algorithm to identify whether a given machine is "circle-free". This represents the first formulation of what we now know as the Halting problem.

Chapter 11 - Of Machines and Men. The focus of Chapter 11 is the Turing's philosophical comparison of the human "computer" with his computing machine. In essence, a human performing a given task is reduced to a representation such that the task can be modeled on a finite-state device using an infinite tape. In preparation for the Entscheidungsproblem, Hilbert functional calculus is also discussed. From a historical perspective, Turing's view of artificial intelligence and the now-famous "Turing test" are discussed. The chapter concludes with biographical details of Turing's suicide at the age of 41, which occurred in 1954, through the ingestion of cyanide.

Chapter 12 - Logic and Computability. Chapter 12 presents a 20-page introduction to mathematical logic in preparation for the discussion of the remainder of Turing's paper. This chapter begins a discussion of Hilbert functional calculus involving a finite number of symbols and defines a computing machine in terms of first-order logic, paving the way for the next chapter on computable functions.

Chapter 13 - Computable Functions. Chapter 13 describes, in mathematical terms, the concept of computable functions, including dependent and independent variables, and domain and range. The computation of a series of functions using Turing machines is also discussed in detail. A major focus of the chapter is the introduction and proof of several theorems of computability. Dedekind's theorem for real numbers is described along with a similar version valid for computable numbers. This chapter completes all the prerequisites for the proof that the Entscheidungsproblem has no solution.

Chapter 14 - The Major Proof. This chapter represents the culmination of the paper and uses the structure Turing has constructed up to this point to prove that the Entscheidungsproblem for first-order logic has no solution. In this chapter Turing refers to the Entscheidungsproblem as general method, or *decision procedure*. Proof is based on contradiction. He first constructs a theoretical Turing machine capable to solving the problem and then proves that it is not possible to construct such a machine, thus generating a contradiction and proving that the problem is insolvable.

Chapter 15 - The Lambda Calculus. Prior to publishing his paper, Turing became aware of Alonzo Church's Lambda Calculus. Church had used Lambda calculus to prove that there is no general decision procedure for first-order predicate logic. In essence this proof was equivalent to Turing's proof. Prior to publication Turing added an appendix to his paper to elucidate and prove this equivalence.

Chapter 16 - Conceiving the Continuum. Chapter 16 describes some of the controversies and disagreements that took place in history of mathematics, particularly relating to the concept of infinity. Different mathematical/philosophical groups are described including realists (such as Gödel) and the constructivists. Much of the chapter focuses on the controversy surrounding Cantor's work on infinity and the continuum.

Chapter 17 - Is Everything a Turing Machine? This chapter discusses the impact of Turing's paper, how Turing machines relate to other, completing models, and the connection between Turing machines and the human mind. The chapter also articulates the development of automata theory, and briefly discusses the work of Kleene and Shannon. Three equivalent notions are considered: the Turing machine, Gödel's recursive functions, and Church's λ-functions. The chapter ends with a brief discussion of whether the human brain is a classical computer, or a quantum computer.

Chapter 18 - The Long Sleep of Diophantus. Although Turing and Church proved that there could be no general decision procedure for first-order logic, Hilbert's original tenth problem [3] had yet to be proven. It was eventually proven in 1970 by Julia Robinson, a mathematician from Arizona, and Yuri Matiyaysevich, a 22-year-old Russian graduate student.

3 Opinion

This book is a well-researched and extremely thorough analysis of Turing's seminal paper and as such provides a detailed commentary of the paper, clearly explaining the often confusing terminology and notation used by Turing. The author provides substantial theoretical and historical background that allows the reader to understand how Turing's work fits within the overall mathematical universe. In this respect, the book is a valuable resource and a worthy read that falls somewhere between "popular science" and graduate-level text book.

[3] Determination of the solvability of a Diophantine equation. Given a polynomial equation with any number of unknown quantities and with integer coefficients: To devise a process according to which it can be determined by a finite number of operations whether the equation is solvable in integers.

It is easy to take Turing's work for granted as several decades of Computer Science education has solidified both our understanding and acceptance of this innovative work. At the time of publication, Turing's work was completely new, and this is somewhat reflected in the inconsistent use of notation in the paper. This unfortunately results in arguably the most important contribution to modern Computer Science being largely ignored by most undergraduate and graduate students. Most cull a basic understanding of the key results from standard "theory of computation" text books.

To a large extent, Petzold succeeds in bringing Turing's paper to life, and certainly provides a comprehensive running commentary that guides the reader through Turing's paper. It should be noted that while this is a well-written book that clearly achieves its goals in a clear and occasionally humorous fashion, it is in no sense meant for a general audience. Rather, the interested reader will most likely be the general Computer Science graduate student who is looking for an accessible, yet rigorous introduction to Turing's work. The book will also be a useful complement to a course in the theory of Computation, filling in many of the details that are often glossed over due to limits of time.