CHURCH-TURING THESIS: THE TURING IMMORTALITY PROBLEM SOLVED WITH A DYNAMIC REGISTER MACHINE

Introduction SECTION 0

The results demonstrated here address the foundations of mathematics and computer science, building upon the prior work of Hilbert, Godel, Church, Kleene, Turing, Post and others. See [HILBERT], [GODEL], [CHURCH], [KLEENE], [TURING] and [POST].

New results are reached by studying the Turing machine from a dynamical systems point of view, focusing on the periodic behavior of the Turing machine. Classical dynamical systems theory has been successful in the past by studying periodic behavior. See [ARNOLD], [BROUWER], [BOWEN], [HOPF], [POINCARE], and [SMALE].

A method for solving the Turing Immortality Problem [HOOPER] is presented at the end of Section 8 based on the mathematical results developed in Sections 1 through 8. In Section 9, a *dynamic register machine* is defined as an extension of the classical register machine developed in [STURGIS]. In Section 10, the design of the dynamic register machine program presented in section 11 is explained. In section 11, a dynamic register machine program, called the IDRM, composed of 1590 *dynamic register machine instructions* is presented that can determine in a finite number of computational steps whether a given Turing machine has any immortal configurations.

In light of Hooper's proof [HOOPER] that the Turing Immortality problem is Turing machine-undecidable, it is concluded that this *constructive demonstration* of a dynamic register machine program proves that the Church-Turing thesis is false.