Giuseppe Primiero
Department of Philosophy, University of Milan

On Computational Hypotheses and Computational Experiments

Abstract

The notion of experimental computer science can be traced back to the report of a workshop sponsored by the National Science Foundation (US), the so-called Feldman Report from 1979. A report following the biennial meeting of Computer Science Department Chairmen held at Snowbird, Utah in 1980, presents computer science as both a theoretical and an experimental science.

Since then, the notion of experiment in CS started to appear, but often used with different meanings by different people. some decades after these early attempts at characterizing the experimental nature of computer science, the analogy between the scientific method and the problem-solving process underlying computing still is a tempting proposition.

In this talk, we attempt a clarification of the characteristics of hypotheses and experiments in the context of the computational sciences. In this context, the scientific method seems to offer a parallel with the algorithmic method in computer science: the scientific method consists in formulating a hypothesis for explaining a phenomenon, testing the hypothesis by conducting an experiment, and finally in confirming or rejecting the hypothesis by evaluating the results of the experiment; analogously, problem-solving in computing can be seen as formulating an algorithm for solving a problem, testing the algorithm by writing and running a program, and finally in accepting or rejecting the algorithm by evaluating the results of running the program.

But the properties of the notions of hypothesis and experiment required by a methodological reading of computing still need to be qualified in detail. We overview the origin of the term “experimental Computer Science”, the early positions on experiments in this discipline and introduce two basic notions of computational hypothesis and computational experiment essential to the understanding of computing as an experimental discipline.