Journal of Automata, Languages and Combinatorics **23** (2018) 1–3, 111 – 126

WHAT IS A COMPLEX REGULAR LANGUAGE?

Cezar Câmpeanu

School of Mathematical and Computational Sciences
University of Prince Edward Island
550 University Ave., C1A 4P3 Charlottetown, Canada
ccampeanu@upei.ca

ABSTRACT

If two regular languages have both the state complexity equal to n, how can we further say that one of them is more complex than the other one? In other words, when can we say that a regular language is complex? In this paper, we try to answer this question, by proposing a few possible measures that we can use to quantify the relative complexity of regular languages and compare all these proposals with the classical definition of Kolmogorov complexity.

Keywords: regular languages, state complexity, Kolmogorov complexity, Blum static complexity spaces, encoded spaces, randomness, distinguishability

1. Introduction

In classical algorithmic information theory, we usually work with strings or sequences over a finite alphabet, and we define the complexity of a word as the length of a minimal program or machine that can output that word [18, 19, 25]. We can distinguish a few varieties of definitions for descriptional complexity, depending on the class of machines, programs considered, and some characteristics of the encodings: plain complexity [25], prefix complexity [19, 22, 26], process complexity [30], monotone complexity [26], uniform complexity [27], Chaitin's complexity [20], or Solomonoff's universal complexity [31, 32, 33].

In all these definitions, the principle of minimal description length (MDL) is involved: an object is declared to have a high complexity or is said to be random if the length of its shortest description is one of the highest among the other objects of the same size. For example, if you consider all Turing Machines encoded over a binary alphabet, you can declare a string w to be t-random, $t \in \mathbb{N}$, if the length of the shortest encoding of a Turing machine producing w as output is at least $|w| - t$, where $|w|$ is the length of the string w.

Therefore, we can see that the size of computing machines is involved in the definition of complexity. Blum captured this idea in [3], where he is proposing a set of axioms for defining complexity. He said that "these axioms are all so fantastically