4 C − 1

Complexity Analysis of Boolean Functions by Information Entropy

Nakagawa Akinari, Yoshiyuki Morii, and Hiroshi Hagiwara
Faculty of Science and Technology, Ryukoku University

1. Introduction

We shall present here a new scale of complexity of Boolean functions. This measure is based upon information entropy, and thus will be called *information complexity*. In fact, our definition of information complexity will be given in terms of 'conditional mutual information.' Although we restrict ourselves to the complexity of Boolean functions, information complexity is also applicable to the investigation on the complexity of any type of functions. Being defined in terms of information entropy, it can be considered as an ultimate scale of complexity of functions over probabilistic variables. As a specific application, we shall be involved in evaluating the complexity of arithmetic operations on computers. An arithmetic operation between two n-bit numbers can be regarded as a collection of n-Boolean functions having $2n$-input Boolean variables. Complexity of arithmetic operations can thus be derived from that of Boolean functions. We shall here clarify the necessity of introducing this new scale of complexity of Boolean functions, illustrate the essential idea on information complexity, and make some remarks upon representation systems of numbers on which arithmetic operations are defined

2. Information Complexity

A widely used measure scaling the complexity of a Boolean function is that by means of the minimum size of (logical) circuits computing it: *circuit complexity*. This tradition originated from Shannon. In recent literatures, the circuit complexity of Boolean functions has been investigated extensively in relation to algorithmic complexity. However, we do not adopt this measure for our study of complexity. The reason is threefold. First, it depends on a specific selection of a logical basis for the construction of circuits. Second, there is (has not been given) no criteria to choose, so-called, 'universal' basis for the investigation of inherent nature of Boolean functions. Third, even if we fix a logical basis, it is known to be difficult in general to determine the actual minimum size of a circuit to compute a specific function; it is indeed true that there have been several reports devoting themselves to the circuit complexities of arithmetic functions and some of them has succeeded to give lower bounds of circuit complexity, though. Thus, we need a (-n easily computable, if possible) scale of complexity inherent in each Boolean function. Shannon has also made another suggestion in this respect: *information entropy*. Inspired by the concept of information entropy, we shall introduce *information complexity* as for the measure of complexity of Boolean functions; it may be considered as a natural characterization intrinsic to each Boolean function.

We outline the essential idea on information complexity here. For a Boolean function f having an input variable x, the information complexity of f concerning x is to be defined depending upon the degree of the information on x included in f, or, put equivalently, upon the degree of the dependency of f on x. Thus, for example, the information complexity of '$f = x$' concerning x is 1, while that of '$f =$ TRUE' is 0. Notice here that it is not so immediate to conclude the former complexity to be 1. In the discussion above, we in fact assume implicitly that x has a value from one of the two alternatives, TRUE or FALSE, and each has an equal probability of occurrence. Hence, the entropy of this probability distribution is to be given by $h(1/2) = 1$, where

$$h(p) = -p \log_2 p - (1-p) \log_2 (1-p)$$

is the entropy in the case of two possibilities with probabilities p and $1-p$. For a different probability distribution, the complexity of '$f = x$' given above has a value other than 1 accordingly. Observe that 1 is the maximum value of entropy that provability distributions of two possibilities can attain. Then, consider the next example such that '$f = x$ AND y.' In this case, x has full information on f if '$y =$ TRUE,' whereas no information if '$y =$ FALSE.' Thus, the information complexity of '$f = x$ AND y' concerning x must be $(1+0)/2 = 1/2$. Let the 'total' information complexity of a given Boolean function be the sum of information complexities concerning each variable. Thereby, for '$f = x$ AND y,' it is given by $(1/2) + (1/2) = 1$. Here, the simple summation regarding to x and y is to be justified because both input variables are mutually independent.

3. Complexity Analysis

We shall utilize the power of information complexity for analyzing the complexity of Boolean functions in order to evaluate quantitatively the degree of complexity of arithmetic operations on computers. We here comment on representation systems of numbers on which arithmetic operations are defined. Widely used representation systems of real numbers at present, such as IEEE standard, are based upon a floating-point expression and allocate a fixed number of bits for representing an exponent field. Defining a real-number representation system can be considered as a problem to scatter a fixed number of points onto the real-number line. To optimize the distribution of these finite number of points, several proposals have been made with variable-length representation of an exponent field necessarily employed. Among them, two notable and successful ones are the Hamada's system (URR, as he calls) and its extension by Yokoo, and *level-index system* developed by Clenshaw, Olver, Turner and their colleagues. Recently, through the investigation on properties of these representation systems, Akinari and Hagiwara [1] have reached the conclusion that recommended is one of the extensions of the Hamada's system, that is, $H_2(x)$ system, in their notation. By adopting double exponentation, the Hamada's original system, $H_1(x)$ system, in the same notation, has succeeded to embrace the specification of the bit length of an exponent field into the field itself gracefully. While it is possible to reduce the bit length of an exponent field about half by the use of 'triple' exponentation ($H_2(x)$ system), the reduction cannot be sharpened further by the additional multiplication in exponentation ($H_i(x)$ systems for $i = 3, 4, 5, \ldots$). On the other hand, the $H_2(x)$ system can be regarded as one of the variants of level-index system with maximal level being 2. Owing to the fact that the original level-index system adopts base e for exponentation and thus employs 'mantissa' rather than 'fraction,' even small integers, such as 2 or 3, cannot be exact in level-index system. Thus, the $H_2(x)$ system holds a special position among real-number representation systems having a variable-length exponent field. As an inevitable consequence of its elegance in representation, the arithmetic operations of $H_2(x)$-system numbers, however, are expected to be harder than, for example, that of IEEE-standard numbers or of integers. Our original motivation of this study has in fact been to evaluate the degree of this difficulty in arithmetics of $H_2(x)$-system numbers quantitatively. Having thus introduced information complexity for this purpose, we have, however, reached the conclusion opposing the expectation; that is, the arithmetic manipulation of $H_2(x)$-system numbers is not complex at all as compared even with integer addition. Moreover, the arrangement of integer arithmetics such as addition, multiplication, and division has been found to be in order not of simplicity, but of complexity.

References

[1] N. Akinari, and H. Hagiwara, On the real-number representation with variable-length exponent field, *Inform. Proc. Lett.* **52** (1994) 1–6.