Journal of Computer and System Sciences 79 (2013) 152–172

Contents lists available at SciVerse ScienceDirect

Journal of Computer and System Sciences

www.elsevier.com/locate/jcss

Probabilistic automata for computing with words

Yongzhi Cao [a,b,*], Lirong Xia [c], Mingsheng Ying [d,e]

[a] Institute of Software, School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China
[b] Key Laboratory of High Confidence Software Technologies, Ministry of Education, China
[c] SEAS, Harvard University, Cambridge, MA 02138, USA
[d] Center for Quantum Computation and Intelligent Systems, Faculty of Engineering and Information Technology, University of Technology, Sydney, Australia
[e] State Key Laboratory of Intelligent Technology and Systems, Tsinghua National Laboratory for Information Science and Technology, Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China

ARTICLE INFO

Article history:
Received 22 April 2006
Received in revised form 30 September 2011
Accepted 19 September 2012
Available online 3 October 2012

Keywords:
Computing with words
Extension principle
Probabilistic automata
Probabilistic grammars

ABSTRACT

Usually, probabilistic automata and probabilistic grammars have crisp symbols as inputs, which can be viewed as the formal models of computing with values. In this paper, we first introduce probabilistic automata and probabilistic grammars for computing with (some special) words, where the words are interpreted as probabilistic distributions or possibility distributions over a set of crisp symbols. By probabilistic conditioning, we then establish a retraction principle from computing with words to computing with values for handling crisp inputs and a generalized extension principle from computing with words to computing with all words for handling arbitrary inputs. These principles show that computing with values and computing with all words can be respectively implemented by computing with some special words. To compare the transition probabilities of two near inputs, we also examine some analytical properties of the transition probability functions of generalized extensions. Moreover, the retractions and the generalized extensions are shown to be equivalence-preserving. Finally, we clarify some relationships among the retractions, the generalized extensions, and the extensions studied by Qiu and Wang.

1. Introduction

To capture the notion of automated reasoning involving linguistic terms, Zadeh has advocated the idea of computing with words (CW) [23,25–29]. The objects of CW are words and propositions which describe perceptions in a natural language, where the words play the role of labels of perceptions and the meaning of a proposition is expressed as a generalized constraint. Many basic types of constraints have been given by Zadeh; among others, possibilistic constraint characterized by fuzzy sets (possibility distributions) and probabilistic constraint characterized by probabilistic distributions are two most familiar ones. As a methodology, CW has provided a foundation for dealing with imprecise, uncertain, and partially true data which have the form of propositions expressed in a natural language; see [8,10,21,30] for some applications.

Based upon the generalized constraints, one can handle some computations and uncertain reasoning on perceptions. However, in the traditional sense computing is centered on the manipulation of numbers or symbols, and is usually represented by a dynamic model in which an input device is equipped. It is well known that various automata and Petri nets are the prime examples of classical computational systems. Note that the inputs of such models are exact rather than vague data, and thus they cannot serve as formal models of CW. This observation motivated Ying [22] to interpret CW as a computational procedure and propose a formal model of CW in terms of fuzzy automata. The key idea underlying Ying's

* Corresponding author at: Institute of Software, School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China.
E-mail addresses: caoyz@pku.edu.cn (Y. Cao), lxia@seas.harvard.edu (L. Xia), mying@it.uts.edu.au (M. Ying).

0022-0000/$ – see front matter
http://dx.doi.org/10.1016/j.jcss.2012.09.002

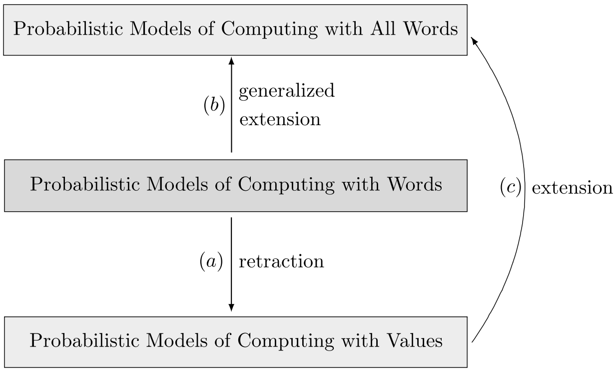

Fig. 1. Interrelation among retractions, extensions, and generalized extensions.

formal model of CW is the use of words in place of values as input symbols of a fuzzy automaton. Further, Wang and Qiu extended the concept of CW to fuzzy Turing machines in [19] and investigated the formal theory of CW in the framework of probabilistic automata and probabilistic grammars [13].

Essentially, the buildings of all the formal models of CW in [13,19,22] go as follows: beginning with a classical computational model with values as inputs and then deriving a formal model with all words (interpreted as probabilistic distributions or possibility distributions) as inputs. From a different perspective, Cao et al. introduced the notion of fuzzy automata for CW (FACWs) in [3]. An FACW is a fuzzy automaton where the input alphabet consists of finite words (possibility distributions) over some crisp symbols. In order to deal with arbitrary words that may be not in the input alphabet of an FACW as inputs, they established the so-called retractions and generalized extensions of FACWs by exploiting the methodology of fuzzy control. Most recently, Cao and Chen developed a concurrency model of CW, fuzzy Petri nets for CW, and compared the language expressiveness of this model with that of FACWs in [2].

The purpose of this paper is to build a general probabilistic model of CW. We introduce probabilistic automata for CW (PACWs) and as well probabilistic grammars for CW (PGCWs) to model formally CW in a probabilistic framework. Probabilistic automata and probabilistic grammars have been studied since the early 1960s [12]. Relevant to our line of interest is the work of Rabin [14]. In the present paper, the words that represent generalized constraints are interpreted as probabilistic distributions or possibility distributions over some crisp symbols. We may think that PACWs and PGCWs are specified by experts, in which only finite words are considered. For example, an expert may express his opinion on a repeated risk investment of a firm in the following proposition: If the firm is in a good situation and if it invests in the projects A and B with probabilities 0.7 and 0.3, respectively, then it will be still in the good situation with probability 0.9 while in a bad situation with probability 0.1. Based upon some analogous propositions, we can build a PACW or PGCW to represent the expert's opinions. In practice, in many areas expert's opinions may be naturally expressed in terms of linguistic uncertainties. Clearly, it is desirable if from such knowledge, we can make inferences about some particular actions that are not specified by the experts. For instance, one may want to assess the situations of the firm when it invests in the projects A and B with probabilities 0.75 and 0.25, respectively, or it invests in the project A with probability 1. This motivates us to consider the so-called retractions and generalized extensions of PACWs and PGCWs.

Roughly speaking, the retraction of a PACW is a probabilistic automaton, called probabilistic automaton for computing with values (PACV), that has crisp symbols as inputs; the generalized extension of a PACW is another probabilistic automaton, called probabilistic automaton for computing with all words (PACAW), that can accept any words as inputs (see Fig. 1). As we will see, the extension from PACVs to PACAWs developed in [13] is a specific case of generalized extensions. By probabilistic conditioning rather than the methodology of fuzzy control used in [3], we establish a retraction principle from CW to computing with values for dealing with crisp inputs and a generalized extension principle from CW to computing with all words for dealing with arbitrary inputs. These principles show that in the probabilistic framework, computing with values and computing with all words can be respectively implemented by computing with some special words. From a modeling viewpoint, the generalized extensions enable infinitely possible inputs to be represented by finite inputs by means of interpolation. Analogously, we investigate the retractions and the generalized extensions of PGCWs. Furthermore, we show that the retractions and the generalized extensions preserve all the three kinds of equivalences among PACWs and PGCWs consisting of the equivalence between PACWs, the equivalence between PGCWs, and the equivalence between PACWs and PGCWs.

The rest of the paper is organized as follows. In Section 2, after presenting some preliminaries in probabilistic automata and probabilistic grammars, we introduce two probabilistic models of CW, PACWs and PGCWs. The retractions of PACWs are established in Section 3. We develop the generalized extensions of PACWs and discuss some related analytical properties in Section 4. Section 5 is concerned with the retractions and the generalized extensions of PGCWs, and Section 6 is devoted to the equivalence preservation under the retractions and the generalized extensions. Some relationships among the retractions, the generalized extensions, and the extensions in [13] are explored in Section 7. We conclude the paper and identify some future research directions in Section 8.