Information Processing Letters 114 (2014) 147–151

Contents lists available at ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

Modeling time criticality of information ☆

Branislav Rovan [a], Marek Zeman [a,*]

[a] *Department of Computer Science, Comenius University, Mlynska Dolina, 84248, Bratislava, Slovakia*

ARTICLE INFO

Article history:
Received 7 May 2013
Received in revised form 21 October 2013
Accepted 24 October 2013
Available online 26 October 2013
Communicated by J. Torán

Keywords:
Formal languages
Information
Time criticality
Advice
Finite automata

ABSTRACT

In this paper, we continue the research on formal treatment of attributes of information, based on the computational approach. In this scenario, the usefulness of advisory information is measured by the decrease in complexity of a problem we need to solve. We propose to model the time criticality via usefulness of a piece of information which is received during the computation. As a modeling tool, we use deterministic finite automata. We give two definitions of time criticality. In the static case, we consider supplementary information which concerns the entire input instance. In the dynamic case, we consider information about the unprocessed part of the input. Despite the simplicity of our model, we shall see that the development of time criticality may exhibit an interesting behavior.

1. Introduction

The research on formal treatment of attributes of information was initiated by Gaži, Rovan and Steskal in [1–3]. Building on the notion of usefulness of information we present a formal approach to the time criticality of information.

Shannon's definition of information [4] was motivated by problems related to information transmission and provides a measure of "the amount" of information. Algorithmic information theory studies the information content in strings by measuring the complexity of their description by machines [5]. The idea of providing "additional information" appeared in probability theory, automata theory (e.g. promise problems [6]) and more recently in on-line algorithms [7,8].

A formal study of the informally used notion of "usefulness" of information was initiated in [1] and later elaborated in [2] and [3]. The usefulness of information is measured by the decrease in complexity of solving a given problem using the information provided, i.e., a computational point of view on the information usefulness is used.

In this paper, we consider another informally used notion not studied yet – the notion of "time criticality" of information. We attempt to capture the fact that the same information may be more or less useful depending on the moment when it is received. We shall use the automata theory setting. The key issue is the choice of the measure of time criticality. We shall base it on the notion of usefulness, as formalized in the work of Steskal [2].

We shall formalize time criticality using the following scenario. Suppose we have a set of machines, which are able to perform a set of algorithms. We are solving a particular instance of a computational problem. During the computation, we obtain some supplementary information (advice) about the input instance. This information might enable us to use a different algorithm (i.e., a different machine) to finish the computation. It may be more or less simple depending on the time when the advice is received.

The setting: The problem to be solved is a regular language L given by a finite automaton A_L. The additional information is provided during the computation of A_L on a given word w via another language L_{adv}. In principle, we could consider L_{adv} to be an arbitrarily complex language.

☆ This work was partially supported by the grant VEGA 1/0979/2012 and the grant UK 242/2013.

* Corresponding author. Tel.: +421 2 60295837.
E-mail addresses: rovan@dcs.fmph.uniba.sk (B. Rovan), zemanm@dcs.fmph.uniba.sk (M. Zeman).

0020-0190/$ – see front matter
http://dx.doi.org/10.1016/j.ipl.2013.10.008

This could render the remaining task in accepting w trivial. We therefore do not want to use advice which is more complex than the original problem. We shall thus focus on utilizing regular advice in this paper.

Our goal is to analyze possible behaviors of usefulness of information as a function of the time when the information is received. We do not aim to deal with particular problems or algorithms. We shall focus on the very substance of the aspect and demonstrate some of its properties.

In Section 2 we give our definition of the time criticality using deterministic finite automata. In Section 3 we show some basic properties of the time criticality. In Section 4 we discuss an alternative definition of time criticality and its aspects.

2. Time criticality of information in finite automata

We shall use the standard notation from automata theory [9]. A deterministic finite automaton is a 5 tuple $(K, \Sigma, \delta, q_0, F)$ with the usual meaning of its components. The transition function δ is required to be total in this article. We denote components of an automaton A by using the subscript A, i.e., $(K_A, \Sigma_A, \delta_A, q_A, F_A)$. We shall extend the transition function δ to words over Σ as usual. Let $R_A(q)$ denote the set of states in A reachable from the state q. The length of a word w is denoted by $|w|$. The number of elements of a set S is denoted by $|S|$. We denote the empty word by ε.

Let A_L denote the minimal automaton for a regular language L. We may omit the index L when it is clear from the context.

Denote by $\mathcal{C}(A)$ the complexity of A. In this paper we shall measure the complexity of A by the number of its states. Thus $\mathcal{C}(A) = |K_A|$. We shall define the complexity of a regular language L by $\mathcal{C}(L) = \mathcal{C}(A_L)$.

In order to define the time criticality of information we shall use the following scenario. Let A be an automaton. Consider the computation of A on $w = uw'$. After processing the prefix u the supplementary information $w \in L_{adv}$ is received. Based on this information it is possible to switch to a (simpler) automaton A', which in case $w \in L(A) \cap L_{adv}$ reaches an accepting state on w'.

We shall first consider the case where the supplementary information is static in the sense that it always concerns the entire input word w. In Section 4 we shall consider the dynamic case where the supplementary information concerns w', the remaining part of the input word.

More formally, let L_{adv} be an advisory language and let A be an automaton. If we receive the information the input word belongs to L_{adv} during the computation, we can switch to a new (simpler) automaton A' satisfying the following properties for every $v \in \Sigma_A^*$: $uv \in L \cap L_{adv} \Rightarrow v \in L(A')$ and $uv \in L^C \cap L_{adv} \Rightarrow v \notin L(A')$. It may behave arbitrarily on words not from L_{adv} because it does not affect its correctness. However, we can also receive the information the input word does not belong to L_{adv}. In this case, we can switch to a new (simpler) automaton A'' satisfying for every $v \in \Sigma_A^*$: $muv \in L \cap L_{adv}^C \Rightarrow v \in L(A'')$ and $uv \in L^C \cap L_{adv}^C \Rightarrow v \notin L(A'')$.

Example 2.1. Let L be a language of words consisting of at least two subwords bc and let L_{bc} be a language of words, where each b is followed by c. Let A be the minimal automaton accepting L. The computation of A can be divided into three phases. In the first phase, we wait for the first b which is followed by c. If we receive the advisory information $w \in L_{adv}$ during this phase, it suffices to check weather there are at least two b's in the input word, so instead of five, we can finish the computation using only three states.

If we do not receive the information during the first phase and A verifies the occurrence of the first bc in the input word, it uses just three states to finish the computation. If the advisory information is received at this point, we would be able to finish the computation using an automaton with just two states.

The preparation for using the additional information L_{adv} may "cost" something. The complexity of preparation for using $w \in L_{adv}$ after t steps may differ from the complexity of preparation for using $w \notin L_{adv}$ after t steps. To compute the worse case, we shall use maximum of these two values. Note that for a particular w, only one of these values exists and makes sense. However, the automaton A has processed only the first t symbols and cannot distinguish the two cases.

For a given automaton A, an input word w and an advisory language L_{adv} we shall define time criticality of L_{adv} as a function $T_{A,w,L_{adv}}(t)$ of time, when the information is received, measuring the usefulness of this information via a possible decrease in the number of states needed to finish the computation.

Definition 2.1. Let A be an automaton, L_{adv} be an advisory language and w be an input word. Let $w = a_1a_2 \dots a_n$, each $a_i \in \Sigma_A$. Let w_t denote $a_1a_2 \dots a_t$ and let $\delta_A(q_A, w_t) = q_t$. We shall define *time criticality* by

$$T_{A,w,L_{adv}}(t) = \frac{|R_A(q_t)| - \max(\mathcal{C}(L_{A,w,L_{adv}}(t)), \mathcal{C}(L_{A,w,L_{adv}^C}(t)))}{|R_A(q_t)|}$$

where $\mathcal{C}(L_{A,w,L_{adv}}(t))$ resp. $\mathcal{C}(L_{A,w,L_{adv}^C}(t))$ represent the complexities of problems to be solved after the piece of supplementary information $w \in L_{adv}$ resp. $w \notin L_{adv}$ is received at time t. $L_{A,w,L_{adv}}(t)$ (similarly $L_{A,w,L_{adv}^C}(t)$) can be solved by any automaton A' which satisfies

(i) $w_t w' \in L(A) \cap L_{adv} \Rightarrow w' \in L(A')$,
(ii) $w_t w' \in L(A)^C \cap L_{adv} \Rightarrow w' \notin L(A')$,

and minimizes $\mathcal{C}(A')$.

Thus, time criticality is a function giving the ratio of the number of states we can spare by obtaining the information at time t to the number of states required without the supplementary information. In our scenario, the obtained information is either $w \in L_{adv}$ or $w \in L_{adv}^C$. We consider the case when the information helps less. Note that time criticality is a rational number between 0 (inclusive) and 1.