Turing Machine Representation in Temporal Concept Analysis

Karl Erich Wolff[1] and Wendsomde Yameogo[2]

[1] Mathematics and Science Faculty,
Darmstadt University of Applied Sciences
Schoefferstr. 3, D-64295 Darmstadt, Germany
karl.erich.wolff@t-online.de
http://www.fbmn.fh-darmstadt.de/home/wolff
[2] Independent Researcher
wendsomde@yameogo.com
http://www.yameogo.com

Abstract. The purpose of this paper is to investigate the connection between the theory of computation and Temporal Concept Analysis, the temporal branch of Formal Concept Analysis.

The main idea is to represent for each possible input of a given algorithm the uniquely determined sequence of computation steps as a life track of an object in some conceptually described state space. For that purpose we introduce for a given Turing machine a Conceptual Time System with Actual Objects and a Time Relation (CTSOT) which yields the state automaton of a Turing machine as well as its configuration automaton. The conceptual role of the instructions of a Turing machine is understood as a set of background implications of the derived context of a Turing CTSOT.

1 Introduction: Computing as a Temporal Activity

Computing is usually understood as a temporal activity, starting with a certain input which is transformed in several steps into the output of the computation. Usually, the common mathematical representations of computations do not make explicit its temporal aspects. For example, while the notion of *time* is employed in nearly all informal descriptions of computations, it is usually not specified in mathematical representations of computations. Similarly the notion of *state* is often used only informally and not as a mathematical term in some specified temporal theory. It is well-known that the notion of *state* is chosen as a primitive notion in the definition of an automaton, but automata theory does not have an explicit time representation [Arb70, Eil74, Mal74]. The notions of *initial state*, *final state* and *transition* emphasize its temporal interpretation as well as the notion of a *successful path* leading from an initial to a final state.

Clearly, the notion of *state* is also used in many other sciences, for example in classical mechanics where the trajectories of particles seem to be quite similar to the paths in automata theory. This similarity has been made explicit in Temporal

B. Ganter and R. Godin (Eds.): ICFCA 2005, LNAI 3403, pp. 360–374, 2005.

Concept Analysis by the introduction of the notion of *state* in a Conceptual Time System (CTS) [Wol00a] and the notion of the *life track of an object* in a Conceptual Time System with Actual Objects and a Time Relation (CTSOT) [Wol02b].

It was shown in the Map Reconstruction Theorem in [Wol02b] that each automaton is isomorphic to an automaton within a suitable CTSOT such that the paths of the given automaton are mapped onto the life tracks of objects of the CTSOT.

In the present paper we show that any Turing machine [Tur36, Loe76] can be represented by a CTSOT together with a specified set of *background implications* interpreting the instructions of the given Turing machine. That temporal representation preserves the state automaton of the given Turing machine as well as its tape automaton (Theorem 1 in section 6).

2 Turing Machines

Alan M. Turing [Tur36, Tur36a] and simultaneously E.L. Post [Pos36] investigated "computing machines" using simple mathematical models of a computer. Turing wrote in the first section of his paper [Tur36]:

> Quote 1:
> *We may compare a man in the process of computing a real number to a machine which is only capable of a finite number of conditions q_1, q_2, ..., q_R which will be called "m-configurations". The machine is supplied with a "tape", (the analogue of paper) running through it, and divided into sections (called "squares") each capable of bearing a "symbol". At any moment there is just one square, say the r-th, bearing the symbol S(r) which is "in the machine". We may call this square the "scanned square". The symbol on the scanned square may be called the "scanned symbol". The "scanned symbol" is the only one of which the machine is, so to speak, "directly aware".*

These "computing machines" are now well-known under the name "Turing machines". There are many slightly different formal definitions of Turing machines. We assume that the reader is familiar with the main ideas and standard notions.

2.1 Definition of a Turing Machines

We first recall an often used definition of a Turing machine [Loe76, HU79]).

Definition 1. ***"Turing Machine"***
A Turing machine is a tuple

$$\mathfrak{T}_0 := (Q, q_0, \Gamma, B, \Sigma, \delta)$$

*where Q is a finite set, called the set of **states**, $q_0 \in Q$ is called the **initial state**, Γ is a finite set, called the set of **tape symbols**, B is an element of Γ,*