Information and Computation 244 (2015) 49–75

Contents lists available at ScienceDirect

Information and Computation

www.elsevier.com/locate/yinco

The dual equivalence of equations and coequations for automata

A. Ballester-Bolinches [a,*], E. Cosme-Llópez [a], J. Rutten [b,c]

[a] Departament d'Àlgebra, Universitat de València Dr. Moliner, 50, 46100 Burjassot, València, Spain
[b] CWI Science Park, 123, 1098 XG Amsterdam, The Netherlands
[c] Radboud Universiteit Nijmegen, Heyendaalseweg, 135, 6500 GL Nijmegen, The Netherlands

ARTICLE INFO

Article history:
Received 26 May 2014
Received in revised form 29 May 2015
Available online 18 August 2015

Keywords:
Automata
Languages
Algebra
Coalgebra

ABSTRACT

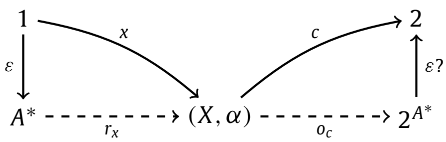
The transition structure $\alpha : X \to X^A$ of a deterministic automaton with state set X and with inputs from an alphabet A can be viewed both as an algebra and as a coalgebra. We use this algebra–coalgebra duality as a common perspective for the study of equations and coequations. For every automaton (X, α), we define two new automata: $\mathsf{free}(X, \alpha)$ and $\mathsf{cofree}(X, \alpha)$ representing, respectively, the greatest set of equations and the smallest set of coequations satisfied by (X, α). Both constructions are shown to be functorial. Our main result is that the restrictions of free and cofree to, respectively, preformations of languages and to quotients A^*/C of A^* with respect to a congruence relation C, form a dual equivalence. As a consequence, we present a variant of Eilenberg's celebrated variety theorem for varieties of monoids (in the sense of Birkhoff) and varieties of languages.

1. Introduction

In this paper, a deterministic automaton is a pair (X, α) consisting of a possibly infinite set X of states and a transition function $\alpha : X \to X^A$, with inputs from an alphabet A. Because of the isomorphism

$$(X \times A) \to X \quad \cong \quad X \to X^A$$

a deterministic automaton can be viewed both as an algebra [1,2] and as a coalgebra [3,4]. This algebra–coalgebra duality in the modelling of automata leads us to the following setting for our investigations:



(1)

In the middle, we have our automaton (X, α). Any function $x : 1 \to X$ represents the choice of a designated *point*, that is, initial state, $x \in X$. Dually, any function $c : X \to 2$ gives us a (binary) *colouring* of the states in X or, equivalently, a set

* Corresponding author.
E-mail addresses: adolfo.ballester@uv.es (A. Ballester-Bolinches), enric.cosme@uv.es (E. Cosme-Llópez), jjmmrutten@gmail.com (J. Rutten).

http://dx.doi.org/10.1016/j.ic.2015.08.001
0890-5401/

$\{x \mid c(x) = 1\}$ of final or accepting states. On the left side of our diagram, A^* is the automaton of all words over A, with transitions

$$v \xrightarrow{a} va$$

and with the empty word ε as initial state. Furthermore, every point $x : 1 \to X$ determines a unique *homomorphism* (that is, transition preserving function)

$$r_x \colon A^* \to X \qquad w \mapsto x_w$$

that sends any word w to the state x_w reached from the initial state x on input w. Dually, on the right side of our diagram, 2^{A^*} is the automaton of all languages over A, with transitions

$$L \xrightarrow{\quad a \quad} L_a = \{v \in A^* \mid av \in L\}$$

and colouring function $\varepsilon?$, asking whether the empty word belongs to a language or not

$$\varepsilon?(L) = \begin{cases} 1 & \text{if } \varepsilon \in L \\ 0 & \text{if } \varepsilon \notin L \end{cases}$$

Every colouring $c : X \to 2$ determines a unique homomorphism

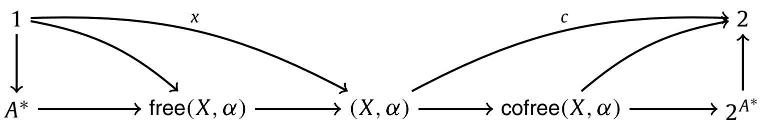
$$o_c \colon X \to 2^{A^*} \qquad x \mapsto \{w \in A^* \mid c(x_w) = 1\}$$

that sends a state x to the language that it accepts.

As it turns out, a pointed automaton (X, x, α) is an algebra (and not a coalgebra); a coloured automaton (X, c, α) is a coalgebra (and not an algebra). And a pointed and coloured automaton (X, x, c, α), which is what in the literature is usually taken as the definition of 'deterministic automaton', is neither an algebra nor a coalgebra.

Now sets of *equations* will live in the left – algebraic – part of our diagram and correspond to the *kernels* of the homomorphisms r_x; that is, sets of pairs of words (v, w) with $x_v = x_w$. Dually, sets of *coequations* live in the right – coalgebraic – part of our diagram and correspond to the *image* of the homomorphisms o_c; that is, sets of languages containing $o_c(x)$, for every $x \in X$. Satisfaction of sets of equations and coequations by the automaton (X, α) will then be defined by quantifying over all points $x : 1 \to X$ and all colourings $c : X \to 2$, respectively.

The main contribution of the present paper will be the observation that equations and coequations of automata are related by a dual equivalence. To this end, we will further refine diagram (1) as follows:



The new diagram includes, for every automaton (X, α) a new automaton free(X, α), which will be shown to represent the *largest set of equations* satisfied by (X, α). And, dually, we will construct an automaton cofree(X, α), which will represent the *smallest set of coequations* satisfied by (X, α). The automaton free(X, α) will turn out to be isomorphic to the so-called *transition monoid* from algebraic language theory [5,6] and as a consequence, cofree(X, α) can be viewed as its dual.

Next, we will show that the constructions of free(X, α) and cofree(X, α) are in fact functorial, that is, they act also on (certain) homomorphisms of automata. If we then restrict the functor cofree to the image of the category of automata under free, we obtain our main result: a dual equivalence. This dual equivalence relates, more precisely, two special classes of automata: on the one hand, the class of quotients A^*/C of the automaton A^* with respect to a congruence relation $C \subseteq A^* \times A^*$; on the other hand, the class of preformations of languages, which in the present paper are defined as subautomata of the automaton 2^{A^*} that are complete atomic Boolean algebras closed under left and right language derivatives. As it turns out, this duality is a lifting of the well-known dual equivalence between sets and complete atomic Boolean algebras: on congruence quotients, cofree acts as the powerset construction, and on preformations, applying free amounts to taking the set of atoms.

We then illustrate the dual equivalence between equations and coequations by applications to both regular languages and non-regular ones, such as context-free languages. Furthermore, we will show how to use the duality to give (co)equational definitions of interesting classes of languages, again not restricted to regular ones. We also present a variant of Eilenberg's celebrated variety theorem [2]. We replace pseudovarieties in the original work of Eilenberg by varieties of monoids (in the sense of Birkhoff [7]). Further, we replace varieties of regular languages by varieties of languages, which are classes of formal languages closed under some properties defined in terms of equations and coequations. Following the spirit of the original result by Eilenberg, we prove that there is a one-to-one correspondence between varieties of monoids and varieties of languages. Finally, we introduce the notion of *equational bisimulation* and a corresponding coinduction proof principle. For a given congruence relation C, we can show that a language satisfies C and hence belongs to the corresponding preformation of languages, by constructing a suitable equational bisimulation.