# A final coalgebra for the k-regular and k-automatic sequences

Joost Winter, Helle Hvid Hansen, Clemens Kupke, Jan Rutten

Centrum Wiskunde & Informatica Radboud Universiteit Nijmegen University of Strathclyde

September 21, 2013

#### Introduction

- k-automatic and k-regular sequences: classes defined by Allouche/Shallit
- ▶ A sequence  $\sigma \in \mathbb{Z}^{\omega}$  is *k*-automatic if generated by a deterministic automaton with output in  $\{0, \ldots, k-1\}$ ...
- ... where  $\sigma(n)$  is output after reading n in base k.

#### Introduction

- ► *k*-automatic and *k*-regular sequences: classes defined by Allouche/Shallit
- A sequence  $\sigma \in \mathbb{Z}^{\omega}$  is k-automatic if generated by a deterministic automaton with output in  $\{0,\ldots,k-1\}$ ...
- ▶ ... where  $\sigma(n)$  is output after reading n in base k.
- ▶ *k*-regular sequences generalize this:

$$\frac{k\text{-regular}}{k\text{-automatic}} = \frac{\text{weighted automata}}{\text{deterministic automata}}$$

► This talk: connecting *k*-regular sequences to (abstract) coalgebra and (concrete) behavioural differential equations.

# k-regular sequences: a definition (for k = 2)

We call a sequence (or stream)  $\sigma$  2-regular when there is a finite family of sequences

$$\Sigma = (\sigma_i) \quad i \leq n \in \mathbb{N}$$

with  $\sigma_0 = \sigma$ , s.t. for all  $i \leq n$  the sequences **even** $(\sigma_i)$  and **odd** $(\sigma_i)$  are linear combinations of sequences from  $\Sigma$ .

Here even and odd are defined by

$$even(\tau)(n) = \tau(2n)$$

and

$$\operatorname{odd}(\tau)(n) = \tau(2n+1)$$

## Derivative and **zip**

We will reason with the stream derivative from the coinductive stream calculus. Definition:

$$\sigma'(n) = \sigma(n+1)$$

We can define streams and operators coinductively by giving the first element and the derivative, e.g.

$$zip(\sigma, \tau)(0) = \sigma(0)$$
  
 $zip(\sigma, \tau)' = zip(\tau, \sigma')$ 

gives

$$zip(\sigma, \tau)(2k) = \sigma(k)$$
  
 $zip(\sigma, \tau)(2k+1) = \tau(k)$ 

and thus

$$zip(even(\sigma), odd(\sigma)) = \sigma$$

# Systems of **zip**-equations

*k*-regular sequences can be seen as solutions to finite systems of equations.

$$\tau_1 = \mathbf{zip}(\tau_1^e, \tau_1^o)$$
 $\vdots$ 
 $\tau_n = \mathbf{zip}(\tau_n^e, \tau_n^o)$ 

**Example:** the sequence of numbers whose base 3 representation does not contain the digit '2'

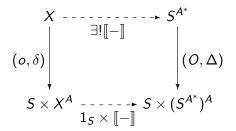
$$0, 1, 3, 4, 9, 10, 12, 13, 27, 28, 30, 31, \dots$$

is a solution to

$$\sigma = \mathbf{zip}(3\sigma, 3\sigma + \mathbf{ones})$$
 
$$\mathbf{ones} = \mathbf{zip}(\mathbf{ones}, \mathbf{ones})$$
 (with  $\mathbf{ones}(0) = 1$ ,  $\sigma(0) = 0$ )

## Automata as coalgebras

- ▶ Automaton (with output in S, input in A) is coalgebra for the functor  $S \times -^A$ .
- Semantics [-] given by unique morphism into final automaton:



Fact:  $[x](w) = o(x_w)$ 

## Streams are an instance of this

If |A|=1, note that  $S^{A^*}\cong S^{\mathbb{N}}$  and we get

$$(o, \delta) \downarrow \qquad \qquad \downarrow (O, \Delta)$$

$$S \times X \longrightarrow S \times (S^{\mathbb{N}})$$

$$O(\sigma) = \sigma(0)$$
$$\Delta(\sigma) = \sigma'$$

# Main result (for case k = 2)

#### **Theorem**

A sequence  $\sigma$  is 2-regular if and only if it is the unique solution to a system of stream differential equations

$$o(x) = k$$
  $x' = zip(x_e, x_o)$ 

for a finite set X, where  $k \in \mathbb{Z}$ , and for each  $x \in X$ ,  $x_e$  and  $x_o$  are given as a linear combination of elements from X.

(also found by Endrullis/Moss/Silva)

# Main result (for case k = 2)

#### **Theorem**

A sequence  $\sigma$  is 2-regular if and only if it is the unique solution to a system of stream differential equations

$$o(x) = k$$
  $x' = zip(x_e, x_o)$ 

for a finite set X, where  $k \in \mathbb{Z}$ , and for each  $x \in X$ ,  $x_e$  and  $x_o$  are given as a linear combination of elements from X.

(also found by Endrullis/Moss/Silva)

Idea: transform flat systems into guarded systems.

or: move from standard base k numeration to bijective base k numeration

Construct a system of stream differential equations from the earlier system:

$$\begin{array}{rcl} \sigma' & = & \text{zip}(3\sigma + \text{ones}, 3\sigma') \\ \sigma'' & = & \text{zip}(3\sigma', 3\sigma' + \text{ones}') \\ \text{ones}' & = & \text{zip}(\text{ones}', \text{ones}) \\ \text{ones}'' & = & \text{zip}(\text{ones}', \text{ones}') \end{array}$$

or

$$w' = zip(3w + y, 3x)$$

$$x' = zip(3x, 3x + z)$$

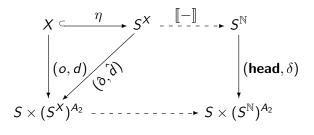
$$y' = zip(y, z)$$

$$z' = zip(z, z)$$

Add output values to specification and you're done!

# A final coalgebra diagram

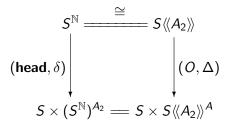
Semantics can be given by the following diagram (initiality + finality):



with

$$\delta(\sigma)(1) = \operatorname{even}(\sigma')$$
  
 $\delta(\sigma)(2) = \operatorname{odd}(\sigma')$ 

# An isomorphism of final coalgebras



Can be proven using the bijective base k numeration between  $\mathbb{N}$  and  $(A_k)^*$ .

Gives correspondence with weighted automata (over any semiring S).

## Application: divide and conquer recurrences

On the Online Encyclopedia of Integer Sequences, some formats for divide and conquer recurrences are given. E.g.

$$a(2n) = Ca(n) + Ca(n-1) + P(n)$$
  
 $a(2n+1) = 2Ca(n) + Q(n)$ 

where P and Q are expressible by a rational g.f.

## Application: divide and conquer recurrences

On the Online Encyclopedia of Integer Sequences, some formats for divide and conquer recurrences are given. E.g.

$$a(2n) = Ca(n) + Ca(n-1) + P(n)$$
  
 $a(2n+1) = 2Ca(n) + Q(n)$ 

where P and Q are expressible by a rational g.f.

Q (asked on oeis.org/somedcgf.html): 'An open question would be whether all sequences here discussed are 2-regular.'

A: if you replace the condition 'expressible by a rational g.f.' by '2-regular' yes (includes all their examples), otherwise no.

## Generalizations, conclusions and future work

- ▶ Everything told here about 2 works for any  $k \ge 2$ .
- ▶ We established a correspondence between rational power series in k (noncomm.) variables and k-regular sequences over arbitrary semirings.
- ...allowing us to translate back and forth between recurrences and systems of stream differential equations.

## Generalizations, conclusions and future work

- ▶ Everything told here about 2 works for any  $k \ge 2$ .
- ▶ We established a correspondence between rational power series in k (noncomm.) variables and k-regular sequences over arbitrary semirings.
- ...allowing us to translate back and forth between recurrences and systems of stream differential equations.
- ▶ Future work: how about *k*-algebraic sequences?
- . . . further investigate the connections with recurrences.