Applications of Algebra and Coalgebra in Scientific Modelling

Illustrated with the Logistic Map

M. Hauhs   B. Trancón y Widemann

Ecological Modelling
University of Bayreuth, D

Coalgebraic Methods in Computer Science
2010–03–26/28
Paphos, CY

Context

This Talk

- systems and models of **ecology**
- exemplified by the **logistic map**

Why at CMCS?

$$\begin{array}{ccc} \text{ecology} & \overset{?}{\longleftrightarrow} & \text{computer science} \\ \text{scientific modelling} & \overset{?}{\longleftrightarrow} & \text{coalgebra} \end{array}$$

Context

Relation of Ecology and Computer Science

Wikipedia

- Nothing too obvious, long term work...

Scientific Modelling

What is Scientific Modelling?

- Real systems are studied using **models**.
- Models are stylized objects that resemble the system in some properties considered **essential**.
- Which properties are essential is determined by the context, in particular by the scientific **discipline**.
- Most model types employed in empirical disciplines belong to either of two (dual?) **paradigms**.

Model Types: State vs. Behavior

Ontological Dispute

- **State**-based or **behavior**-based modelling?
- Scientific disciplines are biased:

Model Types: State vs. Behavior

Ontological Dispute

- **State**-based or **behavior**-based modelling?
- Scientific disciplines are biased:

Physics

State	is fundamental (snapshots of reality)
Behavior	is the effect of natural laws
Functional	models define **variables** and their **dynamics**
Problems	prediction of state and inference of parameters

Model Types: State vs. Behavior

Ontological Dispute

- **State**-based or **behavior**-based modelling?
- Scientific disciplines are biased:

Computer Science

Behavior	is fundamental (specifications of technology)
State	is the artifact of implementation
Interactive	models define **interfaces** and their **protocols**
Problems	assessment of situations and planning of strategies

Model Types: State vs. Behavior

Ecology

- has no theoretical framework of its own,
- is characterized by the interaction of living systems with their environment,
  hence
- calls for both paradigms,
  but
- is traditionally dominated by physicalism (PDEs[a] etc.);
  to the effect that
- **relevance** and **rigor** often appear mutually exclusive.

[a] Partial Differential Equations

Model Types: State vs. Behavior

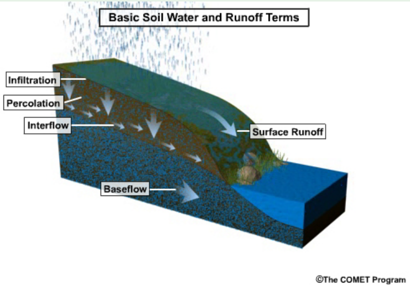
Examples

Environment-Dominated

State soil / groundwater

↓

Behavior runoff from watershed



Life-Dominated

Behavior annual tree growth

↓

State carbon pool in forest

Model Types: State vs. Behavior

Proposal: Formalize the Two Paradigms

$$\begin{array}{ccc} \text{state} & \leftrightarrow & \text{algebra} \\ \text{behavior} & \leftrightarrow & \text{coalgebra} \end{array}$$

- exploit duality for unifying framework

Logistic Map

Definition [R. May, 1976, Nature]

- discrete-time demographical model
- simple real quadratic map

$$f_r(x) = r \cdot x \cdot (1 - x)$$

- with real **growth** parameter $r > 0$
- restricted to the unit interval $\mathbb{I} = [0, 1]$
- iteration produces complex behavior

Logistic Map

Forward Dynamics

- **Trajectories** for given initial value x_0 and parameter r

$$x_0, f_r(x_0), f_r^2(x_0), \ldots \in \mathbb{I}$$

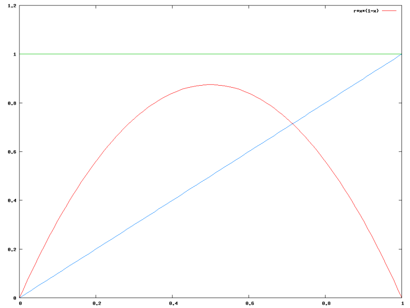
- infinite for $r \leq 4$
- eventually divergent for $r > 4$
- all sorts of long-term dynamics:

  stable/unstable fixpoints, stationary cycles,
  deterministic chaos, strange attractors

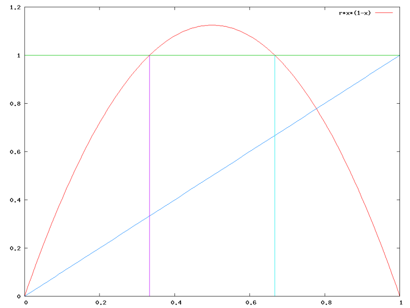
- test case for information/complexity measures [F. Wolf, 1999, Diss.]

Logistic Map

Graph



$r < 4$



$r > 4$

Logistic Map

Backward Dynamics

- Each x has (at most) two preimages under f_r
- infinite binary decision tree **or**
- infinite automaton

  binary input, state space $\mathbb{I}$
- total for $r \geq 4$
- partial for $r < 4$:

  states $x > r/4$ (global max) are error states

Logistic Map

Finite Partitioning (Coloring)

$$c(x) = \begin{cases} 0 & x \in [0, \frac{1}{2}) \\ 1 & x \in [\frac{1}{2}, 1] \end{cases}$$

- simple model of imperfect observation
- generating partition of symbolic dynamics
- complementary to f_r:

  neither c nor f_r is invertible,
  but $\langle c, f_r \rangle$ is

Models as Homomorphisms

Modelling [R. Rosen, 1991, "Life Itself"]

$$
\begin{tikzcd}
\mathrm{Real}_2 \arrow[r, "\textit{abstraction}"] & \mathrm{Abstract}_2 & {} \\
\mathrm{Real}_1 \arrow[u, "\textit{causality}"] \arrow[r, "\textit{abstraction}"'] & \mathrm{Abstract}_1 \arrow[u, "\textit{computation}"'] & {} \arrow[u, dotted, "\text{time}"']
\end{tikzcd}
$$

Wanted abstraction that commutes with dynamics

Test round-trip $\Rightarrow$ prediction

Models as Homomorphisms

Modelling [R. Rosen, 1991, "Life Itself"]

$$\begin{tikzcd}[column sep=huge, row sep=huge]
\mathrm{Real}_2 & \mathrm{Abstract}_2 \arrow[l, red, "\textit{interpretation}"'] \\
\mathrm{Real}_1 \arrow[u, "\textit{causality}"] \arrow[r, "\textit{abstraction}"'] & \mathrm{Abstract}_1 \arrow[u, "\textit{computation}"']
\end{tikzcd}$$

time

Wanted abstraction that commutes with dynamics

Test round-trip $\Rightarrow$ **prediction**

Models as Homomorphisms

Generalization: (Co)Algebras

$$\begin{tikzcd} A_2 \arrow[r, "h"] & B_2 \\ A_1 \arrow[u, "\alpha"] \arrow[r, "h"] & B_1 \arrow[u, "\beta"'] \end{tikzcd}$$

Base Case trivial signature functor (identity)

Algebra signature F encodes model **queries**

Coalgebra signature F encodes model **observations**

Models as Homomorphisms

Generalization: (Co)Algebras

$$\begin{tikzcd} A_2 \arrow[r, "h"] & B_2 \\ F(A_2) \arrow[u, "\alpha"] \arrow[r, "F(h)"] & F(B_2) \arrow[u, "\beta"'] \end{tikzcd}$$

Base Case trivial signature functor (identity)

Algebra signature F encodes model **queries**

Coalgebra signature F encodes model **observations**

Models as Homomorphisms

Generalization: (Co)Algebras

$$\begin{tikzcd} F(A_1) \arrow[r, "F(h)"] & F(B_1) \\ A_1 \arrow[u, "\alpha"] \arrow[r, "h"] & B_1 \arrow[u, "\beta"'] \end{tikzcd}$$

Base Case trivial signature functor (identity)

Algebra signature F encodes model **queries**

Coalgebra signature F encodes model **observations**

Definition: Affine Functor $\mathcal{A}_B^A$

Recursive structure: A-List with B-labelled end

$$\mathcal{A}_B^A = A \times (-) + B$$

$$\texttt{go} : A \times X \to \mathcal{A}_B^A(X) \qquad \texttt{stop} : B \to \mathcal{A}_B^A(X)$$

Agenda

- metaphors $\Longrightarrow$ formal calculus
- all affine functors have initial algebras / final coalgebras
- generate model types by instantiating A/B with simple sets
- take cata-/anamorphisms as modelling maps ((co)induction)

Initial Algebra

$$\left(A^* \times B, [\alpha_1, \alpha_2]\right)$$

$$\alpha_1\big(a, (w, b)\big) = \big(\mathtt{cons}(a, w), b\big) \qquad \alpha_2(b) = (\mathtt{nil}, b)$$

Catamorphism

$$h : \left(A^* \times B, [\alpha_1, \alpha_2]\right) \to \left(C, [\gamma_1, \gamma_2]\right)$$

$$h\big(\mathtt{cons}(a, w), b\big) = \gamma_1\big(a, h(w, b)\big) \qquad h(\mathtt{nil}, b) = \gamma_2(b)$$

Final Coalgebra

$$(A^* \times B \cup A^\omega, \phi)$$

$$\phi(\mathtt{cons}(a, w), b) = \mathtt{go}(a, (w, b)) \qquad \phi(\mathtt{nil}, b) = \mathtt{stop}(b)$$

Anamorphism

$$h : (C, \gamma) \to (A^* \times B \cup A^\omega, \phi)$$

$$\phi(h(c)) = \begin{cases} \mathtt{go}(a, h(c')) & \text{if } \gamma(c) = \mathtt{go}(a, c') \\ \mathtt{stop}(b) & \text{if } \gamma(c) = \mathtt{stop}(b) \end{cases}$$

Instances

$$\mathcal{A}_B^1 \qquad \mathcal{A}_1^A \qquad \mathcal{A}_\varnothing^A \qquad \mathcal{A}_B^\varnothing$$

Instances

$\mathcal{A}_B^1$ | $\mathcal{A}_1^A$ | $\mathcal{A}_\varnothing^A$ | $\mathcal{A}_B^\varnothing$

Algebras: Iteration

General

$$(C, \gamma : 1 \times C + B \to C)$$

$$\gamma \sim (f, g) \qquad f : C \to C \qquad g : B \to C$$

Initial

$$\left(\mathbb{N} \times B, (f_0, g_0)\right) \qquad 1^* \sim \mathbb{N}$$

$$f_0(n, b) = (n + 1, b) \qquad g_0(b) = (0, b)$$

Catamorphism $(\!|\gamma|\!)$

$$i(n, b) = f^n(g(b))$$

Instances

$\mathcal{A}_B^1$ $\mathcal{A}_1^A$ $\mathcal{A}_\varnothing^A$ $\mathcal{A}_B^\varnothing$

Algebras: List Folding

General

$$(C, \gamma : A \times C + 1 \to C)$$

$$\gamma \sim (f, e) \qquad f : A \times C \to C \qquad e \in C$$

Initial

$$(A^*, (\texttt{cons}, \texttt{nil}))$$

Catamorphism $(\!|\gamma|\!)$

$$j(\texttt{cons}(a, w), b) = f(a, j(w)) \qquad j(\texttt{nil}) = e$$

Instances

$\mathcal{A}_B^1$ $\mathcal{A}_1^A$ $\mathcal{A}_\varnothing^A$ $\mathcal{A}_B^\varnothing$

Coalgebras: Stream Unfolding

General $(C, \gamma : C \to A \times C + \varnothing)$

$\gamma \sim (h, t)$ $\quad h : C \to A \quad t : C \to C$

Final $(A^\omega, \mathtt{cons}^{-1})$

Anamorphism $[\![(\gamma)]\!]$ $\quad k(c) = \mathtt{cons}\big(h(c), k(t(c))\big)$

Instances

$\mathcal{A}_B^1$ $\mathcal{A}_1^A$ $\mathcal{A}_\varnothing^A$ $\mathcal{A}_B^\varnothing$

Coalgebras: Degenerate

General

$$(C, \gamma : C \to \varnothing \times C + B)$$

$$\gamma \sim t \qquad t : C \to B$$

Final

$$(B, \texttt{stop})$$

Anamorphism $[\![(\gamma)]\!]$

$$m = t$$

Model Instances

Functional, Deterministic

$$F = \mathcal{A}_B^1 \qquad C = \mathbb{I}$$

$$B = \mathbb{I} \qquad f = f_r \qquad g = \mathtt{id}_{\mathbb{I}}$$

$$\begin{tikzcd}
\mathbb{N} \times B \arrow[r, "i"] & C \\
F(\mathbb{N} \times B) \arrow[u] \arrow[r, "F(i)"'] & F(C) \arrow[u, "(f, g)"']
\end{tikzcd}$$

Given boundary (r) and exact initial (x_0) conditions

Predict state after n steps

$$i(n, x_0) = {f_r}^n(x_0)$$

Model Instances

Functional, Nondeterministic

$$F = \mathcal{A}^1_B \qquad C = \mathcal{P}\mathbb{I}$$

$$B = \mathcal{P}\mathbb{I} \qquad f = \mathcal{P}f_r \qquad g = \mathtt{id}_{\mathcal{P}\mathbb{I}}$$

$$\begin{tikzcd}
\mathbb{N} \times B \arrow[r, "i"] & C \\
F(\mathbb{N} \times B) \arrow[u] \arrow[r, "F(i)"'] & F(C) \arrow[u, "(f,g)"']
\end{tikzcd}$$

Given boundary (r) and potential initial (x_0) conditions

Predict state after n steps

$$i(n, x_0) = \mathcal{P}f_r{}^n(x_0)$$

Model Instances

Functional, Probabilistic

$$F = \mathcal{A}^1_B \qquad C = \mathbb{I}^\sim$$

$$B = \mathbb{I}^\sim \qquad f = f_r^{\sim} \qquad g = \mathtt{id}_{\mathbb{I}^\sim}$$

$$\begin{array}{ccc} \mathbb{N} \times B & \xrightarrow{i} & C \\ \uparrow & & \uparrow{\scriptstyle (f,g)} \\ F(\mathbb{N} \times B) & \xrightarrow[F(i)]{} & F(C) \end{array}$$

Given boundary (r) and distribution (CCDF[a]) of initial (x_0) conditions

Predict state after n steps

$$i(n, x_0) = {f_r^{\sim}}^{n}(x_0)$$

[a] Cumulative Continuous Distribution Function

Model Instances

Inverse Functional

$$F = \mathcal{A}_1^A \qquad C = \mathbb{I} \nrightarrow \mathbb{I}$$

$$A = 2 \qquad f(a,h) = f_r|_a \circ h \qquad e = \mathtt{id}_{\mathbb{I}}$$

$$\begin{tikzcd}
\mathbb{N} \times B \arrow[r, "j"] & C \\
F(\mathbb{N} \times B) \arrow[u] \arrow[r, "F(j)"'] & F(C) \arrow[u, "(f,e)"']
\end{tikzcd}$$

$$f_r|_a(x) = \begin{cases} f_r(x) & \text{if } c(x) = a \\ \text{undefined} & \text{if } c(x) \neq a \end{cases}$$

Given a finite trace of colors

Solve for initial & final states

$$j(a_1 \ldots a_n) = f_r|_{a_1} \circ \cdots \circ f_r|_{a_n}$$

- limit $n \to \infty$?

Model Instances

Interactive [Rutten, 2000]

$$F = \mathcal{A}_{\varnothing}^{A} \qquad C = J_r$$

$$A = 2 \qquad h = c \qquad t = f_r$$

$$\begin{tikzcd} F(J_r) \arrow[r, "F(k)"] & F(2^\omega) \\ J_r \arrow[u, "(h{,}t)"] \arrow[r, "k"'] & 2^\omega \arrow[u] \end{tikzcd}$$

$$J_r = \bigcap_{n=0}^{\infty} \overline{\mathcal{P}} f_r^n(\mathbb{I}) \qquad (r > 4 \Rightarrow \text{Cantor dust})$$

Given the set of states not leading to termination

Represent their fully abstract behavior at interface c

$$k(x)(n) = c\big(f_r^n(x)\big) \qquad (k \text{ iso})$$

Inverse Interactive (Work in Progress)

$$k \text{ iso} \implies (J_r, (c, f_r)) \text{ final} \implies (c, f_r) \text{ iso}$$
$$(c, f_r)^{-1} : 2 \times J_r \to J_r \quad \simeq J_r \to J_r^2$$

$$\begin{tikzcd} F(J_r) \arrow[r, "F(k)"] & F(2^\omega) \\ J_r \arrow[u, leftarrow, "(c,f_r)^{-1}"'] \arrow[r, "k"'] & 2^\omega \arrow[u] \end{tikzcd}$$

Game solitaire with binary moves

Record a particular player's actual moves

Cover reached states by some subsystem (-coalgebra)

Describe subsystem axiomatically by **modal logic**

Attribute axioms as **strategy** to player

simple backward strategies have arbitrarily complicated forward reconstructions

Summary

Conclusion

- Generate family of model types
  (co)inductively from family of functors
- functional–interactive mapped to algebra–coalgebra
- functional & interactive models on par
- counterexamples to arguments from complexity
- toy examples work out fine, so now for...

Outlook

Current & Future Challenges

- modal logic & ecosystem management
- non-well-founded structures & open-ended evolution
- information & complexity of behavior traces
- definition of life
- causality in biology

References

R.M. May (1976).
Simple mathematical models with very complicated dynamics.
Nature 261(5560), pp. 459–467

R. Rosen (1991).
Life Itself: A Comprehensive Inquiry into the Nature, Origin, and Fabrication of Life.
Columbia Press, New York.

F. Wolf (1999).
Berechnung von Information und Komplexität in Zeitreihen – Analyse des Wasserhaushaltes von bewaldeten Einzugsgebieten.
Dissertation, Universität Bayreuth.

J.J.M.M. Rutten (2000).
Universal coalgebra: a theory of systems.
Theor. Comp. Sci. 249(1), pp. 3–80.

B. Trancón y Widemann, M. Hauhs (2009).
Programming as a Model for the Theory of Ecosystems.
In J. Knoop, A. Prantl (eds.). *Programmiersprachen und Grundlagen der Programmierung*; Post-Proceedings KPS 2009. Technical Report 2009-XII-1, pp. 186–201. Institut für Computersprachen, Technische Universität Wien.