

**Jacobs, Bart; Rutten, Jan**

**A tutorial on (co)algebras and (co)induction.** (English) Zbl 0880.68070  
Bull. EATCS 62, 222-259 (1997).

Summary: Algebraic structures which are generated by a collection of constructors – like natural numbers (generated by a zero and a successor) or finite lists and trees – are of well-established importance in computer science. Formally, they are initial algebras. Induction is used both as a definition principle, and as a proof principle for such structures. But there are also important dual “coalgebraic” structures, which do not come equipped with constructor operations but with what are sometimes called “destructor” operations (also called observers, accessors, transition maps, or mutators). Spaces of infinite data (including, for example, infinite lists, and non-well-founded sets) are generally of this kind. In general, dynamical systems with a hidden, black-box state space, to which a user only has limited access via specified (observer or mutator) operations, are coalgebras of various kinds. Such coalgebraic systems are common in computer science. And “coinduction” is the appropriate technique in this coalgebraic context, again both as a definition principle and as a proof principle. The latter involves bisimulations. It is the aim of this tutorial to provide a brief introduction to this relatively new field of coalgebra.

**MSC:**

[68W30](#) Symbolic computation and algebraic computation

Cited in **1** Review  
Cited in **104** Documents

**Keywords:**

[coalgebra](#)

**Software:**

[HOL](#)