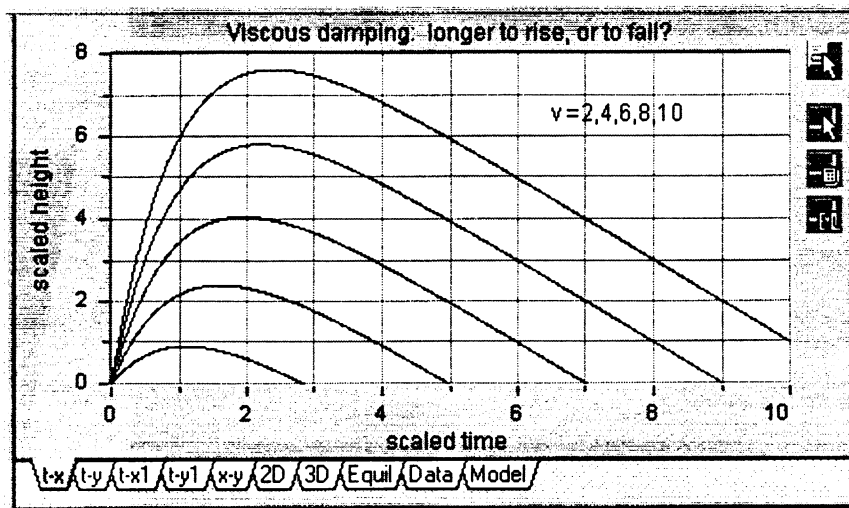


Modeling and Visualization

with

ODE Architect

An Example Set



Robert L. Borrelli
Courtney S. Coleman
Harvey Mudd College

ODE Architect was designed and implemented by CODEE (a consortium of 15 mathematics faculty members at eight colleges and universities) and Intellipro Inc. (an educational software company) with support from the publisher John Wiley and Sons, Inc. and the National Science Foundation. The material in these notes is solely the responsibility of the authors, however, and not the responsibility of any of the four organizations mentioned above.

Send comments and suggestions to

borrelli@hmc.edu
coleman@hmc.edu

For more information about ODE Architect visit the web site

<http://www.intellipro-inc.com>

Preface

The examples in this collection illustrate some of the features of ODE Architect. These units are independent of one another and for convenience are divided into three groups: elementary, intermediate, and advanced. Each unit includes several screen shots from ODE Architect. These shots illustrate how to supply data to the Architect solver tool which result in the pictured graphs.

The units can be used in several ways:

- As a learning tool for exploring the features of ODE Architect
- To construct a model appropriate for computer simulation by ODE Architect
- For student projects in a variety of courses
- For lectures or talks.

There are many other accessible examples and models within the ODE Architect package itself. The ODE Library in the tool contains over one hundred briefly described, illustrated, and editable examples appropriate for differential equations courses. See the Appendix in the User's Guide for a list of the library entries. The User's Guide gives detailed instructions on how to use the various features of the solver tool.

We want to especially thank Professor Michael Moody and our student Nathan Jakubiak for their help in preparing some material for these notes.

Robert L. Borrelli
Courtney S. Coleman
1/99

Table of Contents

Elementary

Lascaux Cave paintings

Radioactive decay. Using computer model to approximate age. Backward IVP. Scaling.

Vertical Motion

Uses Newton's Second Law to model vertical motion. Scaling. ODE Architect is used to answer the question "Does a body take longer to rise or to fall?". Solution formula is not helpful in answering this question if air resistance is taken into account.

Sprint

Models strategies for running 100 meters. Compares results with real data.

Samurai

Uses a computer model to design an efficient cutting instrument. Also derives a solution formula in polar coordinates which describes the sword's edge mathematically.

Intermediate

Lead in the Body

Linear system modeled with the Balance Law. Examines sensitivity of solutions to the environmental parameters. ODE Architect will find eigenvalues and steady-states.

Coaxial Cable

Uses on-off functions as inputs to a simple RC-circuit for sending a coded message. Examines sensitivity of output to the frequency of the input.

Air-Conditioning

Uses Newton's Law of Cooling to describe how an air-conditioner keeps a room's temperature within a prescribed range.

Sky Diver

Determines when a sky diver should pull the rip cord to minimize descent time (and survive). ODE Architect solves the problem by overlapping graphs of a forward IVP and a backward IVP.

Good Solver, Bad Behavior

Things may go wrong even when using a good numerical ODE solver: aliasing, extension, choice of step-size, etc.

Advanced

Van der Pol

A nonlinear model with a limit cycle. Effect of parameter on shape and period of the limit cycle, and animation of Hopf bifurcation. ODE Architect automatically calculates the eigenvalues of the linearized system.

Lorenz

ODE Architect finds equilibrium points and eigenvalues of the linearized Lorenz system and graphically illustrates how parameter changes affect stability. 3-D graphs of chaotic wandering, period doubling sequences.

Battle of the Bulge

Actual combat data from the battle is used to determine the coefficients for a combat model. The modeling linear system is solved by ODE Architect and the results are compared with the actual data.

Autocatalator

Interacting chemical species in an autocatalytic reaction lead to a nonlinear system which exhibits peculiar oscillatory behavior. ODE Architect shows how changing the rate constants can be used to turn the oscillations on and off.

Satiabile Predation

Predator-prey model where the predator's appetite satiates. A sensitivity study shows a Hopf bifurcation followed by a reverse Hopf bifurcation. Illustrated via animation.

Rotational Stability

Examines the stability of steady rotations of a rigid body (like a book or tennis racket) when tossed in the air. 3-D graphics clearly shows which steady rotations are stable and which are unstable.

Fitzhugh-Nagumo (written by Michael E. Moody)

Uses ODE Architect to show how a planar system of ODEs representing a neuron responds to a stimulus.