THE ANNALS OF "DUNAREA DE JOS" UNIVERSITY OF GALATI
FASCICLE XIV MECHANICAL ENGENEERING, ISSN 1224 – 5615
2005

SIMULATION OF DYNAMICAL SYSTEMS WITH LINEAR AND NON-LINEAR BEHAVIOR IN SCICOS ENVIRONMENT

conf.univ.dr.ing. Gilbert-Rainer GILLICH
assist.univ.drd.ing. Cristian Paul CHIONCEL
Universitatea "Eftimie Murgu" Reşiţa

ABSTRACT

The paper presents the SCICOS program and the way how to formalise dynamic system with known physical and mathematical models in this environment. Using simple examples from the field of vibrations, it is shown how "continuous space/discrete time" models can be constructed in this formalism. For instance, it is presented an example of a mass-spring system, working dumped in different ways or excited. In the end of the paper conclusions regarding the SCICOS program are formulated.

1. Introduction

SciLab is a scientific numerical, programming and graphics environment software package for numerical computations, user-friendly, providing a powerful open computing environment for engineering and scientific applications, based on vector/matrix manipulations. It resembles MatLab/Simulink and Matrix/SystemBuild family of products. It is similar in operation to MatLab, being its best clone, and other existing numerical/graphic environments, that can be run on a variety of operating systems including UNIX, Windows, Linux, etc.

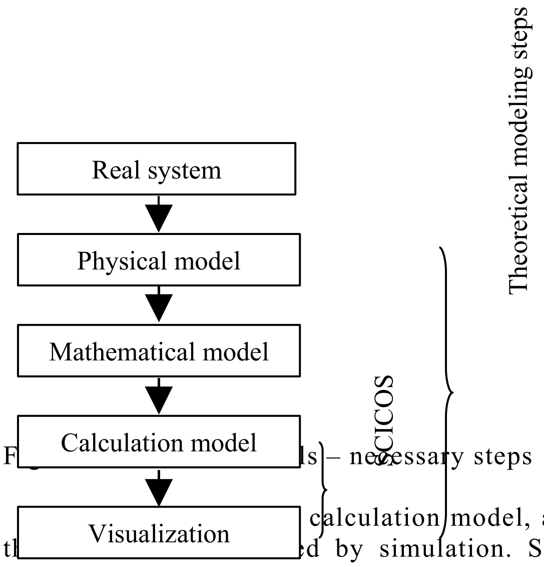
Developed since 1990 by researchers from the French Government's "Institut Nationale de Recherche en Informatique et en Automatique" - INRIA (National Institute for Informatics and Automation Research) SciLab, is now maintained and developed by Scilab Consortium since its creation in May 2003. Distributed freely and open source via the Internet since 1994, SciLab is currently being used in educational and industrial environments around the world. Scilab is made of three distinct parts: an interpreter, libraries of functions (SciLab procedures) and libraries of Fortran and C routines. It includes hundreds of mathematical functions with the possibility to add interactively programs from various languages (C, C++ ...) with sophisticated data structures (including lists, polynomials, rational functions, linear systems...), an interpreter and a high level programming language.

A key feature of the SciLab syntax is its ability to handle matrices: basic matrix manipulations such as concatenation, extraction or transpose are immediately performed as well as basic operations such as addition or multiplication. SciLab also aims at handling more complex objects than numerical matrices. For instance, control people may want to manipulate rational or polynomial transfer matrices. This is done in SciLab by manipulating lists and typed lists which allow a natural symbolic representation of complicated mathematical objects such as transfer functions, linear systems or graphs.

SCICOS is a graphically-based system modelling software for dynamical system, an environment for designing reactive systems modeller and simulator toolbox included in the SciLab engineering and scientific computation software. Based on an open formalism, SCICOS can create block diagrams and be used to model and simulate the dynamics of hybrid dynamical systems and compile models into executable code. SCICOS is used for signal processing, systems control, to study physical and biological systems.

New extensions allow hard real-time control executables, as well as component based modelling of electrical and hydraulic circuits.

Real systems can be transposed in theoretical models following four steps, like it is presented in figure 1. The system is simplified, it results the physical (discrete, continuous or pseudo-continuous) model. For it mathematical equation(s) can be written, which gives the mathematical model.



F... ls – necessary steps

... calculation model, and th... d by simulation. SCICOS is used for the both last steps of theoretical modelling.

2. Physical and mathematical model of the dynamic considered system

Let’s take for example a real technical system which has the physical model consisting in a block with mass *m* moving upon a surface bond on a fix point trough a spring with the spring constant *k*. We take in consideration two situations: the mass moves on a rough surface (figure 2) where the friction coefficient is μ and a viscous surface (figure 3) with the damp coefficient *c*. Both non-linear and linear situations are represented as well.

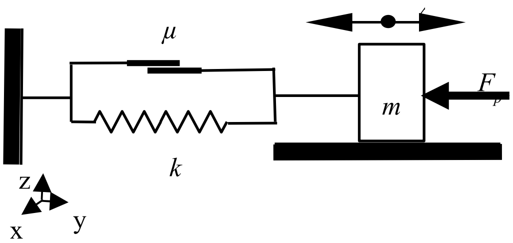


Fig. 2. Physical model of a dumped mass-spring system, with Coulomb friction

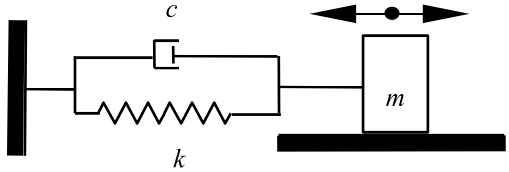


Fig. 3. Physical model of a viscous dumped mass-spring system

If we move the mass *m* on the surface whit a distance *A* from the equilibrium point in direction y and let it free, the mass will begin to oscillate around this point. For the friction respecting Coulomb’s law, the resistant force is:

$$F_r = \mu \cdot m \cdot g(sign\ y') \quad (1)$$

where *g* is gravitation and y' is the velocity.

If the system is dumped in viscous way, the resistant force is:

$$F_r = c \cdot y' \quad (2)$$

The spring acts with a force depending on the distance to the equilibrium point, as follows:

$$F_e = k \cdot y \quad (3)$$

The mass in motion will give following force:

$$F_m = m \cdot y'' \quad (4)$$

where y'' is the acceleration of the mass *m*.

Writing the equilibrium equation for the forces acting on mass *m*, we find out:

$$F_m = -F_e - F_r + F_p \quad (5)$$

where F_p is an eventual external additional force.

The particular equations are in this case:

$$m \cdot y'' = -k \cdot y - \mu \cdot m \cdot g(sign\ y') + F_p \quad (6)$$

for the non-linear system, with Coulomb friction, and

$$m \cdot y'' = -k \cdot y - c \cdot y' + F_p \quad (7)$$

for the linear, viscous dumped system.

In the paper we present the behaviour of this system in three different conditions:

-Coulomb friction in free movement;
-Coulomb friction with external additional force;
-Viscous friction in free movement.

For the numerical example following values are set out:

mass *m* = 1 kg
spring constant *k* = 2 N/m
friction coefficient μ = 0,0254

dump coefficient $c = 0,25$ kg/s
gravity $g = 9.82$ m/s^2
amplitude $A = 1$ m

The system can be driven by an additional force $F_p = F_0 \cdot sin(2\pi ft)$ which act in y direction and has $F_0 = 1.5$ N and frequency $f = 3$ Hz

The values have no physical signification, they are chosen for simplicity.

3. SCICOS formalism

SCICOS construct simply models of dynamical systems, using a graphical editor in which blocks (representing predefined basic functions or user defined functions) are interconnected, but an underlying language exists providing a well defined formalism. This formalism is very simple because it deals exclusively with the reactive part of the design; it does not provide a complete programming language. The blocks are considered as atoms in SCICOS. In SCICOS formalism, the execution of simulation functions are considered instantaneous so SCICOS can be considered a Synchronous language or more specifically an extension of it to handle continuous time systems.

SCICOS provides many elementary blocks organized in different palettes that can be accessed using the operation **Palettes** in the **Edit menu**. This operation opens up a dialog box that includes the list of available palettes. By selecting a palette in the list, a new SCICOS window opens up displaying the blocks available in this palette. Blocks from palettes can be copied into the main SCICOS window by clicking first on the desired block and then at the location where the block is to be copied in the SCICOS window. Block parameters are ready defined, but for most blocks they can be modified by opening the block dialogs by clicking the right mouse button

A SCICOS block can have two types of inputs and outputs:
-regular inputs - usually placed on the sides;
-regular outputs - also on the sides;
-activation inputs - usually on top;
-activation outputs - usually at the bottom.

Regular inputs and outputs are used to communicate data from block to block through regular links. Activation inputs and outputs are connected by activation links which transmit control information (activation). It is also possible to make a link originate from another link (to split a link). The signals have associated a set of time indices, called activation times, on which the signal can evolve. Outside their activation times, SCICOS signals remain constant.

The activation time can be continuous (time intervals), discrete (isolated points called events) or a union of time intervals and isolated points.

Continuous time operations and discrete-time event dependent operations can interact in different ways. Continuous and discrete time signals can be inputs to the same block. In fact fundamentally, there is no difference between a discrete time signal and a continuous time signal. A SCICOS signal can have discrete property over a period of time and later continuous property. This means that in SCICOS we can perform operations (such as addition of) continuous and discrete time signals. Continuous time signals can generate events through zero-crossing blocks.

4. Example of solving the mathematical model for the Coulomb friction case

If a force elongates the spring with one distance unit and than it stops its action, the system will start oscillating. The initial conditions will be: elongation $y = 1$ and velocity $y' = 0$.

Equation (6) will be disposed in an accessible form, about isolating the derivation highest degree:

$$y'' = -\frac{k}{m} \cdot y - \mu \cdot g(sign\ y') \qquad (8)$$

To implement this equation we need to utilize from the SCICOS library (figure 4) two integrators, one to obtain the velocity, the other for the elongation. The first will have as initial condition the value zero (initial velocity); the second will have as initial condition the value 1 (initial elongation). We need also a sum block with two inverting inputs and two amplifiers with factors $\mu \cdot g$, respectively k/m. A Sign block is also used. The Sign block indicates the sign of the input:
-the output is 1 when the input is greater than zero;
-the output is 0 when the input is equal to zero;
-the output is -1 when the input is less than zero.

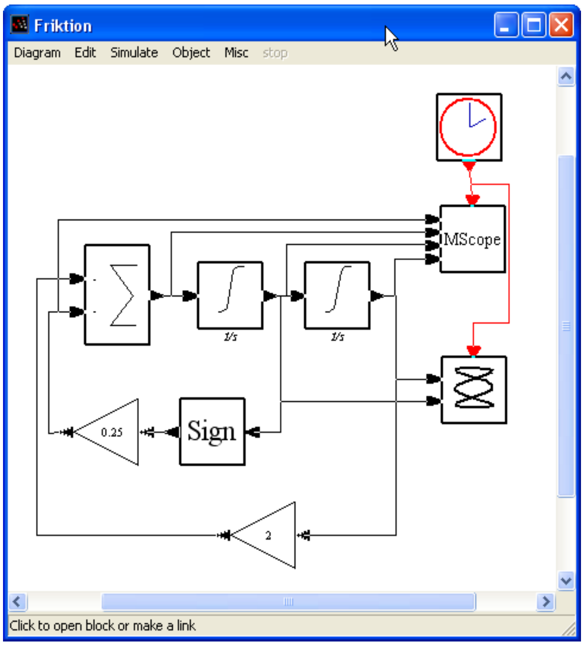


Fig. 4. Scheme block of a dumped (Coulomb friction) mass-spring system

The simulation results are presented on a Multi display scope (our MScpe displays acceleration, velocity and displacement diagrams separately or cumulated) and a XY scope (display phase plain).

The Activation clock block can be find in Pallets - Super Block and is constructed by feeding back the output of an event delay block into its input event port. The unique output of this block generates a regular train of events.

If we particularise for the chosen parameters we can analyse the case of free movement (figure 5 and figure 6). The first diagram in figure 5 presents the evolution of *sign y'* (sense of resistance), the second diagram the evolution of the acceleration *y''*, the third diagram the evolution of the velocity *y'* and the fourth diagram the evolution of displacement the *y*.

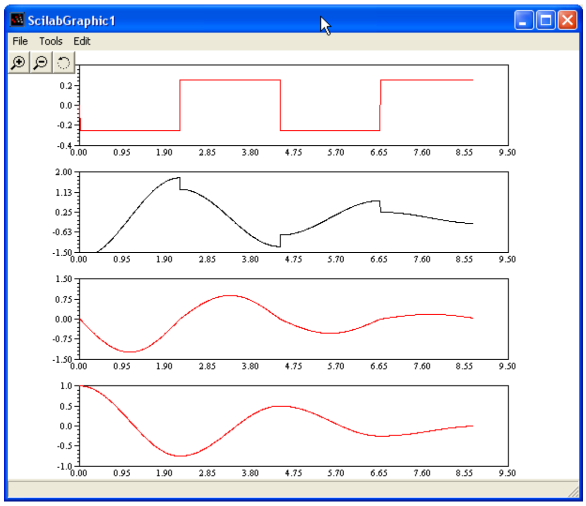


Fig. 5. Simulation of behaviour of a dumped (Coulomb friction) mass-spring system

Let us note $y_s = +\mu \cdot m \cdot g/k$ the distance between the equilibrium point where the spring is not tensioned and the static equilibrium point given by the friction. It can be recognized from the graphs above that the phase portrait for this system is a family of semicircles with centers at $+ y_s$ and $- y_s$. When $y' < 0$ the center is at $+ y_s$ else the center is at $- y_s$.

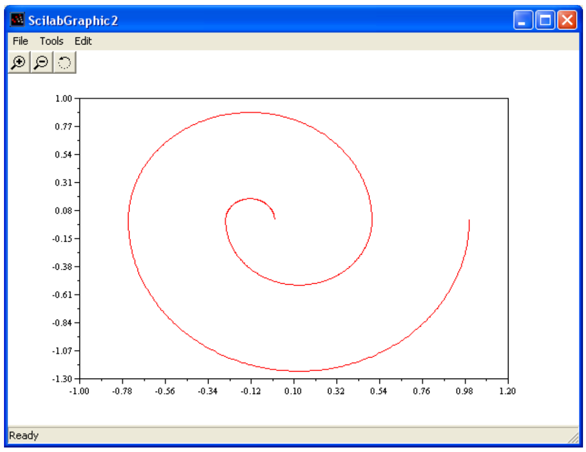


Fig. 6. Phase plain for the dumped (Coulomb friction) mass-spring system

The phase plane shows also the initial values, $y = 1$ and $y' = 0$, i.e. the block is moved to the right and released from rest. The velocity is zero but the acceleration is negative so the block starts moving to the left.

If the system is driven by a force described on the end of chapter 2, the mathematical model of the system will be:

$$y'' = -\frac{k}{m}\cdot y - \mu \cdot g(sign\ y') + \frac{F_0}{m}\cdot sin(2\pi ft) \quad (9)$$

In this case we have to add to the block scheme a block (Sinusoid generator in figure 7), available in the SCICOS library, which gives as output signal a sinusoid.

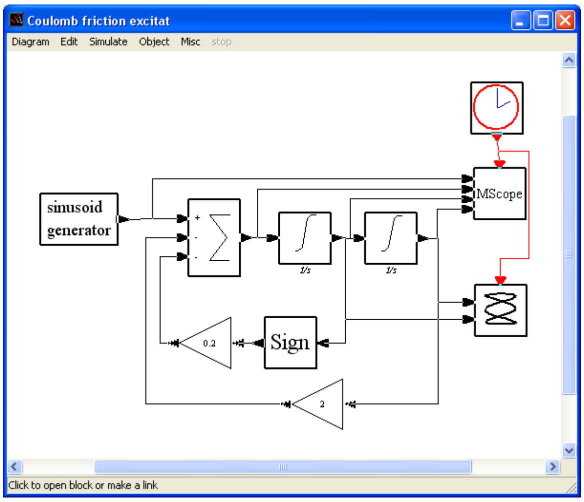


Fig. 7. Simulation of behaviour of a dumped (Coulomb friction) mass-spring system driven by a sinusoidal acting force

The Sinusoid generator can be connected to the clock or it can work by inheritance.

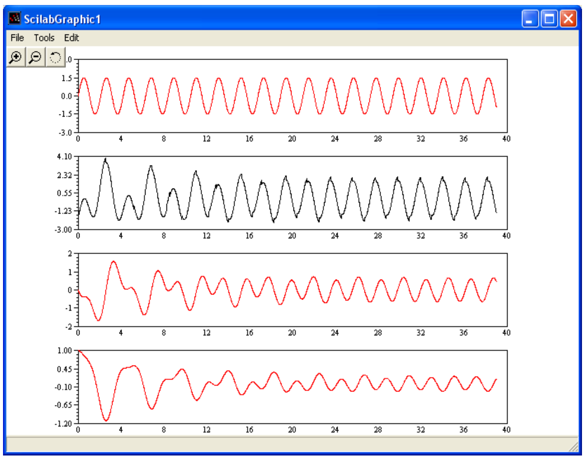


Fig. 8. Simulation of behaviour of a dumped (Coulomb friction) mass-spring system driven by a sinusoidal acting force

The parameters of the signal, first diagram in figure 8, can be defined from the Block menu of the Sinusoidal generator.

If we choose to use a constant force and not a time depending one, we can set the frequency zero in the Sinusoid generator block, or we can use another block which gives a constant signal.

Seeing the diagrams in figure 8 and comparing it with the diagrams in figure 5 we can conclude that, if the force started to act in the same direction like the elastic force, the amplitude of the displacement initially grows and it is necessary a longer time to stabilise the movement.

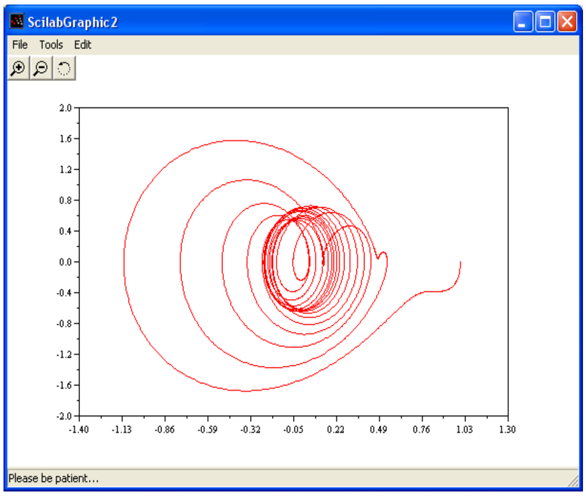


Fig. 9. Phase plain for the dumped (Coulomb friction) mass-spring system driven by a sinusoidal acting force

The phenomena can by amplified if the frequency of the force is equal or close to the proper frequency of the system.

From the phase plain in figure 9 we see that the equilibrium point moves with y_s like in the previous case but because the action o force F_p as well.

5. Example of solving the mathematical model for the viscous dump case

If a force elongates the spring, like in the case presented in chapter 4, with one distance unit and than it stops its action, the system will start oscillating. The initial conditions will be: elongation $y = 1$ and velocity $y' = 0$.

To solve the problem we have to dispose equation (7) in an accessible form, about isolating the derivation highest degree:

$$y'' = -\frac{k}{m}\cdot y - \frac{c}{m}y' \quad (10)$$

In this case, to implement the equation, we need from the SCICOS library two integrators to integrate the acceleration y'' (results velocity y') and the velocity y' (results displacement *y*). We need also a sum block with two inverting inputs and two amplifiers with factors *c/m*, respectively *k/m*.

The parameters set in the blocks depend on the nature of the al system. For the integrators we set: first integrator will have as initial condition the value zero (initial velocity); the second integrator will have as initial condition the value 1 (initial elongation).

The amplifiers will have values 0.25 for the velocity (value of *c/m*) respectively 2.5 (value of *k/m*). Time is set for 0.1, it means the scopes will draw points for the diagrams each 0.1 second.

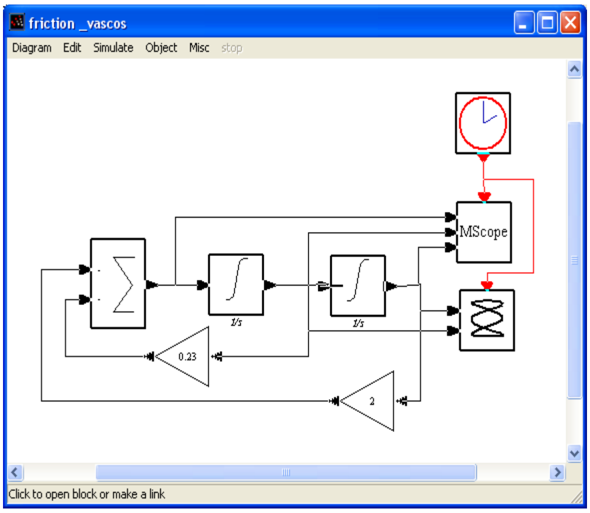


Fig. 10. Simulation of behaviour of a viscous dumped mass-spring system – block system

The diagrams in figure 11 shows that the equilibrium will be found later than in case of dry friction, but the proper frequency is the same.

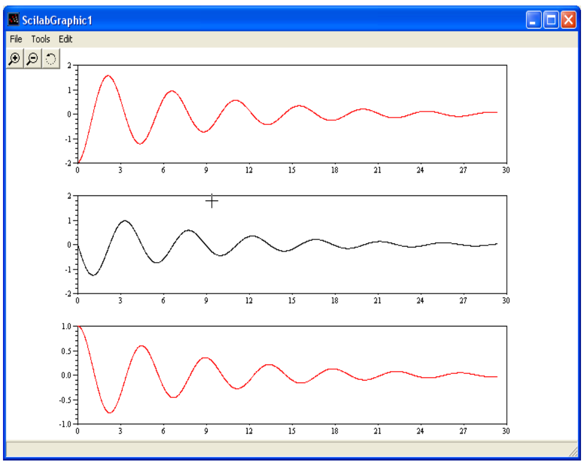


Fig. 11. Simulation of behaviour of a viscous dumped mass-spring system

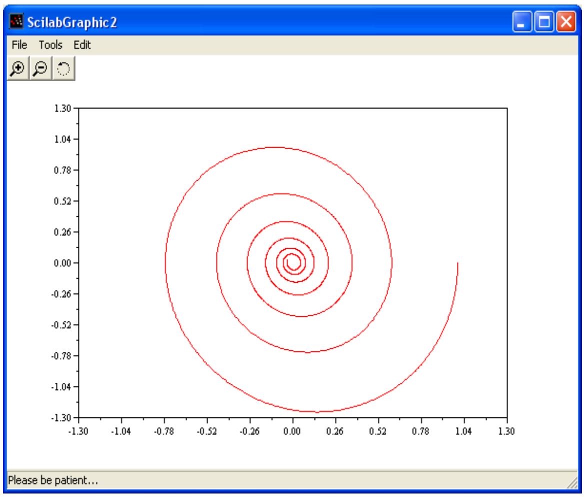


Fig. 12. Phase plain for the viscous dumped mass-spring system

The phase plain diagram is in a way similar to the equivalent diagram for Coulomb friction. It starts also by $y = 1$, $y' = 0$ and with negative acceleration but, because it's no difference between the equilibrium point where the spring is not tensioned and the static equilibrium point given by the friction, the phase portrait maintains the centre constant in the $y = 0$ and $y' = 0$ position. Another difference is that it consists in a spiral and not in a family of semi-circles.

Conclusions

Some aspects of SCICOS formalism have been presented. It is shown how activation and inheritance are used to obtain an environment for modeling dynamical systems in a precise and simple manner. The authors present also a concrete analysis of different technical systems using the SCICOS environment as calculation model and simulation, which are the lest steps by theoretical modeling.

References

[1] **Babeu, T.** – *Teoria elementară a rezistenţei materialelor*, Editura MIRTON Timişoara, 1998

[2] **Bratu, P.** – *Vibratiile sistemelor elastice,* Ed. Tehnica, Bucuresti, 2000

[3] **Chioncel, C.** – *Limbaje de simulare. Aplicatii de modelare si simulare*, Ed. Eftimie Murgu, Resita, 2004

[4] **Gillich, G.R.** – *Dinamica maşinilor • Modelarea sistemelor tehnice*, Editura AGIR Bucureşti, 2003

[5] **Gillich, G.R.** – *Trusses and frames analysis using the Finite Element Method*, JUPITER XXXI Konferencija, Beograd, 12-14 April 2005

[6] **Hollburg, U.** – *Maschinendynamik*, Oldenburg Verlag München Wien, 2002