Simulation Modelling Formalism: Ordinary Differential Equations

The concepts of ordinary differential equations (ODEs) and difference equations (ΔEs) are presented as means of describing the timewise development of dynamic systems as they are commonly found both in nature and in an artificial (human-created) environment.

1. Modelling of Physical Systems

Let us look at a simple example from mechanics first. Two masses sliding on a surface are coupled by springs to each other and to a vertical wall. An external force

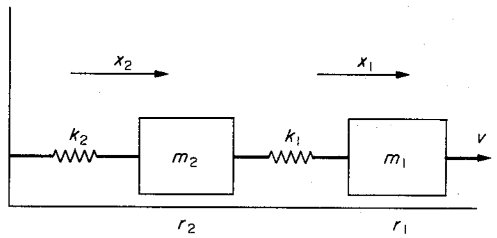

Figure 1
Representation of a simple mechanical system

$u(t)$ is exerted on the mass which is further from the wall. Graphically, this system can be depicted as in Fig. 1. According to Newton's second law, the mass m_i of each of these bodies multiplied by its acceleration a_i must equal the sum of all forces acting on the body. Thus

$$m_1 a_1 = u(t) - r_1 v_1 - k_1(x_1 - x_2) \tag{1}$$

$$m_2 a_2 = k_1(x_1 - x_2) - r_2 v_2 - k_2 x_2 \tag{2}$$

and with $v_i = \mathrm{d}x_i/\mathrm{d}t$ and $a_i = \mathrm{d}v_i/\mathrm{d}t = \mathrm{d}^2x_i/\mathrm{d}t^2$ we find

$$m_1(\mathrm{d}^2x_1/\mathrm{d}t^2) = u(t) - r_1(\mathrm{d}x_1/\mathrm{d}t) - k_1(x_1 - x_2) \tag{3}$$

$$m_2(\mathrm{d}^2x_2/\mathrm{d}t^2) = k_1(x_1 - x_2) - r_2(\mathrm{d}x_2/\mathrm{d}t) - k_2 x_2 \tag{4}$$

where m_i are the masses, r_i are the friction coefficients and k_i are the spring constants.

Using Newtonian mechanics we can describe the dynamics of this system by two second-order ODEs, that is, given the required number of initial conditions (in our case four), we can know the position, velocity and acceleration of each of the two bodies at any moment.

It was one of the great mathematical discoveries that many kinds of different systems show similar behavior, that is, they can be modelled by the same mathematical rules. It is, for example, easy to show that the electrical system of Fig. 2 can be described by the set of ODEs:

$$L_1(\mathrm{d}^2i_1/\mathrm{d}t^2) = \mathrm{d}v(t)/\mathrm{d}t - R_1(\mathrm{d}i_1/\mathrm{d}t) - (i_1 - i_2)/C_1 \tag{5}$$

$$L_2(\mathrm{d}^2i_2/\mathrm{d}t^2) = (i_1 - i_2)/C_1 - R_2(\mathrm{d}i_2/\mathrm{d}t) - i_2/C_2 \tag{6}$$

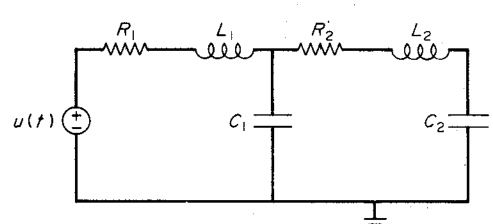

Figure 2
Simple electrical circuit mathematically equivalent to Fig. 1

In other words, from a mathematical point of view these two systems are identical, where the mechanical variable m_i corresponds to the electrical variable L_i, r_i to R_i, k_i to $1/C_i$, x_i to i_i and v to $\mathrm{d}v/\mathrm{d}t$.

2. Mathematical Models Versus Reality

Although we have shown that the same mathematical models may govern a variety of different systems, that is, these models have a large degree of generality that goes far beyond that of the physical systems they represent, one must not succumb to the temptation of identifying reality by the mathematical model representing it or, even worse, trust the computer figures better than one's eyes. For example, in our coupled-masses system we may easily exert a force which drives the masses into the vertical wall or which breaks the springs. Quite obviously, our model is not valid for this type of experiment. Validation of models is discussed by Sargent (1982) and elsewhere in the Encyclopedia (see *Validation of Simulation Models: General Approach; Validation of Simulation Models: Statistical Approach*).

3. The State-Space Description

So far we have dealt with two second-order ODEs. It is common to represent such models in the form of sets of first-order ODEs. An nth order ODE may be transformed into such a representation by solving for the highest derivative and by using the variable itself and all its derivatives except the highest as state variables:

$$x^{(n)} = f(x, \dot{x}, \ddot{x}, \ldots, x^{(n-1)}) \tag{7}$$

$$x = x_1, \quad \dot{x} = x_2, \quad \ddot{x} = x_3, \ldots, x^{(n-1)} = x_n \tag{8}$$

$$\begin{aligned} \Rightarrow \dot{x}_1 &= x_2 \\ \dot{x}_2 &= x_3 \\ &\vdots \\ \dot{x}_{n-1} &= x_n \\ \dot{x}_n &= f(x_1, x_2, x_3, \ldots, x_n) \end{aligned} \tag{9}$$

Applied to the mechanical spring problem, this gives

$$\dot{x} = Ax + bu, \qquad x \in \mathbb{R}^4 \tag{10}$$

where

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\dfrac{k_1}{m_1} & -\dfrac{r_1}{m_1} & \dfrac{k_1}{m_2} & 0 \\ 0 & 0 & 0 & 1 \\ \dfrac{k_1}{m_1} & 0 & \dfrac{-(k_1 + k_2)}{m_2} & \dfrac{-r_2}{m_2} \end{bmatrix}, \quad b = \begin{bmatrix} 0 \\ \dfrac{1}{m_1} \\ 0 \\ 0 \end{bmatrix} \tag{11}$$

This transformation is common use, because all our current numerical integration algorithms are developed to solve problems of the form

$$\dot{x} = f(x, t); \quad x(t = t_0) = x_0; \quad t \in [t_0, t_f]; x \in \mathbb{R}^n$$

(see Lambert 1973 and the article *Ordinary Differential Equation Models: Numerical Integration of Initial-Value Problems*).

It is only fairly recently that integration algorithms for other representations have begun to be developed. For example,

$$A\dot{x} = f(x, t) \quad (A \text{ singular}) \tag{12}$$

arises in the modelling of coupled differential and algebraic systems. More general is the representation

$$f(x, \dot{x}, \ldots, x^{(n)}) = 0 \tag{13}$$

In the near future we expect some special algorithms to be developed for the representation

$$\ddot{x} = f(x, \dot{x}, t) \tag{14}$$

which may be profitable in case of highly oscillatory problems (Petzold 1978).

4. Range of Applications

The concept of differential equations goes back to the time of Newton and the previously demonstrated analogy between mechanical and electrical systems is a child of the last century. However, the concept became really useful only in the middle of the present century when the first analog computers became available which allowed very general nonlinear ODE problems to be solved in a convenient and elegant way (see Korn and Korn 1964 and the article *Hybrid Analog–Digital Computers*). The interest in large-scale models is even younger, as only modern digital computers are capable of treating large sets of ODEs (which are in general very stiff) in an acceptably efficient and accurate manner (Gear 1971, Björck and Dahlqvist 1974).

As interest in these methods grew, the theory was developed further. The analogy between mechanical and electrical models was generalized into a concept which we nowadays call bond graph modelling (see van Dixhoorn 1982 and the article *Simulation Modelling Formalism: Bond Graphs*). Applications were extended recently to cover a wide variety of dynamical systems. The same or a similar set of differential equations could equally well represent a chain of chemical reactions, the ecology of a lake or some facets of Wall Street. We may even note first attempts to model some global quantities, such as global ecology or world food supply (Odum 1983, Frohberg 1982). Few would doubt that our planet really constitutes a dynamical system with all its characteristic properties. Making use of our planet's resources in a careless way can no longer be justified.

5. Difference Equations and their Relation to Differential Equations

So far we have discussed models of the form

$$\left.\begin{array}{ll} \left|\begin{array}{l} \dot{x} = f(x, u, t) \\ y = q(x, u, t) \end{array}\right| & \begin{array}{l} t \in [t_0, t_f] \\ x \in \mathbb{R}^n \\ u \in \mathbb{R}^m \\ y \in \mathbb{R}^p \end{array} \end{array}\right\} \tag{15}$$

where the u vector stands for the m inputs, the y vector represents the p outputs and the x vector denotes the n state variables. In the linear case, this can be written as

$$\left|\begin{array}{l} \dot{x} = Ax + Bu \\ y = Cx + Du \end{array}\right| \tag{16}$$

When solving the ODEs on a digital computer, we have to discretize the time axis. The most direct approach of doing so is by expanding a Taylor series to extrapolate to the point $(t^* + \Delta t)$ from a given time t^*:

$$x(t^* + \Delta t) = x(t^*) + \Delta t \cdot \frac{\mathrm{d}x(t^*)}{\mathrm{d}t} + \frac{\Delta t^2}{2} \cdot \frac{\mathrm{d}^2 x(t^*)}{\mathrm{d}t^2} + \cdots \tag{17}$$

When Δt is made sufficiently small, we may neglect all higher-order terms and obtain the first-order approximation:

$$x(t^* + \Delta t) = x(t^*) + \Delta t \cdot \dot{x}(t^*) \tag{18}$$

which is also known as Euler's integration algorithm. putting in the above ODEs, we can write

$$\begin{aligned} x(t^* + \Delta t) &= x(t^*) + \Delta t \cdot f[x(t^*), u(t^*), t^*] \\ &= f^*[x(t^*), u(t^*), t^*, \Delta t] \end{aligned} \tag{19}$$

In a somewhat simpler form, by representing the time axis through an integer range $t = (0, 1, 2, \ldots, k, \ldots, n)$, we can write

$$\left|\begin{array}{l} x(k+1) = f^*[x(k), u(k), k, \Delta t] \\ y(k+1) = g[x(k+1), u(k+1), k] \end{array}\right| \tag{20}$$

and the linear case

$$\left|\begin{array}{l} x(k+1) = Fx(k) + Gu(k) \\ y(k+1) = Hx(k+1) + Iu(k+1) \end{array}\right| \tag{21}$$

In the linear case, a better solution is obtained by means of the z transform, leading to

$$\left.\begin{array}{ll} F = \mathrm{e}^{A\Delta t}, & G = \displaystyle\int_0^{\Delta t} \mathrm{e}^{A\tau} B \, \mathrm{d}\tau \\ H \equiv C, & I \equiv D \end{array}\right\} \tag{22}$$

(Jury 1964 and the article *Digital Control: Practical Design Considerations*). This method is also sometimes referred to as Tustin's integration algorithm (Howe 1982).

In the case of linear systems, Tustin's integration is often quite a reasonable choice; Euler's method on the other hand is not, due to insufficient accuracy (first-order approximation) and too small a stability domain (Lambert 1973 and the article *Ordinary Differential Equation Models: Numerical Integration of Initial-Value Problems*). Why then has the Euler method received so much attention? There are several reasons.

(a) In a real-time environment (e.g., in a computer-controlled system), we cannot afford variable-order variable-step integration algorithms, as we must be able to guarantee that the time advance in the simulation model (simulation clock) keeps abreast of the real time advance. As these control algorithms are often linear, the linear ΔE is of primary importance here.

(b) Biologists and sociologists, who are more and more interested in analyzing the dynamic properties of their systems, are often not very well trained in numerical mathematics. For such workers, Forrester (1961) developed his notion of levels and rates (see *Simulation Modelling Formalism: Systems Dynamics*) where the level in the following sampling instant is computed from the level and rate at the current instant through the formula

$$\text{level}(k+1) = \text{level}(k) + \Delta T * \text{rate}(k) \qquad (23)$$

This is obviously nothing but a reformulation of Euler's integration. However, persons with weak mathematical background seem to be more at ease with the terms *level* and *rate* than with the term *differential equation*.

(c) More and more our ODE concept is being applied to so-called ill-defined systems, where the model parameters and even the governing equations are known to only a minor extent (Vansteenkiste and Spriet 1982). In such cases, it does not make sense to compute the state equations to 14 digits if perhaps not one of them is significant. It may then make sense to save computer time by discretizing the original ODEs to ΔEs. This is also partly a justification for point (b).

(d) There exist some systems which *are* discrete in nature, for example the population of insects which reproduce in spring only (see *Simulation Modelling Formalism, Discrete Arithmetic-Based*).

6. Difference Equations Revisited

Besides systems which are continuous in nature (at least as long as one does not try to model them down to the level of quantum mechanics), there exists another class of systems which are entirely discrete in nature and which are mostly modelled by a discrete-event mechanism (see *Simulation Models: Taxonomy*).

The simplest representative of this class of models is the single-server–single-queue model. We could obviously model the length of the waiting queue in, say, a barbershop by the ΔE:

$$x(k+1) = x(k) + u_1(k) - u_2(k) \qquad (24)$$

where $x(k)$ is the length of the waiting queue at time $k*\Delta t$, $u_1(k)$ is the number of customers entering the shop between $k*\Delta t$ and $(k+1)*\Delta t$ and $u_2(k)$ is the number of customers serviced between $k*\Delta t$ and $(k+1)*\Delta t$. If, for instance, both the interarrival time and the service time are exponentially distributed, both $u_1(k)$ and $u_2(k)$ will follow a Poisson distribution (Fishman 1973).

Although common in stochastic analysis, this approach has, until now, rarely been applied in the simulation context. It could, however, be automated to a large extent (that is, given a discrete-event model, the set of ΔEs could be generated automatically). Such an approach may prove beneficial for example in computer-aided manufacturing, as we have available many more tools for the analysis and synthesis of ΔEs than for discrete-event models.

Bibliography

Björck Å, Dahlqvist G 1974 *Numerical Methods*. Prentice-Hall, Englewood Cliffs, New Jersey

Fishman G S 1973 Statistical analysis for queueing simulations. *Manage. Sci.* **20**, 363–69

Forrester J W 1961 *Industrial Dynamics*. MIT Press, Cambridge, Massachusetts

Frohberg K 1982 The international linkage of open exchange models. In: Cellier F E (ed.) *Progress in Modelling and Simulation*. Academic Press, London, Chap. 5, pp. 77–84

Gear C W 1971 *Numerical Initial Value Problems in Ordinary Differential Equations*. Prentice-Hall, Englewood Cliffs, New Jersey

Howe R 1982 Digital-simulation of transfer-functions. *Proc. 1982 Summer Computer Simulation Conf.* Society for Computer Simulation, La Jolla, California, pp. 57–63

Jury E I 1964 *Theory and Application of the z-Transform Method*. Wiley, New York

Korn G A, Korn T M 1964 *Electronic Analog and Hybrid Computers*. McGraw-Hill, New York

Lambert J D 1973 *Computational Methods in Ordinary Differential Equations*. Wiley, London

Odum H T 1983 *Systems Ecology: An Introduction*. Wiley-Interscience, New York

Petzold L R 1978 An efficient numerical method for highly oscillatory ordinary differential equations, Department of Computer Science Report UIUCDCS-R-78-933. University of Illinois at Urbana-Champaign

Sargent R G 1982 Verification and validation of simulation models. In: Cellier F E (ed.) *Progress in Modelling and Simulation*. Academic Press, London, Chap. 9, pp. 159–69

van Dixhoorn J J 1982 Bond graphs and the challenge of a unified modelling theory of physical systems. In: Cellier F E (ed.) *Progress in Modelling and Simulation*. Academic Press, London, Chap. 12, pp. 207–45

Vansteenkiste G C, Spriet J A 1982 Modelling ill-defined systems. In: Cellier F E (ed.) *Progress in Modelling and Simulation.* Academic Press, London, Chap. 2, pp. 11–38

F. E. Cellier