Chapter 8

Studying Dynamic and Stochastic Systems Using Poisson Simulation

Leif Gustafsson

1. Introduction

1.1. Different types of models

In order to describe and understand complex systems, different types of models are used to capture different aspects of the real world. In some cases mathematics, statistics or some theoretical approach from operations research may be used, but usually some kind of simulation is required for more complex models.

Depending on the nature of the system and the purpose of the study, the researcher may choose to build a dynamic, stochastic, spatial or other kind of simulation model. In this chapter the focus is on the combined dynamic and stochastic aspects, which are often linked in such a way that a pure dynamic model may give a false dynamic behaviour and a pure stochastic model may give wrong stochastic estimates. The dynamic and stochastic aspects then have to be modeled in one and the same model.

In Fig. 1 the four possible combinations of static/dynamic and deterministic/stochastic models are displayed.

Stochastic models may be based on a strict statistical theory and dynamic models on a strict mathematical theory of difference or differential equations. Here the focus is on combined dynamic *and* stochastic models (the upper right square in Fig. 1) where the theoretical base is more problematic and rapidly becomes complex. In the discrete time domain this class of models can be expressed as: $\mathbf{x}(t+\Delta t) = \mathbf{f}[\mathbf{x}(t), \mathbf{u}(t), \mathbf{w}(t), t]$ where $\mathbf{x}$ is a state vector, $\mathbf{u}$ is an input vector and $\mathbf{w}$ is a vector of stochastic variables (Åström K.J., 1970). (In a linear case, models of the form $\mathbf{x}(t+\Delta t) = \mathbf{F}\,\mathbf{x}(t) + \mathbf{Gu}(t) + \mathbf{w}(t)$ are studied in e.g. automatic control.) Conceptually, such models can be obtained from a stochastic model by adding dynamics or from a dynamic

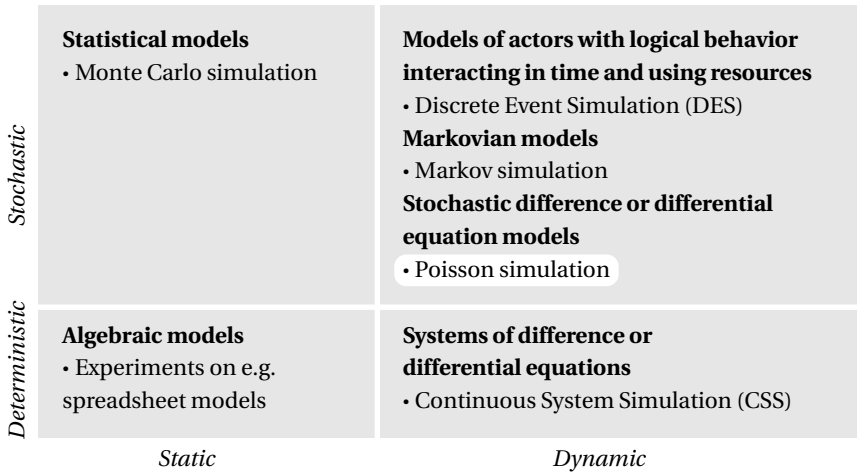

Figure 1. Classification of models in terms of static/dynamic and deterministic/stochastic and examples of simulation techniques. In this chapter the focus is on mesoscale models using Poisson simulation.

model by adding stochastics.

Historically, Discrete Event Simulation (DES) is based on the Monte Carlo-type of model where a network of stations (resources) is passed by some actors (objects, entities). Actors queuing for these resources generate a dynamic behaviour. The results from DES – statistics of number of actors, resource utilization, queue times etc. – also show that the focus is on the stochastic aspects rather than on the dynamic.

Another type of simulation is based on Markovian models, e.g. simulation of Markov chains, which is a stochastic process obeying the Markov condition which means that all information on the history up to the present moment is represented by the state. If a stochastic variable $x(t)$ of the process is discrete, it can take a number of discrete outcomes called states. Transitions between these states are specified in a transition probability matrix.

A challenge is to include an appropriate mechanism to Continuous System Simulation (CSS) so that we get stochastic results compatible with what we get from DES or Markov simulation. This is what we call Poisson simulation (Gustafsson, 2000). Here the foundation is a dynamic Continuous System Simulation model where random generators of proper distributions are used to control flows between the states. Then the results in the form of display of the dynamic behaviour (including the stochastic disturbances) are presented by the CSS language, while statistics within and between simulations have to be added.

1.1.1. Aspects of scale

There is also an interesting difference according to scale (in terms of number of items rather than in terms of size). On a *micro-scale* the focus is on a single or a few objects or events. Although differential equations can then be used to describe physical conditions using e.g. Newtonian equations, this is not the instrument to describe flows of objects or events over time. On a *macro-scale*, on the other hand, the numbers are so large that statistical fluctuations gradually lose their importance in accordance with the Law of large numbers. For example, statistical mechanics then transforms into thermodynamics. However on the *meso-scale*, with moderate numbers of entities or events, dynamics are well treated in terms of differential equations and the numbers are large enough to be treated statistically.

This also becomes clear comparing the DES and the CSS approaches. In DES we have a *micro* view of the system where each object is modeled, whereas conventional CSS, based on differential equations, requires *macro*-scale flows (state derivatives) large enough to be considered continuous. For example, when studying radioactive decay of an element the number of atoms is usually very large and therefore the stochastic variations are negligible. But if the number is not that large, statistical aspects may become crucial for correct estimates.

DES is also a powerful tool for handling the stochastics of a complex model on the *meso-scale*. But in many cases there is a desire to base one's model on well known differential equation relations, including stochastics in an appropriate way. The meso-scale may occur because of limited population, small sample, limited study etc., either because the number of cases are few or because of costs, time consumption, ethical restrictions or other factors. This is often the case in demography, ecology, epidemiology, medical trials, agriculture, forestry and genetics to mention just a few disciplines. Poisson simulation, described below, is then a powerful tool.

1.1.2. Purpose of this chapter

The purpose of this chapter is to demonstrate the importance of treating dynamics and stochastics together in the same model and to show how Poisson simulation can smoothly handle different types of models on the meso-scale. Therefore, a number of examples from various disciplines are given in order to demonstrate different aspects emerging from the interplay between dynamics and stochastics. In these examples it is also demonstrated how statistical aspects can be handled. In the last section of this chapter, there is a discussion of how Continuous System Simulation should be extended or

complemented to also include facilities for multiple runs and subsequent statistical analyses.

1.2. A dynamic and stochastic process

A model of dynamic and stochastic equations has a solution (behaviour) that varies over time but it also gives a new solution for each realization (new simulation run). Mathematically, this is described in the theory of *stochastic processes*, where a stochastic variable $X(t, \omega)$ is a function of two variables t and ω (defined on an appropriate space $\{t \in T, \omega \in \Omega\}$). Then t is time and ω is a realization* (solution, trajectory). See Fig. 2.

If we fix time we get a stochastic variable, and if we fix ω we get a realisation (trajectory).

1.2.1. Dependency between dynamics and stochastics

A dynamic and stochastic model results in stochastic solutions that vary because of the dynamic structure and the stochastic variations, but also because of couplings between the dynamics and stochastics. Thus, the stochastics excite the dynamics and the dynamics change the conditions for the stochastics. This means that there is a kind of feedback between the dynamics and the stochastics. Therefore, in the general case it is *not* possible to separate the problem into dynamic and stochastic parts.

In the general case, the stochastic variable (X) may be discrete or continuous. In addition, the time can be modeled as discrete ($t = 0, 1, 2\ldots$) or continuous ($0 \leq t < \infty$). Since this chapter focuses on the meso-scale in terms of moderate numbers, only the case where the stochastic variable takes discrete numbers in continuous time is treated.

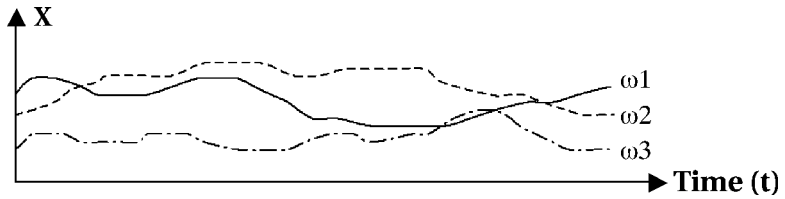

Figure 2. Three realisations of a stochastic process.

* The term "realization" is also used for different model structures in e.g. automatic control. Here, it will only be used in the meaning of stochastic outcome e.g. of a simulati on run.

1.3. Two traditions – focusing on structure or behaviour

One problem in discussing and handling dynamic and stochastic problems is that dynamics and stochastics are studied in two different traditions or cultures.

In mathematics and systems sciences the focus is on the model with an internal structure expressed e.g. in terms of a system of differential equations. The solution (behaviour) over time is then deduced analytically or calculated numerically from this structure. The focus is on modeling; finding a suitable model structure and fitting parameters (so that the studied property expressed as an objective function behaves as similarly as possible to the corresponding quantity of the real system).

In statistics, on the other hand, the connection between structure and outcome was once handled by deducing probability distributions from certain assumptions. However, a statistical model is more of a black box where input (e.g. exposure) generates an output with a certain distribution. The task is to fit the parameters of the distribution to observed data in e.g. a least square sense. (In likelihood statistics, though, the model often becomes more explicit.) Regression and statistical tests are major instruments. This works well when we have a proper standard distribution for the problem at hand, but when the underlying structure, conditions or assumptions become more complex, e.g. dynamics have to be included, we have to go all the way back to the fundamentals.

1.3.1. Reasons for starting with the structure

Let us demonstrate the advantages of starting the modeling work from the structural approach by means of a rather simple, but not trivial, example as shown in Fig. 3.

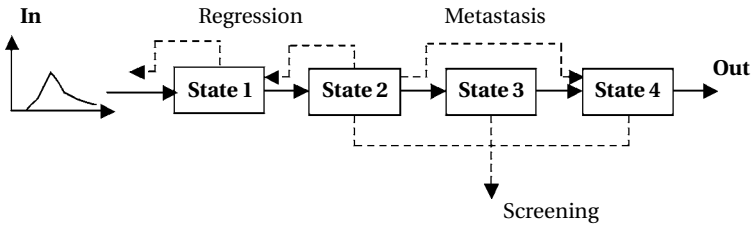

Figure 3. A rather simple dynamic and stochastic model to describe the stochastic process of a cancer in a birth cohort.

To make it more concrete, we assume that the four states depict the pre-stages or stages of cancer development in a cohort. Each state is represented by a box of prevalent cases and the flows between the states are incidences (rates). The input is the incidence of new cases and varies over age for the cohort studied. In State 1 the disease cannot be detected by screening tests, which it can in later states. From States 1 and 2 the disease may spontaneously regress. The output from State 4 represents deaths from the disease. From State 2 and onwards the tumour may produce metastases, taking the disease to State 4. With this (oversimplified) model we want to study the effects of different screening strategies on the mortality. Since we deal with moderate number of cases, statistical tools are needed to calculate averages, variation and confidence intervals for our estimates. However, we get into the following problems that can be handled with a structural approach, but not with a black box model of a distribution:

- There are *no named statistical distributions* to most structures that handle the underlying dynamics. Furthermore, those cases that enter State 1 at a certain time will not all go the same way and the fraction that does will not come out at the same time. (Seen from the other direction: the cases that come out at a certain time did not commence simultaneously.) Even if we knew the average sojourn times in the states, we would still not find a suitable stochastic distribution to describe this system.
- *Inputs* may be empirical functions obtained from measurements. This makes a theoretical approach impossible, including overall distributions between input and output. (In best case we get into convoluted time integrals of a stochastic process.) Numerical integration over time is the only option.
- The *initial value* problem. The states are usually not empty at the start. For example the study may concern a birth-cohort that is, say, 40 years old at start. What is the initial values of each state?
- To *understand/modify/control* etc., an explanation (structural) model – not a "black box" distribution – is required.

However, from a more practical point of view it is also better to focus on structure and to leave the calculations to the computer. Man is good at handling structure – computers can calculate.

1.3.2. Different model structures

A mathematical model relating the dynamics between input and output can be described in terms of a system of differential equations in different ways. In structural terms, the states can be arranged in series and in parallel. However even though different model structures can generate the same dynamic behaviour, these structures may differ with regard to sensitivity to disturbances, observability, controllability and other aspects. This is well known in systems sciences such as automatic control, (where the term *realisation* is used for different ways to structure a model – a term we avoid in this presentation). As will be seen later on, when stochastics are added the distinction between different (deterministically equivalent) model structures becomes important.

1.4. Poisson simulation

In Continuous System Simulation the model is composed of a system of differential (and algebraic) equations that are solved repeatedly for a small step of time until the end time is reached.

Poisson simulation is a method to include stochastics into Continuous System Simulation in a realistic way. This technique is treated in detail in Gustafsson (2000) and will only be sketched here. In the modeling, a system of differential equations can be described in terms of states and flows, see the Appendix. These flows can be handled stochastically by including random numbers of proper distributions. Thereby the stochastics become modeled as opposed to just adding input noise.

The basic idea is that any change of a state value occurs only through inflows and outflows to that state. Deterministically, each flow is regarded as having a constant flow rate or intensity (λ) (objects or events per time unit) during a very short time step, Δt. If this flow is composed of events that happen independently and one at a time, the flow is a Poisson process defined by the single intensity parameter, λ. The probability of an event during the finite time interval Δt then becomes proportional to the length of the interval. In terms of a statistical distribution, the number of events during the time interval Δt then becomes Poisson distributed and described by Po(λ*Δt). In the case where we have a discrete outcome it is thus natural to base the stochastics on the Poisson distribution (although other distributions may sometimes be justifiable).

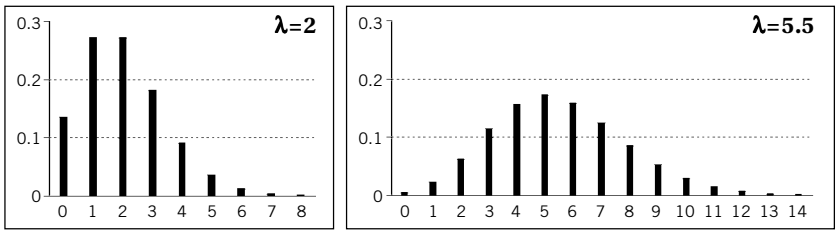

Figure 4. The Poisson distribution for $\lambda = 2$ and $\lambda = 5.5$. Notice that the intensity λ may have any positive value while the outcome always is a non-negative integer.

1.4.1. The Poisson distribution

The Poisson distribution, $\mathrm{Po}(\lambda)$, is a *discrete* distribution that describes the number of events that occur in a unit interval of time for a certain value of the intensity parameter λ when the events occur independently and one at a time. In Fig. 4 the Poisson distribution is shown for $\lambda = 2$ and $\lambda = 5.5$.

The Poisson probability function is $p(x) = e^{-\lambda} \cdot \lambda^x / x!$ if $x \in \{0, 1, ...\}$ and otherwise 0, while the mean and variance of the Poisson distribution are both equal to the intensity λ.

Since the interest is on the number of events during the time step Δt, we will in the following deal with the expression $\mathrm{Po}(\Delta t \cdot \lambda)$.

1.4.2. Poisson distribution or not?

The simplest case is when events happen *independently* and *one at a time* (Poisson process). Often the intensity varies over time, implying a non-stationary process where $\lambda = \lambda(t)$. This causes no problem because $\lambda(t)$ can be stepwise constant during the short time interval Δt – just like other quantities in Continuous System Simulation.

Of course, there are cases when the assumptions on independency and one at the time are not fulfilled or when the content of a state changes momentarily. In such a case it is up to the modeler to attach an appropriate distribution. If a Poisson distribution is not appropriate, any of the following may be used:

- any distribution D(..).
- sum of distributions $\mathrm{D}_1(..) + \mathrm{D}_2(..)$.
- product of distributions $\mathrm{D}_1(..) \cdot \mathrm{D}_2(..)$.
- distribution of distributions $\mathrm{D}_1(\mathrm{D}_2(..))$.
- distributions with several parameters $\mathrm{D}(p_1, p_2, ... p_n)$ where the parameters can be modeled separately.

1.4.3. Adding stochastics to a simple CSS model

A CSS model is represented by a system of ordinary differential equations. Each equation can be written as: $dx / dt = f(x, t)$ together with an initial value $x(0) = x_o$. In CSS languages this dynamic equation is sometimes written as: $x = \text{INTEG}(f, x_o)$, or in the Euler approximation: $x(t + \Delta t) \approx x(t) + \Delta t \cdot f(t)$, where f is the net flow rate to the state x. For the sake of simplicity, this is exemplified by a first order system written in the Euler form.

Example 1. *How stochastics are introduced – Radioactive decay*

Study a system of 100 radioactive atoms that will decay with a time constant of $T = 50$ time units. This means that at each time unit the fraction $p = 1 / T$ of the remaining atoms are expected to decay. This system has one state x and one outflow rate F which is proportional to the state value. Neglecting statistical fluctuations, a conventional CSS model might have the structure:

```
  x = 100                 [Initialization]
  Goto *                  [Since x(0) is given it remains to calculate
                           F at time=0.]
AGAIN:
  x = x + Δt · (–F)       [x is updated by Δt times the outflow rate –F.]
* F = p · x               [F is prop. to the remaining number of atoms (x).]
  time = time + Δt        [Time is advanced one small time-step.]
  if time ≤ Tend then Goto AGAIN
```

For each time step Δt, a certain fraction, $\Delta t \cdot p$, of the state value, x, leaves the state as an outflow. When the number of atoms in the state is large, the stochastic variations can be neglected. However when the number is not that large, the stochastic variations have to be considered. The expected outflow is still $F = p \cdot x$ per time unit or $\Delta t \cdot F = \Delta t \cdot p \cdot x$ during the time interval Δt. Since the properties of single events and independency, discussed above, are fulfilled, the number of events during Δt should be Poisson distributed with the intensity $\lambda = F = p \cdot x$. Thus, the outflow during the time interval Δt has a Poisson distributed variation denoted $\text{Po}(\Delta t \cdot \lambda)$. The flow rate then becomes: $\text{Po}(\Delta t \cdot \lambda) / \Delta t$. Therefore, the model is reformulated as:

```
x = 100
Goto *
```

AGAIN:
$x = x + \Delta t \cdot (-F)$
* F = Po$[\Delta t \cdot p \cdot x]$ / Δt [F is now stochastic. The rest is unchanged.]
$time = time + \Delta t$
if $time \leq Tend$ then Goto AGAIN

Po[~] means that for each time-step Δt, a random number is sampled from a Poisson distribution with the actual parameter value specified in the expression within the brackets. One advantage with this mechanism to introduce randomness is that the time step Δt can be changed to handle the dynamics properly without distorting the stochastics.

An alternative representation of a differential or difference equation model, often used in CSS languages like Powersim or STELLA, is a graphical description in terms of states and flows. The states are then represented by boxes (containing a certain amount at initial time) and inflows and outflows to the boxes. (There are also auxiliary symbols that just compute the algebraic relations.) A die in the flow symbol means random sampling from a specified distribution. A short description of this is given in the Appendix. In Fig. 5, the deterministic and stochastic models of radioactive decay just described are shown together with a graph of the results of a simulation. This model reveals the stochastic properties with regard to the number of items in state x. ■

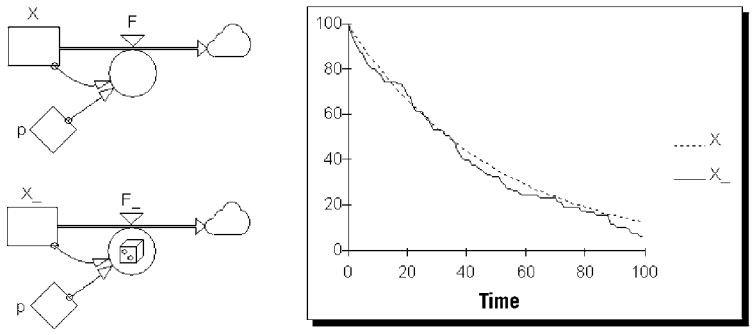

Figure 5. The radioactive decay model in CSS and as a Poisson simulation model. (See Appendix for understanding the dynamic diagram.) The deterministic solution is shown as a dashed line and one stochastic realisation is shown as a solid line.

In general, the states in a Poisson model are updated by inflows and outflows. Some flows may be deterministic, some Poisson distributed, and for other flows noise of any kind may be used.

Poisson simulation thus handles the case where the stochastic difference equation $\mathbf{x}(t + \Delta t) = \mathbf{f}[\mathbf{x}(t), \mathbf{u}(t), \mathbf{w}(t), t]$ can be written as $\mathbf{x}(t + \Delta t) = \mathbf{f}[\mathbf{x}(t), \mathbf{w}(x(t), \mathbf{r}, t), t]$ where $\mathbf{r}$ is a parameter vector. Notice that the vector $\mathbf{w}$ of stochastic variables here depends on the state vector $\mathbf{x}$. This approach preserves the difference equation model in a form that is mathematically well understood compared with e.g. a discrete event simulation model of the same system. At the same time, Poisson simulation offers a number of attractive features and solves several types of problem in an efficient way, which will be demonstrated in the next section.

2. Examples of meso-scale models using Poisson simulation

2.1. Poisson simulation examples

In the following, a number of small models will be treated in order to demonstrate new qualities that occur when dynamics and stochastics are modeled by Poisson simulation. The models are intentionally kept as simple as possible. This means that some of the models are oversimplified to be easily comprehensible. This will, however, not distort the purpose – to show how adding stochastics gives new qualities and possibilities to better discriminate between alternative structures of the real system.

Example 2. *Stochastics give information which can be interpreted in terms of model structure – the Gompertz model*
Suppose you want to understand the growth of microbes or tumour cells in some environment and that data show a growth up to a certain level and seem to fit a Gompertz model (Waggoner, 1986; Braun, 1993) which has the solution:

$$V(t) = V_o \cdot exp[(\lambda / a) \cdot (1 - exp(-at))] \quad (1)$$

where V is the number of e.g. microbes, $V_o = V(t = 0)$ and λ and a are positive constants. The aim is to understand the underlying structural conditions giving this behaviour. Therefore, we differentiate the equation [using $de^u / dt = e^u \cdot du/dt$] and get

$$dV / dt = V_o \cdot \lambda \cdot exp(-at) \cdot exp[\lambda / a \cdot (1 - exp(-at))] = \lambda \cdot exp(-at) \cdot V \quad (2)$$

However, the Gompertzian relation is sometimes expressed as:

$$dV / dt = a \cdot V \cdot \ln(K / V) \tag{3}$$

where a is the growth rate and K is the maximum size. This equation gives exactly the same solution if we set the constant $K = V_o \cdot exp(\lambda / a)$.

Thus, even for a first order model, the same solution (1) can be derived from two different differential equations (2) and (3). The differential equation (2) describes the changes in terms of an exponentially decaying growth parameter of the process, whereas equation (3) describes the "damping" in terms of the number of microbes. When V exceeds the capacity limit K the expression $\ln(K / V)$ becomes negative and equation (3) gets a negative feedback. However before displaying what happens when stochastics are added we will go one step further, interpreting the equations (2) and (3) in structural terms.

Equation (2) can be written as

$$dV / dt = [\lambda \cdot exp(-at)] \cdot V \tag{2a}$$

interpreting the factor $[\lambda \cdot exp(-at)]$ as an increasing mean generation time for the microbial growth, perhaps because of increasing resistance from the environment. However it can also be written as:

$$dV / dt = \lambda \cdot [exp(-at) \cdot V] \tag{2b}$$

with a constant mean generation time λ, but where the increase in microbes is hampered by death for some reason. (This latter effect is also seen among tumour cells where a necrotic region can develop in the tumour or with mushroom circles in a lawn.) If this is the case we have in fact two "flows" hidden in the net flow in equation (2b). One flow, F_1, describes the increase and the other, F_2, the necroses. One mathematically possible model structure is then:

$$\begin{aligned} dV / dt &= F_1 - F_2 \\ F_1 &= \lambda \cdot V \\ F_2 &= \lambda \cdot [1 - exp(-at)] \cdot V \end{aligned} \tag{2b'}$$

Then $dV/dt = \lambda \cdot V - \lambda \cdot [1 - exp(-at)] \cdot V = \lambda \cdot [exp(-at)] \cdot V$ still constitutes equation (2b) above.

The three dynamic structures of the Gompertz relation (1) above based on the equations (2a), (2b') and (3) are shown in Fig. 6 a–c as dynamic diagrams where the state V is represented by a box and inflows and outflows by arrows. Their deterministic behaviours are exactly the same.

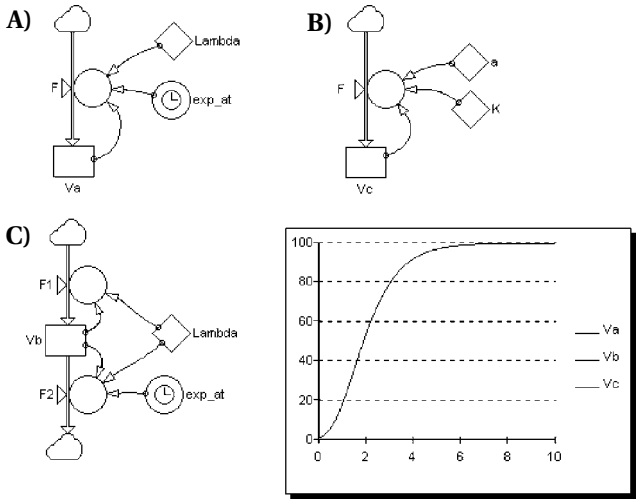

Figure 6. Three equivalent structures a, b and c for equation (2a), (2b') and (3), respectively, all giving the same behaviour $V(t) = V_0 \cdot exp[(\lambda / a) \cdot (1 - exp(-at))]$ over time.

Adding stochastics

However, results from experiments on the real system vary within and between the experiments. Therefore, such variations should also be generated in the model. (Here, we assume that this randomness occurs from stochastics in the behaviour outside our control, and not from observational errors.)

This means that the content of a flow F_x at each point in time becomes a stochastic variable. For simplicity's sake, we assume that the numbers of births or deaths during a short interval Δt are independent and occur one by one and thus we regard these flows as Poisson processes. For the equations (2a), (2b') and (3), presented as difference equations in Euler's form, the flow equations are modified as:

(2a) $\Delta V = \Delta t \cdot F$

$F = \lambda \cdot V \cdot exp(-at) \quad \Longrightarrow \quad F = \mathrm{Po}[\lambda \cdot V \cdot exp(-at) \cdot \Delta t] \,/\, \Delta t$

(2b') $\Delta V = \Delta t \cdot (F_1 - F_2)$

$F_1 = \lambda \cdot V$ $\qquad\qquad$ $F_1 = \mathrm{Po}[\lambda \cdot V \cdot \Delta t] \,/\, \Delta t$

$F_2 = \lambda \cdot (1 - exp(-at)) \cdot V \;\Longrightarrow\; F_2 = \mathrm{Po}[\lambda \cdot (1 - exp(-at)) \cdot V \cdot \Delta t] \,/\, \Delta t$

(3) $\Delta V = \Delta t \cdot F$

$F = a \cdot V \cdot \ln(K / V) \;\Longrightarrow\; F = \mathrm{Po}\ [a \cdot V \cdot \ln(K / V) \cdot \Delta t] \,/\, \Delta t$

Since the systems of equations are now stochastic we get different results for each simulation run.

In Fig. 7 a number of realizations are made for each of the models (2a), (2b') and (3), (which without stochastics gave identical results) with initial values $V(0) = 1$:

In the realisation shown in Fig. 7a, the "time controlled damping" is sensitive for randomness in the beginning. If the first events come rapidly, the microbes will grow fast and reach a high value before the process is strangled. Therefore, we get a wide variation between runs. Within a run, however, the variations will disappear when the exponential damping increases, i.e. when $exp(-at)$ goes to zero.

In the realisation in Fig. 7b the inter-variations are very wide. When the exponential term in $(1 - exp(-at))$ disappears, there remains both an inflow and an outflow of the same expression $(\lambda \cdot V)$. The net flow from these may then stochastically drift to any non-negative value. (A more realistic model would here be based on the logistic equation where the outflow is related to a quadratic competition term which would stabilize the model around the deterministic end level, but still with intra-variations – see Example 3.)

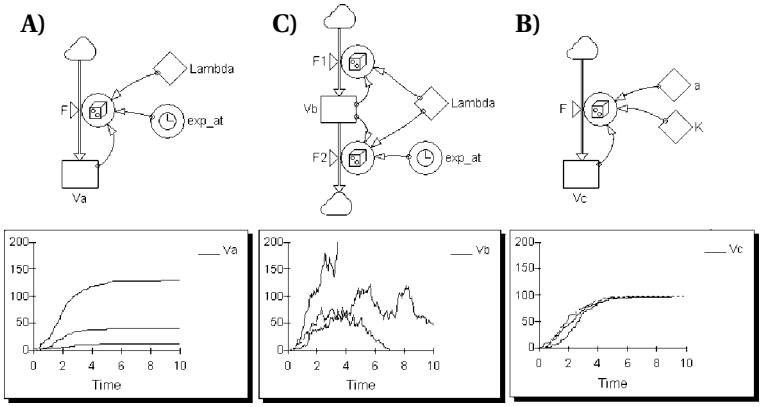

Figure 7. A number of simulations made for each of the models (2a), (2b') and (3). The dashed line is the behavior of the deterministic model.

In the last realisation, shown in Fig. 7c, the inter-variations arise early because of randomness in the start, just like in 7a. However this time the damping is because of negative feedback when the number of microbes exceeds the equilibrium level. At that time all further variations disappear.

As seen, modeling dynamics and stochastics together makes it possible to discriminate between different underlying structures. Carrying out a number of experiments on the real system gives us information about inter and intra variations that have to occur from an underlying structure (still assuming the stochastics to be inherent in the system and not a result of bad observational accuracy). With Poisson simulation such information can be modeled and interpreted. ■

Example 3. *Adjusting the model structure to fit observed stochastic variations – A logistic model*

Since the stochastic equation (2b') in Example 2, above, drifts freely, it doesn't seem to be a realistic model of a biological system. Therefore, we keep only its main structure of separate inflow and outflow, but replace the Gompertz model with a logistic model of the form:

$$dV/dt = F_1 - F_2$$
$$F_1 = a \cdot V$$
$$F_2 = b \cdot V^2$$

The change in the state value V thus increases proportionally to V and decreases because of competition proportionally to the term V^2 (meaning that each of the V organisms competes with all the others).

But how does competition work in a given case? Does it result in an outflow of dying individuals or does it hamper the growth process or is it involved in both? From a deterministic point of view the resulting behaviour will be the same. But stochastically, competition resulting in reduced growth will give small variations gradually dying out when the system reaches equilibrium, while competition implying deaths will give the largest variations for the largest value of the state, i.e. around equilibrium.

It is also possible to add an extra, proportional outflow of deaths (and compensate for it by a larger inflow) keeping the above net equation. This will make the variations still larger. In practice, fitting the model to real data (including the stochastic variations) will help in discriminating among these options.

In Euler's form, these three structures may be modeled as:

Competition as outflow:

$\Delta V = \Delta t \cdot (F_1 - F_2)$

$F_1 = a \cdot V$ $\qquad F_1 = \text{Po}\,[a \cdot V \cdot \Delta t]\ /\ \Delta t$

$F_2 = b \cdot V^2$ $\qquad F_2 = \text{Po}\,[a \cdot \text{V}^2 \cdot \Delta t]\ /\ \Delta t$

Competition included in a net inflow:

$\Delta V = \Delta t \cdot (F)$

$F = a \cdot V - b \cdot V^2$ $\qquad F = \text{Po}\,[(a \cdot V - b \cdot V^2) \cdot \Delta t]\ /\ \Delta t$

Competition as outflow and extra flows:

$\Delta V = \Delta t \cdot (F_1 - F_2 - F_3)$

$F_1 = (a + c) \cdot V$ $\qquad F_1 = \text{Po}\,[(a + c) \cdot V \cdot \Delta t]\ /\ \Delta t$

$F_2 = b \cdot V^2$ $\qquad F_2 = \text{Po}\,[a \cdot V^2 \cdot \Delta t]\ /\ \Delta t$

$F_3 = c \cdot V$ $\qquad F_3 = \text{Po}\,[c \cdot V \cdot \Delta t]\ /\ \Delta t$

In the last equation the parameter c can be adjusted to fit the observed variations. In addition, the amount of competition referring to input or output can be parameterized in this way.

With this approach each model can be optimally fitted to real data, not only for mean values but also for intra and inter variations. This gives a base for testing various structural hypotheses. ■

Example 4. *How stochastics excite dynamic variations and how new qualities are added – A Lotka-Volterra model*

The Lotka-Volterra equations describe a prey-predator system for two species e.g. Rabbits (X) and Foxes (Y) by differential equations (Volterra,1926; Luenberger, 1979). The rabbits breed at a rate proportional to their number X. They die because of encounters with the foxes, which is proportional to $X \cdot Y$. There is also competition among the rabbits, where each rabbit competes with all the others. Competition, therefore, is proportional to X^2. The encounters with rabbits give the foxes energy to breed, so they increase in proportion to $X \cdot Y$. Finally, the fox' death rate is proportional to the number of foxes, Y. The Lotka-Volterra model therefore has the form:

$$dX / dt = aX - bXY - kX^2$$
$$dY / dt = cXY - dY$$

where a and c are fertility constants, b and d mortality constants, and k is a constant for competition.

By setting the derivatives dX / dt and dY / dt to zero and solving for X and Y we obtain three possible stationary solutions (of no rabbits and foxes, only rabbits or both rabbits and foxes):

1) $X = 0$ and $Y = 0$, 2) $X = a / k$ and $Y = 0$, 3) $X = d / c$ and $Y = (a - kd / c) / b$.

Setting e.g. $a = 0.2$, $b = 0.005$, $c = 0.005$, $d = 0.3$ and $k = 0.001$ gives in the second case (foxes become extinct) $X = 200$ and $Y = 0$, and in the third case (both species survive) $X = 60$ and $Y = 28$ as stationary solutions.

To treat the system numerically we rewrite the equations in Euler's form:

$$\begin{aligned}
\Delta X &= \Delta t \cdot (F_1 - F_2 - F_3) \\
F_1 &= aX \\
F_2 &= bXY \\
F_3 &= kX^2 \\
\Delta Y &= \Delta t \cdot (F_4 - F_5) \\
F_4 &= cXY \\
F_5 &= dY
\end{aligned}$$

Initializing this simulation in steady state with $X(0) = 60$ and $Y(0) = 28$ gives the trivial results of two constant lines for X and Y in a time plot. Even if we disturb the system to generate variations, these will die out as shown in Fig. 8.

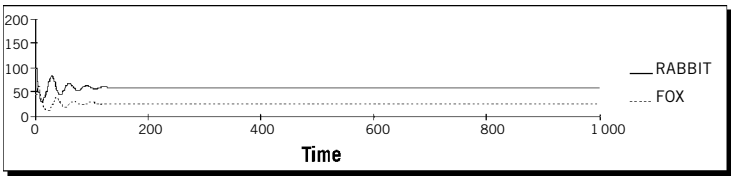

Figure 8. The deterministic system is damped by the competition and then stays in an equilibrium.

Now assume that the flows of births, deaths and competition follow the Poisson requirements (this can be debated – but the qualities discussed below don't require just a Poisson distribution). Then each flow F_x is transferred into the form: Poisson($\Delta t \cdot F_x$) / Δt. The model, with stochastic flows, is shown in Fig. 9.

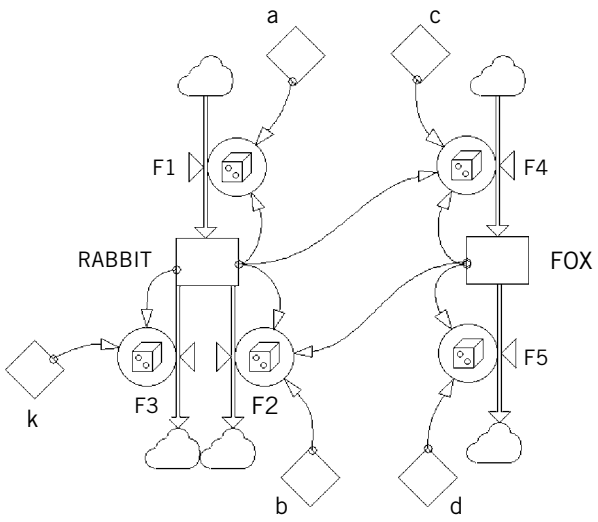

Figure 9. The structure of Volterra's equations. Randomness is introduced in each flow.

Starting the simulation from the equilibrium values $X(0) = d / c = 60$ and $Y(0) = (a - kd / c) / b = 28$ gives a behaviour like that in Fig. 10.

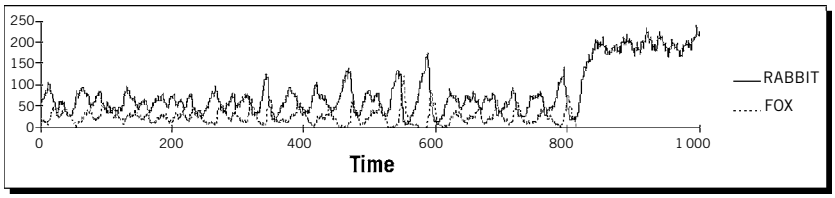

Figure 10. An example of the behavior of the Poisson model starting at equilibrium.

In the Poisson simulation the inherent dynamics of the system are excited by the stochastic fluctuations. We also see that these dynamics not only cause the numbers of rabbits and foxes to vary, but they also display the periodic pattern and period length that is typical of the deterministic model starting off equilibrium.

Note also that in some simulations all foxes will starve to death, making the system vary around the second stationary solution. In the simulation shown in Fig. 10, this happens at about time 820, making the rabbits increase and fluctuate around the steady state $X = a / k = 200$.

If we instead had just added noise to the results, it would only have resulted in a superimposed noise without exciting the dynamics.

Including noise of some distribution (say Normal(m_i, σ_i)) to each of the five flows would excite the dynamics so that they oscillated with the periodicity given by the dynamics. It would however add several problems. 1) it would give ten constants m_i and σ_i to be estimated. 2) We would then have to normalize the variation for changing Δt (otherwise the noise influence would change with the computational step-size used). 3) The foxes might recover from zero or even negative values. There is also a substantial risk that a negative state value makes the model unstable and causes numerical overflow. However the results would, for example, never have revealed a switch to the mode $X = a / k$ and $Y = 0$ which may happen in the real system. This is because the variance would then not vary in accordance with the state values and also because we would then have lost the intrinsic integer mechanism of the Poisson distribution.

This example demonstrates several qualities. Firstly, it illustrates the fact that dynamics and stochastics may generate persistent oscillations even when starting at an equilibrium and even though a corresponding deterministic model would damp them out. Secondly, the model may flip to a mode where foxes become extinct – a quality that could not happen in a deterministic model. Thirdly, the Poisson mechanism preserves integer numbers of the rabbits and the foxes. ■

Example 5. *Experimental design*
A common problem in medical, biological, ecological, agricultural and many other types of systems is to understand and estimate the effects of some action, "exposure" (e.g. medication, plant thinning, hunting, pesticide spraying, acidification, liming, irradiation, stimulus etc.) that via a dynamic process causes some effects (like healing, growth, migration, sickness, mortality, gene modifications, response etc.). A dynamic model is then needed to understand the system, calculate the damage or benefit from the exposure over time, and a statistical treatment is needed to handle variations and uncertainties.

A study of cause and action has an experimental design where the effects on an exposed study system and an unexposed ("background exposure" only) reference system are compared. These systems should in other respects be as similar as possible. Treating this problem with *only* statistical methods causes problems. For example in an epidemiological study, the participants are randomized into a study and a control group. The controls

only receive background exposure while the study group is given, say, twice this exposure. The numbers of cases (sick, dead, with some symptoms etc.) are cumulative for each group. But what is the *sojourn time* for an exposed individual to become a case, and does this time depend on the amount of exposure? Even if the sojourn time is known on average, there is another problem. Since the process between exposure and outcome is dynamic, the cases exposed at the same point in time will emerge successively over time. Thus, for how long a time should the follow up be performed? If a too short time is chosen, most of the exposed individuals will not get enough time to become cases and there will be an underestimation. If a too long time is used, most of the cases induced by the extra exposure will show up, but they will be diluted by the many cases who get the disease from the ever-present background exposure. This weakens the statistical estimates.

To handle these problems properly, a dynamic model is needed to handle the time evolution and to understand the process. However, we also need to estimate the stochastic variations to "guarantee" the results, in terms of

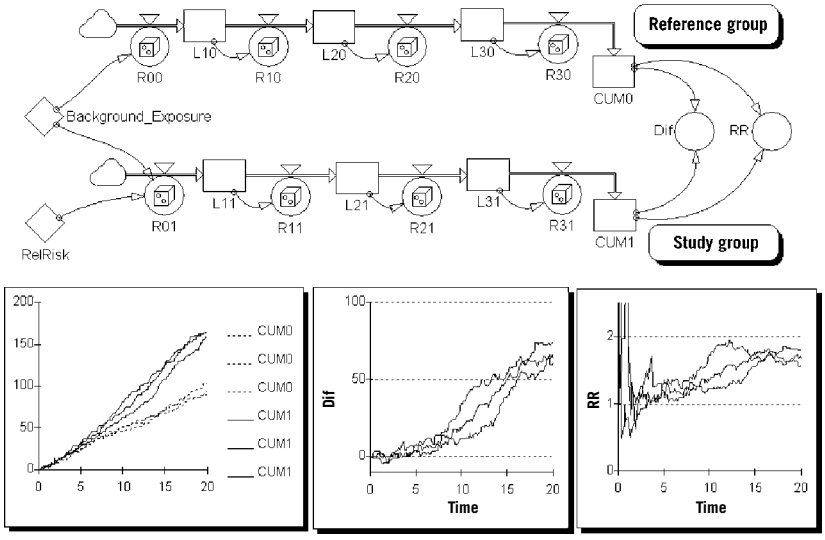

Figure 11. A Poisson simulation model of a reference system (0) only exposed to background exposure and a study system (1) in which the total exposure is twice the *Background Exposure*. The studied effects are cumulative number of cases over 20 years. Three simulation runs of the cumulative effect for the two systems are shown in the first diagram.The second diagram shows the results for the difference (*Dif*) and the third shows the ratios (*RR*) between cumulative numbers of cases from the study group and reference group for each of the three simulation runs.

variations, interval of significance or hypothesis tests. Furthermore, dynamics and stochastics have to be treated together to get realistic results. Thus we use a Poisson simulation model of this experimental design as shown in Fig. 11.

Assume that the disease from exposure to outcome can be described by a third order dynamic model and that the known sojourn time (average time in all the three states) is six years. The goal is to estimate the effects of the extra exposure imposed on the study group. "*RelRisk*" is an a priori unknown parameter that tells how many times the background exposure level the study group is exposed to. (*RelRisk* = 2 means that the *total* exposure of the study group is twice the *Background Exposure*.)

After fitting this model to real data, the effects of the exposure developing over time can be simulated. By making many runs the stochastic variations of the exposure in terms of mean, average, confidence intervals etc. can be estimated in absolute terms or in terms of difference (*Dif*) or ratio (*RR*). With a model that reasonably well describes the dynamics and stochastics, this is readily done by making some hundreds of simulations and calculating the mean, variance, confidential interval or testing a hypothesis. It can also be seen that although the sojourn time was only six years, a follow up period of 20 years is not enough to get a good estimate of *RR* = *Cum 1* / *Cum 0*. We get a value around 1.7 to 1.8 instead of 2.0.

A more powerful approach is *to estimate parameters involved* (like sojourn times or fractions that progress or regress). Such results relate to the underlying structure of the disease and are more comparable between studies than statistical measures that vary with the follow-up time used. Parameter estimation would also give an unbiased estimate of *RelRisk* independent of the length of the follow-up period. A longer follow-up time then does not change the expected value but only reduces the variances of the estimates.

Experimental design is for many sciences by far the most important method of studying a dynamic system. However, experiments on the real system then have to be interpreted with the help of a model that preserves both the dynamic and stochastic aspects. Poisson simulation here offers a powerful technique for modeling and analysis. ■

Example 6. *Interpretation of Life Table information*
Five-year survival, LD-50 and other scalar measures are often diffuse measures of a disease, but are still in wide use. A more powerful description of survival in a disease is to use *life tables*. This means that you present the number of survivors as a function of time after diagnoses (or treatment). At time zero

you have a closed cohort of diseased objects (patients, animals or plants). As time goes by, a number of the objects die. Unfortunately, this curve of the number of survivors over time is not only a measure of the disease. The distribution of diagnoses in different stages and treatment also affects the life table. To further interpret life table data, a realistic underlying model of the disease to be fitted to actual data is needed.

At first glance the life table may look somewhat like an exponential decay. However, a closer study reveals that the disease process has to be described by several consecutive states like those in Fig. 12. In this case we assume that a certain amount are successfully treated in Stages 1 and 2 but that Stage 3 is untreatable.

Conclusion: With the Poisson simulation we can model and analyze both the dynamics and the variations of a life table. ■

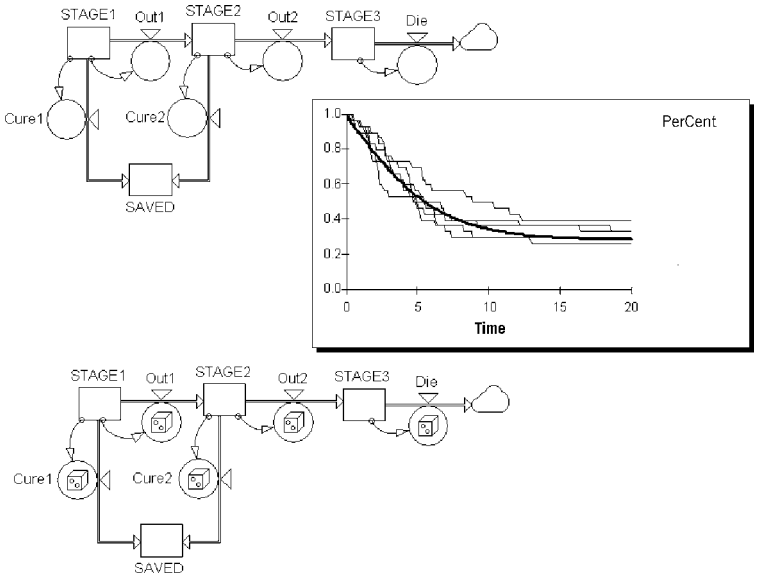

Figure 12. An example of a deterministic life table model and its Poisson equivalent. The behavior of these models is also shown by a thick line for the deterministic solution and thin lines for three stochastic realizations. (The numbers are presented in per cent.)

Example 7. *Replacing a large number of Markov simulations – Ion channels in a neurone*

Ion channels transporting sodium or potassium ions play a fundamental role in transmitting signals in neurones. By attaching a probe to a neurone the current from a huge number of sodium ion channels can be recorded. During an experiment where a neurone is stimulated, the non-stationary current from the sodium channels changes over time but also fluctuates stochastically. This is interpreted as a varying number of ion channels being open at a given point in time. To explain the results from such measurements dynamically and stochastically, a Markovian model of a single ion channel is used. This model may look like that shown in Fig. 13.

The number of ion channels in the studied part of the neurone and contributing to the current is in the order of 1 000 to 100 000. This may look like a huge number, well beyond the meso-scale, implying a dynamic but deterministic model. However, only a small fraction of the ion channels are open at a particular point in time. Since the current is proportional to (the negative value of) the number of open channels, fluctuations from a moderate number of channels are seen.

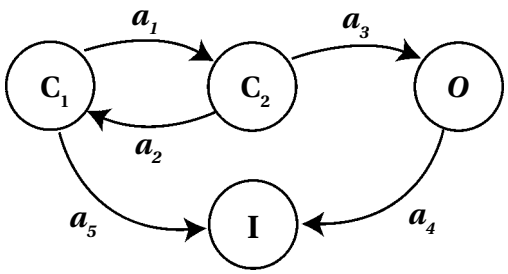

Figure 13. A possible Markov model of a single ion channel. The channel has a number of inner gates that may be open or closed. The states C_i refer to combinations of the states (open or closed) of these gates. The state O means that all gates, i.e. the channel, are open. The state I means that the channel is inhibited and cannot open again during the short experiment. The numbers a_1 to a_5 are transition probabilities between states. At start the initial state is C_1. (Note that in a Markov model the term *state* refers to a possible outcome of the stochastic variable, and does not represent the variable itself as in e.g. differential equation models or CSS simulation.)

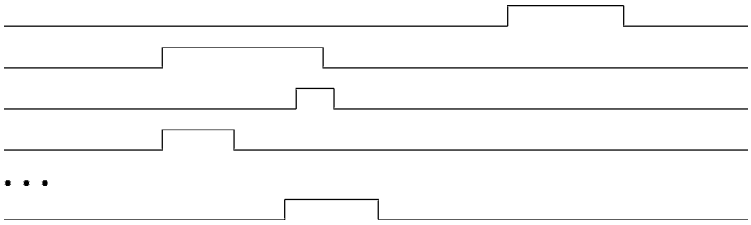

Figure 14. Results from N realizations of the Markov model. By summing up for each point in time, the sum of open channels, i.e. (minus) the current can be displayed revealing the dynamics and stochastics of the process.

By building 10^3 to 10^5 copies of the Markov model, or more cleverly, by simulating the Markov model 10^3 to 10^5 times, we get that number of time plots, as in Fig. 14.

Imagine now that the Markov model instead could be initiated by N, instead of one, item in state C_1 and that the transitions (still independent and one at the time) are random. Then the number of transitions from one state to another during a time interval Δt becomes Poisson distributed and proportional to the number in the departing state, to the transition probability,

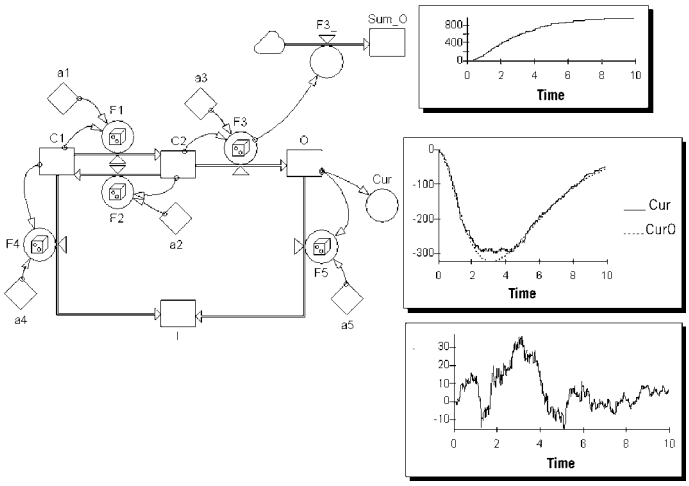

Figure 15. A Poisson simulation model of the N ion channels. C_1 is initiated to N. The current (*Cur*) is displayed to the right of the model (solid line) together with the results from the corresponding deterministic model (*Cur0* displayed as a dashed line). This model is not shown. To calculate the number of ion channels that have participated during the simulation, an extra, deterministic, accumulator (*Sum_O*) is introduced. From the top diagram it is seen that this total number is about 150–200 (meso-scale!).

and to the time interval Δt. However this is exactly Poisson simulation in its straight forward setting. Thus, let us make an equivalent model of the N ion channels, as shown in Fig. 15.

As demonstrated, when studying the aggregated results of many Markov processes, Poisson simulation can readily handle the problems much more efficiently than Markov studies. The same results are obtained with a much smaller and more flexible model, and the calculations take a small fraction of the time required using a Markov model. It is also easy to change the structure, parameters, number of channels, or to add extra facilities like accumulators for number of channels open at any time during the simulation, or to compare with deterministic results. ■

Example 8. *Testing a hypothesis – Lanchester's Quadratic Law*
In warfare as well as disarmament, it is important to understand the balance between forces in dynamic and stochastic terms. A fundamental principle of concentration was formulated by F. Lanchester in 1916 and is known as Lanchester's Model of Warfare or Lanchester's Quadratic Law (Lanchester, 1916). See also (Newman, 1956; Saaty, 1968).

The underlying assumption is that two forces are in combat where each unit (soldier, vessel or aircraft) can fire against any unit on the other side. The number of casualties per time unit on the enemy side is then proportional to the number of firing units and to the power of their guns and vice versa. This can be mathematically formulated as:

$$dx / dt = -\alpha \cdot y$$
$$dy / dt = -\beta \cdot x$$

where $x(t)$ and $y(t)$ are the number of units and β and α are the power of their arms.

If each unit has the same power ($\alpha = \beta$) we can write $dx / (\alpha y) = dy / (ax)$ or $x \cdot dx = y \cdot dy$. Integrating this from time zero when the forces are complete until time T when one of the forces is completely eliminated gives:

$$\int_0^T x \cdot dx = \int_0^T y \cdot dy \qquad \text{implying } x^2(T) - x^2(0) = y^2(T) - y^2(0)$$

where the smaller force, here y, is eliminated at time T and thus $y(T) = 0$. Then we can write: $x^2(0) = y^2(0) + x^2(T)$ which is Lanchester's Quadratic Law of Warfare.

In words this says that after the weaker force is defeated the number of units remaining in the stronger force can be calculated from the "Pythagorean" statement where $x(0)$ is the hypotenuse and $y(0)$ and $x(T)$ the shorter sides. This theorem shows the importance of keeping the forces concentrated. A weaker army can beat a stronger one if the latter is divided into parts.

Thus if Force x has 13 units and Force y has 12, Lanchester's model predicts 5 remaining units for x when the battle is over ($13^2 = 12^2 + 5^2$). However even if all other conditions are equal, randomness is part of the game. Therefore, the model is also reformulated as a Poisson model. This means that during each increment of time, $\Delta t \cdot \alpha \cdot y$ units on the x side and $\Delta t \cdot \beta \cdot x$ units on the y side will *on average* be eliminated. Using Poisson distributions (the most realistic distribution to be used in a particular case can be discussed) we have:

$$\begin{aligned} \Delta x &= \Delta t \cdot F_1 \\ \Delta y &= \Delta t \cdot F_2 \\ F_1 &= \text{Po}\,(-\alpha \cdot y \cdot \Delta t)\,/\Delta t \\ F_2 &= \text{Po}\,(-\beta \cdot x \cdot \Delta t)\,/\Delta t \end{aligned}$$

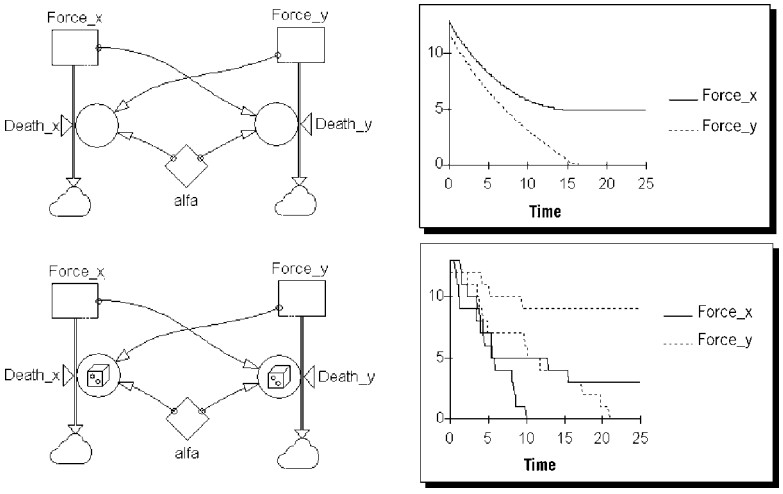

Figure 16. Lancaster's Quadratic Law. a) As a deterministic model with the results for Force X shown as solid lines and the results for Force y shown as dashed lines . b) As a Poisson model with two realizations where x wins one and y wins the other battle. (Technically, the outflows are also limited so that the states cannot become negative.)

Setting $\alpha = \beta = 0.1$ (in some time unit, e.g. minutes or hours) and the initial values of the forces to $x(0) = 13$ and $y(0) = 12$ units gives the model and results shown in Fig. 16.

Here again, a new quality enters the picture when randomness is added. For the deterministic model the outcome was given. After adding stochastics, this is no longer the case.

By making a large number of simulations, we can study the outcome in statistical terms and test the hypothesis that Force x will win, for example with a probability of 75 %. Making 200 runs gave the victory to x in 126 of the battles and to y in 74. A rough estimate is that x wins in about 63 % and y in 37 % of the battles. An initial number of 13 units to 12 is thus not enough to win with a probability of 75 %.

Conclusion: This example shows a general method to include stochastics in an appropriate way and to run the model a large number of times so that statistics are obtained. From these statistics you may e.g. test a hypothesis. You may also calculate what force you need to win with, for example, 95 % probability. ■

Example 9. *Obtaining statistical estimates from N simulations – A queuing model*

Queuing systems are theoretically studied in queuing theory and in more complex cases by using Discrete Event Simulation. Such studies are frequent for e.g. manufacture- or service systems.

In Discrete Event Simulation the model jumps to the next event of interest like: arrival of a new customer, service for a customer is just finished etc. Therefore, the mechanism to calculate the next arrival has to be different from that in Markov or Poisson simulation. In the simplest case of independent arrivals and one at a time, i.e. a Poisson process, the *time to the next arrival* is drawn from a negative exponential distribution. (This is identically similar, but saves us from stepping time Δt by Δt.) However, when the arrival intensity, $\lambda(t)$, varies over time we get a problem. Say that the arrival intensity to a lunch bar is very low at 10 a.m. and thus the next arrival is scheduled for 3 hours later, but at 11 o'clock the intensity rises dramatically. Then the mechanism where each customer at arrival schedules his successor will not work. We then have to work with thinning or some other mechanism. Although Poisson simulation was not invented for queue studies, it can easily include queues as part of a model and handle time varying intensities.

Now assume a M/M/1 queuing system (M stands for Markovian and

M/M/1 means exponentially distributed inter-arrival and service times and one service station. Thus, we have a Poisson processes.) The arrival intensity varies over the day (8*60 = 480 minutes) like: $0.1 \cdot (1 + \sin(time / 50))$ customers per minute. The service rate has a capacity of 0.12 customers per minute which on average is somewhat more than the arrival intensity. A mechanism to prevent departures from an empty state is also included in the last equation below, but not shown in the dynamic diagram.

$$\Delta QandS = \Delta t \cdot (In - Out)$$
$$In = \text{Po}[0.1 \cdot (1 + \sin(time / 50)) \cdot \Delta t] / \Delta t$$
$$Out = \text{Po}[0.12 \cdot \Delta t] / \Delta t \text{ if } Out \leq QandS \text{ else } Out = QandS$$

In Fig. 17, the M/M/1 system is shown.

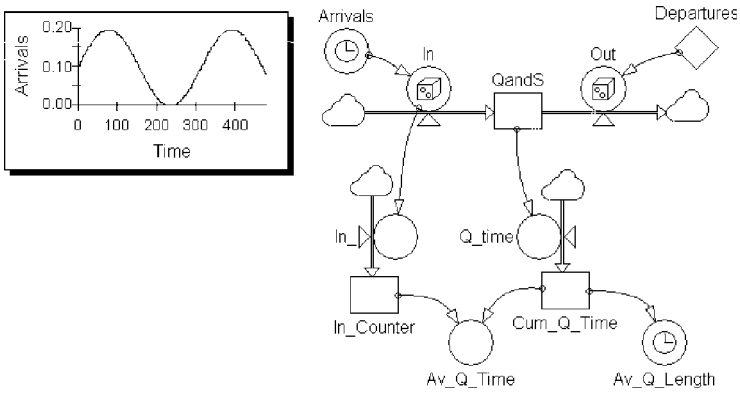

Figure 17. A M/M1 queue system when the arrival intensity rate varies sinusoidally, and some devices for calculation of numbers, queue-time and queue-length.

Below the queue model statistics are calculated. The flow of arrivals, *In*, is accumulated over time in *In_Counter*. The queuing customers (all but one in *QandS* who is currently being served) are also integrated over time to *Cum_Q_Time*. By dividing by the *In_Counter* and by time, the average time in queue, *Av_Q_Time*, and the average queue length, *Av_Q_Length*, are obtained. One realisation of these quantities as functions of simulation time is shown in Fig. 18.

To estimate various statistical quantities such as mean, standard deviation, confidence intervals or quintiles, the model is run a large number of times. For example, to find the 5 and 95 per cent quintiles of the average

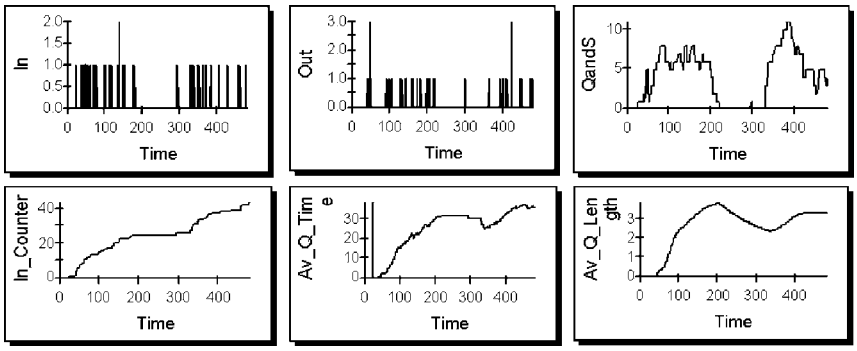

Figure 18. In, Out, QandS, IN_Counter, Av_Q_Time and *Av_Q_Length* from one simulation run where the intensity rate varied sinusoidally.

queue length at the end of a day, the model is run say one thousand times. Removing the simulations with the 50 smallest and the 50 largest end values gives the quintiles required. In Fig. 19 we have just done 50 simulations so as not to get a too blurry a figure. Of course we could use these simulation results to calculate mean, variance, confidence intervals or some other statistical measures.

Conclusions: Firstly, Poisson simulation can efficiently include queuing systems. Secondly, all kinds of statistics can be estimated from the results of a number of Poisson simulations. ■

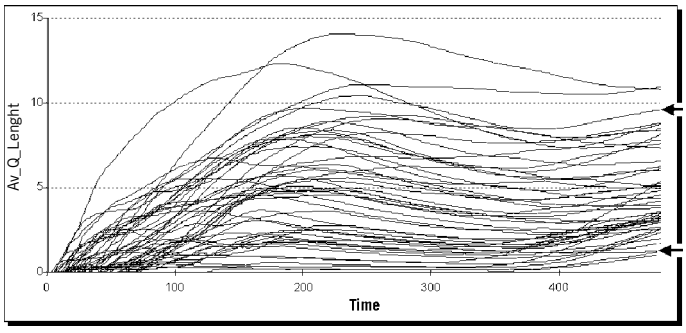

Figure 19. The average queue length during the day from 50 realizations of the queuing system. The 5 and 95 per cent quintiles of the queue length at the end of the day can be roughly estimated to be between 2 and 8.5, see arrows.

Example 10. *Estimating confidence intervals for model parameters*
As we have seen above, calculating variations or confidence limits for outcome quantities is straight forward – just make N runs and use appropriate statistical formulas. Calculation of confidence intervals for *model parameters* is a more complicated and time consuming task since these are *inputs* to the model. This requires a procedure in several steps. The ideas are well described in Press *et al.* (1989), Chapter 14.5 "Confidence Limits on Estimated Model Parameters". This description refers to Monte-Carlo models but is equally valid for Poisson simulation.

To demonstrate the idea, assume that a real dynamic and stochastic process like a radioactive decay described in Example 1 above is being observed. Using a first order deterministic model: $dx / dt = -b \cdot x$ and $x(0) = a$ gives $x(t) = a \cdot exp(-b \cdot t)$. Fitting this expression to observed data in e.g. a least-square sense gives estimates of the parameters a and b, say with values a_1 and b_1 respectively. Repeating the experiment on the real system and the parameter fitting procedure N times gives N estimates of the two parameters. From these N estimated values of the parameters a and b, statistical estimates of mean, variance or confidence interval for a and b can be obtained. Alternatively, they can be presented together as N pairs (a_i, b_i) in a two-dimensional plot where e.g. ellipses including a certain fraction of the pairs can be drawn.

Repeated experiments on the real system is, of course, the ideal procedure. In practice, however, an epidemiological study, an ecological experiment etc. cannot usually be repeated say 100 times, or we have just one unique observation. To get some kind of statistical estimates the second best is to *repeat the experiments on a model.* ("Resampling" from the model.) The quality of such estimates then depends on how realistic and accurate the model is. (Of course it would be better to have many real observations, because the single experiment might have been rather extreme, thus making the estimates of a and b extreme. If we have a few real experiments, we can also validate the model and see if the variations in and between experiments are about the same for system and model.) However this is the usual statistical reality – now we have to do the best with just a single real observation.

In the radioactive decay example, we know from physics that the decay is a Poisson process. Therefore, the deterministic model: $dx / dt = b \cdot x$ (with the estimated initial value $x(0) = a$) is transferred into a Poisson simulation model given as:

$$x = x + \Delta t \cdot (-F)$$
$$F = \mathrm{Po}(\Delta t \cdot b \cdot x) / \Delta t$$

Now, making N simulations with the Poisson model gives N realizations. The idea is that these N realizations can be treated as real experiments on the system. Then a new parameter estimation is performed for each of these realizations. Hereby, we get a set of N estimates (a_1, b_1), (a_2, b_2), … (a_N, b_N), on which appropriate statistics can be applied to calculate mean, variance or confidence limits of the parameters. Note also that even when the estimates are uncertain because of few experiments on the real system, the estimates of the variations can be good if the (dynamic and stochastic) model structure is realistic. ■

2.2. Other examples from the meso-scale world

The examples above are selected to demonstrate different points when dynamics and stochastics meet in the meso-scale world. For example in genetics or in demography, the development is often treated generation by generation in statistical terms. Using Poisson modeling, however, separate descriptions of male and female generation times as well as variations within each sex can be smoothly included.

Many multi-component systems are very reliable and do not fail until a number of components are broken. An example of this from the medical world is multi-hit models, i.e. models where the genome needs several hits in random order before a disease like cancer will develop. Poisson modeling is then a nice base for studying such stochastic developments within a dynamic frame. Since every combination of n out of N places may be hit, we have to make N^2 states in the model. This is easily handled with vector representation which is available in most modern simulation languages.

In biology, not only predation, but also competition, symbiosis and other relationships between species, can be modeled.

Infectious diseases may be treated. Classically there is a threshold limit of susceptible people that deterministically determines if an infected person triggers an epidemic. With Poisson simulation, stochastics affect the size of the epidemic and also if it occurs at all (the infected person may recover before infecting another person). Furthermore, diseases in animals or plants can be handled in the same way. Sexually transmitted diseases such as gonorrhoea or AIDS with different behaviours and properties among the sexes or between groups can also be handled in dynamic and stochastic terms.

In epidemiology and occupational health there is often a need to incorporate dynamics. Finding an optimal follow-up time is often crucial, because too short a follow-up gives bias and too long dilutes the results statistically.

The results of different follow-up times can be clearly displayed, even for complex models, using Poisson simulation techniques. When designing an experimental study, it is important to get sufficient numbers of participants, plants, or whatever to get a good statistical power in the analysis. In epidemiology for example, this can be achieved from tables in very simple situations where dynamics are not involved. With Poisson simulation this can be handled in a straight forward way, including any a priori information or assumption.

2.3. Generalizations

2.3.1. Not always a Poison distribution

As already stated above, not all distributions fulfil the Poisson assumptions. When for example the units in a state are momentarily removed, a pulse-function controls the outflow during an "infinitesimal" time, Δt. This may refer to some mechanical action, to the effects of a bomb or pollution etc. Assume for example that we are studying the effects of screening for a disease. A usual trick is then to study what happens over time *after screening*. This means that the individual time-scales are synchronized according to time of screening. Therefore, *in the model* screening occurs simultaneously for the persons at risk.

Now, say that the subjects having a symptom free pre-stage that should be found and treated are screened by a test with a sensitivity of 75 %. (On average you detect a pre-stage or disease with a probability of $S = 75$ %.) Thus the outcome of testing N subjects with the pre-stage, gives a binary outcome

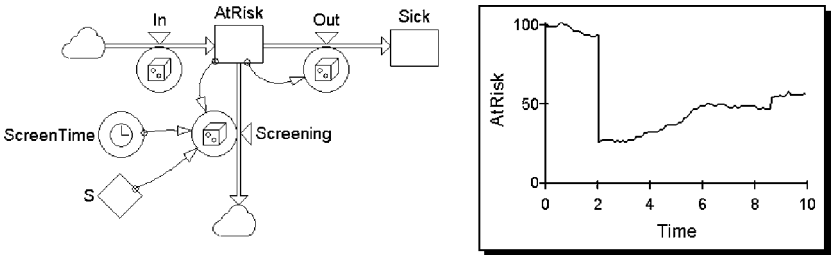

Figure 20. An oversimplified Poisson simulation model of a study of screening effects. The screening outflow is drawn from a binomial distribution. When screening with a test sensitivity of $S = 0.75$ takes place, the prevalence (*AtRisk*) is momentarily reduced by 75 %.

of positive ($p = S = 0.75$) or negative ($1-p$) tests. The number of positive tests will then be binomially distributed as Binormal (N, p). For large numbers (say $N > 10$) this can be approximated by the Normal distribution Normal ($N \cdot p$, Sqrt($N \cdot p(1-p)$)). This is shown in Fig. 20.

2.3.2. *Continuous variables*
One interesting question is whether Poisson simulation can be extended to continuous stochastic variables. In many cases we do not have a well defined number of microbes, but some fungus whose biomass grows continuously. Technically, it is not a problem to use a continuous statistical distribution to control an inflow or outflow, but two kinds of problem complicate the situation. The first is to find a good statistical distribution, and the second is to handle the distribution's parameters – often mean and variance. However, when studying continuous processes we leave the meso-scale in the meaning of moderate number of entities or events as used in this chapter.

3. Conclusions and discussion

3.1. Conclusions

This paper shows how Poisson simulation is used to extend Continuous System Simulation by modeling stochastic aspects of a system, as contrasted to just adding noise to input, states or output. This technique integrates an appropriate stochastic description into the dynamic structure of differential equations and thus preserves the close relationship between such equations and mathematical analysis and understanding.

As demonstrated in the examples above, this approach is powerful, versatile and flexible. It solves a number of problems on the meso-scale region where dynamics and stochastics are both important.

- We have demonstrated how different structural representations, giving the same dynamic behaviour, differ considerably when stochastics are added (Gompertz model in Example 2). This captures important information, helping to select a feasible structure or to disqualify infeasible options.
- Variance may be adjusted by balancing inflows and outflows in a model (Logistic model in Example 3).
- The interplay between dynamics and stochastics is demonstrated (e.g. the Lotka-Volterra model in Example 4).

- New qualities, not existing in pure dynamic models, can appear – such as extinction in the Lotka-Volterra model (Example 4).
- In experimental design (Example 5), where the statistical aspect is in focus, it is important that the dynamics are also handled properly. In this example the follow-up time is an important factor affecting the bias of statistical estimates. Poisson simulation also enables more powerful, unbiased analysis based on parameter estimation. A further advantage of this example is the possibility of treating the initial distribution over states realistically.
- A Poisson simulation model makes it possible to interpret data against a structural model in a more powerful way (Life tables in Example 6).
- A huge number of Markov simulations can be replaced by one Poisson simulation. (Ion channels in neurones in Example 7).
- Variable intensity over time, like changing inter-arrival or service times, is more easily dealt with than in DES and can therefore easily be incorporated if needed (Queuing model in Example 9).
- Statistics of different kinds can be easily obtained. After running the model a suitable number of times, the results of the simulations can then be treated statistically and presented in terms of mean value, variance, min, max, confidence interval, quintile etc. It is also possible to test hypotheses of the resulting quantity being below or above a certain limit. Examples of statistical treatment are Lanchester's Quadratic Law in Example 8 and the queuing model in Example 9.
- With the technique described in Example 10, *model parameters* can also be obtained in terms of statistical estimates. Used together with identification techniques, this creates an extremely powerful concept. This technique, suggested for Monte Carlo simulations, is described in Press *et al.*, 1989 (Chapter 14.5) but is equally applicable for Poisson simulation. In that book it is described as *"the only game in town"* and extremely powerful. *"In fact, the ability to do Monte Carlo simulations in this fashion has revolutionized many fields on modern experimental science."*
- Compared to methods based on statistics in for example demographic or genetic problems, the possibility of adding dynamics gives new opportunities. For example, letting the generation time vary or be different between males and females enables more realistic modelling.

Poisson simulation opens a new approach for modeling a variety of systems from e.g. biology, medicine, agriculture or society where dynamics and stochastics can be readily treated together in a powerful and flexible way. At the same time this type of model preserves its near connection to the mathematical and the statistical theory. This opens new opportunities to make more realistic and accurate models of complex systems without making the model technically complicated. It also creates new means to analyze stochastic data to draw conclusions about the real system.

3.2. Methods and tools

During the past year, the author has tested Poisson simulation in various programming tools such as Pascal, MATLAB and Powersim. Programming Poisson simulation models is straightforward for all these tools but depends somewhat on the prerequisites of the tool used.

Poisson-simulation is based on continuous system simulation models where stochastics are added. This means that methods, tools and results presentation to handle the dynamic aspects are already present, while the stochastic aspects in the form of multiple realizations and statistical measures and procedures have to be added.

A study of a dynamic and stochastic process is performed by a series of N simulations which display dynamic and stochastic variations. To draw significant conclusions various statistical measures have to be calculated. These measures are of two kinds emerging from variations *within* a realisation and those *between* the N realizations.

Mechanisms and estimates to be calculated *within* a simulation are constructed in the simulation model. This refers, for example, to the implementation of the stochastic flows of a given distribution and to estimates (mean, variance etc.) updated during the simulation. In a Poisson model (except for mechanisms for handling Poisson, normal, uniform and other distributions usually already implemented in the simulation language) counters, accumulators, collectors of statistics, calculators of least-square errors etc. may be needed. In the queuing model in Example 9 such mechanisms are included and are shown in Fig. 17.

Tools for controlling a series of realizations, handling various stochastic aspects and analysis of stochastic estimates *between* those realisation have to be added. This can be implemented within the simulation language. However, it is usually easier and more flexible to make a superior, controlling program that runs the model a specified number of times, collects the

results and calculates statistical estimates from the N simulations and presents them in tabular or graphical form. The statistical analysis is based on statistical formulas found in elementary books on statistics or in simulation books like Bratley *et al.* (1983) or Law and Kelton (1991).

Thus, the controlling program has to communicate with the simulation language where the Poisson model is written, so that dynamics and stochastics can be studied in one context.

Acknowledgement

The author wishes to acknowledge Kjell Pernestål and Mikael Sternad for valuable discussions and suggestions. A project on Poisson simulation was supported through a grant from the former Swedish Council for Planning and Coordination of Research.

References

Åström K.J. (1970): *Introduction to Stochastic Control Theory*. New York, NY: Academic Press.

Bratley P., Fox B.L. & Schrage L.E. (1983): *A Guide to Simulation*. New York, NY: Springer-Verlag.

Braun M. (1993): *Differential Equations and heir Applications*. New York, NY: Springer-Verlag.

Gustafsson L. (2000): "Poisson Simulation – A Method for Generating Stochastic Variations in Continuous System Simulation". *Simulation* **74:5**, 264–274, May 2000.

Lanchester F. (1916): *Aircraft in Warfare, the Dawn of the Fourth Arm*. London: Constable.

Law A.M. & Kelton W.D. (1982): *Simulation Modeling and Analysis*. New York, NY: McGraw-Hill Inc. (1991).

Luenberger D.G. (1979): *Introduction to Dynamic Systems. Theory, Models and Applications*. New York: John Wiley & Sons, Inc.

MM High Performance Systems Inc. Getting Started with STELLA v. 6.0. (Can be downloaded from the net together with a demo version of STELLA from <www.hps-inc.com>.)

Newman J.R. (1956): *SIGMA – The World of Mathematics. Part XX*. New York, NY: Simon & Schuster.

Powersim Corporation. *Powersim Reference Manual, Powersim Press1996*. Powersim Corporation, 1175 Herndon Parkway, suite 600, Herndon, VA 22170, USA. (A free demo to test the examples in this chapter is found at: <www.powersim.com>.)

Press W.H., Flannery B.P., Teukolsky S.A. & Vetterling W.T. (1989): *Numerical Recipes in Pascal – The Art of Scientific Computing*. Cambridge, UK: Cambridge University Press. (The book is also available in FORTRAN or C.)

Saaty T.I. (1989): *Mathematical Models of Arms Control and Disarmament*. New York, NY: Wiley.

Volterra V. (1926): "Variazioni e fluttuazioni del numero d'individui in specie animali conviventi." *Memoire della R. Accademia Nazionale dei Lincei,anno CCCCXXIII*, II. pp 1–110.

Waggoner P.E. (1986): "Progress curves and foliar diseases: Their interpretation and use". In: Leonard K.J. & Fry W.E. (eds): *Plant Disease Epidemiology – Population Dynamics and Management, Volume 1*. New York, NY: Macmillan Publishing Company.

Appendix: The Dynamic Diagram

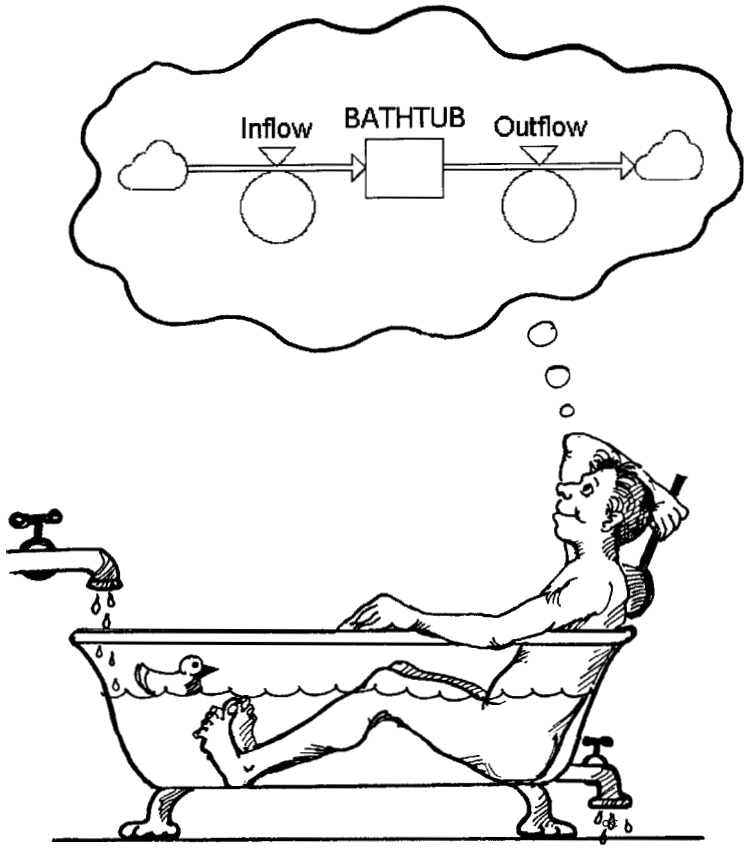

Figure A1. The bathtub analogy. (Drawing by Liss Timner).

The dynamic diagram is an alternative way to present a system of differential and algebraic equations. It is based on a bathtub metaphor, see Fig. A1. It has significant pedagogic merits why it is often used by biologists, ecologists, medical people, economists and others where the mathematical background is not as large as for example that of technicians or physicists. A dynamic diagram is constructed with the following components:

- A **state** is represented by the "bath tub" symbol () which holds the amount or number of objects.
- The content in the state changes because of "physical" **inflows** and **outflows** ().
- Other symbols are **constants** () and **auxiliary quantities** for algebraic computations ().
- To show the limits of the model we let the flows origin from or end in **cloud** symbols (). (We are not interested in where rabbits go after death!)
- Everything (except physical flows) affecting a quantity is displayed by single **arrows** () from the affecting quantity to the affected one.

To stress that e.g. a flow is random or time dependant the flow symbol may include a die or a clock:

- A **die** () in a flow (or auxiliary) symbol denotes that the value of the flow rate (for each time step) is drawn from a probability distribution.
- A **clock** () in a symbol denotes that it is a time function.

In simulation languages like Powersim (Powersim Corporation, 1996) or STELLA (MM High Performance Systems, Inc.) the model is constructed by placing suitable symbols (icons) on the screen. The exact mathematical expression and initial values for the states are then defined after double clicking the quantity (but are not seen in the diagram). Then the model can be simulated by calculating time-step (Δt) by time-step. A free copy of a Powersim demo, enough to test Poisson simulation, can be downloaded from <www.Powersim.com>. (Note however that the time-step is denoted *"TIME-STEP"* in Powersim.)

Example A1. *A time continuous process*
Suppose we want to construct a simple model with one state, one inflow and one outflow. This can be written as:

$$dx/dt = F_1(x,t) - F_2(x,t)$$
$$x(0) = x0$$
$$F_1 = Some\_expression$$
$$F_2 = c^*x$$

This process may be *numerically* described as (Euler):

$\Delta x = \Delta t \cdot (F_1 - F_2)$
$x(0) = x0$
$F_1 = Some\_expression$ [Defined in the *Auxiliary* below]
$F_2 = c \cdot x$ [c is defined in the *Constant* below]

Such a model is composed of the components treated above and may look like that in Fig. A2. ■

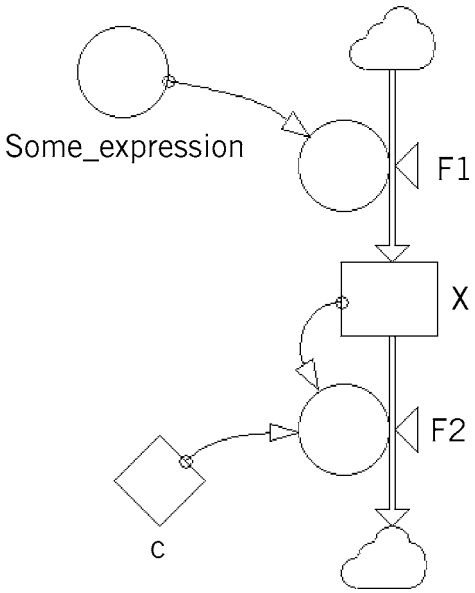

Figure A2. A dynamic diagram as it looks in the CSS language Powersim.