Mathematical Biosciences 225 (2010) 94–107

Contents lists available at ScienceDirect

Mathematical Biosciences

journal homepage: www.elsevier.com/locate/mbs

Consistent micro, macro and state-based population modelling

Leif Gustafsson *, Mikael Sternad

Signals and Systems, Dept. of Engineering and Sciences, Uppsala University, P.O. Box 534, S-751 21 Uppsala, Sweden

ARTICLE INFO

Article history:
Received 21 August 2009
Received in revised form 25 January 2010
Accepted 13 February 2010
Available online 19 February 2010

Keywords:
Model comparison
Poisson Simulation
Population model
Simulation methodology
Stochastic integration
Stochastic time handling

ABSTRACT

A population system can be modelled using a micro model focusing on the individual entities, a macro model where the entities are aggregated into compartments, or a state-based model where each possible discrete state in which the system can exist is represented. However, the concepts, building blocks, procedural mechanisms and the time handling for these approaches are very different. For the results and conclusions from studies based on micro, macro and state-based models to be *consistent* (contradiction-free), a number of modelling issues must be understood and appropriate modelling procedures be applied. This paper presents a uniform approach to micro, macro and state-based population modelling so that these different types of models produce consistent results and conclusions. In particular, we demonstrate the procedures (*distribution*, *attribute* and *combinatorial expansions*) necessary to keep these three types of models consistent. We also show that the different time handling methods usually used in micro, macro and state-based models can be regarded as different integration methods that can be applied to any of these modelling categories. The result is free choice in selecting the modelling approach and the time handling method most appropriate for the study without distorting the results and conclusions.

1. Introduction

This paper focuses on *population models* – which are defined as models of dynamic systems with an integer number of discrete entities (individuals) such as plants, animals, patients, vehicles, molecules, atoms, data packages or entities of any kind. Such models are frequently used in ecology, epidemiology, demography and queuing systems, and are also important in physics, chemistry, biology, traffic planning, production and many other fields.

The crucial task in modelling is to preserve the characteristics of interest of the system under study. Four fundamental properties of a population system under study are of special interest for preservation in the model:

- The integer non-negative quality of the entities in the population.
- The continuous nature of time, which should at least be sufficiently well approximated in the model.
- The structural and temporal relations creating the dynamics of the system.
- Important irregularly occurring events of the system. These have to be characterised by an appropriate probabilistic representation in the model, because they cannot be described in detail.

This list could be extended with spatial aspects – but we restrict this paper to non-spatial models.

A system of interacting entities, a *population system*, can be modelled in three different but related ways: by a *micro approach* where each entity is separately described, by a *macro approach* where similar entities are lumped into compartments so only the number of entities in each compartment is recorded, or by a *state-based approach* where each discrete state in which the system can exist is explicitly represented and every transition between these states is specified by a conditional probability. The behaviour of micro and macro models is typically calculated by a numerical method (simulation), while state-based models, if sufficiently simple, can also be analysed analytically.

In various applications it is often found that e.g. a micro model produces results that are inconsistent with those from a macro model based on stochastic differential (or difference) equations. Furthermore, an analytic approach based on a state-based model may produce still other results. A number of studies compare the results from different approaches of modelling the same system and discuss the pros and cons of these approaches [1–6]. However, a vast number of studies in which the results and conclusions are dependent on the modelling approach lack such a discussion. Little has been done to explain how different approaches are related and what is needed for different approaches to produce consistent results.

The purpose of this paper is to provide a base for *consistent* (contradiction-free) micro, macro and state-based population

* Corresponding author. Tel.: +46 18 500061.
E-mail address: Leif.Gustafsson@bt.slu.se (L. Gustafsson).

0025-5564/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.mbs.2010.02.003

modelling and simulation. Two models will here be said to be consistent if their outputs in terms of probability distribution/density functions are contradiction-free for relevant output quantities. This means that we allow for comparison between a micro model based on identifiable individuals that can be studied individually and macro or state-based models that only produce aggregated results. In particular, we will demonstrate the procedures (*distribution*, *attribute* and *combinatorial expansions*) necessary to keep these three types of models consistent. We will also show that the different time handling methods usually used in micro, macro and state-based models can be regarded as different integration methods that can be applied to any of these modelling categories.

An advantage of having consistency between micro, macro and state-based population modelling is that it allows for *an appropriate choice of type of model* instead of selecting it by routine. There are many aspects to this choice such as: nature of the system, purpose of the study, size of the model, execution time in computer simulation, possibility of including both discrete and continuous quantities, simplicity, transparency and communicability of the model, parameterisation and possibility of estimating parameters, validation, and possibility of simplifying the model without distorting the results and conclusions. Some of these issues are discussed later on. Furthermore, consistency between the three model types provides a powerful context where different approaches can contribute different types of insights to a study (see Fig. 1).

The possibility of combining theory and practice in a consistent way for micro and macro population modelling and simulation and having access to important results from the theory of stochastic processes are major advantages. In some cases deterministic models, embedded in the stochastic model, can be of value for e.g. mathematical analysis, model fitting, optimisation and sensitivity analysis – see [7].

To avoid making the presentation longer and more detailed than necessary, no distinction is made between the *continuous time* of the system under study and the *almost continuous time* using sufficiently small time-steps in the numerical model. Therefore, the notation $x(t)$ is usually used rather than x_t. So, for the sake of simplicity, this paper refers to exponential distributions even in the case of models with almost continuous time. To avoid the linguistic similarity between 'state variable' in a macro model and 'state' in a state-based model, the term *compartment* is in this paper used instead of state variable.

This paper is organised in the following way. In Section 2, micro, macro and state-based modelling are presented, while in Section 3 three possible time handling principles are introduced and consistency is discussed. In Section 4 we consider the merits and demerits of the three approaches. Finally, in Section 5 the findings are discussed in a broader perspective.

2. Micro, macro and state-based modelling

2.1. Introduction

A system under study is in general composed of so many pieces (e.g. atoms), has so many characteristics and is so complicated that it can never be modelled in all its details. The very essence of modelling is to build a parallel description, called the *model*, which is much simpler but otherwise preserves important characteristics and mechanisms of the system under study.

For population systems in general, the nature of the system under study includes discrete entities that interact irregularly in a dynamic context over a continuous time. This nature has to be preserved in the model unless it can be shown that further simplifications can perform the task in accordance with the overall aim.

Before scrutinising the different approaches it is necessary to use distinct terminology for the entities, their characteristics and changes over time. In the literature there are many synonyms and homonyms, but we use the following:

The system under study consists of *entities* of the same or different kinds. These entities may have a number of *characteristics* that are permanent or can be changed. The entities may interact with other entities and with the environment so that *changes* occur, or they may remain in a situation for a longer or shorter time.

A conceptual model is a *specification* of what to include or exclude from the system under study in accordance with the purpose of the study and practical considerations. This is a complete specification, but the conceptual model cannot be executed. The task is then to map the properties of the conceptual model into an executable micro, macro or state-based model.

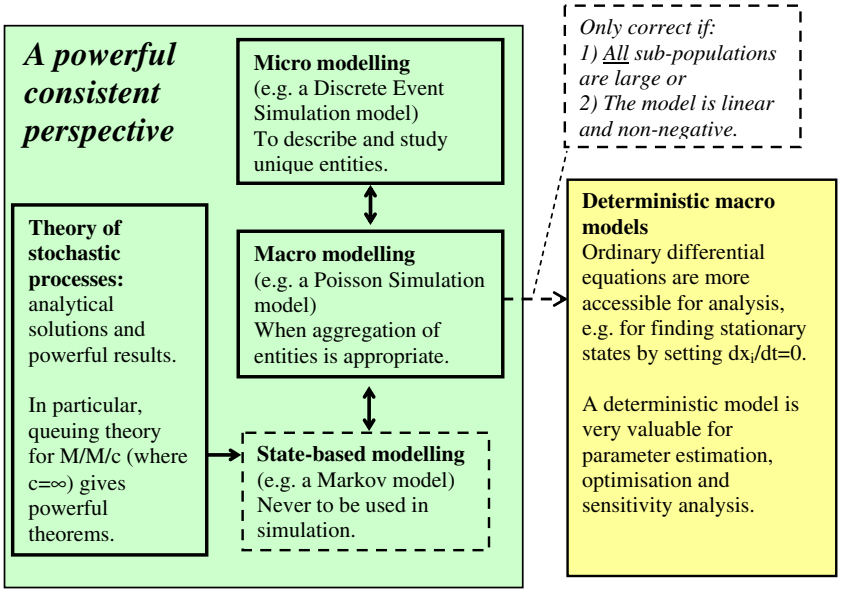

Fig. 1. Consistency brings a situation where theory and different simulation approaches are at hand for a powerful investigation of the system under study. It also shows when simplifications can be made. However, the simplification to a deterministic model is beyond the scope of this paper.

A micro model is based on the main concepts of *entities* and *events*. Each entity here becomes an individual unit with its *attributes* and *behavioural procedures*. Entities can compete for resources of different kinds, or wait in a 'queue' for a resource. They can also cooperate with each other, etc. A change is called an *event* and affects at least one of the entities. An event occurs at a point in time called the *event time*. The entity may remain in a certain condition (*stage*) for a certain period of time. This time period is called the *sojourn time* and may be stochastic according to any specified distribution.[1]

A macro model is based on considerably fewer concepts than a micro model. It is composed mainly of *compartments* (state variables) holding the entities, and *flows* into, out of or between compartments. Entities with the same characteristics are here lumped together and the model describes changes in the compartments caused by the flows. The individual entity can no longer be distinguished. Only the number of entities in a compartment is recorded.

Finally, a *state-based model* is composed of *states* and *transitions* between these states. This type of model is closely related to the macro model. The main difference is that every combination of values (number of entities) that the compartments of a macro model can take has to be explicitly represented as a separate state in the state-based model. This usually means that the state-based model is a huge disaggregation of the macro model.

Below, the micro model is compared with the macro model and the state-based model. Section 2.2 discusses the mapping of a conceptual model of the system under study into a micro model. In Section 2.3, the micro model is transformed into a macro model and in Section 2.4 the macro model is transformed into a state-based model. The reason for these successive transformations is to focus on what is necessary in order to keep the consistency, to simplify the presentation and to avoid repetition.

2.2. Micro modelling

Micro modelling is the approach that is by far the closest to a direct mapping of the original problem. The individual entities with their attributes and behaviours are directly mapped in a straightforward manner from the conceptual model of the system under study to the model in almost a one-to-one way. These entities might represent plants and animals in ecosystems, vehicles in traffic, people in crowds, patients with a disease, data packets in a network, genes in a population, etc. A frequently used metaphor for micro simulation is a theatrical performance on a stage. Actors arrive on this stage, act together with other actors, wait for and use resources in the form of props, change their characteristics, leave the stage, etc.

There are many settings and different terminologies of the micro approach, e.g. Micro-simulation, Discrete Event Simulation, Individual-based Modelling, Agent-based Modelling, Multi-agent Based Modelling, etc. The expression micro modelling is used in this paper.

Modelling individual entities with their interactions, queuing for resources, competing, co-operating, etc. first became possible with the advent of digital computers and has mainly been developed along two lines: The first line includes passive entities with attributes that travel in a network of stations where they may have to wait for resources. Examples of this type are GPSS [8], SIMSCRIPT [9], SIMAN [10] and Arena [11] to mention but a few. An alternative line started with the object-orientated programming language Simula with its class Simulation [12,13], and later on the more user-friendly class DEMOS [14]. Here the entity is an *object* that has both attributes and internal procedures controlling its behaviour. The concept *agent* is an extension of the object concept.

Micro modelling naturally includes the four fundamental properties presented in Section 1. The entities are discrete and dynamics occur when the entities interact in continuous (or almost continuous) time. Probabilistic descriptions of the properties are implemented in a straightforward way by including appropriate statistical distributions from which random numbers are drawn. In this way it is possible to describe, e.g. the actor's chance of success in a situation, its choice between options, or the sojourn time during which it will remain in a certain condition. In particular, the possibility of directly using any statistical distribution (possibly empirical) for the sojourn time in a stage is a powerful asset in micro modelling.

The micro approach is versatile since sample paths and statistical estimates of unique individuals as well as aggregated estimates and distributions can be obtained. In micro modelling, variations among individuals, local interactions, complete life cycles, utilisation of resources, how properties emerge from the behaviour of individuals and how the system affects the individuals of a population can be studied. Therefore, a number of counters, clocks, tallies, etc. are included to record what happens, The result is usually presented as statistics in the form of mean, standard variation, min, max, confidence interval, histogram, etc. of performance, numbers, transit times, utilisation of resources, etc.

2.3. Macro modelling

A macro model is composed mainly of *compartments* (state variables) holding all entities in the same condition, and *flows* into, out of or between compartments. The structure of a compartmental stochastic population model is the same as for a deterministic compartment model, but the content of a compartment is an integer number and the flows are stochastically controlled. Examples of such models are Poisson Simulation models [15] and compartmental models controlled by the Stochastic Simulation Algorithm [16,17].

2.3.1. Distribution and attribute expansions

The aggregation of entities into compartments has two consequences:

1. The individual entities cannot be observed and they have *no personal attributes*. Instead, each compartment is defined by a *set* of attribute values (e.g. female and elderly) that its entities must possess. Therefore, the value (content) of the compartment represents only the total number of entities with the specific set of attribute values. The number of compartments has to be large if many attributes are to be included (*attribute expansion*).
2. A compartment has no counterpart to *the general sojourn time distribution* of a stage in a micro model. In the case of an outflow equation that is linear in the state and time-invariant, the compartment is bound to produce an exponential sojourn time. A desired sojourn time distribution, therefore, has to be generated by a structure of compartments in series, parallel and/or feedback (*distribution expansion*).

2.3.2. Modelling aspects

Since only the numbers of entities in the compartments are recorded, the data needed for a macro model and the information from such a model are usually considerably smaller than for a

[1] *Sojourn*, *residence* or *dwell time* are often used as synonyms. In most instances in this paper this is the case as well. However, in queuing theory the waiting time before service and the time in service (residence time) has to be separated. The sojourn time then stands for the sum of the waiting and residence times. We follow this latter convention. By using the term *sojourn time* we allow for – but do not require – 'waiting before service'.

corresponding micro model. This also means that a macro model does not answer micro questions about individual entities.

The macro model is well suited for a comprehensive graphical representation of compartments and flows. For example a model with three successive compartments and two flows can be represented in a Forrester diagram [18] as shown in Fig. 2. The flows in this figure are in general dynamically dependent and time-variable functions. For a deterministic model, the flow intensities could be expressed as $F1 = \lambda(x, y, z, t)$ and $F2 = \mu(x, y, z, t)$. During a sufficiently short time interval, the flows can be regarded as constant. In the stochastic case a probabilistic mechanism is needed for generating random and integer numbers of entities to the flows for each time-step. Such a probabilistic mechanism is an integral part of time handling and is discussed in Section 3.

2.4. State-based modelling

State-based population models are based on the two concepts: *state* and *transition*. Such models can be of many different kinds, e.g. Markov chain models, continuous-time Markov models, semi-Markov models, generalised semi-Markov process models, chain-binomial models, etc. [19–22].

The theory of state-based models provides a powerful analytical device with a number of useful results. However, from a *practical aspect*, construction and simulation using a state-based model is inferior to using a macro model [23].

2.4.1. Theoretical value of state-based models

The analytical power of, e.g. a Markov chain model originates from the fact that matrix calculations operate on the complete probability distribution function (pdf) of the state vector. This contrasts with a simulation of a stochastic model. Analytically, the transition matrix $\boldsymbol{P}$ is used to update the initial distribution row vector $\boldsymbol{p}^{(0)} = \left(p_1^{(0)}, p_2^{(0)}, \ldots\right)$ that represents the probabilities $p_j^{(0)}$ of being in state j at time t=0, so that $\boldsymbol{p}^{(1)}=\boldsymbol{p}^{(0)}\boldsymbol{P}$. The pdf of the state distribution at integer time index n is then obtained according to $\boldsymbol{p}^{(n)} =\boldsymbol{p}^{(0)}\boldsymbol{P}^n$.

A number of issues can be analytically examined within Markov theory. For example if the process has a stationary distribution $\boldsymbol{\pi}$, it is obtained by solving the eigenvector equation: $\boldsymbol{\pi} = \boldsymbol{\pi}\boldsymbol{P}$.

Another example is from the field of queuing theory. Here powerful results for $M/M/c$ queuing systems (where the M/M stands for exponential inter-arrival-time distribution and exponential service–time distribution and c represents the number of existing parallel service channels) are directly applicable. In particular, for $c = \infty$ there is no waiting for service [22].

Furthermore, Burke's theorem and feedforward Jackson networks in queuing theory can be used to reveal when flows are independent and Poisson-distributed [24,25] for a stationary situation. If so we have a network of $M/M/c$ nodes. The content of entities in an $M/M/c$ node when entities that cannot be served directly are rejected is given by the Erlang's loss formula [26]. For the case $c = \infty$ there are no rejections or waiting for service and the content of entities in the node is Poisson distributed in the stationary case.

In particular, the stationary results of an $M/M/\infty$ node are also valid for a compartment in a linear macro model in e.g. Poisson Simulation [27]. They can therefore be used for assigning initial values to corresponding compartmental or state-based population models starting from a stationary situation [27,28].

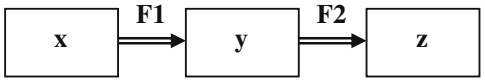

Fig. 2. Backbone of a Forrester diagram of three successive compartments, x, y and z, and two flows, $F1$ and $F2$. (In a complete diagram, single line arrows are used to indicate the quantities that will affect each flow.)

2.4.2. Distribution and attribute expansions

When constructing a state-based model, it is necessary to include the *distribution* and *attribute expansions* previously discussed in Section 2.3.1.[2] A practical way to accomplish this is to start with a compartmental model where these expansions are already implemented.

2.4.3. Combinatorial expansion

Combinatorial expansion is the process of constructing a state-space, starting from a given macro model. The k compartments of a macro model correspond to k dimensions of a state-space and the prescribed number n of entities bounds the size of the state-space. The state-space then represents all possible combinations of n entities distributed over k compartments. Each state of a state-based model then corresponds to a k-tuple $(n_1, n_2, \ldots, n_k)$, where the integers $n_1, n_2, \ldots, n_k$ add up to n.

For example, consider the model with three compartments x, y and z as shown in Fig. 2. In the special case when there is *only one entity* in the model, it is located in compartment x, y or z. The possible states are then $(1, 0, 0)$, $(0, 1, 0)$ and $(0, 0, 1)$ and the number of states equals the number of compartments in a corresponding macro model. With 10 entities distributed over the three-compartment model there will be 66 states in a state-based model. In the general case with n entities in k compartments, there will be $\binom{n+k-1}{k-1}$ states, which is a huge number unless n or k is small.

For all but the smallest systems, it is a formidable task to explicitly represent each state of the state-space in a state-based model. Furthermore, in a growth model the modeller cannot say how many entities there will be during a simulation, so provision has to be made for the worst case. In addition, *the probability of every possible transition* has to be quantified in the transition matrix $\boldsymbol{P}$ with $\binom{n+k-1}{k-1} \times \binom{n+k-1}{k-1}$ elements, which in the best case is sparse. Because of complexity, the values of the transition probabilities are difficult to analytically deduce, nor is it easy to fit a large number of transition elements to data from the system under study. To make the modelling problem still worse, the values of the transition probabilities are generally functions of the time-step used. There are further disadvantages of using a state-based model instead of a stochastic macro population model, such as the difficulty in modifying or extending the state-based model, the intuitive distance between the transition matrix and the model structure, the lack of a smooth way to adjust the implicit time-step used, etc. See [23].

3. Stochastic time handling principles

The non-executable conceptual model should define the appropriate probability distribution/density functions that describe the probability of any relevant event under given conditions. To obtain an executable simulation model, there is a need for a mechanism that generates time evolution that is in accordance with the assumptions in the conceptual model.

[2] In practice, state-based models are often heavily simplified in order to keep the model within a reasonable size. For example the sojourn time of a stage in the real system is often erroneously represented by the equivalence of a single compartment and thus an exponential sojourn time distribution (in the linear case) is implicitly assumed. In addition, different characteristics (attributes) of e.g. different age groups are often ignored.

Such a mechanism is obtained by regarding the simulation model as a generator of interacting stochastic processes.[3] Events are then generated randomly, with an intensity λ that may be constant, time-varying or dynamically dependent. Repeated replications of the stochastic processes should give different event sequences, but the statistics of these multiple replications should be in accordance with the pdfs specified by the conceptual model.

The fundamental issue for updating (simulating) a population model over time is to handle the sequence of events that is generated by the dynamic model structure and/or originates from external influence. This updating can be performed in different ways, that can be thought of as different integration methods of stochastic models.

3.1. Stationary and non-stationary Poisson processes

To understand how stochastic micro, macro and state-based population models are related, it is necessary to understand how time can be handled when event times may not be predetermined, but must occur randomly with a given intensity. The natural starting point and mathematical foundation of this presentation is the stationary Poisson process.

Definition 1. A *stationary Poisson process* is defined in the following way:

Let $X(t)$, $t \geqslant 0$ be the number of times an event occurs in the time interval $(0,t)$. If the stochastic process $\{X(t), t \geqslant 0\}$ has the properties:

(1) The process has independent increments.
(2) $P[\text{an event occurs exactly once in the interval } (t,t+h)] = \lambda \cdot h + o(h)$.
(3) $P[\text{an event occurs more than once in the interval } (t,t+h)] = o(h)$.

then the process is a stationary Poisson process with intensity λ.

A *stationary* Poisson process (constant intensity λ) has a number of useful properties [29]:

- The number of events during a time interval (t_1,t_2) is *Poisson-distributed*, i.e. $X(t_1,t_2) \in Po(\lambda \cdot (t_2 - t_1))$. In particular, for a time-step of length Δt we have $X(t,t+\Delta t) \in Po(\lambda \cdot \Delta t)$.
- If $X \in Po(m)$ then $E[X] = m$ and $Var[X] = m$.
- If $X \in Po(m_1)$ and $Y \in Po(m_2)$, where X and Y are independent, then $X + Y \in Po(m_1 + m_2)$. Therefore, independent Poisson processes can be superimposed or subdivided.
- If $X \in Po(m_1)$ and $Y \in Po(m_2)$, where X and Y are independent processes, then the next event originates from the X process with probability $m_1/(m_1 + m_2)$ and from the Y process with the probability $m_2/(m_1 + m_2)$.
- For a Poisson process with intensity λ the time between consecutive events is exponentially distributed with expected value $1/\lambda$.

In a dynamic relationship, the intensity is in general a function $\lambda(\boldsymbol{x})$, where $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)^T$ is a vector of entities/compartments/states affecting the intensity. The intensity may also vary in time because of factors external to the model as $\lambda = \lambda(t)$. So in the general case the intensity is given by $\lambda(\boldsymbol{x}(t), t)$.

In this paper a pragmatic approach, relying on numerical methods, is employed. The idea is to use the stationary Poisson process as an approximation during appropriately short time intervals, even when the intensity λ changes because dynamics or time are involved. The powerful properties of the stationary Poisson process can thereby be maintained as good approximations within each updating time-step.

3.2. Principles for generating and handling irregularly occurring events

Four different ways of generating time sequences of irregularly occurring events with a stationary intensity of λ events per time unit are illustrated in Fig. 3. They provide a key for understanding our subsequent discussion of consistency between different modelling approaches and are further discussed below. Of these, methods (i), (ii) and (iii) can be used in a dynamic non-stationary case by regarding the process as stepwise stationary.

- *Uniform*: A simple way to obtain randomly distributed events over the time interval (t_1,t_2) is to draw a number of $N = INT[\lambda \cdot (t_2 - t_1)]$ *uniformly distributed* random numbers between t_1 and t_2. (To also get the *number* of events random one can e.g. use a longer time interval and keep a midsection of length $t_2 - t_1$.) This method is included only for pedagogic reasons. It is never used in dynamic simulation because the locations of the events in a simulation depend on the model evolution and cannot be known in advance.

(i) *Bernoulli*: According to the definition of the stationary Poisson process, the probability of an event occurring during a very short time interval is proportional to the length of the interval. When the time increment is a very small time-step (h), time is updated as: $time := time + h$. A sequence of events can then be located by a sequence of points in time where single events that may happen are handled at the time increments.

In a numerical context *the Bernoulli distribution* with probability $\lambda \cdot h$ is used to decide whether an event will happen during $(t,t+h)$. For example, by introducing a stochastic variable X that accumulates the number of events, the second condition in the Poisson process definition can be reformulated as $P[X(t+h) = i + 1 \,|\, X(t) = i] = \lambda \cdot h$. This displays a similarity to the likewise memoryless Markov chain of a birth process. This method of time handling also works for micro and macro modelling.

However, in a numerical context the Bernoulli method is a very inefficient way of handling time, because if several events occur during $(t,t+h)$ the model only captures one of them. Therefore, h has to be so short that it guarantees that more than one event almost never occurs during an interval $(t,t+h)$, which can be expressed as $\lambda \cdot h \ll 1$. This very short time-step implies that the vast majority of intervals will be empty.

(ii) *Exponential*: The temporal distances between successive random events of a stationary Poisson process are distributed according to an *exponential* probability density function given by $\lambda e^{-\lambda t}$ where $t \geqslant 0$. By drawing a number from an exponential random number generator $Expo[1/\lambda]$ with this probability density function, the next event is scheduled at $time := time + Expo[1/\lambda]$.[4]

This method is often used in micro simulation. For example, in Discrete Event Simulation it is the standard method, where e.g. an actor schedules his successor [11,13,30]. For macro models this was the idea behind the first stochastic integration algorithm, presented by Daniel Gillespie in 1976 [16]. The time interval to the next event, $Expo[1/\lambda]$, may be long. This generates no problem from a dynamic point of view since nothing is dynamically changed until

[3] A stochastic process is a sequence of stochastic variables, separated by an index t and defined on a sample space Ω. The process is denoted $\{X(t,\omega), t \in T, \omega \in \Omega\}$. If t is fixed one obtains a stochastic variable and if ω is fixed a realisation/replication.

[4] Technically, this can be realised by drawing a uniformly distributed random number (u) between 0 and 1, and applying the inverse transform method [30]. The inter-event time is then obtained from: $Expo[1/\lambda] = -ln[u]/\lambda$ [30,31].

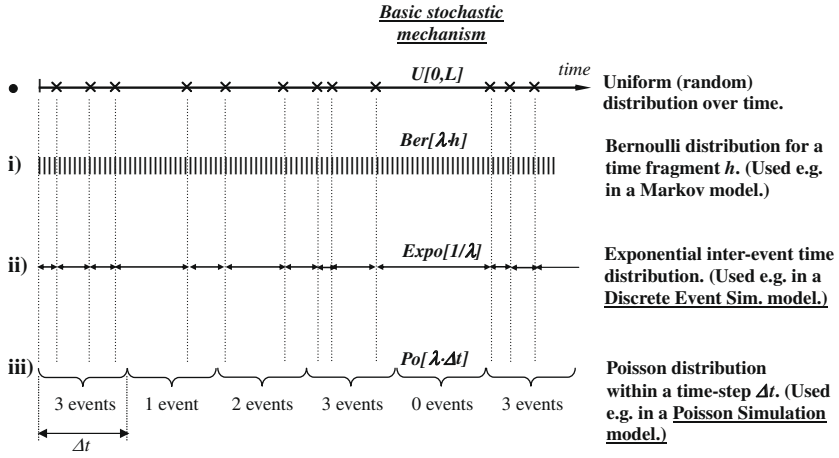

Fig. 3. A stationary Poisson process with intensity λ can be generated in different ways. Events are marked by '×' on the time axis. Note that the stochastic mechanisms are calls to random number generators with the argument given in the brackets. In particular, *Expo[mean]* (where $mean = 1/\lambda$) calls a random number generator with an exponential probability density function given by: $\lambda e^{-\lambda t}$.

the next event occurs. For models where the external conditions do not change, this is an exact integration method – although slow when there are many events during a simulation. Only if the external conditions have changed $\lambda(.,t)$ considerably, a problem arises, but technical solutions for such problems exist.[5] For state-based models this method of time-steps of irregular size gives a simpler model structure and significantly speeds up the simulation compared with using a short fixed time-step h. Another great advantage is that there is no time-step to adjust. However, it requires the model to be defined for continuous time rather than for discrete, equally spaced points in time [20].

(iii) *Poisson*: An efficient time-driven schedule is $time := time + \Delta t$, where Δt may be so large that many events can occur during a time-step. However, the step Δt must be constrained to be sufficiently shorter than the shortest dynamic time-constant of the system. Otherwise the dynamics of the model, which influence the number of events during the time-step, are too much changed during the time-step.

When a set of events can be described as mutually independent within a time interval Δt and occur with intensity λ, then the number of events during Δt will be *Poisson distributed* $Po[\Delta t \cdot \lambda]$. Random number generators that generate Poisson distributed numbers of events can then be used for relatively large time-steps Δt, if the intensity λ can be approximated as constant within the time-step.

The underlying idea for this time-handling method is the same as when solving a *deterministic* differential equation model using e.g. Euler difference equations:

$$x_i(t+\Delta t) = x_i(t) + \Delta t \cdot \lambda_i(\boldsymbol{x}(t),t) - \Delta t \cdot \mu_i(\boldsymbol{x}(t),t) \quad \text{for } i = 1..n,$$

where $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)^T$. Then the equations can be stepwise calculated by assuming the input intensity $\lambda_i(\boldsymbol{x}(t),t)$ and the output intensity $\mu_i(\boldsymbol{x}(t),t)$ fixed during the time interval $(t, t+\Delta t)$. In the *stochastic* case the Euler difference equations take the form

$$x_i(t+\Delta t) = x_i(t) + Po[\Delta t \cdot \lambda_i(\boldsymbol{x}(t),t)] - Po[\Delta t \cdot \mu_i(\boldsymbol{x}(t),t)],$$

where the values of the $Po[]$ terms are obtained from a random number generator [15,33].

For the sake of clarity and simplicity, the Euler integration scheme is used throughout this paper. Other single-step integration methods, such as the Runge–Kutta methods, are also possible. A Runge–Kutta algorithm is then used to estimate the argument of the stochastic function. Thereafter, a Poisson-distributed random number is drawn using the estimated argument.

As just demonstrated, compartmental macro modelling can use this scheme smoothly. Then any number of events in each of any number of flows can be handled during a time-step [15]. It can also be used in micro modelling and, in principle, also in state-based modelling [23].

3.3. Implementation of time handling in micro, macro and state-based models

3.3.1. Time handling in micro models

For micro models the exponential form of time handling is usually the most efficient. The simulation process dynamically generates new events that are sorted in an event list. The execution proceeds by jumping to the point in time for the *next event* and the consequences of the event are then executed. [11,13,34]

Time control by $time := time + h$, where h is very small, is possible, but would be very inefficient. A Poisson approach using $time := time + \Delta t$ is also possible, but nothing would be gained compared with the exponential approach, since every entity concerned would still have to be individually updated.

3.3.2. Time handling in macro models

In order to illustrate the consequences of different time handling approaches in a macro model, the structure in Fig. 2 and an example of bacterial growth are used.

3.3.2.1. Bernoulli time handling in a macro model.

Example 1 (*a three-compartment model*). Assume events to be handled by the Bernoulli mechanism for a population model with the structure described by Fig. 2, where the flows are assumed to be random sequences of events. Using a very small time-step h, an accurate model that describes the time evolution at the three compartments would then be given by

[5] For example if the intensity $\lambda(t_1)$ of customers arriving at a lunch restaurant is very low before lunch, then the next customer, scheduled to arrive at $t_2 = t_1 + Expo(1/\lambda(t_1))$ may come late in the afternoon, implying that the peak of lunch customers is missed. This can be handled by e.g. techniques for over-sampling and thinning [11,30,32].

$$\begin{cases} x(t+h) = x(t) - h \cdot F1, \\ y(t+h) = y(t) + h \cdot F1 - h \cdot F2, \\ z(t+h) = z(t) + h \cdot F2, \\ h \cdot F1(t) = Ber[h \cdot \lambda(x,y,z,t)], \\ h \cdot F2(t) = Ber[h \cdot \mu(x,y,z,t)], \end{cases}$$

where time is updated according to $t := t + h$. The Bernoulli distribution $Ber[h \cdot \lambda]$ can e.g. be implemented as

If $Uniform[0,1] < h \cdot \lambda$ *then Event*.■

This approach is of only theoretical interest. Numerically, it is extremely inefficient since the time-step h has to be so small that an event rarely occurs. Otherwise the risk becomes non-negligible that more than one event per time-step should – but cannot – happen in the model, which gives biased results. Such a case is demonstrated in Example 2.

Example 2 (*bacterial growth*). Assume that a single bacterium starts breeding with on average one cell division per hour. To make matters simple (but not very realistic) in order to get a rough estimate of the numerical consequences, we reason deterministically here. After three days (72 h) there will be about $2^{72} \approx 5 \times 10^{21}$ bacteria. Now, assume that Bernoulli time handling with a time-step of 0.1 h (a 10th of the time constant) is used in a compartment model. Because the Bernoulli outcome is restricted to zero or one for each time-step, the model would produce a maximum of $72 \times 10 = 720$ bacteria. To get an accurate result it would be necessary to use a time-step less than 10^{-22} h so that $h \cdot \lambda \ll 1$ (where $\lambda = \lambda(x(t))$ is an increasing intensity). Then about 72×10^{22} time-steps are needed!■

3.3.2.2. Exponential time handling in a macro model. This method was first published by Gillespie [16] and is known as the Stochastic Simulation Algorithm (SSA). The idea is to divide the algorithm into two parts by answering the questions: 1. When does the next event occur? and 2. Where does it occur (i.e. which flow will produce the next event)?

Example 1 (*revisited*). The model in Fig. 2 will handle exactly one entity transfer in either *F1* or *F2* for each event. This can be implemented as

$$F1 : x \to x - 1 \ \& \ y \to y + 1 \quad (\text{occurs with intensity } \lambda(x,y,z,t)),$$
$$F2 : y \to y - 1 \ \& \ z \to z + 1 \quad (\text{occurs with intensity } \mu(x,y,z,t)).$$

The events will then occur with a total intensity of $\lambda + \mu$. Therefore, the next event comes at $t := t + Expo[1/(\lambda + \mu)]$. Furthermore, the probability that this event belongs to *F1* is $\lambda/(\lambda + \mu)$. Thus the randomisation between *F1* and *F2* can be done by

If $Uniform[0,1] < \lambda/(\lambda + \mu)$ *then execute F1 else execute F2*.

The method can be extended to any number of flows with intensities $\lambda, \mu, \nu, \ldots$ [16,35].■

This method is surprisingly efficient for small intensities. It also has the advantage of being exact if the external conditions do not change (i.e. when $\lambda(x,t) = \lambda(x)$ only). Also the time-step (h or Δt) is eliminated from the model. This works since the dynamics are frozen until the next event.

Example 2 (*revisited*). The bacterial growth will be correctly handled by the exponential time-handling method, but the time to run the model over some 5×10^{21} events (time-steps) will be considerable.■

3.3.2.3. Poisson time handling in a macro model. Poisson Simulation was first described by Gustafsson in 2000 [15] and in the following year by Gillespie [33] under the name 'tau-leaping'. (The tau is the Δt leaping over many events.)

The fundamental idea is that *during a sufficiently short time-step, each intensity can be considered constant and can therefore be modelled by a Poisson process*. The simulation can then be regarded as sequences of Poisson processes. Furthermore, the number of events in each flow is realised by drawing a Poisson-distributed random number of the form $Po[\Delta t \cdot intensity]$ from a random number generator for every time-step, Δt.

Example 1 (*revisited*). When time is updated with a time-step Δt and flows between compartments are described by the Poisson distribution mechanism, the model of Fig. 2 becomes

$$\begin{cases} x(t+\Delta t) = x(t) - \Delta t \cdot F1, \\ y(t+\Delta t) = y(t) + \Delta t \cdot F1 - \Delta t \cdot F2, \\ z(t+\Delta t) = z(t) + \Delta t \cdot F2, \\ \Delta t \cdot F1(t) = Po[\Delta t \cdot \lambda(x,y,z,t)], \\ \Delta t \cdot F2(t) = Po[\Delta t \cdot \mu(x,y,z,t)], \end{cases}$$

where time is updated according to $t := t + \Delta t$.■

Example 2 (*revisited*). The bacterial growth is easily simulated with Poisson time handling using e.g. $\Delta t = 0.01$ h. Then 7200 time-steps are needed, compared with some 72×10^{22} for the Bernoullian or 5×10^{21} for the exponential time-handling approaches. This represents an improvement of the order of 10^{20} or 10^{18}-fold for this example.■

The Poisson Simulation approach can handle any number of events in any number of flows for each time-step. It largely requires the same considerations as the numerical solving of deterministic differential equations. Deterministic and stochastic differential equations can also be combined in the same model in so-called combined simulation [27,36].

The form $x_i(t + \Delta t) = x_i(t) + Po[\Delta t \cdot \lambda_i(\boldsymbol{x}, t)] - Po[\Delta t \cdot \mu_i(\boldsymbol{x}, t)]$ for $i = 1..n$, where $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)^T$ is also very intuitive since it resembles the embedded deterministic difference equation: $x_i(t + \Delta t) = x_i(t) + \Delta t \cdot \lambda_i(\boldsymbol{x}, t) - \Delta t \cdot \mu_i(\boldsymbol{x}, t)$, obtained by stripping off the stochastic Po-parts. This also provides a bridge between stochastic and deterministic macro modelling, see Fig. 1. Gustafsson [7] discusses when consistency is maintained for the embedded deterministic model.

3.3.3. Time handling in state-based models

Now consider the three different ways of handling time in state-based models. To be specific, we focus on a Markov chain model where the step-size is short enough to regard the time as almost continuous.

A *Markov chain* is a stochastic process with discrete states $\{X(t), t = 0,1,2,\ldots\}$ in discrete time for which the Markov condition: $P[X(n) = x_n | X(0) = x_0, X(1) = x_1, \ldots, X(n-1) = x_{n-1}] = P[X(n) = x_n | X(n-1) = x_{n-1}]$ is valid. This condition simply states that the process is memoryless and that all there is to know is represented by the actual state the system is in. (A compartmental model also has this property.)

The time unit for the sequence $t = 0,1,2,\ldots$ implicitly constitutes the time-step between the time-points enumerated. The time-step can be very small (h) or quite large (Δt), but must be smaller than the shortest time constant of the dynamic changes.

3.3.3.1. Bernoulli time handling in a state-based model. Let us consider a birth–death Markov chain model where it is sufficient to regard only transitions between 'neighbouring' states. This will make the transition matrix sparse. Since a transition corresponds to zero or one event, the transition probabilities will be of Bernoulli type: $p_{i,i-1} = \mu h$ and $p_{i,i+1} = \lambda h$, as obtained directly from the definition of the Poisson process. The diagonal elements of the state transition matrix will then be $1 - \Sigma_{j \neq i} p_{ij}$ so that the row elements add up

to unity. The Bernoulli time handling is only valid for very small time-steps so the model becomes very inefficient to execute.

3.3.3.2. Exponential time handling in a state-based model. The Markov chain definition above only allows events to occur at regular time-points $t = 0, 1, 2, \ldots$ This condition of advancing time in equal time segments can be relaxed [20]. The irregularly updated time is then handled separately [20]. Stepping time irregularly from one event to the next is not only more realistic, but also has a number of advantages.

- First, the matrix will be sparse compared with the method to be described in Section 3.3.3.3 below.
- Second, since an event happens at the end of the interval, a '*new*' state is *always* reached. The diagonal elements a_{ii} of the transition matrix represent the probability of the system remaining in the state. They are therefore equal to zero, unless there is an absorption state. (Then the corresponding diagonal element is equal to 1, preventing any further events to happen).
- Third, when the diagonal elements are all zero, all the elements of a row are *proportional* to the probabilities of the respective transitions. The transition probabilities of rows then only have to be scaled so that they add up to unity (without involving a time-step). This also makes the transition matrix easier to construct.
- Fourth, the problem of finding an appropriate step-size is eliminated.

3.3.3.3. Poisson time handling in a state-based model. Using Poisson time handling with a larger time-step Δt enables correspondence to multiple transfers between compartments and results in a dense transition matrix. This speeds up the execution of the model. However, the assignment of values to the transition probabilities p_{ij} is a formidable task for all but the simplest models [23].[6]

For both the h and the Δt approaches, the transition probabilities are furthermore functions of the time-step used, which means that an adjustment of the time-step would require a complete reconstruction of the transition matrix.

3.3.4. Summary

Table 1 provides a summary of micro, macro and state-based population modelling, each of which may use the three ways of handling time.

3.4. Consistency for micro, macro and state-based population models using different time handling principles

When performing a model study, the purpose of the study requires a desired level of accuracy in the outcome (averages, variations, some functions of the joint pdfs, etc.). The desired accuracy then refers to the difference in behaviour between the real system and the model.

When comparing two stochastic simulation models for consistency, one considers the 'error' that represents the difference between the outcomes of the two models. For two models that possibly belong to different model types, it should be possible to reduce this inter-model error to below a pre-specified value. If so, the two models are said to be consistent. The issue is now whether this can be achieved by certain measures, such as sufficiently small step-size, sufficiently accurate representation of the sojourn time distribution in the form of structures of compartments or states, etc.

It is of course not feasible to demonstrate this property for every conceptual model. Instead, a stepwise analysis is performed here, where the focus is on one type of model (micro, macro, and state-based in turn) to show that different models within the selected model type that differ only in their time-handling principles can be made consistent. Thereafter, the different model types are compared with respect to consistency.

3.4.1. Time handling and consistency

The Bernoulli, exponential and Poisson principles are consistent ways of handling time for stationary intensities (Poisson processes). In particular, when the intensities $\lambda_i(\boldsymbol{x}(t))$ are dynamically related to entities, compartments or states but not explicitly to time, then the exponential principle is exact. Consistency with the Bernoulli and Poisson principles is therefore obtained by holding the time-step h and Δt small enough, which reduces the differences in outcome to arbitrarily small levels.

When the intensities are functions of time $\lambda_i(\cdot, t)$ because of varying external conditions, this creates no additional problem with the Bernoulli and Poisson methods. However, in such cases the exponential method may sample too long an inter-event time to the next event if the intensity is increasing with time. As discussed in Section 3.2 and the accompanying footnote, this can be corrected for by e.g. over-sampling and thinning [11,30,32].

Since population models that use different time handling principles can be made to produce consistent results within one model type, the extent to which micro, macro and state-based models can be made mutually consistent remains to be discussed.

3.4.2. Model type and consistency

The conceptual population model has by definition the four properties: *discrete entities*, *continuous time*, *dynamic relations* and *stochastic variations*.

Furthermore, the entities of the conceptual population model may have a number of properties, logical connections and interactions that together determine their behaviour:

(a) They may represent a non-homogeneous population characterised by different attributes (such as sex, age, health status, etc.).
(b) They may remain in a situation (stage) during a sojourn time according to a specific statistical distribution.
(c) They may remain in a situation (stage) for a deterministic time T.
(d) They may take different paths for reasons that are not exactly known and must therefore be expressed in probabilistic terms.
(e) They may take different paths for reasons that can be formulated as logical relations.
(f) They may support or block other entities. For example, they may influence each other by signals.
(g) They may form pairs or groups with other entities.
(h) They may need resources to perform a task and have to wait for that resource ('queue for it') if not available.

A *micro model* can map points (a)–(h) of the conceptual model in a one-to-one manner. Different technical realisations may de-

[6] For example, consider a two-compartment model containing a large amount of molecules that can be in either of these compartments. The states are accordingly numbered from 0 to n, representing the number of molecules in the first compartment. Then let p_{ij} denote the probability of going from i to j ($i > j$) molecules during Δt. If the net transfer of $i - j$ molecules during Δt only went from the first compartment to the second, then we would have the transition probability $p(i-j) = e^{-\Delta t \cdot i \cdot \mu}(\Delta t \cdot i \cdot \mu)^{i-j}/(i-j)!$, which is the $i - j + 1$:*th* term of a Poisson distribution with intensity μ during the interval Δt. However, a decrease of $i - j$ molecules in the first compartment during Δt will also occur if $i - j + 1$ molecules depart and 1 molecule arrives, or if $i - j + 2$ molecules depart and 2 arrive, etc. So even in this extremely simple model the transition probabilities become complicated expressions of terms from Poisson distributions. Furthermore, the length of the time-step Δt appears in a nonlinear way in such expressions.

Table 1
Nine possible ways of combining micro, macro and state-based models using Bernoulli, exponential or Poisson type time handling. The 3×3 alternatives for constructing stochastic simulation models become mutually consistent if correctly modelled.

Model type	Time handling		
	Bernoulli (h) *Usually zero and occasionally one entity transferred per very small time-step*	**Exponential** (*next event*) *Exactly one event in the model is generated per time-step*	**Poisson** (Δt) *Many entities may be transferred in many flows for each time-step*
Micro model *n entities with attributes that may change at events* *1:1 mapping of the conceptual model describing the system under study*	Possible but very inefficient	Efficient for e.g. Discrete Event Simulation models [11–13]	Usually less efficient – but a practical way to handle time-varying intensities. See also the micro model in Example 4
Macro model *k compartments connected by flows* *Distribution and attribute expansion of micro model*	Possible but very inefficient	Gillespie's SSA method [16]	Poisson Simulation model [15,33]
State-based model $\binom{n+k-1}{k-1}$ *states, and transitions between states* *Combinatorial expansion of macro model*	A sparse Markov model [20,21]	Zero diagonal and sparse Markov model [20]	Dense and extremely complicated Markov model [23]

pend on the *micro* language used, but no specific approximations are needed when performing this task.

When electing to use a *macro model*, the choice is already made to aggregate the individual entities into compartments where the individuals are not identifiable. The consistency issue is therefore limited to comparing results on an aggregated level.[7] To exemplify this, we assumed the *macro model* to be realised as a Poisson Simulation model in a common Continuous System Simulation language e.g. Powersim [37] or Stella [38] or directly written in a general purpose programming language.

In [36], the consistency between *micro* and *macro models* is demonstrated by giving a Poisson Simulation model in a general setting and discussing how to consistently accommodate the dynamics and the stochastics of the underlying conceptual model. This was accomplished by defining Poisson Simulation in mathematical terms as a series of Poisson processes that generate sequences of Poisson distributions with dynamically varying parameters.

In general, the *macro model* can handle the set of aspects (a)-(h) of a conceptual model in the following way (using e.g. Poisson Simulation):

(a) A non-homogeneous population is handled by attribute expansion.
(b) A sojourn time distribution other than exponential is handled by distribution expansion.[8]
(c) A static delay of length T can be modelled by a buffer of $N = INT[T/\Delta t]$ positions (compartments). Such pipeline delays are often predefined functions in a Continuous System Simulation language.
(d) Different paths, determined by probabilistic decision criteria, are obtained by giving the compartment two or more outflows and by adjusting the parameters that control the outflows.
(e) Logical functions (like the if-statement) are part of the Continuous System Simulation language to operate on the flows. Any part of the *macro model* can control any flow rate.
(f) All kinds of influences between entities can be modelled in accordance with the level of aggregation. Any compartment or group of compartments may control any flow rate.
(g) Formation of pairs or groups is common in e.g. chemical reactions where two or more molecules form another molecule or molecules. In such an application, the required number of ingoing molecules must be taken out of their compartments and the appropriate number of new molecules added to the respective compartment(s). The same idea is used for e.g. mating.
(h) Queues for handling waiting for resources can be built in a straightforward way, see [27]. In that publication, it is shown that $M/M/1$, $M/M/c$ and $M/M/\infty$ queuing models are easily constructed using Poisson Simulation. It is also shown how other kinds of queues can be formed. Adding counters and tallies is also straightforwardly realised within the *macro model* [27]. (Of course, the possibility of following *micro* properties, such as queuing discipline, is restricted to the level of aggregation.)

Selecting a *state-based approach* often results in an unrealistically large model. However, in principle it is possible to implement a conceptual model in a consistent way.

A *macro model*, represented by a Poisson Simulation model, can be mathematically transformed into a consistent *state-based model* represented by e.g. a Markov model [23]. In [23] it is discussed in detail how the rows of transition elements are obtained from the stochastic flow rates of the Poisson Simulation model so that consistency is obtained. To make the discussion lucid, it is assumed that a *macro model* (in the form of a compartment model) is to

[7] Of course, in the extreme case one might choose to give every entity a unique set of compartments – meaning that micro modelling is performed using a *macro* tool.

[8] Our experience is that a good approximation of sojourn time distributions can usually be obtained by a rather simple structure of compartments in series, parallel and/or feedback. This type of approximation can always produce arbitrarily small deviations from the desired distribution by increasing the number of compartments. An alternative, and sometimes more efficient, solution based on a shift buffer of elements (compartments) in series, that is fully consistent is presented in [27]. There, multiple inflows to different elements of the buffer are used, where the integer valued inflows to the buffer elements are selected in accordance with the specified statistical distribution.

be rewritten as e.g. a Markov model. The necessary combinatorial expansion is then assumed to be performed. In addition:

(a) Attribute expansion is already performed in the formulation of the *macro model*.
(b) Distribution expansion is already performed in the *macro model*.
(c) Any static delay described by $N = INT[T/\Delta t]$ compartments is already part of the combinatorial expansion.
(d) Different paths due to stochastic reasons are implemented by appropriate values of the transition elements for the currently relevant row of the transition matrix.
(e) A logical condition to control the flow rates of a compartmental model is handled by the values of the transition elements of the currently relevant row.
(f) In the compartment model, influences between different groups of entities are handled by appropriate influences from compartments to flow rates. In a *state-based model*, each situation (state) is separately described. It then remains only to assign the appropriate transition probabilities (or compulsory transition by an '1' in the appropriate element) to the row of transition elements of that state.
(g) The extra compartments needed to form pairs or groups in the compartment model are taken care of by the combinatorial expansion.
(h) Queues (used for waiting for resources) bring nothing special to the compartment model so there is no problem involved for the Markov model.

There may be further aspects of a conceptual model, but to the best of our knowledge there are no structures, logics or mechanisms for a conceptual population model that cannot be accurately represented in a micro, macro and (in principle because of size) state-based model.

The conclusion is that consistency between micro, macro and state-based models using Bernoulli, exponential or Poisson methods for time handling can be obtained provided that: (1) Appropriate distribution, attribute and (for state-based models) combinatorial expansions are performed; and (2) the time handling is appropriately performed (not too large time-step, oversampling and thinning if needed).

4. Merits and demerits of micro, macro and state-based population models

4.1. A cautionary example

Not all types of stochastic differential or difference equations are consistent with a conceptual population model. For example, in many common types of stochastic differential or difference equations, the change of a quantity is described by separate deterministic and stochastic terms (or factors) on the right-hand side of the equation. This is an excellent form for many problems in e.g. automatic control where signals are separated additively from noise, but is not appropriate for population models, because the stochastics are an intrinsic part of the changing process originally defined by events. As a simple example, consider the artefacts that may be generated by a stochastic difference equation model with an additive noise term.

Example 3 (*radioactive decay*). Radioactive decay, where $x(t)$ represents the number of radioactive atoms with a decay fraction a per time unit, should in a macro model with Poisson Simulation be modelled by

$$\Delta x(t) = -Po[\Delta t \cdot a \cdot x(t)].$$

We may contrast this to a model with additive noise perturbations:

$$\Delta x(t)/\Delta t = -a \cdot x(t) + b \cdot e(t),$$

where $e(t)$ is a zero-mean discrete-time white noise. The latter form would produce a number of unfeasible phenomena such as: (1) Non-integer numbers of atoms[9]; (2) stochastic variations unrelated to the remaining number of atoms; (3) sudden increases in the number of atoms; (4) continued variations around the equilibrium state even when there are no atoms left; (5) the number of atoms may become negative; (6) the ensemble of trajectories from many replications not having the correct distribution; and (7) the extent of randomness being strongly dependent on Δt. Such artefacts are all eliminated by e.g. the Poisson Simulation approach. ■

The artefacts in this example may seem innocent and obvious, but when hidden within parts of a larger model, such artefacts may generate severe consequences. Variations without appropriate reasons may excite other parts of the model. Negative numbers of entities may trigger different kinds of phenomena.

4.2. Choice of model type

The consistency between micro, macro and state-based models brings the great advantage that the type of model that gives the *most appropriate realisation* of the conceptual model can be selected. This issue has many aspects, such as model size and construction effort, homogeneity of population, average or individual issues to study, execution time, transparency and communicability of the model, access to data for building the model, validation, etc.

To exemplify how a conceptual model can be realised in a micro, macro or state-based form, an epidemic model can be considered. Models of infectious diseases are generally based on a sequence of *stages* from Susceptible via Infectious to Removed. Such a model is therefore denoted a SIR model. The first SIR model was published by Kermack and McKendrick in 1927 [39].

4.2.1. Example 4: An epidemic SIR model

An epidemic is a process that affects a population of individuals. From the system under study and the purpose of the study, a conceptual model is defined where the fundamental structure and the assumptions about the population and the disease are stated. Such a conceptual model is realised here as micro, macro and state-based models.

4.2.1.1. The conceptual model. The conceptual model of our example has the following setting: The population, consisting of n individuals, is exposed to an infectious disease. Every individual of the population meets every other individual under equal conditions in each time unit. The population is here assumed to be non-homogeneous. There are differences between individuals due to sex and age. Each individual is classified according to sex = {male, female} and age group = {child, adult, elderly}. The attributes sex and age group are regarded as constant over the time of study.

The disease has the three consecutive stages ($\mathbf{S} \rightarrow \mathbf{I} \rightarrow \mathbf{R}$). The change of stage for an individual takes place instantaneously. Furthermore it is assumed that:

- An event is *statistically independent* of all other events at that point in time.
- The probability of a Susceptible individual being infected by an Infectious individual is $p_{s,a}$ per time unit.
- The time during which an individual resides in stage **I** has a 3-Erlang distribution with the mean of $T_{s,a}$ time units, i.e. a $\Gamma(3, T_{s,a}/3)$ distribution.

[9] We here ignore the decay products, why, $x(t)$ is denoted '*the number of atoms*'.

4.2.1.2. The micro-model. Implementation of the conceptual model as a micro-model is a straightforward one-to-one mapping of the individuals with their attributes and behaviours. Each individual has the variable attribute *stage*, which can take the values '**S**', '**I**', '**R**', and the fixed attributes s = sex and a = age group, which affect the probability of getting infected, $p_{s,a}$, and the expected sojourn time in stage **I**, $T_{s,a}$. The behaviour is: 'everybody meets everybody at each time unit' and 'when a Susceptible meets an Infectious the former becomes infected and is transferred to stage **I** with probability $p_{s,a}$'. Furthermore, when an individual enters into stage **I**, a random number for the sojourn time in the stage is drawn from a 3-Erlang distribution with the time parameter set to $T_{s,a}/3$.

For practical reasons a Poisson-distributed time handling approach is used here, so that several events may happen during a time-step. A micro-model may then have the following structure.

- Generate the n individuals as entities and assign initial values to the variable attribute *stage*{**S**,**I**,**R**}, and to the fixed attributes *sex*{male, female} and *age group*{child, adult, elderly}.

While time < Simulation_time, loop over the following three steps.[10]

- *Time := time + Δt.*
- *For each entity in stage **S**, test if it will become infected during Δt. If so, change its stage to stage = **I** and draw a value for the sojourn time in **I** from a 3-Erlang[$T_{s,a}$/3] distribution.*
- *For each entity in stage **I**, test whether the sojourn time is up. If so, its stage = **R**.*

4.2.1.3. The macro-model. Let S, I and R stand for *numbers of individuals* in the consecutive stages **S**, **I** and **R**.

Because the macro model is based on compartments, the required 3-*Erlang*[T/3] distribution must be created or approximated by *distribution expansion*. In this case the 3-*Erlang*[T/3] can be exactly obtained by expansion to three compartments in a series with an expected sojourn time of T/3 for each compartment [40]. The Infectious stage therefore expands into three compartments denoted I1, I2 and I3 – all equally infectious.

Furthermore, the attributes sex and age group cause an *attribute expansion* of the number of compartments. Since the processes for the subpopulations are similar in structure, the simulation model should be vectorised by giving compartments and flows indices s and a to loop over. Using the Euler algorithm, the stochastic macro model then takes the form

$$\begin{cases} S_{s,a}(t+\Delta t) = S_{s,a}(t) - \Delta t \cdot F1_{s,a}(t), \\ I1_{s,a}(t+\Delta t) = I1_{s,a}(t) + \Delta t \cdot F1_{s,a}(t) - \Delta t \cdot F2_{s,a}(t), \\ I2_{s,a}(t+\Delta t) = I2_{s,a}(t) + \Delta t \cdot F2_{s,a}(t) - \Delta t \cdot F3_{s,a}(t), \\ I3_{s,a}(t+\Delta t) = I3_{s,a}(t) + \Delta t \cdot F3_{s,a}(t) - \Delta t \cdot F4_{s,a}(t), \\ R_{s,a}(t+\Delta t) = R_{s,a}(t) + \Delta t \cdot F4_{s,a}(t), \\ \Delta t \cdot F1_{s,a}(t) = Po[\Delta t \cdot S_{s,a}(t) \cdot (\Sigma_{m,n}(I1_{m,n}(t) + I2_{m,n}(t) \\ \quad + I3_{m,n}(t))) \cdot p_{s,a}], \\ \Delta t \cdot F2_{s,a}(t) = Po[\Delta t \cdot I1_{s,a}(t)/(T_{s,a}/3)], \\ \Delta t \cdot F3_{s,a}(t) = Po[\Delta t \cdot I2_{s,a}(t)/(T_{s,a}/3)], \\ \Delta t \cdot F4_{s,a}(t) = Po[\Delta t \cdot I3_{s,a}(t)/(T_{s,a}/3)]. \end{cases}$$

Thus a threefold *distribution expansion* of the Infectious stage is first made and then the *attribute expansion* for sex and age group is applied for the five compartment equations above. This results in $5 \times 2 \times 3 = 30$ compartments. The four flow equations expand to $4 \times 2 \times 3 = 24$ flow equations, where the flow equations are denoted $Fi_{s,a}(t)$. The total model thus consists of 54 equations.

4.2.1.4. The state-based model. A state-based model can be constructed by a *combinatorial expansion* of the compartment model above. The size of the state-based model depends both on the dimensions of the state-space (i.e. the number of compartments in the macro model) and on the size of the population. The population of n individuals is partitioned into six subpopulations with fixed attributes {sex} × {age} so that $n=n_1+n_2+n_3+n_4+n_5+n_6$ entities. Each individual of a subpopulation can then be in any of the $k = 5$ compartments {S, I1, I2, I3, R}. Thus each of the six subpopulations needs $\binom{n_i+k-1}{k-1}$ states independently of each other. Therefore, a total state-space of $\Pi_{i=1..6}\binom{n_i+k-1}{k-1}$ states is needed.

For example, consider a small population with only $n_i = 100$ individuals in each of the six subpopulations $i = 1,2,\ldots,6$. The state-based model is then composed of $\binom{104}{4}^6 \approx 10^{40}$ states, so a transition matrix would need to have 10^{40} rows of transition probabilities.

Let us sum up this epidemic SIR modelling example for the case where initially 100 individuals are in each of the six subpopulations. With a *micro model* there is one entity class (prototype) with the fixed attributes sex and age, the dynamic attribute stage, and a procedure controlling the transfers to a new stage. This single prototype is then copied into 600 entities, which are initiated with respect to sex and age and with respect to the initial stage.

For the *macro model*, distribution and attribute expansions result in a model with 30 compartments independently of the population size. Finally, a *state-based model* requires (in addition to distribution and attribute expansions) a combinatorial expansion which results in about 10^{40} states.

4.2.2. Model size of micro, macro and state-based models

Fig. 4 shows the number of entities in a micro model, the number of compartments in a macro model, and the number of states in a state-based model for various sizes of the population (n) and categories/compartments (k). In the last case the formula $\binom{n+k-1}{k-1}$ is used, which is correct when every entity can be in every category/compartment. (As seen from the epidemic SIR modelling example, this is not always exactly the case.)

Thus the number of states, even for quite a small state-based model, can become astronomically high, while for macro modelling the model size grows only linearly with the number of compartments (k) and not at all with the population size (n). The micro model usually requires only one or a few entity prototypes with attributes and behaviours from which it is trivial to generate any number (n) of entities.

Conceptual complexity is of importance beside computational complexity. An example of when a micro model may be preferred to a macro model from a conceptual complexity perspective is the following.

Example 5 (*micro or macro models of vaccination*). A vaccine against infection by Human Papilloma Virus (HPV), which causes cervical cancer, was developed by Merck & Co. At Merck Research Laboratories, a macro model for assessing HPV vaccination strategies was developed [41].

From a conceptual point of view this model describes the stages: susceptible, three stages of HPV infection, eight pre-stages of cancer, and invasive cancer. The stages are also subdivided into different HPV types, and all possible progressions and regressions

[10] Faster execution is obtained with: *While I ≠ 0 loop over the model*, because the simulation can then be terminated when I becomes zero.

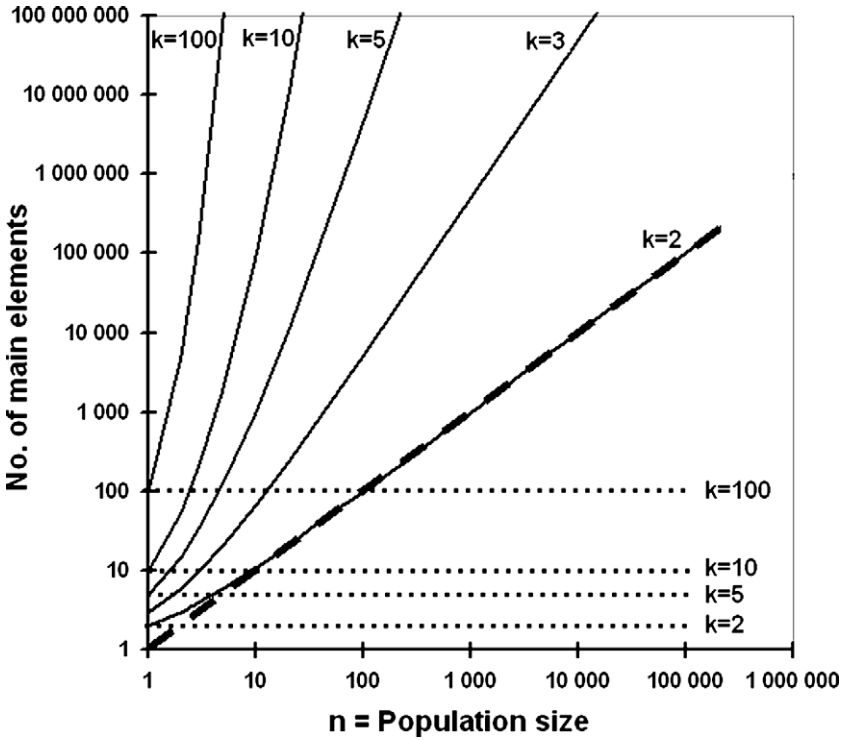

Fig. 4. Approximate size of a micro, macro, and state-based model. For a micro model the model size grows linearly with the number of *entities* (n) (dashed line). For a macro model the size grows with the number of *compartments* (k) after distribution and attribute expansions are performed (dotted lines). For a state-based model the number of *states* grows in combination with the number of entities (n) and the number of state-space dimensions (k) (solid lines).

between these stages are also modelled. Furthermore, a population of 100,000 individuals is classified by sex, 17 age groups, vaccinated or susceptible, habits of attending screening, screening outcome, degree of sexual activity, previous treatment such as hysterectomy, etc.

This led to the construction of a deterministic compartmental model which, because of attribute expansion, consists of 7191 differential equations (compartments)! [42]. The documentation of the model comprises 50 pages, dominated by formulae with multiple indices.

This model was reformulated to a micro model by the Danish National Board of Health [43] and implemented as a computer program [Code and personal communication]. To us, there is no doubt that the Danish micro model is considerably easier to understand, build, validate and communicate and better structured than the original Merck model.

As a final detail, let us consider the smooth extendability of a micro model. Assume that age was originally omitted from the vaccination model, but is now to be included. In the micro model one static attribute '*TimeOfBirth*' is to be added and age can then be calculated when needed from $Time - TimeOfBirth$, while the number of compartments in the macro model has to be expanded by, say, a factor of 20 if the population is subdivided into 5-year age classes between age zero and a hundred.■

4.3. Consequences of omitting distribution and attribute expansions

4.3.1. Distribution expansion and feasibility

Knowledge of the time an entity stays in a stage may be represented by any probability distribution (uniform, exponential, gamma, Weibull, etc. or even empirical), whereas a single compartment holds the entities in accordance with an exponential distribution (in the linear and time invariant case). For example, if the sojourn time in a stage has a probability distribution differing from exponential, then modelling a stage by a single compartment, without checking the consequences, is a serious and frequent error in both macro and state-based modelling.

4.3.1.1. Feasible and unfeasible representations of sojourn time in a stage. Most processes in real life require a non-zero time. For e.g. biological or medical processes, a non-zero time is needed for digesting, transportation of nutrients, defeating a disease, developing a cancer, growing an embryo, etc. For such a process a first order process is biologically *unfeasible* because it means that the probability of performing the process in almost zero time, i.e. finishing it within $(0,\varepsilon)$, is larger than finishing it in any other time interval $(t,t+\varepsilon)$ of length ε.

Example 6 (*distribution expansion of the infectious stage of a SIR model*). Many model studies produce strongly biased results due to modelling the sojourn time of a stage by a single compartment. One example is the stochastic SIR model, discussed in Section 4.2. Depending on the disease, the sojourn time distribution of the infectious stage must be realistically described. For example the average size of a new epidemic produced by the model strongly varies with the number and structure of compartments, even when the average sojourn time in the stage is the same. ■

4.3.2. Attribute expansion

The importance of attribute expansion is easy to understand. If entities with different attributes (such as sex, age, etc.) behave differently, then it is often important to preserve these differences in the model – rather than modelling a homogeneous population of 'average individuals'. For example a population of two strains with fertility *f1* and *f2* will grow faster than one strain with fertility $(f1+f2)/2$ and the strain with the largest fertility will soon dominate.

Example 7 (*attribute expansion to explain why a disease will not vanish*). If an infectious disease has a basic reproduction number less than unity (an infectious individual infects less than one other person on average) in a homogeneous population, the disease will die out. However, if there is a risk group while the rest of the population is at low risk, then the disease may remain in the population.■

5. Discussion and conclusions

Construction of models and their use in simulation is a powerful extension of mathematical and statistical methods for studying complex systems. Unfortunately, model construction and simulation are often practised by researchers with insufficient understanding of the implicit meaning of applying different modelling approaches. Too often, some standard model is selected because it is simple and commonly used, but it often has absurd consequences on closer inspection. There is a lack of a scientific base that considers the underlying qualities of different types of modelling and the relationships between different types of models.

This paper presents a uniform foundation for population modelling based on consistency and discusses the fundamental relationships between micro, macro and state-based population models. The focus is on the most fundamental aspects of population modelling and simulation, such as primary building blocks and time handling, rather than on technical issues and implementation with more complicated situations such as queue handling, use of resources of different types, co-operation between entities, etc.

The message of this paper is twofold. On the theoretical side it is shown that, based on the concept of events, there are three ways to handle time that can each be applied to micro, macro and state-based population modelling. Furthermore, it is shown that by applying distribution and attribute expansions, micro and macro models can be made consistent. Finally, by applying combinatorial expansion to the macro model, a consistent state-based model can be obtained.

On the practical side, important findings are as follows:

1. We demonstrate how to realise a conceptual model in the form of a micro, macro or state-based population model, including necessary expansions.
2. We discuss the size of a model in terms of the number of main building blocks for a micro, macro or state-based population model and its consequences for the model building effort. See also Fig. 4.
3. We demonstrate how time can be handled and its consequences.
4. We present an overall perspective (Fig. 1) where the type of modelling and simulation can be selected rather than being set routinely. This selection may be affected by the purpose of the study, the effort required to build the model, size of model, execution time, data available, parameterisation and parameter estimation, simplicity and transparency of the model, validation aspects, etc.

In the not recent, but still relevant, book '*System Simulation – Programming Styles and Languages*' [13], Kreutzer identifies four model 'paradigms': Monte Carlo Simulation, Continuous System Simulation, Discrete Event Simulation and Combined Simulation. It is interesting to see that all four of these paradigms can be related in our uniform base to modelling and simulation (see Fig. 1).

In Kreutzer's presentation, state-based models are not represented. The reason for including these models in the present paper is that they constitute a strong connection to analysis from fields such as stochastic processes and queuing theory, where theoretical knowledge can be obtained. For example, Burke's theorem [22,24] on stable stationary $M/M/c$ queuing systems states that a Poisson process driving an exponential server generates a Poisson process for the departures with the same intensity as in the driving process. Furthermore, the input and output processes are independent. In fact many multiple server nodes ($M/M/c$) can be connected in a feed-forward network and still preserve a node-by-node decomposition. Jackson [25] reports that for a network of queues with many nodes and different service times, each node behaves as if it is an independent $M/M/c$ system with a Poisson input rate.

However, these results from queuing theory are equally valid for e.g. Poisson Simulation models, since the compartments in such a model are $M/M/\infty$ queues, i.e. there is no waiting for service. Such results not only contribute to the understanding of stochastic modelling but also have direct implications in assigning initial values [27,28].

We here also wish to address the misunderstanding that micro models in general are more complex than macro models. For example, it has been stated (where micro models are exemplified by Individual-Based Models – abbreviated to IBM and differential equation models are named analytical models):

'However, the great potential of IBMs comes at a cost. IBM models are necessarily more complex in structure than analytical models. They have to be implemented and run on computers. IBMs are more difficult to analyze, understand and communicate than traditional analytical models (Grimm et al. 1999)' [44], and 'Particularly critical is the problem of communication. Analytical models are easy to communicate because they are formulated in the general language of mathematics. Their description usually is complete, unambiguous and accessible to the reader. In contrast, published descriptions of IBMs are often hard to read, incomplete, ambiguous, and therefore less accessible. Consequently, the results obtained from IBMs are not easily reproduced (Hales et al., 2003)' [44].

We do not share the view that a micro model is necessarily more complex than a macro model. Behind such an opinion there is usually a skewed comparison where more complex micro models are compared with relatively simple macro or state-based models, and where the latter type often lacks appropriate expansions. Letting all distributions implicitly be exponential and removing the attributes certainly makes the model simple, but not consistent with a more detailed micro model. An illustrative example of this is the model of vaccination against cervical cancer given in Example 5.

In our opinion, the documentation and communication of a model should be on at least two levels. A presentation in broad outline should describe the conceptual model regardless of how it is implemented. On a technical level the implementation should be described in detail in accordance with the claims of the selected micro, macro or state-based approach.

Finally, we again caution against the routine use of modelling without first having a conceptual model that is appropriate for the intended purpose. A similar danger is when a 'confection model' is picked by routine and perhaps slightly modified, rather than tailor-made from a conceptual model. It is the responsibility of the modeller to select the appropriate type of modelling and time handling and to check that any simplifications made do not distort the model and its results. This paper hopefully contributes to this in the form of theoretical understanding and insights into practical population modelling.

References

[1] D. Claessen, J. Andersson, L. Persson, A.M. deRoos, Delayed evolution branching in small populations, Evol. Ecol. Res. 9 (2007) 51.
[2] L. Fahse, C. Wissel, V. Grimm, Reconciling classical and individual-based approaches in theoretical population ecology: a protocol for extracting population parameters from individual-based models, Am. Nat. 152 (1988) 838.

[3] P. Gómez-Mourelo, From individual-based models to partial differential equations. An application to the upstream movement of elvers, Ecol. Model. 188 (2005) 93.
[4] V. Grimm, Ten years of individual-based modelling in ecology: what have we learned and what can we learn in the future?, Ecol Model. 115 (1999) 129.
[5] S. Huet, G. Deffuant, Differential equation models derived from an individual-based model can help to understand emergent effects, J. Artif. Societies Soc. Simul. 11:2, 10 (2008) 1. http://jasss.soc.surrey.ac.uk/11/2/10.html.
[6] W.G. Wilson, Resolving discrepancies between deterministic population models and individual-based simulations, Am. Nat. 151 (1998) 116.
[7] L. Gustafsson, Consistency or not between deterministic and stochastic population models? Technical Report, Signals and Systems, Uppsala University, 2009, p. 1. Available at: www.signal.uu.se/Research/simulation.html.
[8] P.A. Bobillier, B.C. Kahan, A.R. Probst, Simulation with GPSS and GPSS V, Prentice-Hall, Englewood Cliffs, NJ, 1976.
[9] A.M. Law, An introduction to simulation using SIMSCRIPT II.5, C.A.C.I, 1984.
[10] C.D. Pegden, R.E. Shannon, R.J. Sadowski, Introduction to Simulation Using SIMAN, McGraw-Hill, New York, 1995.
[11] W.D. Kelton, R.P. Sadowski, D.A. Sadowski, Simulation with Arena, WCB/McGraw-Hill, New York, 1998.
[12] G.M. Birtwistle, O.J. Dahl, B. Myrhaug, K. Nygaard, SIMULA BEGIN, Auerbach/Studentlitteratur, Philadelphia, 1973.
[13] W. Kreutzer, System Simulation: Programming Styles and Languages, Addison-Wesley, Sydney, 1986.
[14] G.M. Birtwistle, DEMOS – A System for Discrete Event Modelling on Simula, Macmillan Press, London/Basingstoke, 1979.
[15] L. Gustafsson, Poisson Simulation – a method for generating stochastic variations in Continuous System Simulation, Simulation 74 (2000) 264.
[16] D.T. Gillespie, A general method for numerically simulating the stochastic time evolution of coupled chemical reactions, J. Comput. Phys. 22 (1976) 403.
[17] D.T. Gillespie, Exact stochastic simulation of coupled chemical reactions, J. Phys. Chem. 8 (1977) 1.
[18] J.W. Forrester, Industrial Dynamics, MIT Press, Cambridge, MA, 1961.
[19] A.A. Markov, Extension of the law of large numbers to dependent events (in Russian), Bull. Soc. Phys. Math. Kazan 2 (15) (1906) 155.
[20] P. Brémaud, Markov Chains: Gibbs Fields, Monte Carlo Simulation, and Queues, Springer, New York, 1999.
[21] J.R. Clymer, Systems Analysis Using Simulation and Markov Models, Prentice-Hall International Series in Industrial & Systems Engineering, NJ, 1991.
[22] L. Kleinrock, Queueing Systems, Theory, vol. 1, Wiley, New York, 1975.
[23] L. Gustafsson, Poisson Simulation outperforms Markov simulation, Technical Report, Signals and Systems, Uppsala University, 2008, p. 1. Available at: www.signal.uu.se/Research/simulation.html.
[24] P.J. Burke, The output of a queueing system, Operat. Res. 4 (1956) 699.
[25] J.R. Jackson, Networks of waiting lines, Operat. Res. 5 (1957) 518.
[26] A.K. Erlang, Solution of some problems in the theory of probabilities of significance in automatic telephone exchanges, Electrotekniken (in Danish) 13 (1917) 5. English translation. P.O. Elec. Eng. J. 10 (1918) 189.
[27] L. Gustafsson, Poisson Simulation as an extension of Continuous System Simulation for the modeling of queuing systems, Simulation 79 (2003) 528.
[28] L. Gustafsson, Studying dynamic and stochastic systems using Poisson Simulation, in: H. Liljenström, U. Svedin (Eds.), Micro–Meso–Macro: Addressing Complex Systems Couplings, World Scientific, Singapore, 2005, p. 131.
[29] J.L. Devore, Probability and Statistics for Engineering and the Sciences, sixth ed., Thomson Learning, Toronto, Canada, 2004.
[30] A.M. Law, W.D. Kelton, Simulation Modeling and Analysis, McGraw-Hill, New York, 1991.
[31] W.H. Press, B.P. Flannery, S.A. Teukolsky, W.T. Vetterling, Numerical Recipes in Pascal – The Art of Scientific Computing, Cambridge University Press, Cambridge, UK, 1989.
[32] P. Bratley, B.L. Fox, L.E. Schrage, A Guide to Simulation, Springer, New York, NY, 1983.
[33] D.T. Gillespie, Approximate accelerated stochastic simulation of chemically reacting systems, J. Chem. Phys. 115 (2001) 1716.
[34] G.S. Fishman, Principles of Discrete Event Simulation, Wiley, New York, 1978.
[35] E. Renshaw, Modelling Biological Populations in Space and Time, Cambridge University Press, UK, 1991.
[36] L. Gustafsson, M. Sternad, Bringing consistency to simulation of population models – Poisson Simulation as a bridge between micro and macro simulation, Math. Biosci. 209 (2007) 361.
[37] Powersim Corporation, Powersim Reference Manual, Powersim Press, Powersim Corporation, 1175 Herndon Parkway, suite 600, Herndon, VA 22170, USA, 1996.
[38] High Performance Systems Inc., Stella II Tutorial and Technical Documentation, High Performance Systems, Hanover, NH, 1992.
[39] W.O. Kermack, A.G. McKendrick, Contributions to the mathematical theory of epidemics, Proc. Roy. Soc. A 115 (1927) 700.
[40] M.S. Hamilton, Estimating length and orders of delays in system dynamics models, in: J. Randers (Ed.), Elements of the System Dynamics Method, MIT Press, MA, 1980, p. 162.
[41] E.H. Elbasha, E.J. Dasbach, R.P. Insinga, Model for assessing human papillomavirus vaccination strategies, Emerg. Infect. Diseases 13 (2007) 28.
[42] E.H. Elbasha, E.J. Dasbach, R.P. Insinga, A technical report accompanying manuscript (50 pages): Model for accessing human papillomavirus vaccination strategies, Emerg. Infect. Diseases 13 (2007) 28. Supplementary online appendix: http://www.cdc.gov/ncidod/EId/13/1/28-app.htm.
[43] Sundhedsstyrelsen, Reduktion af risikoen for livmoderhalskræft ved vaccination mod humant papillomvirus (HPV) – En medicinsk teknologivurdering (in Danish), Medicinsk Teknologivurdering 2007; 9 (1) (2007) 1.
[44] V. Grimm, U. Berger, F. Bastiansen, et al., A standard protocol for describing individual-based and agent-based models, Ecol. Model. 198 (2006) 115.