Bridging the micro-macro gap in population forecasting

Contract no. SP23-CT-2005-006637

**Deliverable D8**

# Description of the Microsimulation Model

Authors:

Jutta Gampe
Sabine Zinn

MPIDR

June 2007

# Description of the Microsimulation Model

Jutta Gampe, Sabine Zinn
Max Planck Institute for Demographic Research
gampe@demogr.mpg.de, zinn@demogr.mpg.de

## 1 Introduction

This document discusses technical background and basic design aspects of the microsimulation software in the MicMac-project. The software is based on the general multistate model as laid out in [6]. For a general discussion of biographic forecasting see [5].

The Mic-software will contain three different components: the pre-processor, the Mic-core, which runs the actual Monte Carlo simulation, and the post-processor. The Mic simulation-software is based on a general multistate model, hence state-space, the transition intensities and the specification of a starting population are the key ingredients for an actual microsimulation.

Although the pre-processor will provide tools that allow to estimate transition rates from data, the specification of the actual model and the derivation of the empirically based transition rates will not be part of this document. These aspects heavily depend on the specific application, the availability of the respective data and how they were collected.

In the following section the necessary terminology and notation will be recalled. Mic is designed as a continuous time microsimulation model, therefore section 3 reviews some basic background and practical considerations to simulate arbitrarily distributed durations. Section 4 gives basic design considerations for the Mic-software and some extensions to the pure multistate-model that are currently explored.

## 2 Individual life-courses in the multi-state model

The focus of a micro-simulation model are the trajectories of the individual entities, which compose the aggregate system under study. In Mic this core entity will be an individual whose life-course will be conceptualized as a sequence of transitions between different possible states. The life-course modelled in Mic will evolve in continuous time. The transitions between states will be triggered by characteristics of the individual but also of

its environment, the latter one being the dimension where assumptions on future developments will enter.

To formalize an individual's trajectory we will need the states, that he or she may occupy. The potential states are summarized in the state-space, which we denote by $\mathcal{S} = \{1, \ldots, I\}$. The states give the demographic characteristics of the individuals considered relevant for the specific application. Transitions between states characterize demographic events. The most simple setting would be a two state-model, with states 'alive' and 'dead', the latter being absorbing. Realistic applications will include the combination of several demographic dimensions in the state-space, such as educational achievement, living arrangement or civil status. The design of the Mic-software does not hinge on any specific state-space definition. In general, Mic will contain death as a state in the state space to guarantee finite individual trajectories.

Transitions between the potential states will not be fully deterministic, that is, transitions and consequently individual life-course trajectories will evolve stochastically. As a stochastic micro-simulation model Mic will hence also capture uncertainty in future population development. The transition rates have to be provided by the user and data availability may limit the complexity of the envisaged models. For two states $i$ and $j$ both in $\mathcal{S}$, we denote the age-specific transition rates for an exit from $i$ to $j$ by $\lambda_{ij}(x)$, where $x$ denotes the age of the individual, the principal demographic time-scale.

The notation $\lambda_{ij}(x)$ indicates that the basic assumption is that the multistate process is Markovian, that is that the relevant demographic characteristics are summarized in the states and that the transition rates depend, besides age, only on the exit state $i$ and the destination state $j$.

Models that extend to higher order dependencies, i.e. the history of an individual as expressed in the sequence of previously visited states, is in principle possible but not considered in the current model design.

The life-course of an individual can be characterized by the sequence of states he or she visits during life and the so called sojourn times, which give the durations how long the individual stays in a state before making a transition. How to determine the sojourn times and the sequence of visited states from the transition rates will be discussed in section 3.

## 2.1 Time scales

Three time-scales will be important to distinguish within the multistate model:

- age of the individual $x$
- calendar-time $t$, and

- time since entry into the current state $s$.

As we will simulate individual life courses the natural time scale of the stochastic process described by the multistate model will be age $x$. Transition rates will always be age-specific, i.e. $\lambda_{ij}(x)$. Calendar time $t$ necessarily will be involved if assumptions about future behaviour, i.e. changes to the age-specific transition rates, will be made for projection purposes. There may be cases where simulations with time-constant rates may be of interest, but in general transition rates will be functions of age $x$ and calendar time $t$, i.e. $\lambda_{ij}(x,t)$.

While age-specific patterns mostly will be derived, that is estimated, from current (or past) data, the dependence on $t$ will be imposed externally by the user who has some view on the future development.

The third time-scale $s$, time since entry into current state, may be irrelevant for many transitions, where dependence on $x$ and $t$ sufficiently describes the process, however, for some demographic transitions, like divorce rates, this third time-scale will be relevant, too. It should be stressed that incorporating transition rates $\lambda_{ij}(x,t,s)$ that depend on all three time-scales do not pose particular problems for the microsimulation set-up. It should be clear though that data requirements to estimate transition rates that allow dependence on $x$ and $s$ are much higher than age-specific rates only.

# 3 Simulation and random numbers

Creating life courses, which reflect behaviour as summarized in a multistate model, requires the simulation of transitions between states and lengths of stays in these states, as specified by the transition intensities of the model. How such individual trajectories can be created is described in this section. First we will reiterate the basic mathematical fact that allows to simulate random numbers from arbitrary continuous distributions. Then we will summarize the basic and crucial identities that can be derived for the different characterizing functions of duration variables. We will show these relations for the most simple case of a two-state model with one absorbing state, i.e. survival data. Thereafter we will demonstrate how we can extend this approach if there are multiple alternatives to exit from each state, the so called competing risks setting. Finally, we will return to a more practical aspect, namely the implementation of arbitrary distributions.

## 3.1 Random numbers for continuous distributions

As was already pointed out in the introduction, MIC will model processes in continuous time. Hence any duration of stays in one of the states takes values that are generated by

a continuous random distribution. To actually simulate random durations from arbitrary distributions, we make use of the following basic relationship:

If $X$ is a continuous random variable with distribution function $F(x) = P(X \leq x)$, then the random variable $Y$, which is obtained by transforming $X$ via *its own* distribution function $F$, has a standard uniform distribution, that is

$$Y = F(X) \;\Rightarrow\; Y \sim U(0,1) \quad \text{or} \quad P(Y \leq y) = y\,. \tag{1}$$

The fact that the distribution function of $Y$ is the identity characterizes the uniform distribution. Equation (1) can be reverted whenever $F$ has an inverse function $F^{-1}$. This is the case whenever $F$ is strictly increasing, i.e. over intervals where the density $f(x) = F'(x)$ is non-zero. If we transform $Y$, which, as we know, follows a $U(0,1)$-distribution, by the inverse distribution function $F^{-1}$ of $X$, then we obtain

$$F^{-1}(Y) = F^{-1}(F(X)) = X. \tag{2}$$

This demonstrates that we can – in principle – simulate random numbers from an arbitrary distribution $F$, if we can

- obtain random numbers from a standard uniform distribution $U(0,1)$ and
- invert the distribution function to obtain $F^{-1}$, which can be done analytically, where possible, or numerically, if no closed inversion formula exists. (For some details on this issue see section 3.4.)

Random number generators for the standard uniform distribution are provided in any major programming language or statistical software. Therefore, to obtain arbitrarily distributed random durations, we have to link uniformly distributed random numbers with the given transition intensities. This procedure will be described in the following sections.

## 3.2 Random durations

The basic principles are most easily described if we confine ourselves to the simplest setting: A two-state model, where one state is absorbing. The most prominent example is, of course, the transition from the state ‘alive’ to ‘dead’, i.e. the modelling of survival data. The sojourn time $S$ in this case is life span and the transition intensity is the hazard of death (the force of mortality). The duration in state ‘alive’ is age, so in this simple case the process time-scale age and the duration since entry into this state are identical. As a duration, $S$ can take only positive values and we denote its density by $f(s)$ and its distribution function by $F(s) = P(S \leq s)$. The survivor function $\bar{F}(s) = P(S > s)$.

The hazard (transition intensity) is given by

$$\lambda(s) = \lim_{ds \to 0} \frac{P(s < S \leq s + ds \,|\, S > s)}{ds} \;=\; \frac{f(s)}{\bar{F}(s)}. \tag{3}$$

Together with the fact that

$$\frac{d}{ds} \ln \bar{F}(s) = \frac{-f(s)}{\bar{F}(s)} = -\lambda(s) \tag{4}$$

we obtain the useful identity

$$\bar{F}(s) = \exp\{-\int_0^s \lambda(u)\, du\} = \exp\{-\Lambda(s)\}\,. \tag{5}$$

The integrated hazard $\Lambda(s) = \int_0^s \lambda(u)\, du$ will turn out to be the key function for the simulation of sojourn times.

To see this we pursue the idea laid out in section 3.1. If we have created a random number $u^*$ from a $U(0,1)$-distribution, we can obtain a random duration from the distribution of the sojourn time $S$ if we can find the corresponding (random) number $t^*$ for which

$$u^* = F(t^*) \quad \text{or} \quad F^{-1}(u^*) = t^*\,.$$

Using (5), we see that

$$u^* = F(t^*) = 1 - \bar{F}(t^*) = 1 - \exp\{-\Lambda(t^*)\} \tag{6}$$

or

$$\Lambda(t^*) = -\ln(1 - u^*) = v^* \quad \text{or} \quad t^* = \Lambda^{-1}(v^*)\,. \tag{7}$$

In this way, we have shifted the necessity of inverting the distribution function $F$ to inverting the integrated hazard $\Lambda$. As the transition intensity $\lambda$ is the key characteristic for describing the stochastic process that governs the transition between states (and hence durations in each state), it is natural to directly simulate by employing the function $\Lambda(s)$. The complexity of the problem is similar: For some simple distributions the integrated hazard $\Lambda$ can be inverted analytically, or numerically otherwise. A simple approximate solution will be demonstrated in section 3.4.

## 3.3 Multiple destinations and competing risks

The previous section demonstrated that simulation from a single duration variable can in principle be reduced to (a) drawing random numbers from a standard uniform distribution and (b) providing a way to evaluate the inverted integrated hazard of the distribution. In

this section we will consider how this basic procedure can be used to create random durations that describe the time until the next transition, which may have several alternative destination states. The potential transitions are characterized by their respective transition intensities. The presentation follows [4] and [1].

To simplify the presentation and also notation, we focus on the specific state, e.g. state $i$, into which an individual has just entered. If this state is not death, the only absorbing state in the state space, then the individual will exit out of the current state after a random duration $S$ and will move to one of the remaining $I-1$ states in the state space. Which destination state the individual will actually move to, depends on the transition rates $\lambda_j(x,t)$, $j \neq i$. For ease of notation we have dropped the current state index $i$ here, and we will also drop the dependence on the calendar time variable $t$, assuming that the transition rates are appropriately chosen to reflect current (and future) environment. Furthermore, if the individual has entered into the current state at age $x = x_0$, then we may shift the time scale to $\tilde{x} = x - x_0$, and hence $\lambda_j(\tilde{x} = 0) = \lambda_j(x - x_0)$, so that entrance into current state equals age zero on this scale. Otherwise the transition rates remain unchanged. In this way, length of stay in $i$ and age $\tilde{x}$ share the same origin.

A setting in which individuals will exit to one of several possible detination states is called a 'competing risks' setting. The sojourn time $S$ has a density denoted by $f(s)$, and the survivor function $\bar{F}(s) = P(S > s)$. The hazard is $\lambda(s)$. Again, for the duration $S$, which the individual will spend in the current state, we have $\lambda(s) = f(s)/\bar{F}(s)$. The interpretation of $\lambda(s)$ is exactly like in the previous section.

For known $\lambda(s)$ we could therefore simulate how long the individual would stay in the current sate. What would still be missing is the state $j \neq i$, to which the individual would exit.

The transition intensities $\lambda_j(s)$ summarize the rate of an exit to state $j$ at time $s$, which here is time units since entry age $x_0$, given the individual has not moved out of the current state before $s$. The hazard function $\lambda(s)$ of the sojourn time $S$, which is irrespective of the exit state, is linked to the transition intensities in the following way:

$$\lambda(s) = \sum_{j \neq i} \lambda_j(s)\,. \tag{8}$$

As the states other than the current one are exhaustive and mutually exclusive, and exit from the current state must be to one of the states $j \neq i$, the hazard of the sojourn time $S$ is given by (8).

To obtain the actual exit state $j \neq i$, we would need the probability distribution of the $(I-1)$ possible destination states. If we denote by $p_j$ the probability that departure from the current state is to state $j$, then we can derive these probabilities by the following steps:

For any time $s$, the probability to exit to $j$ in the interval $(s, s+ds)$ is given by

$\bar{F}(s)\lambda_j(s)ds$. Therefrom, by integrating over all $s$, we obtain the probabilities $p_j$:

$$p_j = \int_0^\infty \lambda_j(s)\bar{F}(s)\, ds\,. \tag{9}$$

It is important to note that these probabilities not only depend on the transition rate $\lambda_j(s)$ itself but also, via $\bar{F}(s)$, on all the other transition rates. In principle, at least numerically, it would be possible to calculate the $p_j$ and then randomly draw the exit state from this probability distribution.

There is, however, a different way to simulate both the sojourn time $S$ and the exit state $j$ simultaneously by directly using the $\lambda_j(s)$.

The justification is a formal equivalence between the above probabilites and the following setting: We postulate the existance of $I-1$ independent latent random variables $S_1, \ldots, S_{I-1}$, where the hazard for the random variable $S_j$ is equal to $\lambda_j(s)$. The stay in the current state by the indivdual is ended by the shortest of these latent durations, i.e. $S = \min S_j$. The exit state $j$ is the one whose corresponding latent duration variable was shortest.

It is this formal relationship that allows a simple and straightforward way to simulate the sojourn time $S$ and the exit state $j$ by simulating $I-1$ random durations based on the intensities $\lambda_j(s), j = i$.

In summary, all that is technically needed to simulate the transitions and sojourn times of an individual are uniformly distributed random numbers and procedures to invert the $\Lambda_j(s)$.

## 3.4 Approximations to integrated intensities

As has been mentioned before, it may well be possible that the integrated intensities $\Lambda(s)$ may not allow a closed-form inversion formula. This may be particularly true, if the transition intensities are empirically estimated based on rather complex hazard models.

One possibility to flexibly model intensities, without the need to use fully parametric distributions, is the specification of piecewise-constant intensities. In this case the integrated intensities are piecewise-linear and inverting such a function $\Lambda$ is extremely simple, as will be demonstrated in an example below. Additionally, this example is to demonstrate that approximating an integrated hazard via a piecewise-linear function leads to almost identical random numbers, rendering this approximation a potential general strategy.

In the example we assume that the intensity has the form shown in Figure 1. This choice does not refer to any particular application but was chosen to cover two aspects, namely a nonlinear hazard of non-standard shape and a hazard that declines to zero. Analytically this intensity is given as a multiple of the density of a Normal distribution with mean zero and standard deviation $\sigma$: $\lambda(s) = c \cdot \phi(s; \mu = 0, \sigma)$. This intensity $\lambda(s)$ has the

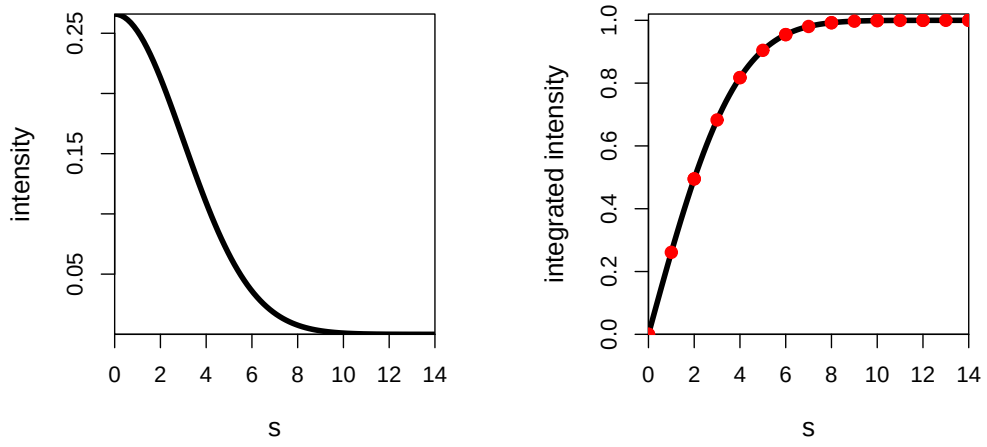


Figure 1: Intensity $\lambda(s)$ (left) and integrated intensity $\Lambda(s)$ (right). Red dots mark the points between which the integrated hazard was linearly interpolated.

corresponding integrated intensity $\Lambda(s) = c \cdot [\Phi(s; \sigma) - 0.5]$, where $\Phi$ is the distribution function of the Normal distribution (with mean zero and standard deviation $\sigma$).

Because of the specific shape of $\lambda(s)$, which converges to zero, the integrated hazard flattens and levels off at $c \cdot 0.5$ (In the example we chose $c = 2$). Consequently, the survivor function does not drop down to zero but levels at $\bar{F} \to \exp\{-0.5 \cdot c\}$, implying that a certain fraction will never experience the transition described by $\lambda(s)$. If such an intensity is one out of several competing risks, it may happen that the simulated latent random duration for this very transition would be infinitely long, corresponding to an individual not experiencing the event. In this case the latent duration of one of the other competing exit alternatives will show a shorter duration and hence will turn out to be the actually realized transition. If death is included as a competing risk, then at least one integrated intensity will increase to infinity (and not level off) and will therefore give a finite random duration.

To invert a piecewise-linear integrated hazard, two steps are necessary (see Figure 2): First we have to identify in which of the linear sections the random value $v^*$ falls (cf. equation (7)). From this information, we can identify the lower and upper values $\tau_L$ and $\tau_U$ and the corresponding values $\Lambda(\tau_L)$ and $\Lambda(\tau_U)$. In case that $v^*$ lies above the upper limit to $\Lambda$, the transition will not occur and the respective duration is set to infinity. From basic geometry we obtain that

$$t^* = \tau_L + [v^* - \Lambda(\tau_L)] \, \frac{\tau_U - \tau_L}{\Lambda(\tau_U) - \Lambda(\tau_L)} \, . \tag{10}$$

This demonstrates that simulating from distributions, whose integrated hazards is piecewise-

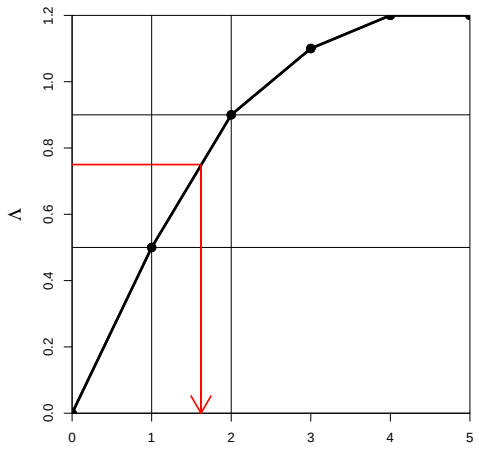


Figure 2: Inverting a piecewise-linear function: First the correct segment is identified (here between $\tau_L = 1$ and $\tau_U = 2$), then the linear function is inverted by basic geometric relations to obtain $t^*$ (here about 1.6, see also equation (10)).

linear, is particularly simple.

To compare the random durations that are obtained by this linear procedure with the ones that would use the more accurate inversion techniques, we created $L = 1000$ random draws from the distribution characterized by the hazard in Figure 1. The inversion based on the Normal distribution function lead to the relation

$$\Phi(t^*; \sigma) = (0.5 + v^*/c) \text{ or } t^* = \Phi_\sigma^{-1}(0.5 + v^*/c)\,. \tag{11}$$

Here $\Phi_\sigma^{-1}$ denotes the quantile function of the Normal distribution with mean zero and standard deviation $\sigma$, which is readily available in R. Figure 3 compares the two ways of simulating. The maximal relative error among the 1000 replications was 0.017.

# 4 Software design

Although the software developed within MIC intends to provide the future user with tools that

- allow to prepare data and estimate transition rates (the *pre-processor*),
- to perform the actual micro-simulation, that is create the individual life-courses (theMIC-*core*), and
- to evaluate and analyse the output (the *post-processor*),

we will keep these three parts strictly separate. This implies that the actual micro-simulation, which will be performed in the MIC-core (see section 4.1), will take as only input the

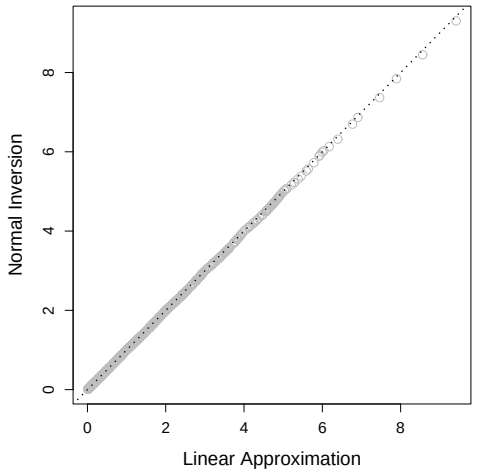


Figure 3: $L = 1000$ random durations corresponding to $\Lambda$ from Figure 1, sampled by linear approximation or Normal inversion.

state-space of the model, a specified starting population, and tables (currently in ASCII format) that contain the values of the corresponding integrated transition intensities for the respective transitions, in a linear approximation as described in section 3.4. Depending on whether the transition rates depend on age, time, and/or duration, these tables may have a one-, two- or three-dimensional structure. No context-related information will be transferred to the MIC-core and all relevant model assumptions will have to be built into the transition rates. This strict separation is in line with the general objective of the MicMac-project, namely, to develop a methodology for detailed demographic projection that does not hinge on any specific application context.

The pre-processor will facilitate data handling and statistical model building to estimate transition rates from data. It will also provide functions that transform estimated transition rates into the format needed by the MIC-core. However, the user will be free to simply provide state-space and transition rates in the required format without the pre-processor.

In a similar way, all output from the micro-simulation will be stored without any further aggregation, that is individual sequences of states and the duration between transitions. The current plan is to store this information in a database.

The post-processor can be used to retrieve and aggregate information from the database. It will also provide tools for some standard analyses. User may, however, decide to access and process the Mic-output by other software.

As a general principle, the MicMac-project intends to be as little restrictive as possible with respect to software licenses, operating systems, and the final product should be computationally efficient and transparent at all levels. Therefore the software to be

produced by MicMac will use free software. Specifically, for the pre- and post-processor, where the user can interact with the core of the micro-simulation model, Mic will use R, which is available as Free Software under the terms of the Free Software Foundation's GNU General Public License.

## 4.1 The MIC-core

To make microsimulation an attractive tool for population projections, the software that takes care of the Monte Carlo simulation has to be extremely efficient so that large-scale problems can be processed in acceptable time. The software should be portable to different operating systems and should, if possible, offer the option of distributed computing.

To take advantage of up-to-date simulation technology, the MPIDR currently collaborates with researchers of the Modelling and Simulation Research Group, Institute of Computer Science, University Rostock, under the leadership of Prof. A. Uhrmacher. This group is developing the project *JAMES II* – a Java-based Agent Modelling Environment for Simulation.

The simulation system *JAMES II* has been designed for supporting a large variety of simulation algorithms for different modelling formalisms. It has a modular system design and therefore easy to extend. It is using a plugin based scheme (for a description (see [3]) and allows parallel and distributed execution, if necessary (see [2]).

The competing risks setting as described in section 3.3, by using piece-wise linear integrated intensities as in section 3.4, has already been successfully implemeted in Java and incorporated into *JAMES II.* The database link, to store the microsimulation output, has also been implemented already.

## 4.2 Additional aspects: Linked lives

As long as independent individuals are considered, a starting population and a specification of the transition rates is all that is needed to perform the Monte Carlo simulation. New individuals, resulting from births to 'old' individuals, can automatically be created and their life-course be simulated onwards.

A more complex problem is the area of union formation. If transitions into marriage (or cohabitation) should not only change an individual's state to 'married' and leave transition rates unchanged otherwise, then, realistically, some attributes of the partner, i.e. the state information of another life course, will have to modify transition rates. This aspect of linked lives is somewhat outside the narrow range of multi-state models.

To make this a realistic process some kind of mate matching strategy will have to be implemented. Currently algorithms based on compatibility indexes combined with

realistic search patterns are explored for their feasibility. Again data of sufficient detail would be required to base these algorithm on empirical patterns.

# References


[1] Per Kragh Andersen, Steen Abildstrom, and Susanne Rosthoj. Competing risks as multi-state model. *Statistical Methods in Medical Research*, 11:203–215, 2002.

[2] Jan Himmelspach, Roland Ewald, Stefan Leye, and Adelinde M Uhrmacher. Parallel and distributed simulation of parallel devs models. In *Proceedings of the SpringSim '07, DEVS Integrative M&S Symposium*, pages 249–256. SCS, 2007.

[3] Jan Himmelspach and Adelinde M Uhrmacher. Plug'n simulate. In *Proceedings of the 40th Annual Simulation Symposium*, pages 137–143. IEEE Computer Society, 2007.

[4] Tony Lancaster. *The Econometric Analysis of Transition Data*. Econometric Society Monographs. Cambridge Unversity Press, 1990.

[5] Frans Willekens. Biographic forecasting: Bridging the micro-macro gap in population forecasting. *New Zealand Population Review*, 3:77–124, 2005.

[6] Frans Willekens. Multistate model for biographic analysis and projection. Technical report, NIDI, 2006.