MODELLING AND SIMULATION E-LEARNING SET OF HYDRAULIC MODELS

M. Bicher[(a)], U. Fitsch[(b)], M. Atanasijević-Kunc [(c)] , N. Nagele[(d)] , W. Wild [(e)] , F. Breitenecker [(f)]

[(a),(b),(d),(e),(f)] Vienna University of Technology, Institute of Analysis and Scientific Computing, Austria
[(c)] University of Ljubljana, Faculty of Electrical Engineering, Slovenia

[(a)]martin.bicher@tuwien.ac.at, [(b)]ursula.fitsch@tuwien.ac.at, [(c)]maja.atanasijevic@fe.uni-lj.si, [(d)] nagele@asc.tuwien.ac.at, [(e)] wild@asc.tuwien.ac.at, [(f)]fbreiten@osiris.tuwien.ac.at

ABSTRACT

As providing e-learning opportunities for students is getting more and more important, the so called MMT system provides an user-friendly online modelling and simulation laboratory. Here a course consisting of a set of experiments is shown to give an example how e-learning can be used in a modern way. A series of tank experiments is used, each of them with different difficulties and increasing complexity, to teach modelling and simulation in fluid dynamics.

Keywords: fluid dynamics, e-learning, MMT, hydraulic, MATLAB, tank

1. INTRODUCTION

Teaching modelling and simulation especially to students who are not permanently confronted with the topic is always a challenge. On the one hand, reasonable models are often too difficult to present them in detail, either for mathematical reasons, lack of programming skills or timing problems. On the other hand presenting a simulation for simplified problems, which cannot be used directly in reality, causes students to suggest, that modelling is a rather theoretical part of science and underestimate the value of good simulations and the complexity and dangers involved.

Thus as a part of the Bologna study a collaborating group of researchers from the University of Ljubljana and the Vienna University of Technology assembled a constructive, well organized e-learning course consisting of a set of high quality simulation models. The course was assembled taking into account the following main ideas:

1. The complexity of the examples is increasing during the course.
2. The examples are strongly related and constructive.
3. Each example deals with different important questions about modelling and simulation.
4. The course can be taught to different fields of study.
5. The examples are available to each student taking part in the course on an internet platform.
6. Students can experiment with the models at home.
7. All students interested in this topic can take part in the course independent from their already existing programming skills.

Especially point 6 and 7 are usually inconsistent or at least hard to manage, so therefore the Vienna University of Technology uses the so called MMT system for teaching, testing and e-learning purposes.

2. MMT SYSTEM

The MMT system - “Mathematics, Modelling and Tools” - is a MATLAB based online simulation platform which provides a user-friendly environment on the one hand for lecturers, who need to present simulation examples in their course, on the other hand for students, who want to experiment with well implemented models at home.

The platform was created in a collaboration between the Drahtwarenhandlung (DWH) and the Vienna University of Technology for many different purposes:

- As an e-learning opportunity for students
- As a virtual modelling and simulation laboratory
- To extend MATLAB, SIMULINK and even JAVA and ANYLOGIC programming skills by downloading and modifying source codes
- As an environment for presentations and lectures
- As a supporting platform for tests and exams
- As a place where well implemented simulation examples by advanced students e.g. for their bachelor or diploma thesis finally get to a good use.

So far about 300 different modelling and simulation examples have been loaded up to the server and the number is steadily increasing. Therefor it is getting more and more difficult for lecturers to find the most appropriate assembly of examples for their course. Thus the upload of a single model or an idealess mix of simulation examples to the server is not enough. A well structured educational aim, e.g. as the one we defined in

the last section, is required to help lecturers taking the correct choice without getting intimidated by the whole variety of the MMT server.
The following set of models shall give an example of how a course can be arranged according to the above mentioned ideas and how the MMT system can be used to assist the lecturer.

3. THE THREE TANK MODEL

The process shown in Figure 1, is located at the Faculty of Electrical Engineering, University of Ljubljana, and consists of three cylindrical plexiglas tanks connected with pipes which can smoothly be closed or opened by valves.

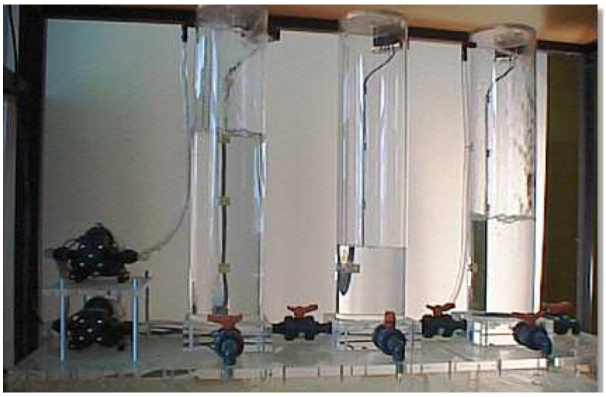


Figure 1: Photograph Of The Process

When water is pumped into the first or the third of the three tanks, assuming that at least those two valves necessary for the flux between the tanks are open, the system complies to the laws of communicating vessels and is thus highly non-linear. Regarding that a model of the whole process is very complex and might be too difficult to explain it is useful to separate the problem.

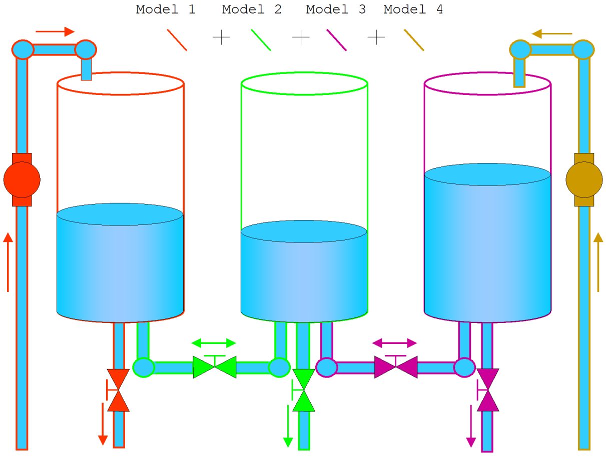
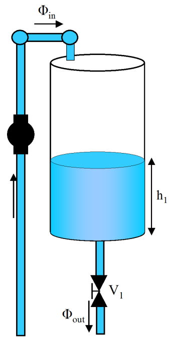


Figure 2: Partition Of The Model

As indicated in Figure 2 the system can be partitioned, hence providing a better understanding of the different subsystems of the tank. The different models discussed later on in the paper are marked in different colours.

3.1. One Tank Model



The easiest model to deal with is defined if the valve, connecting tank one and two, is closed. Though this is a simple first order system it is still non-linear and thus surely a challenge to students inexperienced in modelling. A dynamic model can be found if the relation between the flux into the tank to the derivation of volume and the relation between filling level to the speed of the streaming out liquid are combined:

$$\Phi_{in}(t) - \Phi_{out}(t) = \frac{dV(t)}{dt} = S\frac{dh_1(t)}{dt}$$

$$C\sqrt{2gh_1(t)} = v_{out} = \frac{1}{A}\Phi_{out}(t)$$

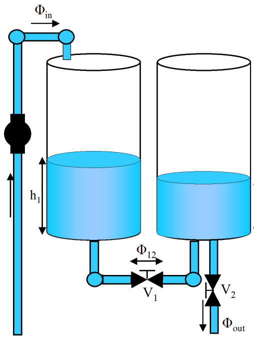
In these formulas S donates the cross section area of the cylindrical tank g defines the gravitational acceleration and A donates the cross section area of the valve. The second formula is called Torricelli's Law for water streaming out of a vessel and is a consequence of the famous Bernoulli equation for fluid dynamics. The constant C (close to one) depends on the liquid and on the form of the opening. The resulting non-linear first order differential equation is:

$$\frac{dh_1(t)}{dt} = \frac{\Phi_{in}(t) - C_{V_1}\sqrt{h_1(t)}}{S}, C_{V_1} = AC\sqrt{2g}$$

The following topics are interesting to deal with in a lecture:

- Physical derivation (Torricelli's Law)
- Valve opening ↔ flux of water (empty tank ↔ spilling over)
- Linearised model
- Comparison with MATLAB's own *linmod()* function
- Transfer function modelling
- Experimenting equilibrium states with the MMT system
- Experimenting with different kinds of input functions (unsteady, steady, smooth) to the pump
- Comparison to real measured data from the process in Ljubljana (Experiment parameters of the simulation at the MMT system)
- Comparison between measured parameters and theoretical calculated parameters (friction, turbulences, incompressible liquid)
- Controlling e.g. by PID (and other) controller

3.2. Two Tank Model



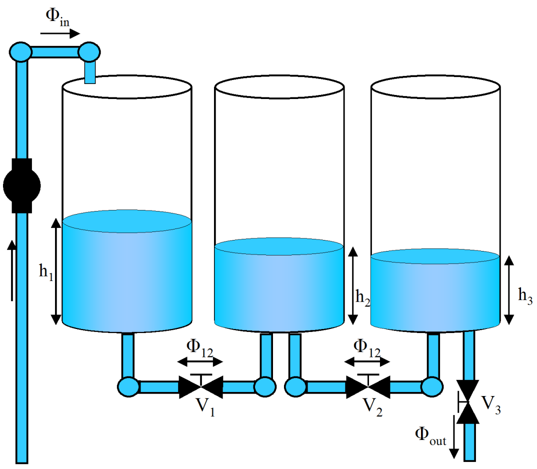
Adding the second tank leads to a second order system which raises additional issues. First of all linearisation leads to a two dimensional Taylor series expansion which is a really good repetition to students lacking of basic math skills. Also it becomes a little bit more difficult for students to determine steady states, on the one hand due to an unsteady signum function and on the other hand due to the definition of the working point regarding the degrees of freedom. From the physical point of view the determination of the differential equations needs a little bit extension because the flux between the two vessels has to be calculated. The problem can be solved using Bernoulli's equation. This leads to the following system of coupled differential equations:

$$\frac{d}{dt}\begin{pmatrix} h_1 \\ h_2 \end{pmatrix} = \begin{pmatrix} \dfrac{\Phi_{in} - \operatorname{sgn}(h_1 - h_2) C_{V_1}\sqrt{|h_1 - h_2|}}{S} \\ \dfrac{\operatorname{sgn}(h_1 - h_2) C_{V_1}\sqrt{|h_1 - h_2|} - C_{V_2}\sqrt{h_2}}{S} \end{pmatrix}$$

Because liquid can pass the valve in both directions the signum function is important here. The following points are taken into account.

- Physical derivation of flux between communicating vessels (Bernoulli's principle, Torricelli's law)
- Calculation of linearised models in form of state-space and transfer function models
- Experimenting with valve openings at the MMT system
- Tuning of the controller
- Difference between SISO and MIMO problems

3.3. Three Tank Models – SISO



Adding the third tank leads to a third order system but is still a SISO problem. The differential equations are quite similar to the two tank system:

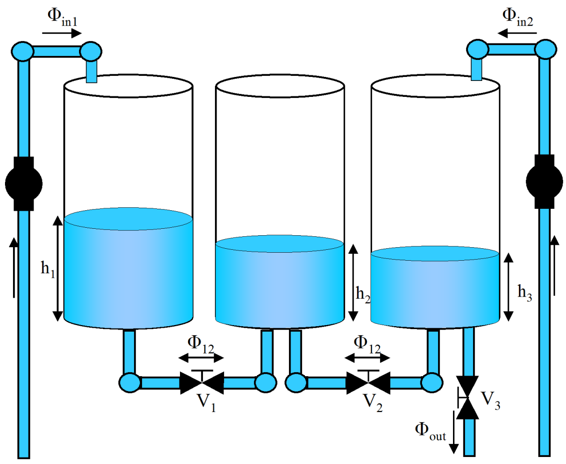
$$\frac{d}{dt}\begin{pmatrix} h_1 \\ h_2 \\ h_3 \end{pmatrix} = \begin{pmatrix} \dfrac{\Phi_{in} - C_{V_1}\sqrt{|h_1 - h_2|}}{S} \\ \dfrac{C_{V_1}\sqrt{|h_1 - h_2|} - C_{V_2}\sqrt{h_2 - h_3}}{S} \\ \dfrac{C_{V_2}\sqrt{h_2 - h_3} - C_{V_3}\sqrt{h_3}}{S} \end{pmatrix}$$

(The signum functions are not shown here)

Due to the fact that the third equation does not depend on h_1 the extension of the two tank model to this third order model is extremely easy. This model can be a perfect example that regarding the modelling point of view a higher order does not always increase the complexity of the problem as the SIMULINK model of the non-linear model needs only to be extended by an additional loop. But the controller needs to be modified and linearisation is getting slightly more difficult. The following points are dealt with.

- Calculate linearised models
- Modification of the controller
- Why is the system SISO

3.4. Three Tank Model – MIMO



In the fourth model the second pump is activated and causes a second input parameter. As a result the system suddenly becomes MIMO and requires completely different controlling. It is important to use and discuss the second degree of freedom which grants that not one but two of the three filling levels can be controlled at once and that there are different options to do so. On the one hand two independent SISO controllers can be used, on the other hand there are several options for really difficult multivariate controllers too. The corresponding differential equation system is:

$$\frac{d}{dt}\begin{pmatrix} h_1 \\ h_2 \\ h_3 \end{pmatrix} = \begin{pmatrix} \dfrac{\Phi_{in1} - C_{V_1}\sqrt{|h_1 - h_2|}}{S} \\ \dfrac{C_{V_1}\sqrt{|h_1 - h_2|} - C_{V_2}\sqrt{h_2 - h_3}}{S} \\ \dfrac{\Phi_{in2} + C_{V_2}\sqrt{h_2 - h_3} - C_{V_3}\sqrt{h_3}}{S} \end{pmatrix}$$

Model 4 is the most frequently used and simulated experiment also in Ljubljana. Therefore exists a great number of real measurement data from the actual plant. So the comparisons of the simulation to the measurements can be performed at the MMT system which can be a real deal for students which are solely used to deal with theoretical models. Discussing and extending Model 4 and the possibilities for controllers is the final target of the course.

- Comparisons between linearised and non-linear model
- Comparison of different controllers at the MMT system
- Create a linearised model
- Comparison to real measured data at the MMT system
- Experimenting with controller parameters

3.5. Model Summary

Each of the four discussed models has its own special properties and raises additional questions. A short summary is given in the following table.

Model	Order	Type	Controller
One Tank	1	SISO	SISO
Two Tank	2	SISO	SISO
Three Tank	3	SISO	SISO
Three Tank + 2nd pump	3	MIMO	MIMO/ SISO

Table 1: Summary Of The Models

4. RESULTS

4.1. Main Pages

The following screenshot (Figure 3) shows the main page of the course at the MMT system.

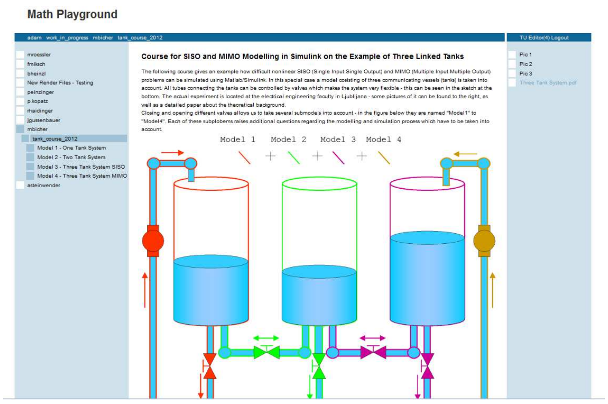


Figure 3: Main Page Of The Course

The area at the left side of the picture respectively of the “Adam-Riese Math Playground”, as the MMT server is called, shows the links to the four model sections with each containing several targeted examples dealing with the already discussed points. It can be seen here, that the MMT system is built up strictly hierarchically using a structure similar to LaTex:

1. Book
2. Chapter
3. Section
4. Subsection

In this case the whole course constitutes a chapter and the four sub-models are created as sections with each containing several subsection examples.

The centre area of the page contains a first raw explanation for the main topic and the goal of the course and the well known sketch of the three tanks. Due to a LaTex environment provided by the MMT server also formulas can be included into this part of the page (see Figure 5). As the graphical environment of the page is detailed and colourful, it can also be used instead of slides for an introduction to the course.

The right hand area of the page represents a download section. Files like slides, PDF-files, pictures etc. can additionally be loaded up to the server and offered for download at this area. The links at the main page of this course offer three photographs of the original plant in Ljubljana and a paper containing detailed theoretical information about the system and how to derivate a simulation model.

Using the links in the left hand area lead to the main pages of each sub-model containing more detailed information about each experiment like a sketch of the physical derivation and a short introduction to the different subsection examples.

4.2. Linearisation Examples

Understanding the usage and creation of linearised models poses an important aim of the course. Often linearisation and especially the role of working point and equilibrium state are misinterpreted and used in a wrong way (a correct interpretation can be found in Figure 4).

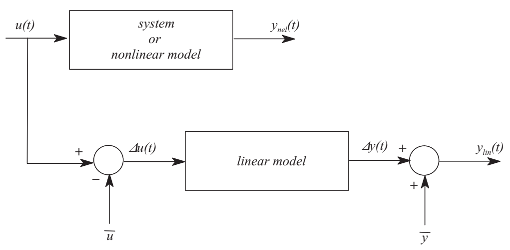


Figure 4: Interpretation Of A Linearised Model

In case of the three tank system, in which all of the differential equations have square-root characteristics and linearisation is not possible or, at least, very bad close to the origin, this is even more important. Figure 5 shows an example of how the process deriving a linearised model could be trained at the example of the one tank system.

The example deals with the calculation of the correct parameters for the linear state space model at a certain working point h_0:

$$\frac{dh}{dt} = \frac{\Phi_{in} - C\sqrt{h}}{S} \qquad \Rightarrow \qquad \dot{x} = Ax + Bu$$

So first of all the equilibrium state has to be calculated and afterwards, Taylor series expansion leads to A and B. The calculated parameters can be compared to MATLAB's own *linmod()* function.

The function *[A,B,C,D]=linmod(Mdl, Wp, Displ)* calculates the linearised State Space model

$$\dot{x} = Ax + Bu$$

$$y = Cx + Du$$

out of the, usually non-linear, model *Mdl* at working point *Wp* with input displacement *Displ*.

As the experiment requires to calculate the correct parameters the user has to put them into the textboxes and press the "ok" button. Doing this, MATLAB is started and the content of the boxes is used as input parameters for a MATLAB function. This function starts SIMULINK, deals with the results and performs the final plots. In this case (Figure 5) the choice for the three input parameters was, taking the output curves compared to the *linmod()* function into account, not perfectly correct.

As the improvement of MATLAB and SIMULINK programming skills is also a target of the MMT server usage, the links to all source files can be found at the right hand side. As all MATLAB codes also the SIMULINK models can be downloaded in text form (which can easily be converted to the classic .mdl-surface again by just saving the file as "filename".mdl).

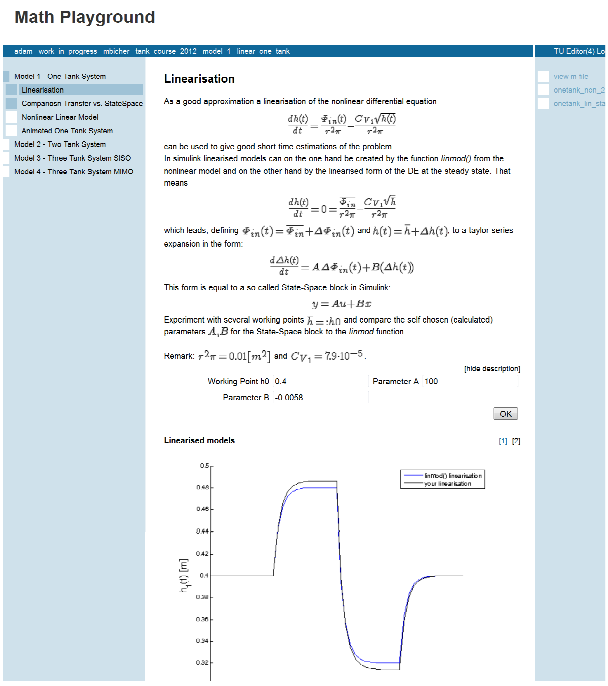


Figure 5: Linearised One Tank model

At the right hand side directly above the figure of the two graphs, a small "<1>" and "<2>" button indicates that there is a second figure available. Switching to the second figure, the graph for the input function (flux into the tank), in this case, a sum of rectangular functions, can be seen. So the MMT server supports multiple output figures represented by a slideshow.

4.3. Non-linear examples

In an example of section "Model 4" several input functions can be tested and the results of the non-linear model is compared to the results of the linearisation. (Indicated in Figure 6)

As there is a choice between a sine-cosine wave function, a sum of rectangular functions (shown in the plot in Figure 6), a sum of sigmoid functions and a pulse of triangular functions, the user can compare the results of steady and unsteady input functions. Because the working points in tank 1 and tank 3 can manually be chosen by the user, comparisons between the non-linear model and the linearisation can also be performed. The plot in Figure 6 shows the results of the non-linear and the linearised model if a sum of rectangular input functions, respectively an unsteady changing but otherwise constant influx into both tanks (one and three), is used.

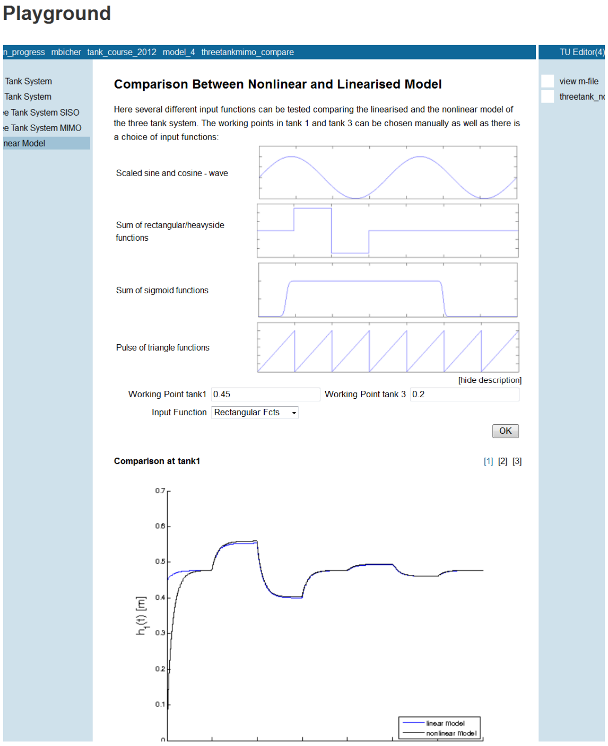


Figure 6: Comparison Between Non-Linear And Linear Three Tank Model Simulation At Tank 1

4.4. Animated examples

As an additional bonus to the graphical plots, the MMT server also supports animated output figures created in MATLAB. In case of the tank system, the scientific value of these examples might not be very high, but they can be used to encourage students and improve the quality of lectures. An example is shown in Figure 7.

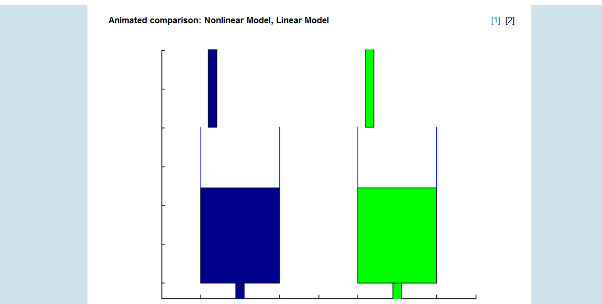


Figure 7: Animated Comparison Between Linear And Non-Linear One-Tank Model

In the animation the water-levels change proportional to the calculated simulation results. A series of plots is used to create a so called animated-".gif" by a function supported directly in MATLAB.

SUMMARY AND FURTHER DEVELOPMENT

The assembled set of models shall give an example of how flexible the MMT server is used in modelling and simulation lectures as well as in tests or exams.

But yet many developments extending the course are planned too. On the one hand a whole variety of multivariate controllers opens up a very difficult topic for advanced students e.g. examples dealing with parameterisation of different MIMO controllers and comparisons between them. On the other hand, the University of Ljubljana does studies dealing with remote control of real experiments. Thus the MMT could be used to collect measured data from Ljubljana which could directly be used for model comparisons at the server.

However it has to be kept in mind that none of the mentioned points necessarily require MATLAB or SIMULINK programming skills, hence encouraging students to deal with MATLAB and SIMULINK as the MMT system provides great flexibility. This work has been realized in the context of the 'Applied Modelling Simluation And Decision Making' project and funded by means of the City of Vienna by the ZIT GmbH - The Technology Agency of the City of Vienna, a subsidiary of the Vienna Business Agency.

REFERENCES

Atanasijevi´c-Kunc M., Logar V., Karba R., Papi˘c M., Kos, *A. Remote multivariable control design using a competition game*. IEEE Trans. Ed., Feb. 2011, Vol. 54, No. 1, 97-103.

Logar V., Atanasijevi´c-Kunc M., Karba R., Papi˘c M., Bešter J. *E-learning game using virtual and remote laboratory environment.* In: Preprints of the 6th Vienna Conference on Mathematical Modelling, 2009, Vienna University of Technology, Austria. MATHMOD Vienna 09 : proceedings, (ARGESIM Report, no. 35). Vienna: ARGESIM, cop. 2009, 2100-2105.

Körner A., Hafner I, Bicher M., Winkler S., Breitenecker F. *MMT- A Web Environment for Education in Mathematical Modelling and Simulation*, ASIM 2011.

Winkler S., Körner A., Hafner I. MMT - *A web-based e-learning System for Mathematics, Modeling and Simulation using MATLAB*, In: Preprints of EUROSIM Congress, Prague, Czech Republic, 2010, paper 231.

AUTHORS BIOGRAPHY

Martin Bicher, born 12.05.1988, is currently writing his diploma thesis at the Vienna University of Technology about Agent-Based modelling.
He is a member of the ARGESIM Master College and in this position one of the main administrators of the MMT server, responsible for Tuwel (Moodle) based e-learning and math tutor for electrical engineering students. He is currently living in Vienna and this is his second paper.

A MODULAR ARCHITECTURE FOR MODELLING OBESITY IN INHOMOGENEOUS POPULATIONS IN AUSTRIA WITH SYSTEM DYNAMICS – FIRST STEP: A POPULATIONMODEL AND HOW TO INTEGRATE IT IN A DISEASEMODEL

Barbara Glock[(a)], Patrick Einzinger[(b)], Felix Breitenecker[(c)]

[(a), (c)]Vienna University of Technology, Institute for Analysis and Scientific Computing, Vienna, Austria
[(b)]dwh Simulation Services, Vienna, Austria

[(a)]barbara.glock@drahtwarenhandlung.at, [(b)]patrick.einzinger@drahtwarenhandlung.at, [(c)]felix.breitenecker@tuwien.ac.at

ABSTRACT
When it comes to modeling diseases with inhomogeneous populations a modular setup can be useful, because specific parts of the model can be easily exchanged or altered. We propose a modular setup consisting of a population model and a separate disease model, which interact on a specifically defined interface. After that an economic cost model can be built and integrated into the modular setup. Aim of this work is to build a population model, in this case with Austrian data, and to define the interface with the disease model, in this case an obesity model. Obesity has become a major problem in Austria. According to (Rieder 2006), over 30% of the Austrian population is overweight and over 9% is obese, resulting in over 227.7 million euro health care costs in 2004.

Keywords: Modular Modeling, Obesity, System Dynamics, Population Model

1. INTRODUCTION

Obesity, defined as the presence of excess adipose tissue, is a health concern of paramount importance in Austria and also worldwide. Next to the long list of co-morbidities associated with obesity, like pulmonary disease, coronary heart disease, diabetes mellitus type 2, orthopedic problems, hypertension, etc. obesity is a great cost factor for health care systems because of weight reduction medications, hospitalizations, laboratory costs, inability to work, loss of productivity, early death, pain and reduced quality of life (Dieterle 2006).

A System Dynamics model for the prevalence of obesity in Austria is needed to show the trend of the prevalence of obesity in the next few years and decades, to identify the most important influences on the disease and to test possible political strategies preventing and treating obesity. The usage of System Dynamics is due to a better overview for non-experts and easier administration for the modeler.

Because of the fact that the model will show the prevalence of obesity in the population over at least the next 50 years, a population model, predicting the change of the Austrian population dependent on births, deaths and migration is necessary. How this affects the population is seen in figure 1. In Austria especially migration is very important, because without it the population would be reduced significantly due to low birth rates.

Figure 1: The Growth Of The Population Is Dependent On Births And Immigration. The Contraction Of The Population is Dependent On Deaths And Emigration.

In this paper we demonstrate why those parts of the model, especially the population model and the disease model, should be independent and therefore a modular setup, as seen in figure 2, is important and how it is done, first by explaining the structure of the disease model and in more detail the structure of the population model and by defining the points of intersection of those two models. Advantages and disadvantages of this modular setup will be discussed too.

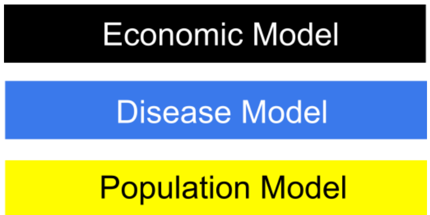


Figure 2: A Modular Setup Of the Simulation Model. An Independent Population Model Is The Basis, Connected Through Defined Points Of Intersection With The Disease Model. Furthermore An Economic Model Can Be Built Individually Based On The Disease Model.

2. THE METHOD: SYSTEM DYNAMICS

The modeling method used for all parts of the whole model is called System Dynamics and was developed in 1956 by Jay W. Forrester (Forrester 1961). It is a top down modeling method, where aggregated states are looked at and not individuals. For example all male persons aged 15-45 could form a single state, but also all male overweight persons aged 15-45. System Dynamics is a commonly used approach to understand the behavior of complex systems. It is different to other approaches because it allows the usage of feedback loops, which, due to the fact that this is a very descriptive modeling method - easily to be overviewed by non-experts - can be better understood and researched. Sometimes a specific state of the system has an effect (feedback) on itself over (a long) time, which is not recognized easily (Sterman 2000).

Especially the disease model for the prevalence of obesity requires feedback loops, to represent the main influence factors on obesity and how they are influenced in return. An example of a possible reinforcing feedback loop, represented in a *causal loop diagram*, can be seen in figure 3.

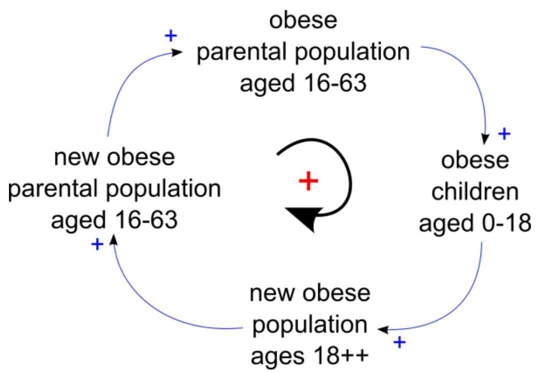


Figure 3: A Causal Loop Diagram Of A Simple Feedback Loop Of the Disease Model.

It shows that the fraction of the obese population, who are parents, has a positive feedback on itself over time: The more obese parents there are, the more obese children they will get, by passing their eating habits on to their children. Therefore the fraction of the obese adult population will increase and also the fraction of the obese parents. Linkages are shown by the blue arrows. A plus-sign represents a proportional effect from quantity at the arrows end to the quantity of the arrows top, as described before.

These causal loop diagrams are a useful tool for developing a model. A System Dynamics model consists of three main elements, described as following and also seen in figure 4:

- *Stocks* describe the state of the system in each time step. They are represented as rectangles.
- *Parameters* are constants.
- *Auxiliaries* are mostly used for mathematical formulas combining stocks and parameters. Special auxiliaries are *flows*, which describe the changes of stocks. Flows are only allowed between two stocks or from a *source* (represents systems of levels outside the boundary of the model) to a stock or from a stock into a *sink* (where flows terminate outside of the system)

Furthermore most System Dynamics Simulators provide a great number of tools, like delay functions, table functions etc. to allow an easier administration of the modelling process.

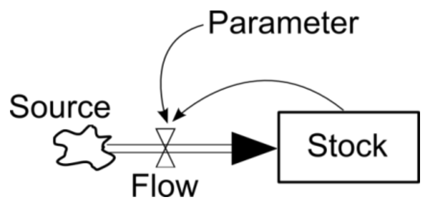


Figure 4: A Simple System Dynamics Stock And Flow Structure With One Parameter And a Source. The Stock Could Represent The Population And The Inflow Births. The Parameter Could Be The Birthrate Depended On The Actual Size Of The Population.

Mathematically System Dynamics models are systems of differential equations with a given simulation time t and initial conditions for the simulation start at t_0. Each stock represents a single equation. The mathematical formulas for the simple stock and flow structure for time t from figure 4 are shown in equation 1 and 2:

$$Stock(t) = \int Flow(t)\ dt \quad + \quad Stock(t_0) \qquad (1)$$

$$Flow(t) = Parameter \cdot Stock(t-1) \qquad (2)$$

The integral for the stock from equation 1 can be transformed into a differential equation, describing the net change of the stock as seen in equation 3.

$$\dot{Stock}\ (\mathrm{t}) = +\ Flow\ (\mathrm{t}) \qquad (3)$$

When building a system dynamics model, the modeler does not need to know exactly how differential equations work or how they are solved, because the modeling device is graphical and the model can simply be built by dragging and dropping stocks, flows, parameters and auxiliaries onto the main user interface.

3. THE OBESITY DISEASE MODEL

First the main structure of the disease model needs to be discussed shortly. Before starting to build the actual model we specify the research questions as following:

- What are the main factors that influence body weight? As a result: How is the obese (resp. normal-weight and overweight) fraction going to develop in the population?
- What can be changed (after identifying the main influences) and more important what is the effect of those changes in the obese (resp.

normal-weight and overweight) population compared to the costs?

Overweight and obesity are defined as abnormal or excessive fat accumulation that may impair health (WHO 2000). Obesity is measured by the so call *Body Mass Index* (BMI), which is calculated by dividing body mass *m* (in kg) through the square of body height *l* (in meters) as shown in equation 4 and categorized as seen in table 1.

$$BMI = \frac{m}{l^2} \tag{4}$$

Table 1: BMI classification: The BMI is divided into 4 main categories of weight classification for adults (WHO 2000).

Classification	BMI
underweight	<18.50
normal weight	18.50 – 24.99
overweight	25.00 – 29.99
obese	≥ 30.00

Furthermore the changes in body weight, and therefore in the BMI are also dependent on the age and sex of the person. For example the basal metabolic rate is different for different ages and also for different gender. Knowing this, we can start to identify the stocks of the System Dynamics disease model: The population is divided by the severity-degree of the disease, age and sex. The idea of this partitioning seen in figure 5 is adapted from Homer et al. (Homer 2006), with the difference that we do not need 100 one-year age-classes, but 12 aggregated five- to ten-year age-classes (except age 0, which is a one-year age-class because of the high infant mortality), because we want to get a look on the trend in general and the main influences on obesity.

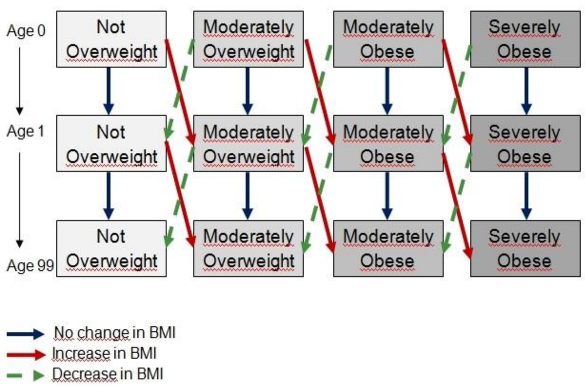


Figure 5: Partitioning Of The Population In The Disease Model (Homer 2006). Persons Of The Same Age, Sex And Severity Degree Of The Disease Build A Separate Stock.

The reasons of crossing over into another BMI-category – which in our model can also be within one age-class, because for example in a 10-year age-class a person can significantly reduce or gain weight – are due to physical activity (caloric expenditure) and eating habits (caloric intake), but they will not be discussed in this paper.

Data is available for the years 1999 and 2006/07 for the partitioning of the population by gender, some age-classes and the four severity degrees of obesity seen in figure 5.

4. THE POPULATION MODEL

As mentioned before the main parts that influence the change of the population are births, deaths and migration. Figure 6 shows the population model as a causal loop diagram. The black loops in the center of such a circle with blue arrows show the polarity of the corresponding feedback loop. A positive feedback portrays a self-reinforcing process, for example: the more people live, the more births will occur and therefore the more people will live in the future. The opposite is a negative feedback loop, or also called a balancing loop, that stabilizes the system. For example: the more people live, the more deaths will occur and the fewer people will live in the future.

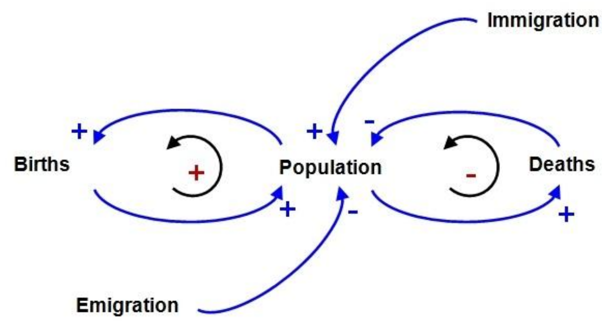


Figure 6: Causal Loop Diagram Of The Population Model Showing The Causal Links And Dependencies Of Variables.

4.1. Available Data

For the partitioning of the population as seen in table 2 data from Statistics Austria (Statistics Austria Database 2012) is available.

The simulation starts in 1999, because for the disease model data is only available since 1999, so the initial partitioning of the population is for this specific year. Due to the fact that migration is not dependent on the population, real data for the years 1999 until 2010 for immigration and emigration (with same partitioning as the population) is also taken from Statistics Austria Database (Statistics Austria Database 2012). The simulation will run 50 years. Therefore the migration data for the next 50 years has to be provided too and a prognosis for immigration and emigration is available in the database, but only for different sexes. To get the partitioning for the age-classes, the distribution of the previous known years is taken, averaged, and assigned to the upcoming years.

Table 2: Partitioning Of The Population By Age And Sex

Age-classes	Sex
Age 0	male
Age 1- 4	female
Age 5 – 9	
Age 10 – 14	
Age 15 – 19	
Age 20 – 24	
Age 25 – 34	
Age 35 – 44	
Age 45 – 54	
Age 55 - 64	
Age 65 – 74	
Age 75+	

The death rate is given for the known years, but for the years from 2010 until 2060 the predicted trend of the life expectancy, which increases by the years and was also taken from Statistics Austria Database (Statistics Austria Database 2012), was used together with a *Weibull* distribution to calculate the death rates for the upcoming years.

Due to the fact that fertility rates did not change significantly over time they were assumed to be constant for the next years as those from 1999. Statistics Austria (Statistics Austria 2011) provides the number of live births per 1000 women per year for the four relevant age-classes in which women can bear a child, namely 15 – 19, 20 – 24, 25 – 34 and 35 – 44. These probabilities *p(i)*, *i=1,2,3,4*, meaning the number of live born singletons divided by 1000 for each of the *i* age-classes, have to be transformed into rates *r(i)* as seen in equation 5.

$$r(i) = \ln\big(1 + p(i)\big) \tag{5}$$

The aging-rates are calculated to 1/*groupsize* for each age-class and sex, where *groupsize* is the length of the interval of the age-class.

Furthermore all rates have to have the same, according to the simulation time step, right dimension, in this case months.

4.2. Implementation of the model

The population model is implemented in Anylogic® University 6.5.0 and can be seen in figure 7.

For a better overview and the fact that similar equations and a lot more arrows are saved to be built AnyLogic provides arrays. As seen in figure 5, there would be 12 (for aggregated age-classes) x 2 (for sex) = 24 stocks for the population, but as seen in figure 7, there is only one stock, called P*opulation*. An array is a container with linear storage of fixed size, called dimension. In this case it is a 2x12 array and holds the 24 entries of the partitioned population. Since the population in general is not the important part of the model, it is “hidden” in one stock and helps non-experts to understand the model better.

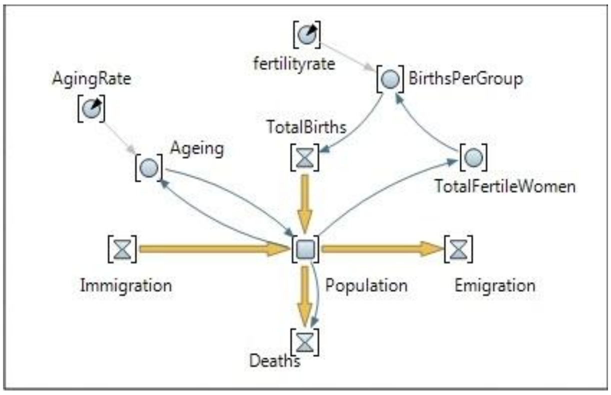


Figure 7: Population Model In Anylogic® University 6.5.0.

Furthermore there are 4 flows, which change the stock (in each dimension of the array). A function provides the input for immigration and emigration and they are not dependent on the population. Two auxiliaries *TotalFertileWomen* and *BirthsPerGroup* are used – also for better overview and understanding, because formally they could be integrated in the equations for *TotalBirths* – in each dimension of the array. *TotalFertileWoman* copies the entries from *Population* that are representative for the fertile women in the previously mentioned four relevant age-classes. *BirthsPerGroup* stores in each time step *t* the number of total births (male and female together) per age-group of fertile women as calculated in equation 6.

$$\begin{aligned} &BirthsPerGroup[FertileWomen](t) \\ &= TotalFertileWomen[FertileWomen]\,(t) \\ &\cdot\ fertilityrate[FertileWomen]\,(\mathrm{t}) \end{aligned} \tag{6}$$

FertileWomen in the brackets represent the dimension of a sub array from the array for population, which in this case is a 4x1 array for females in the four relevant age-classes of fertile women.

The auxiliary *Ageing* stores the number of people (in each index of the dimension of the array) that cross over into the next age-class within the next time step, which is calculated as seen in equation 7.

$$\begin{aligned} &Ageing[Age, Gender](t) \\ &= Population[Age, Gender](t) \\ &\cdot\ AgingRate[Age, Gender](t) \end{aligned} \tag{7}$$

Age represents the dimensions of the 12 age-classes and *Gender* represents the dimensions of the two sexes.

The mathematical formulas behind the stock *Population* are stated in equation 8, 9 and 10. Because of the fact that new born enter the population at age 0, there is a separate equation (9) for the first age-class, including births. And because of the fact that once a person has arrived in the last age-class, he/she stays there or dies, there is s separate equation (10) for the last age-class.

$$\frac{dPopulation[AgeAllBut0\_11, Gender]}{dt} = Immigartion[AgeAllBut0\_11, Gender](t) - Emmigration[AgeAllBut0\_11, Gender](t) - Deaths[AgeAllBut0\_11, Gender](t) + Aging[AgeAllBut0\_11 - 1, Gender](t) - Aging[AgeAllBut0\_11, Gender](t) \quad (8)$$

$$\frac{dPopulation[Age0, Gender]}{dt} = Immigartion[Age0, Gender](t) - Emmigration[Age0, Gender](t) - Deaths[Age0, Gender](t) + Births[Age0, Gender](t) - Aging[Age0, Gender](t) \quad (9)$$

$$\frac{dPopulation[Age11, Gender]}{dt} = Immigartion[Age11, Gender](t) - Emmigration[Age11, Gender](t) - Deaths[Age11, Gender](t) + Births[Age11, Gender](t) - Aging[Age11, Gender](t) \quad (10)$$

The equation for the two flows (one for female newborns entering the population and one for male newborns entering the population) of the *TotalBirths* flow from figure 7 is simply the sum of all births from the *BirthsPerGroup* auxiliary. It is assumed that of all newborns 49% are female and 51% are male.

At last the formula for the flow *Deaths* from figure 7 is calculated as seen in equation 11.

$$Deaths[Age, Gender](t) = Population[Age, Gender](t) \cdot Mortality\_Function[Age, Gender](t) \quad (11)$$

Mortality_Function is a function that represents the time dependent, Weibull-fittet mortality rates as mentioned before and is not seen in figure 7.

5. DEFINING POINTS OF INTERSECTION WITH THE DISEASE MODEL

The changes in the stock of the population model (in each age- and sex-class) are calculated continuously. The disease model has the same partitioning concerning age- and sex-classes (also with arrays for sex and age), but is furthermore divided by severity-degrees of the disease (in this case each BMI-category has a separate stock graphically). Therefore an additional auxiliary *PopulationChange*, which is also an array variable with dimensions *Gender* and *Age* as in the population model, receives the changes in the population from the population model continuously as seen in figure 8. The changes are calculated by the net migration (immigration minus emigration) and the changes within the stock *Population* (Ageing). In the disease model there are flows, as seen in figure 5, from one severity degree to another one, and also from one age-class to the next. The *PopulationChange* affects the flows from one age-class to the next of the disease model, but has to be split up for the different severity degrees for each (Age, Gender)-array-entry. As a result for each severity degree (and (Age, Gender)-array-entry) an additional flow is implemented that calculates the corresponding change by taking into account the current size of the stock and the current size of the part of the population from the population model.

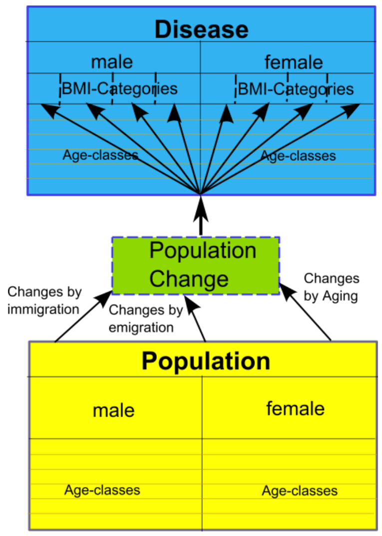


Figure 8: Auxiliary *PopulationChange* is implemented, which has the same structure (dimensions) as *Population*. It stores in each time step the changes of the stocks “hidden” in *Population* and transfers them to the disease model, by splitting them up into BMI-categories.

The equation for the flow *ChangeOverweight*, which calculates the changes within the overweight population (again: the stock *overweight* is an array with dimension *Age* and *Gender* and therefore the calculation is made separately for each entry of this 2x12) is seen in equation 12.

$$ChangeOverweight[Age, Gender](t) = PopulationChange[Age, Gender](t) \cdot \frac{Overweight[Age, Gender](t)}{Population[Age, Gender](t)} \quad (12)$$

The changes for the other BMI-Categories are analogue.

This technique leads to some assumptions, which are discussed in the following.

5.1. Assumptions for the disease model

There are some assumptions concerning the points of intersection:

- Obesity (resp. the other BMI-categories) among persons who enter the system of the population model through immigration (resp. exit the system through emigration) is considered equally distributed as in the main population.
- Obesity (resp. the other BMI-categories) among deaths is considered to be distributed as it is in the main population. Obesity-related deaths are not considered here separately.

5.2. Advantages and disadvantages of a modular setup

Keeping in mind that the model should include a separate cost model for calculating the costs assigned to obesity (e.g. costs for hospitalizations, medications, or even surgery as for example gastric banding), we need to research the conditions (concerning the BMI) on which such medications are prescribed or surgeries are done. The evidence based guidelines of the Austrian society for obesity-surgery (Miller 2005) suggest performing surgery only with a BMI > 40 or with BMI > 35 and additional obesity-related co-morbidities, like hypertension or diabetes mellitus type 2. Therefore the classification of "obese" can be split into 3 more categories as seen in table 3.

Table 3: BMI Classification Of The Obese Category: The Obese Can Be Separated Into 4 More Categories For Adults (WHO 2000).

obese	BMI
obese class I	30.00 – 34.99
obese class II	35.00 – 39.99
obese class III	≥ 40.00

The advantages of a modular set up are, that if one wants to change the disease model afterwards, because of new or altered research questions, only the disease model (but with the same assumptions from chapter 5.1.) needs to be altered. In this case, the obese fraction needs to be split up in three stocks and initialized again and no new mortality rates, fertility rates or migration data have to be calculated again.

Furthermore the population model can be used in another disease model where this partitioning of the population is used. If the partitioning of the age-classes is not needed in so much detail, the classes only have to be aggregated.

For some diseases people develop co-morbidities, which result in higher costs for health care systems. The modular setup allows an easy change or integration of additional stocks for people with co-morbidities without changing the population model.

Furthermore an easier change of the population, by changing the rates and initial conditions, can be done, if one wants to calculate the prevalence, costs etc. for another population.

Tests with the separate parts of the model can be done to look for failures more easily.

A disadvantage of this setup is that disease-related fertility rates and mortality rates cannot be tested or modeled.

6. CONCLUSION

A modular structure can be more understandable, especially for non-mathematicians, when they look at two separate models, one concerning the population and the other one concerning the disease. Secondly the modular parts are easily exchangeable, for example if the disease model shall simulate a different kind of disease, with different severity degrees, but the same population. This increases the reusability of models.

REFERENCES

Dieterle, C., Landgraf, R., 2006. *Folgeerkrankungen und Komplikationen der Adipositas*. Internist (2). 141-149.

Forrester, J., 1961. *Industrial Dynamics*. Cambridge, MA. ISBN: 9780915299881

Homer, J., Milstein, B., Dietz, W., Buchner, D., Majestic, E., 2006. *Obesity Population Dynamics: Exploring Historical growth and Plausible Futures in the U.S.*. 24th International System Dynamics Conference, Nijmegen. Available from: http://www.systemdynamics.org/conferences/2006/proceed/papers/HOMER141.pdf [Accessed 14 May 2012].

Miller, K., 2005. *Chirurgische Therapie der Adipositas. Evidenz Basierte Leitlinien der Österreichischen Gesellschaft für Adipositas-Chirurgie.*

Rieder, A., 2006. *Erster Österreichischer Adipositasbericht 2006.* Altern mit Zukunft. 28-58.

Statistics Austia, 2011. http://www.statistik.at [Accessed December 2011]

Statistics Austria Database, 2012. http://sdb.statistik.at/superwebguest/login.do?guest=guest&db=debevstprog. [Accessed 24 January 2012].

Sterman, J.D., 2000. *Business Dynamics - Systems Thinking and Modeling for a Complex World*. McGraw-Hill Companies. USA.

WHO, 2000. *Obesity: Preventing and Managing the Global Epidemic*. WHO Technical Report Series 894 – Part 1. Geneva. ISBN: 924120894

AUTHORS BIOGRAPHY

Barbara Glock. She was born in Vienna on 5th of March, 1984, started school in 1990 and passed with distinction 2002 from secondary school. After that, she studied Technical Mathematics in Computer Sciences, which she will finish in the end of 2012. Her recent work includes her diploma thesis about a System Dynamics Modeling approach of the prevalence of obesity in Austria.