

## **An Environment for the Development of Operator Training Systems (OTS) from Chemical Engineering Models**

Volker.C. Hass<sup>a)</sup>, Florian Kuhnen<sup>a)</sup>, K.-M. Schoop<sup>b)</sup>

<sup>a)</sup>University of Applied Sciences Bremen  
Neustadtswall 30, 28199 Bremen

<sup>b)</sup>Ingenieurbüro Dr.-Ing. Schoop GmbH  
Riechelmannweg 4, 21109 Hamburg

### **Abstract**

In chemical engineering, process models obtain increasing importance.

There are process models for the optimisation of the process strategy as well as process models for the development of advanced control strategies. Another example of the use of process models is the assessment of potential operational risks, hidden in complex reaction dynamics. These examples of process models may be referred to as engineering type models. Educational type process models represent another type. Both types of models serve specific sets of requirements. However, the frequent use of process models is restricted by the high investments, necessary for their development, combined with a relatively short lifetime.

Reuse of a process model in an operator training system (OTS) is an efficient measure to prolong the lifetime, especially of an engineering type process model.

The reuse task will be largely simplified, by an adequate development environment.

The development environment introduced here is a combination of a coding framework, designed for the rapid design of process models and a commercial process control system. The coding framework provides tools for parameter estimation and verification, while the process control system is equipped with various tools for process control and process visualization. Models, developed in the coding framework are easily incorporated into the process control system, enabling software engineers to quickly generate a most realistic look and feel of the process. In combining the coding framework and the process control software, an elegant way to prolong the lifetime of a process model is established.

**Keywords:** model visualization, model lifetime, operator training

### **1. Introduction**

In a world of increasing economical competition and shortage of resources, the demands on the optimisation of chemical production processes increase. Due to the complexity of the reaction dynamics, the effects of operational settings in general cannot be foreseen.

Process models offer a high potential to access optimisation potentials, hidden in complex reaction dynamics. Additional benefit is given by the fact, that process models allow to simulate a broad range of operational modes allowing to discover potential operational risks. Process models can also be the part of advanced control systems or they can serve as a test and verification tool for the development of control systems.

Nevertheless, the high costs of simulation software, combined with a relatively short lifetime restricts the use of process models more or less to large companies.

Besides the layout and optimisation aspect of process models, an aspect, gaining more and more importance is their employment in an educational environment (Schaich 2003). The educational aspects of process models reach from basic education of fundamental aspects up to operator training for specific production plants.

The constraints and demands of educational type process models largely intersect with the constraints of standalone process models. Consequently, an elegant way of increasing the benefit of standalone process models is their reuse as an operator training simulator.

## **2. Process Model Generation**

Potential requirements of a process model, utilized for the layout or optimisation of a chemical or biochemical production process is the accurate description of the process dynamics for the expected range of operational conditions. The optimisation process affords an adequate possibility of parameterisation of the model as well as the possibility to specify operational conditions. Generation of a process model usually is divided in the three subtasks model formation, model verification and model integration.

Model formation usually is a task of research and paper work. Partial processes of the production process have to be ordered by their importance of the overall process. The kinetics and dynamics of the transformation process and associated reactions have to be determined by literature research and by adequate experiments. Once if the model is stated, the model has to be implemented as a simulation program. This task will be largely simplified and supported by an adequate development environment or coding framework.

This environment should already provide and unify the common tasks of model generation like model parameterisation, result documentation and parameter estimation. Model verification usually is done by comparison of simulation results with experimental data. Model integration means integration of the transformation part of the model in the peripheral parts of a reactor like heating and cooling units substrate supply etc. Model integration will be simplified by the presence of a library of validated building blocks of peripheral units.

## **3. Life Cycle of a Process Model**

During its lifetime, a process model has to at least return the investment supplied in the step of model generation. When used in an early step of process engineering e.g. plant layout, a high amount of flexibility is necessary in order to allow an intuitive approach with even conceptual modifications. If a process model is used in conjunction with a

fine tuning process to find an optimised mode of operation, model precision and accuracy is in the foreground, together with computational speed, when systematic and automate parameter studies are needed.

However, in contrast to a frequent use at the beginning of their lifetime, the use of process models will fade out to the end of their lifetime.

#### **4. Prolongation of Model Lifetime by Transformation into an Operator Training System**

The investments in a process model are high, so that it's reasonable to not only use it in conjunction with process design tasks. If the lifetime of the model can be extended by its use in an operator training system (OTS), a higher return of investment will be achieved and the range of persons who benefit from the process knowledge, represented by the process model, will be extended.

The benefit of an operator training system cannot be over estimated. Besides the education of new staff, OTS can serve a deeper understanding of the process, even of the experienced plant operators and process engineers.

Since the model process is virtual, critical states of the plant or process can be trained, or the anticipation of critical states can be improved.

The set of requirements for an operator training system largely intersects with the set of requirements for an optimisation type process model.

The most important extension of requirements of an operator training system compared to an optimisation type process model is, that it has to be equipped with a realistic user interface. Ideally this interface exactly matches the interface of the process control system, attached to the real production plant.

The suggested approach is to simply import a process model into a process control system. The process model will be implemented in a specialized environment, while the operation of the process model as OTS will take place in an other environment, the process control system itself.

Once if a process model is present, the generation of an operator training system is reduced to interface the internal state variables of the model in a way to represent the measured quantities of the production process. The design of the GUI will then take place in the process control system.

#### **5. Example for the Inclusion of a Process Model in the Process Control System WinErs to generate an Operator Training System**

An example of an operator training system is a Virtual Bioreactor, developed at and distributed by the University of Applied Sciences Bremen.

The underlying process model describes the bio technical production of yeast . The model itself is subdivided into model parts, interacting by mass flow and mass balance conditions. The two main parts of the model consist of the biological model, based on a theoretical description by Bergter and Knorre (1972) and a model of the reaction vessel. The biological part of the model accounts for aerobic and anaerobic growth of yeast, as well as for the Crabtree effect.

The reactor model provides balances for volume, heat and mass balances of the gaseous and liquid phase, including oxygen and carbon dioxide content. Physical properties of matter are represented as temperature dependent, while heat and mass transfer rates mainly respond to stirrer frequency.

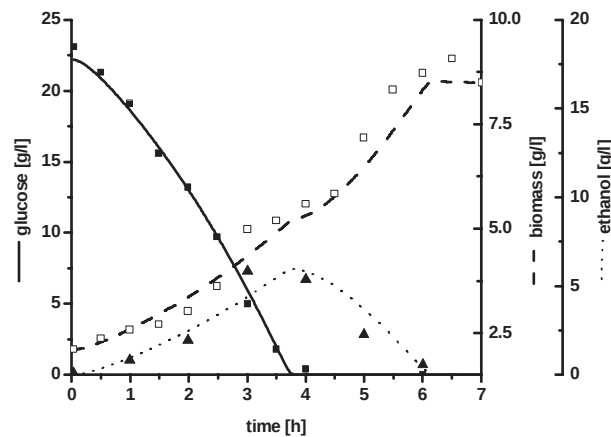


Figure 1. Experimental (dots) and simulated (lines) data of the bio technical yeast production

The model itself is formed, parameterised and tested, using eStIM, a coding framework, specialized for the rapid prototyping of dynamical models.

A process model formed with eStIM can be run in parameter estimation mode, to find parameters by automatically adjusting simulation data to experimental data.

Figure 1 shows how experimental data and simulation data match for the virtual bioreactor.

The user interface which transforms a standalone process model into an operator training system is formed with the help of the process control software WinErs®.

WinErs is a highly flexible modular process automation system. It serves process visualisation and control and it can be used to archive measurements. It is running under Windows 95/98/NT/2000/XP and can be easily learned.

WinErs provides a well defined interface for external DLLs which allow for example to include a process model.

eStIM is equipped with an easy to use mapping mechanism, to export arbitrary simulation quantities from the model to WinErs. This type of job sharing between modelling environment and process control system allows to benefit from the specialisation of both sides.

The coding framework eStIM allows to rapidly design the process model, while the process control software allows to set up the visualisation of the reactor etc.

The GUI of the virtual bioreactor (fig. 2) consists of a simplified graphical representation of the fermentation vessel and peripheral devices as the heater or flasks with feed solutions. Operational quantities can be easily set by clicking to the appropriate input field.

All measured quantities, utilised for process monitoring, are present in numerical form and in graphical form. Offline data like biomass content or substrate concentration are

synchronised with a simulation of a probe taking process which has to be manually invoked.

Since all modern fermentation systems have predefined control loops for sensitive quantities, control loops are established for the virtual bio reactor too. They include temperature, oxygen concentration and antifoam control. The control loops can be accessed via own windows.

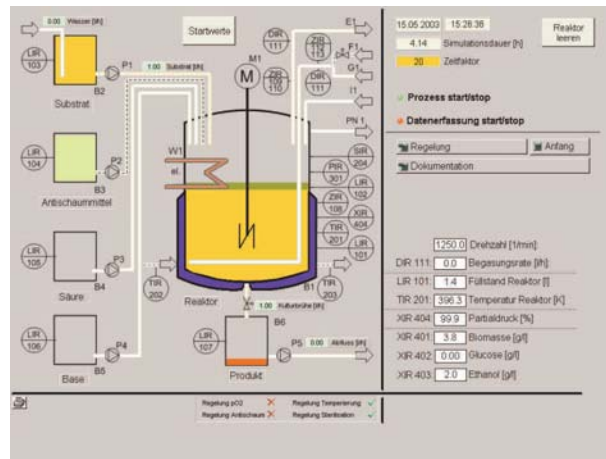


Figure 2. The graphical user interface of the virtual bio reactor

## 6. Teaching Experience with the Virtual Bio Reactor

First teaching experience with the virtual bioreactor at the University of Applied Sciences in Bremen shows promising results.

Without exception, the students accepted the virtual device, and could start to solve their exercises after a short phase of introduction. In addition, they soon began to run experiments on their own in order to explore the dynamics of fermentation.

## 7. Concluding Remarks

The combination of a coding framework and a process control system is a promising attempt for the rapid design of OTS. The effort of adapting the process model to the process control system WinErs is very low. The generation of a GUI, the design of a visualisation and the design of control loops can be easily done in WinErs. One major advantage of this approach is the generation of an OTS in a production environment. Consequently with comparably low investments, the benefit of process models can be largely extended, by transferring them into an operator training system.

## References

- Berger, F. and W.A. Knorre, 1972, Z. f. allg. Mikrobiol. 12, 613-629.  
 Schaich, D. and M. Friedrich, 2003, Automatisierungstechn. Praxis atp, 245 (2), 38-49.

