Data Mining for Bioprocess Optimization: Application of Wavelet Analysis and Decision Trees

Reinhard Guthke, Peter-Jürgen Müller, Heike Rodig, Stephan Lau, Dominik Driesch*
Hans Knöll Institute for Natural Product Research, Beutenbergstr. 11, D-07745 Jena, Germany,
Phone: +49-3641-656820, Fax: +49-3641-656800, email: rguthke@pmail.hki-jena.de
* BioControl Jena GmbH, Wildenbruchstr. 15, D-07745 Jena, Germany,
Phone: +49-3641-675511, Fax: +49-3641-675512, email: biocontrol@t-online.de

ABSTRACT: Process data of 10 fermentation runs of the micro-organism *Streptococcus agalactiae* used for the production of the enzyme hyaluronidase (hyaluronate lyase) were analyzed using the *ProcessAnalyzer* and the *DecisionXpert*, two PlugIns for the software tool *DataEngine*. Attributes determined by wavelet analysis were used for the training and testing of decision trees. These trees were grown in order to predict the process outcome from the growth kinetics observed. Validation results of the trees generated were studied with respect to the mode of classification (i.e. process outcome was classified into two or three classes with low, medium or high product yield) as well as to the splitting mode of the original data set into two subsets for training and testing. The classification mode of the fermentation runs was found to be crucial to the validity of the results.

KEYWORDS: Data Analysis, Knowledge Discovery, Fed-Batch Fermentation, Growth Kinetics

1 INTRODUCTION

Process knowledge described by rules can be used for on-line expert system control. In order to generate such rules from archived data various methods were developed and applied. Some of them are not tree oriented (Guthke, 1992; Krone and Kiendl, 1994; Guthke et al. 1998). However, the most widely used algorithms for the generation of rules are tree oriented (Quinlan, 1986; Quinlan, 1993; Borgelt, 1998). The induction of decision trees will be studied in this paper, too. Features for this learning procedure can be generated by clustering methods, such as the fuzzy-C-means algorithm (Guthke et al., 1998). An alternative method is based on the wavelet decomposition and triangular representation (Daubechies, 1988; Stephanopoulos et al., 1995; Locher et al., 1996).
The paper demonstrates the application of the following three data mining methods

- wavelet decomposition,
- triangular representation and
- learning of decision trees

to the analysis of a fermentation process of hyaluronidase, an enzyme that is used for different therapeutic applications in medicine (dermatology, oncology, etc.).

2 MATERIALS AND METHODS

Experimental materials and methods used during hyaluronidase (hyaluronate lyase) fermentations were described by Rodig (1998) and Guthke et al. (1999). Data analysis focused on potential relations between time series of the growth kinetics $c_X(t)$ and the yield c_P of the product hyaluronidase of 10 fermentation runs as shown in Figure 1.

The software tool *DataEngine 3.0* (MIT GmbH, Aachen, Germany) together with the PlugIns *ProcessAnalyzer* and *DecisionXpert* were used for all calculations. Figure 2 and Table 1 show the modules used. The module *'DXpert Prune'* for the pruning of trees was

used alternatively and linked to the module *'DXpert Grow'*. (This module is not shown in Figure 2 because pruning did not improve the validation results.)

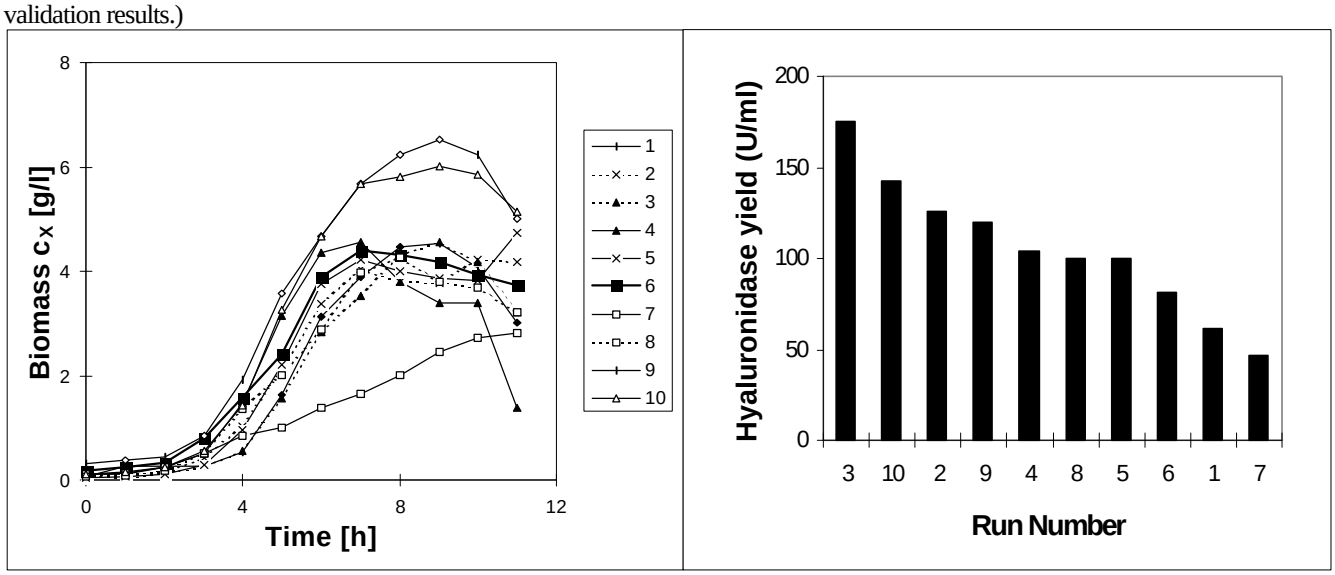
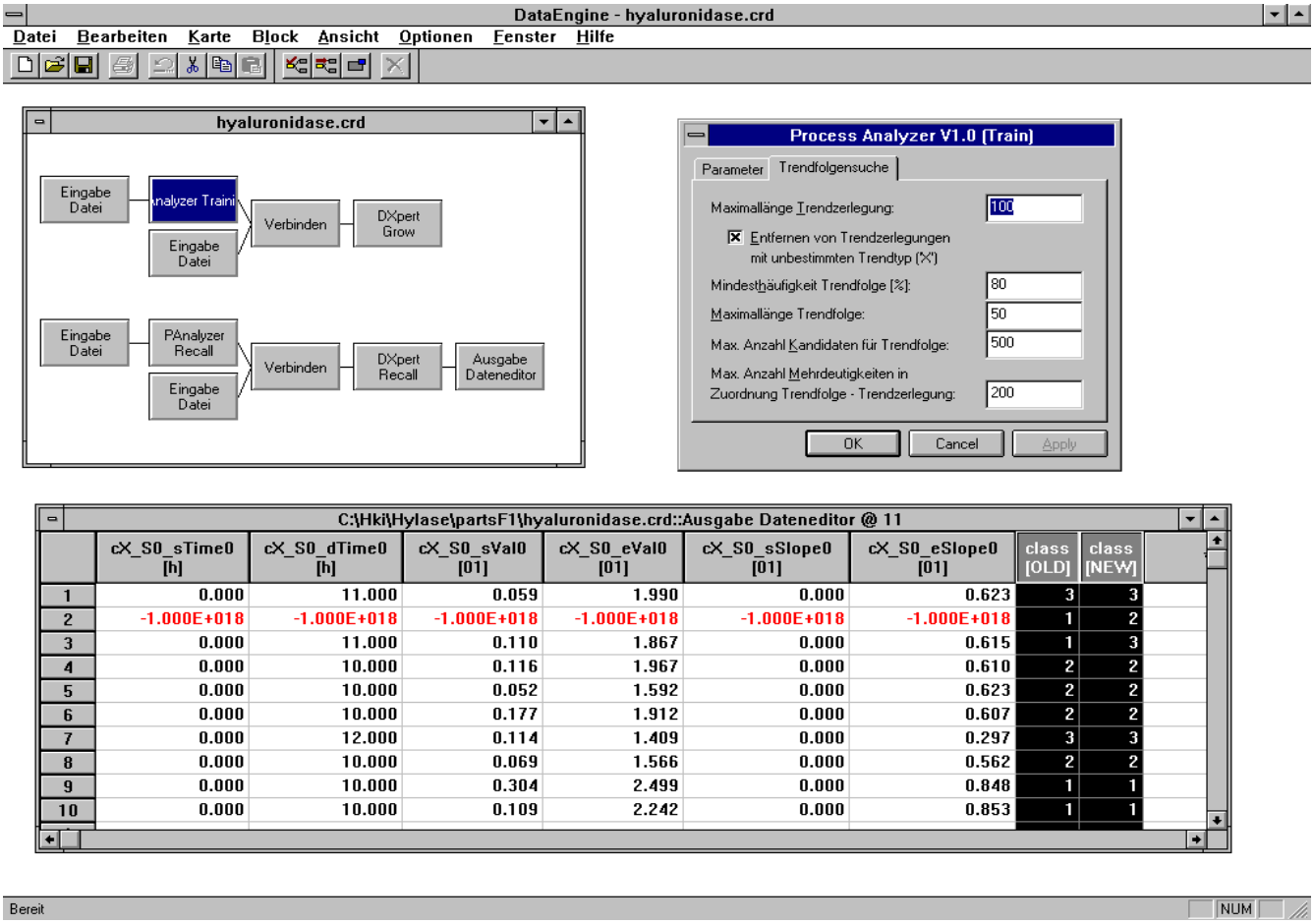


Figure 1: Growth kinetics (left) and product yield (right) of 10 fermentation runs of *Streptococcus agalactiae*



	cX_S0_sTime0 [h]	cX_S0_dTime0 [h]	cX_S0_sVal0 [01]	cX_S0_eVal0 [01]	cX_S0_sSlope0 [01]	cX_S0_eSlope0 [01]	class [OLD]	class [NEW]
1	0.000	11.000	0.059	1.990	0.000	0.623	3	3
2	-1.000E+018	-1.000E+018	-1.000E+018	-1.000E+018	-1.000E+018	-1.000E+018	1	2
3	0.000	11.000	0.110	1.867	0.000	0.615	1	3
4	0.000	10.000	0.116	1.967	0.000	0.610	2	2
5	0.000	10.000	0.052	1.592	0.000	0.623	2	2
6	0.000	10.000	0.177	1.912	0.000	0.607	2	2
7	0.000	12.000	0.114	1.409	0.000	0.297	3	3
8	0.000	10.000	0.069	1.566	0.000	0.562	2	2
9	0.000	10.000	0.304	2.499	0.000	0.848	1	1
10	0.000	10.000	0.109	2.242	0.000	0.853	1	1

Figure 2: *DataEngine* Card: Visualization of modules for training and testing used as described in Table 1; configuration of the *ProcessAnalyzer (Train)* and the resulting file (bottom) containing selected attribute values and class numbers for 10 fermentation runs

DataEngine Module	Function
Eingabe Datei	Input of data sets for training or testing or classification results
PAnalyzer - Train	Wavelet decomposition and trend extraction for training
PAnalyzer - Recall	Calculation of attribute values from test data
DXpert - Grow	Generation of decision trees from training data
DXpert - Recall	Application of the grown decision tree
Ausgabe Dateneditor	Output of results

Table 1: Modules of *DataEngine* and PlugIns used and linked as shown in Figure 2

3 RESULTS

The data set containing 10 hyaluronidase fermentation runs was analyzed in three steps:

(i) Data selection and preprocessing (interpolation of time series c_X , classification of runs with respect to product yield c_P)

(ii) Wavelet decomposition and extraction of trends from time series $c_X(t)$

(iii) Training of decision trees that describe the relation between $c_X(t)$ and c_P and validation of results

3.1 DATA SELECTION AND PREPROCESSING

For the supervised learning of decision trees fermentation runs have to be classified into a number of classes. According to the distribution of product yield as shown in Figure 1, the 10 runs were divided into the following three classes (case I)

- class 1("high product yield"): run 2, 3, 9 and 10
- class 2 ("medium product yield"): run 4, 5, 6 and 8
- class 3 ("low product yield"): run 1 and 7

or into the following two classes (case II)

- class 1("high product yield"): run 2, 3, 9 and 10
- class 2 ("low product yield"): run 1, 4, 5, 6, 7 and 8

Other classification modes were studied too. They will not be discussed here, because validation results did not improve using these modes.

For the cross validation of results the data set was split in a set A for training and a set B for testing. For the training (learning) 7, 8 or 9 runs (set A, m^A=7 or 8 or 9) were taken into account, whereas the remaining runs were used for testing (set B, m^B=3 or 2 or 1). The case m^A=9 and m^B=1 will be discussed in detail. Learning and testing of trees was performed 10 times with 10 different sets A and B, where set B consists of one of the 10 fermentations runs.

Before the application of the software modules the time series $c_X(t)$ were preprocessed by linear interpolation to substitute missing values and to obtain equidistant values.

3.2 WAVELET DECOMPOSITION AND TREND EXTRACTION

From set A of m^A runs k attributes and their values $X^A_{i,j}$ (i=1,...,m^A; j=1,...,k) were generated by wavelet analysis using the module *PAnalyzer - Train* of the PlugIn *Process Analyzer*. The values $X^B_{i,j}$ (i=1,...,m^B; j=1,...,k) of the same k attributes were calculated for set B using the module *PAnalyzer - Recall* of the same PlugIn. Before filtering the software modules add to the 12 (=n) measured values 12 artificial values (labeled 'x' in Figure 3), i.e. 6 (=n/2) data points before the beginning (inoculation at t=0 h) and 6 data points after the end (t=11 h) of the analyzed time interval.

Two filter stages (called *S0* and *S1*) with two trends (0 and 1) were found for the example discussed here with the default configuration of the software modules. For the triangular representation the software modules calculated 6 parameters for each trend called *sTime, dTime, sVal, eVal, sSlope* and *eSlope*. Thus, k=24 values called *S0_sTime0, ..., S1_eSlope1*, i.e. 6 values for the two fiilter stages S0 and S1 as well as for the two trends 0 and 1, were identified for each fermentation run. The 60 parameters for the 10 fermentation runs of the first filter stage S0 and the trend 0 are shown in the Figure 2. The 12 parameters of the two trends of the low pass filtered first signal stage

(S0) are shown in the Table 2. The *ProcessAnalyzer* calculated the time related attributes (*sTime, dTime, sSlope, eSlope*) with respect to the number of sampling intervals. For example, the value *dTime* = 10 calculated for the duration of the first trend as shown in Table 3 corresponds to 10 h which means that the first trend runs from its beginning at t = - 6 h to its end at t = 4 h (see Figure 3, Trend 0). The second trend with a duration of *dTime* = 6 runs from t = 4 h to t = 10 h (see Figure 3, Trend 1).

Value	Trend 0		Trend 1	
sTime	0	[= -6.0 h]	10	[=4 h]
dTime	10	[=10 h]	6	[=6 h]
sVal	0.177	[=0.177 g/l]	1.912	[=1.912 g/l]
eVal	1.912	[=1.912 g/l]	3.939	[=3.939 g/l]
sSlope	0.00	[=0.000 g/l/h]	0.607	[=0.607 g/l/h]
eSlope	0.607	[=0.607 g/l/h]	0.000	[=0.000 g/l/h]

Table 2: Attribute values of the first low pass filter stage (S0) obtained from data of fermentation run 6 by the *ProcessAnalyzer*

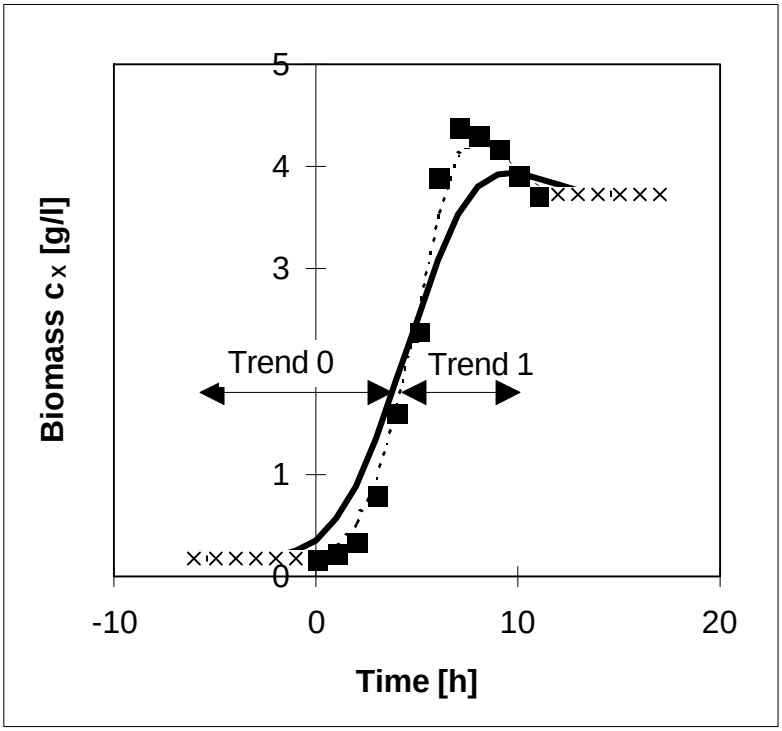


Figure 3: Kinetics of biomass growth of fermentation run 6: measured data (■), added data (×), kinetics of two trends of the first (S0, solid line) and second filter stage (S1, dotted line)

3.3 TRAINING OF DECISION TREES AND VALIDATION OF RESULTS

After classification of case I (i.e. 3 classes, see 3.1) the decision tree shown in Figure 4 with the rules shown in Table 3 was obtained from 7 different data sets A for learning (i.e. from data of all 10 fermentation runs and data of 9 out of the 10 fermentation runs without one of the 6 runs 2, 3, 4, 5, 6 or 8).

After pruning of the decision tree T an identical tree T or trivial trees (without branches) were found depending on the configuration of the module *DXpert Prune*. Using the grown decision tree T for the prediction of the fermentation outcome (product yield), the class was predicted correctly for the 4 fermentation runs 4, 5, 6 and 8 which were not used for training (i.e. run 4, 5, 6 or 8 formed set B) and also for the 4 fermentation runs 1, 7, 9 and 10 which were already used for training (set A). Prediction for the two fermentation runs 2 and 3

however failed. If one of the 4 runs 1, 7, 9 or 10 was excluded from training then two trees T_1 or T_2 were generated that differed from tree T shown in Figure 4. The validation of these trees T_1 and T_2 failed.

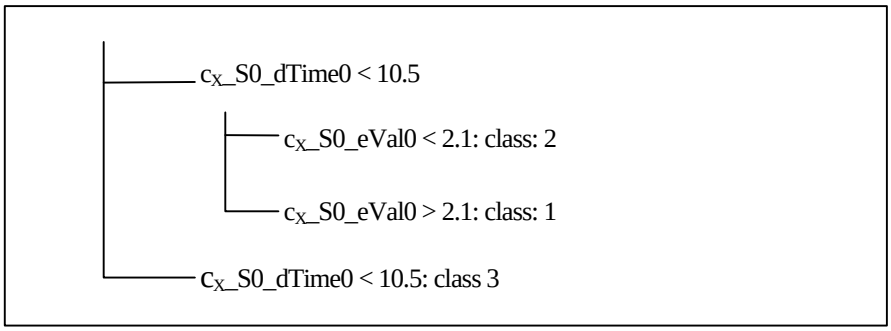


Figure 4: Decision tree T for case I (3 product yield classes considered) identically grown from 7 different data sets (set A containing all or 9 out of 10 fermentation runs without run 2, 3, 4, 5, 6 or 8, respectively)

- IF the first trend of biomass growth finishes before 4.5 h AND the final biomass is higher than 2.1 g/l THEN a high product yield is predicted.
- IF the first trend of biomass growth finishes before 4.5 h AND the final biomass is smaller than 2.1 g/l THEN a medium product yield is predicted.
- IF the first trend of biomass growth finishes after 4.5 h THEN a low product yield is predicted

Table 3: Rules of the decision tree T shown in Figure 4

For case II (with only 2 product yield classes, see 3.1) the validation of the generated decision trees could not be improved: For the 10 different data sets A with a different one of the10 fermentation runs excluded in each case (and used for testing) 10 different trees were generated. One of them was tree T_0 as shown in Figure 5. It predicts classes 1 or 2 for all 10 fermentation runs correctly. This tree was also generated when all 10 runs were included in data set A for training.

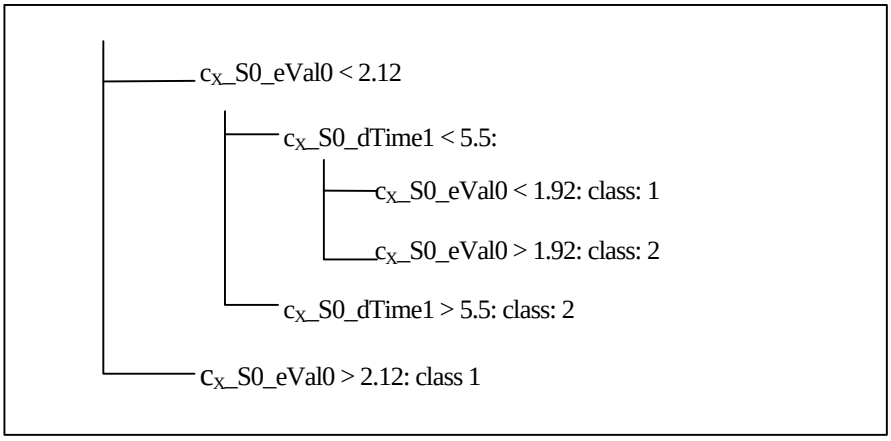


Figure 5: Decision tree T_0 for case II (only 2 product yield classes considered) identically grown from 2 different data sets (set A containing 9 out of 10 fermentation runs without run 6 and set A containing all 10 runs)

4 CONCLUSIONS

The data analysis of a hyaluronidase fermentation process using the PlugIns *ProcessAnalyzer* and *DecisionXpert* together with the software tool *DataEngine* resulted in rule sets which express the relations between growth kinetics and product yield (see Table 3).

These rules can be interpreted physiologically as growth associated product formation. The results may be used for a more detailed modelling and model based experimental design (Guthke et al., 1999; Berkholz et al., 1999).

The trees generated depend on the training data set A. In this respect and in the cases studied here, there was a higher sensitivity (i.e. trees depend more on the training data set A) in case II with only two classes of process outcome compared to case I with three classes: In case II only 3 out of the 11 different data sets A generated the same tree. In case I, however, the trees generated proved more stable: 7 out of the 11 different data sets A studied generated the same tree as shown in Figure 4 and Table 3.

The prediction of the fermentation outcome using the rule set of Table 3 which was generated from 7 different data sets failed for two fermentation runs. However, the tree shown in Figure 5 was found that predicts the outcome of all 10 fermentation runs correctly. For the learning of this tree the data of run number 6 was left out. But this tree was not found when another run than run 6 was excluded from training. Then, trees were grown whose prediction of fermentation outcome failed for one or more runs. Therefore, the information content of runs 1 to 5 and 7 to 10 appears to be essential for a sufficient training .

Thus, the validation of the rule based prediction of the outcome of the hyaluronidase fermentation studied in this paper did not prove entirely satisfactory. The results did not improve when the data of growth kinetics were pre-processed by logarithmic transformation In another case (Guthke et al., 1998) the portion of correct prediction was found to be higher and the obtained decision tree did not depend so much on the input data as shown here. However, the results obtained there were more trivial.

The data of the analyzed time series had to be equidistant. Then, wavelet analysis can filter out noise while retaining distinguishing features. But the number of features generated is so much (six for each trend and each filter stage) that 10 fermentation runs are not enough for sufficient learning.

5 REFFERENCES

Berkholz, R., Guthke, R., Schmidt-Heck, W., 1999, Sequentielle Versuchsplanung zur Optimierung von Bioprozessen mit Produktbildung im Übergangszustand, DECHEMA/GVC-Tagung "Wechselwirkung zwischen Biologie und Prozeßführung", Erfurt, 10.-11.05.1999.

Borgelt, C., 1998, A Decision Tree Plug-In for DataEngine, EUFIT'98, Aachen (Germany), pp. 1299-1303.

Daubechies, I., 1988, Orthonormal Bases of Compactly Supported Wavelets, Comm. Pure Applied Math., Vol. XLI, 909-996.

Guthke, R.,1992, Learning of rules from fermentation data. In: Karim, M.N., Stephanopoulos, G. (Eds.), Modeling and Control of Biotechnical Processes, Colorado (USA), pp. 403-405.

Guthke, R., Schmidt-Heck, W., Pfaff, M., 1998, Knowledge acquisition and knowledge based control in bioprocess engineering, J. Biotechnol. Vol. 65, 37-46.

Guthke, R., Schmidt-Heck, W., Müller, P., Rodig, H., 1999, Data and knowledge based experimental design for bioprocess optimization. Proc. 9th European Congress on Biotechnology, Brussels, Belgium, 11-15 July, 1999.

Krone, A., Kiendl, H.,1994, Automatic generation of positive and negative rules for two-way fuzzy controllers. EUFIT'94, Aachen (Germany), pp. 438-447.

Locher, G., Bakshi, B., Stephanopoulos, G., Stephanopoulos, G., Schügerl, K.,1996, Ein Ansatz zur automatischen Umwandlung von Rohdaten in Regeln. Teil 1+2, Automatisierungstechnik 44, Nummer 2+3, pp. 61-70 + 138-145.

Quinlan, J.R., 1986, Induction of decision trees. Machine Learning, Vol. 1, 81-106.

Quinlan, J.R., 1993, C4.5: Programs for Machine Learning. Morgan Kaufman.

Rodig, H., 1998, Fermentationskinetik und Charakterisierung eines hyaluronsäureabbauenden Enzyms von *Streptococcus agalactiae*. Hans Knoll Institute for Natural Product Research, Jena (Germany).

Stephanopoulos, G., Locher, G., Duff, M., 1995, Pattern Recognition Methods for Fermentation Database Mining. In: Munack, A. and Schügerl, K., Proc. 6th Int. Conf. on Computer Applications in Biotechnology, Garmisch-Partenkirchen (Germany), pp. 195-198.