Pergamon

Computers chem. Engng Vol. 22, Suppl., pp. S587–S593, 1998

Printed in Great Britain
PII: S0098-1354(98)00104-5
0098-1354/98 $19.00 + 0.00

Intelligent Modelling in the Chemical Process Industry with Neural Networks: a Case Study

Kürt Meert and Marcel Rijckaert

Expert Systems Applications Development Group, Department of Chemical Engineering
Katholieke Universiteit Leuven, de Croylaan 46, B-3001 Heverlee, Belgium

Abstract

Nowadays the increasing complexity of most processes increases the demand for performant models. Most of these processes are highly non-linear and dynamic ones, which require complex modelling techniques. Neural networks are eligible modelling candidates for such processes, since they have the ability to map a variety of input-output patterns quite easily. Moreover certain types of networks (the so-called spatio-temporal networks) can not only model spatial but also temporal patterns. Nevertheless a continuous search for improvement is mandatory. Therefore in this paper combinations of spatio-temporal neural network types with other modelling techniques are discussed whilst applied to a complex problem from the chemical process industry, i.e. a polymerisation reactor. © 1998 Elsevier Science Ltd. All rights reserved.

Keywords: Recurrent neural networks, Modelling, Polymerisation reactors.

Introduction

The chemical process industry encounters many problems in controlling and monitoring polymerisation reactors. This has always been a challenging industrial problem due to its complex process dynamics, strong nonlinearities and interaction between process variables. It is therefore an ideal testing field for different kinds of AI techniques. A lot of effort has already been put into these specific problems because of their frequent appearance in the process industry.

One of the major problems that arise during polymer production is the estimation of the average chain length of the polymer. Such an estimation is quite elaborate and time consuming. A MWD (Molecular Weight Distribution) measurement with an on-line size exclusion chromatograph detector causes a time delay of 10 to 15 minutes, which is the main reason why industrial viscosimeters and some empirical correlations are used to estimate the MWD (Ponnuswamy et al., 1987). If the inertia of the process itself (e.g. residence time) is added up to the measurement time delay, one can imagine the consequences a disturbance has on the output of the reactor with such a large total time delay. Furthermore, an erroneous output at the reactor (an unwanted chain length) can cause a lot of losses (production as well as financial). Therefore newer methods for real-time estimation of the chain length distribution are widely developed (Ellis et al, 1994; van Dootingh et al., 1992). In (Dimitratos et al., 1994) a broad review is given on control of polymerisation reactors.

One of the most promising methods for application in the chemical process industry, and more specific to polymerisation reactors, are neural networks. They not only have the ability to model complex, nonlinear processes, but they can also map temporal relationships between input and output parameters. Neural networks which can map spatial as well as temporal relationships are called spatio-temporal networks (Pearlmutter, 1995). These spatio-temporal networks can be divided into two classes, i.e. pseudo spatio-temporal and real spatio-temporal networks (abbreviated spatio-temporal networks). Real spatio-temporal networks, e.g. Elman (Elman, 1990) and the RTRL (Real-Time Recurrent Learning) network (Williams & Zipser, 1989), recycle their activities indefinitely, thus enabling them to capture any temporal phenomenon without requiring some a priori process knowledge. On the other hand pseudo spatio-temporal networks acquire their dynamic mapping abilities by using process specific knowledge (e.g. process time delays) to construct one or other complex architecture, e.g. BP (BackPropagation) networks using a time window (Waibel et al., 1989). Hence their temporal memory capacity is limited, e.g. in the case of a time window, to the length of the time window. This paper focuses on the use of real spatio-temporal networks for modelling a complex problem of the chemical process industry, i.e. a polymerisation reactor. Two types of spatio-temporal networks are applied to this problem, the RTRL network and the MLRN (Multi-Layer Recurrent Network) (Meert & Ludik, 1997). Nevertheless, for referential purposes, a BP network will also be applied to this modelling

problem.

To improve the modelling performance of the spatio-temporal networks, they were introduced into the NMAX model framework (Chen & Billings, 1992). NMAX stands for Nonlinear, Moving Average with exogenous inputs. This means that we have a nonlinear model, where the model error is fed back into the network as an input and which uses some external parameters as inputs. Hence these five network types, i.e. BP, RTRL, MLRN, NMAX-RTRL and NMAX-MLRN (in order of rising complexity), are used to model the polymerisation reactor.

Even for spatio-temporal networks, additional improvements can be obtained by using process specific knowledge. In this paper we make use of the isotime lines of the process. Isotime lines connect points which have an equal time delay. Hence the process parameters (input and output) can be aligned in time. This means, e.g. that an output parameter of the reactor at time t corresponds with an input parameter at time t minus the average residence time of the reactor.

Apart from using these neural models as state estimators, they can be introduced e.g. into predictive control architectures or optimisation schemes. In these cases some additional restrictions have to be imposed on the input parameters, since some of these input parameters cannot be altered independently from other input parameters. Hence the input parameters must be divided into two classes, i.e. controlling and controlled input parameters. Whereas the first class of parameters can be varied arbitrarily, the behaviour of the second class of parameters, like the system's output, depends on the first class of parameters. Moreover the actual values of the controlled parameters are rarely available and they cannot be applied directly as input parameters. However they are generally stored in the plant's historical data base and thus can be used for constructing a neural network model. To incorporate this additional information in the model, a hybrid architecture is developed, which consists of two modules, each of them being a spatio-temporal network. Both modules are explicit models, where the first one predicts the controlled "input" parameters as a function of the controlling "input" parameters and the second one predicts the output parameter as a function of all the input parameters (controlling and predicted controlled input parameters).

In the next part, a brief description is given of the polymer reactor. In section three the modelling results for a variety of network types and architectures are discussed. In the fourth section we conclude with a final discussion.

Polymer production plant description

A continuous stirred tank reactor is used for the free radical polymerisation of methylmethacrylate (MMA). Polymethylmethacrylate (PMMA) is produced in a toluene solvent and benzoyl peroxide (Ponnuswamy et al., 1987) is used as an initiator. Monomer and initiator product are fed into the reactor continuously. The input concentration and flow rate of monomer and initiator can be varied. The reactor temperature can also vary due to a heating jacket. A valve controls the level in the reactor. The average residence time is 15 minutes. Based on the average residence time the isotime lines of the polymer reactor can be derived (Figure 1). All these variables have a strong impact on chain length distributions. An overview of the most important reactions is given below:

$$\textit{Initiation:} \qquad aM + bI \xrightarrow{k_i} P_1$$

$$\textit{Propagation:} \qquad P_1 + M \xrightarrow{k_{pr}} P_2$$

$$P_{n-1} + M \xrightarrow{k_{pr}} P_n$$

$$\textit{Termination by addition:} \qquad P_n + P_m \xrightarrow{k_t} M_{n+m}$$

In this reaction scheme M and I respectively denote the monomer and initiator, P_i the produced polymer with chain length i and k_i, k_{pr}, k_t, the reaction rate constants for the initiation, propagation and termination reactions. To avoid useless overloading of the simulation program some termination reactions are omitted from the basic reaction scheme.

Together with these reactions a number of mass and energy balances are obtained. A general overview can be found in (Froment & Bischoff, 1979) and (Holland & Anthony, 1979). The most important equations, the monomer and initiator mass balances, are given by

$$\frac{d(VM)}{dt} = F_{iM}M_i - F_oM - k_{pr}\left(\frac{2fk_i}{k_t}\right)^{1/2}(I)^{1/2}MV \qquad (1)$$

$$\frac{d(VI)}{dt} = F_{il}I_i - F_oI - k_iIV \qquad (2)$$

with V the reactor volume and f the initiator efficiency. F_{iM}, M_i represent the monomer input flow rate and concentration, F_{il}, I_i the initiator input flow rate and concentration, and F_O, M, I the output flow rate, monomer and initiator concentration.

Equations (1) and (2) lead to the following equation (3) for calculating the average number of monomer units in the polymer chain P_N

$$(\bar{P}_N)_0 = \frac{k_{pr}}{\sqrt{2fk_ik_t}}\frac{M}{I^{1/2}} \qquad (3)$$

The equations listed above are used to simulate the polymerisation reactor. This simulation calculates the

reactor volume, the output concentration of monomer and initiator, the output flow rate and the average number of monomer units in the polymer chain and the number average molecular weight. Furthermore some measurement noise is added to F_{iM}, M_i, F_{il}, I_i and T.

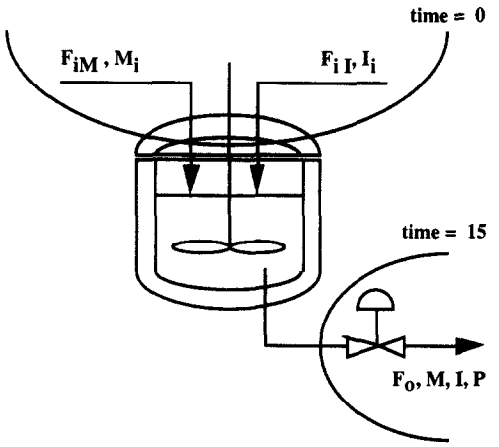

Figure 1. The polymer reactor with its isotime lines.

We are now using the simulation to generate the different input patterns for the neural nets to estimate the average number of monomer units in the polymer chain (or average chain length). One of the major advantages of using such a simulation is the ability to vary and "measure" all the parameters so as to get a broad and general training file, where every possible technique or architecture can be tested on without any limitations whatsoever. Additionally a variety of special problems can be included in the simulator, e.g. measurement noise, etc.

Modelling results for a variety of neural networks

This section is divided into 3 subsections. In each of these subsections the polymer reactor is modelled by different types of nets or architectures (BP, RTRL, MLRN) and from different points of view (making use of a priori knowledge, different sets of input parameters, etc.).

Modelling of the polymer reactor

In this subsection, five different network types or architectures are applied to the polymer reactor: a backpropagation net, a RTRL and a MLRN with or without error feedback, i.e. NMAX(∞,∞). However, no use is made of the a priori knowledge that the average residence time approximates 15 minutes. This signifies that the **actual** average chain length is determined based on the **actual** measurements of the process parameters: no time shift is applied. This leads to the following formula

$$\bar{P}_N(t) = NN(F_{iM}(t), M_i(t), F_{il}(t), I_i(t), V(t), T(t), F_o(t), M(t), I(t)) \quad (4)$$

The train and test results are listed in Table 1.

Network type	*train*	*test*
BP	0.0308	0.0613
RTRL	0.0294	0.0507
MLRN	0.0287	0.0495
NMAX-RTRL	0.0185	0.0315
NMAX-MLRN	0.0177	0.0246

Table 1. MAE of five different network types and architectures for train and test case.

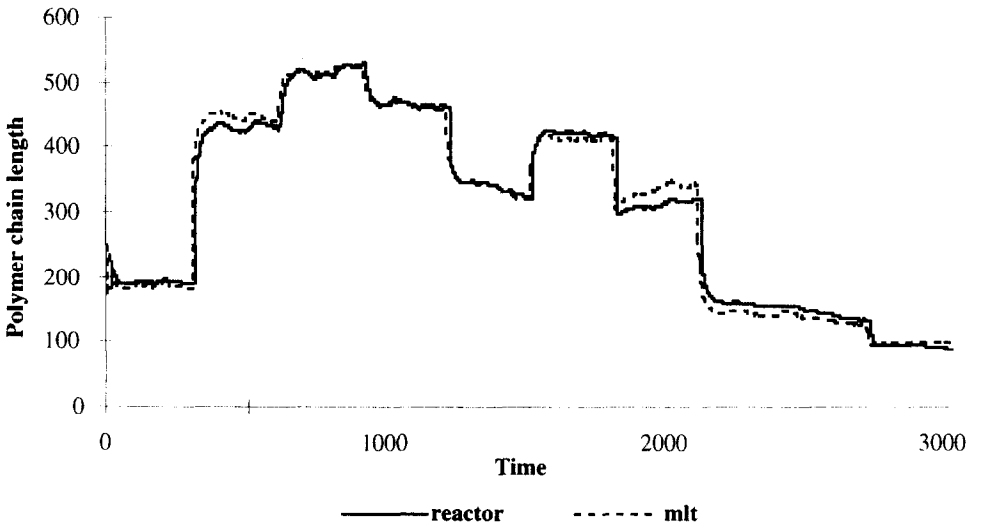

Figure 2. The evolution of the correct and predicted chain length for the MLRN (mlt).

The results gradually improve when the complexity of the network structure rises. The RTRL network outperforms the BP network on the test set by 17%.The error for the RTRL and the MLRN are almost equal for the training and the test set. By applying the previous model error to the two recurrent networks, the average error for the NMAX-RTRL and NMAX-MLRN are respectively reduced by 38% and 50%. Figure 2 shows the evolution of the correct chain length and of its prediction by the MLRN.

Making use of isotime lines

Unlike in the previous case, a priori knowledge is used to reduce the modelling error. This is done by making use of the isotime lines of the process. In this case only two isotime lines are relevant, i.e. the 0 and the 15 minutes isotime line. The latter is based on an estimate of the reactor's average residence time. A time shift 15 minutes is applied between input and output variables. Formula (4) is converted to the following equation

$$\begin{aligned}\bar{P}_N(t) = NN(&F_{iM}(t-15), M_i(t-15), F_{il}(t-15),\\ &I_i(t-15), V(t-15), T(t-15),\\ &F_o(t), M(t), I(t))\end{aligned} \quad (5)$$

While varying the input flow, the output flow and the volume are changed and thus also the residence time. Therefore we assume that the applied changes do not affect significantly the average residence time. If this assumption does not hold, the network performance would deteriorate rapidly and eventually another model should be applied. As an alternative one could also use the minimal residence time of the process, thus avoiding non causal behaviour of the model, which might occur when using the average residence time. The results of the simulation runs of the five different network types are listed in Table 2.

Network type	*train*	*test*
BP	0.0279	0.0564
RTRL	0.0275	0.0489
MLRN	0.0270	0.0478
NMAX-RTRL	0.0181	0.0303
NMAX-MLRN	0.0177	0.0241

Table 2. MAE of five different network types and architectures for the training and test set, when using isotime lines.

Similar conclusions as in the previous section can be drawn when analysing the errors of the networks which use isotime lines. The improvement, however, obtained by increasing the structural complexity of the networks, is less in this case than it was in the former case: e.g. the test error for the RTRL network is only reduced by 13% compared to that of the BP network. If the errors for both modelling cases are compared to each other (see Table 1 and 2), the results of the networks with only a limited dynamic capacity have improved the most: 8% for the BP net and 4% for the RTRL network. The results for the NMAX-MLRN are almost identical for both cases. This can be explained

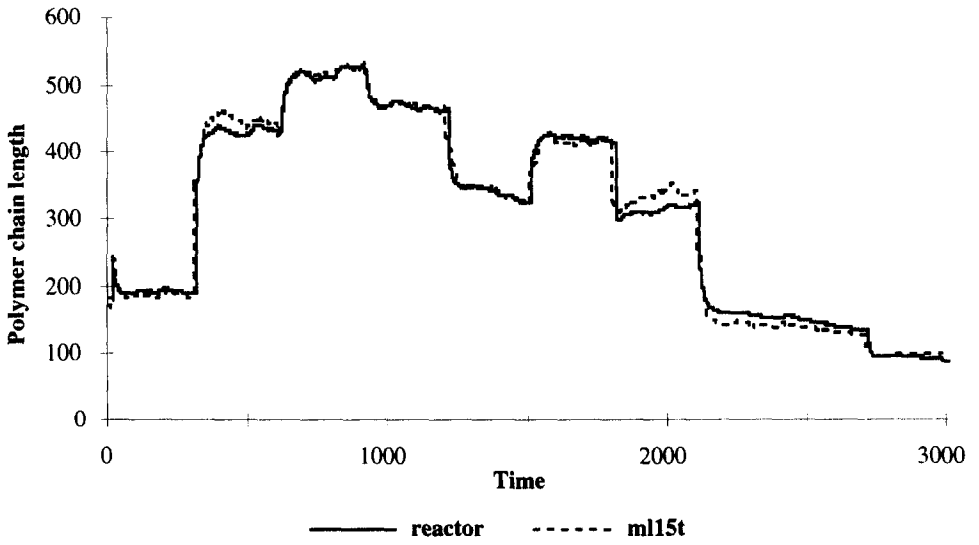

Figure 3. The evolution of the correct and predicted chain length of the MRLN which uses isotime lines (ml15t).

by the fact that in the previous experiment no temporal information was provided to the BP network. On the other hand, the spatio-temporal networks were already able, without the aid of process specific knowledge, to capture (part of) the existing temporal relationships, hence that is why their experimental results only show some relatively small improvements over the results obtained in the previous experiment. Figure 3 shows the evolution of the correct chain length and of its prediction by the MLRN.

Figure 4 shows the evolution of the correct chain length and of its prediction by the BP networks for both modelling cases. A detail of Figure 4 has been blown up.

Notice the phase lead between the output of the BP network which does not make use of the isotime lines, and the correct chain length, which appears during transitions of one steady-state to another. This phase lead approximates the residence time (≈ 15 minutes) of the reactor. The output of the BP network anticipates the changes in the actual values of the input parameters before these really affect the output of the reactor. On the other hand no phase lead emerges for the network which uses isotime lines. Nevertheless some small phase lead or lag can appear since the residence time is subject to changes in the process input or output.

Controlling and controlled parameters

Until now we have only treated those cases where the actual values of all parameters were available. After a small analysis of the problem, the process parameters can be divided into two classes, the controlling and the controlled parameters. The first class of parameters can be altered at one's own discretion, whereas the

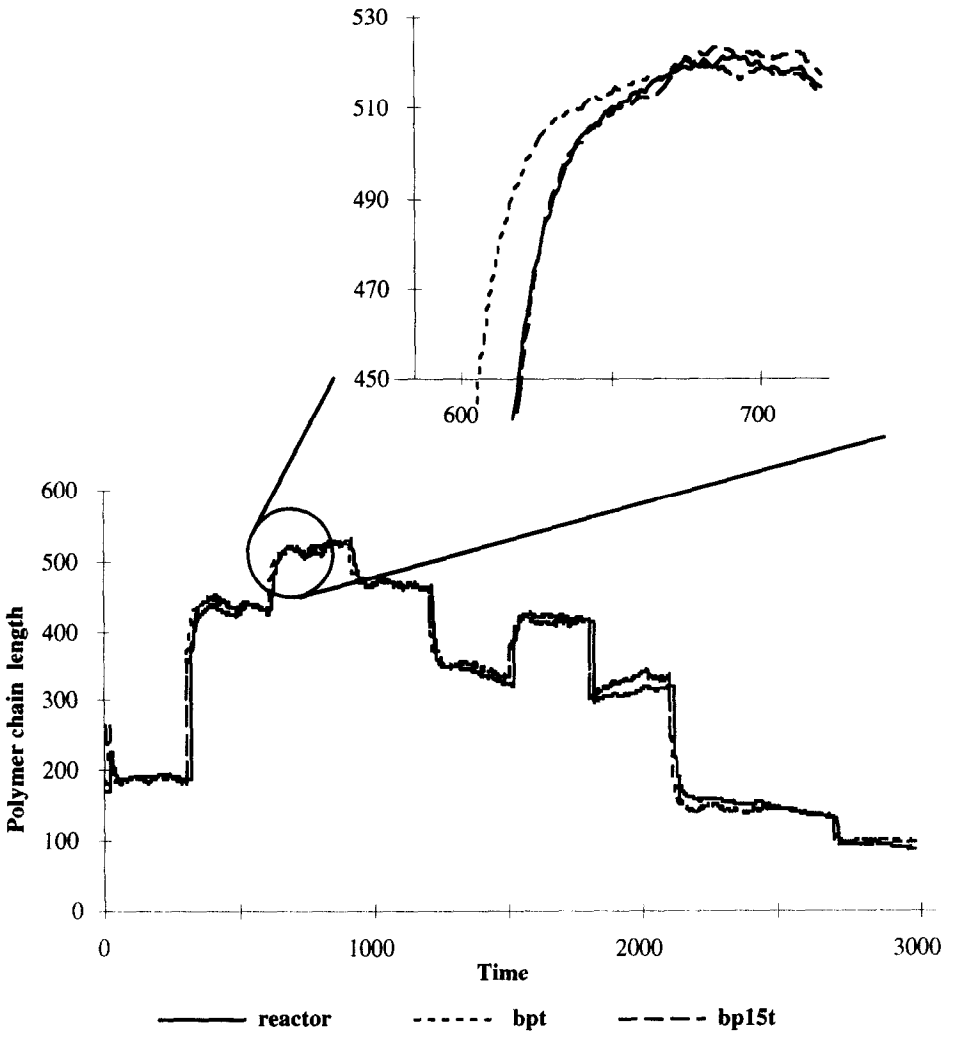

Figure 4. Comparison of the evolution of the BP network with and without the use of isotime lines (respectively (bp15t) and (bpt)).

second class of parameters have to obey the laws of nature and are, as well as the output parameter, controlled by the first class of parameters. Moreover their actual value is not on-line available and/or cannot be altered independently from the other input parameters. Therefore it would be unrealistic to use these parameters directly as input parameters for the network. To solve this issue a modular network structure (Figure 5) is developed.

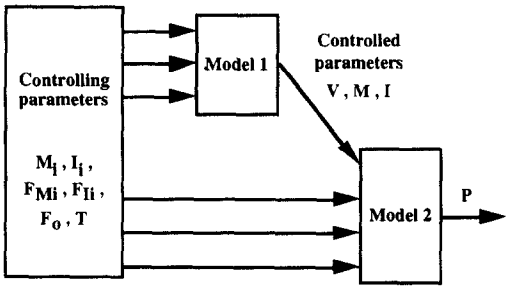

Figure 5. The modular network structure: the first network is used to predict the controlled parameters whereas the second network calculates the output parameter based on the controlling and the controlled parameters.

First of all a model is built to predict the three controlled parameters, V, M, I, based upon the six controlling parameters, F_{iM}, M_i, F_{il}, I_i, F_o, T. Then both classes of parameters, controlling and (predicted) controlled parameters, are fed into a global model, similar to the one built in the previous section, to calculate the average chain length.

This gives rise to the following set of equations:

$$V_p(t), M_p(t), I_p(t) = NN_1(F_{iM}(t-15), M_i(t-15), F_{il}(t-15), I_i(t-15), T(t-15), F_o(t)) \quad (6)$$

$$\bar{P}_N(t) = NN_2(F_{iM}(t-15), M_i(t-15), F_{il}(t-15), I_i(t-15), T(t-15), F_o(t), V_p(t), M_p(t), I_p(t)) \quad (7)$$

Observe that equation (7) resembles equation (5) except for the subscript p (which stands for predicted) for V, M, I. In both cases an identical model network is used.

Network type	*train*	*test*
Modular network	0.0840	0.0796
Single network	0.1003	0.1360

Table 3. MAE of the modular and single network for train and test case.

In Table 3 the results are listed for the modular network as well as for a single neural network model, which predicts the average chain length merely based upon the controlling parameters. For both experiments RTRL networks were used.

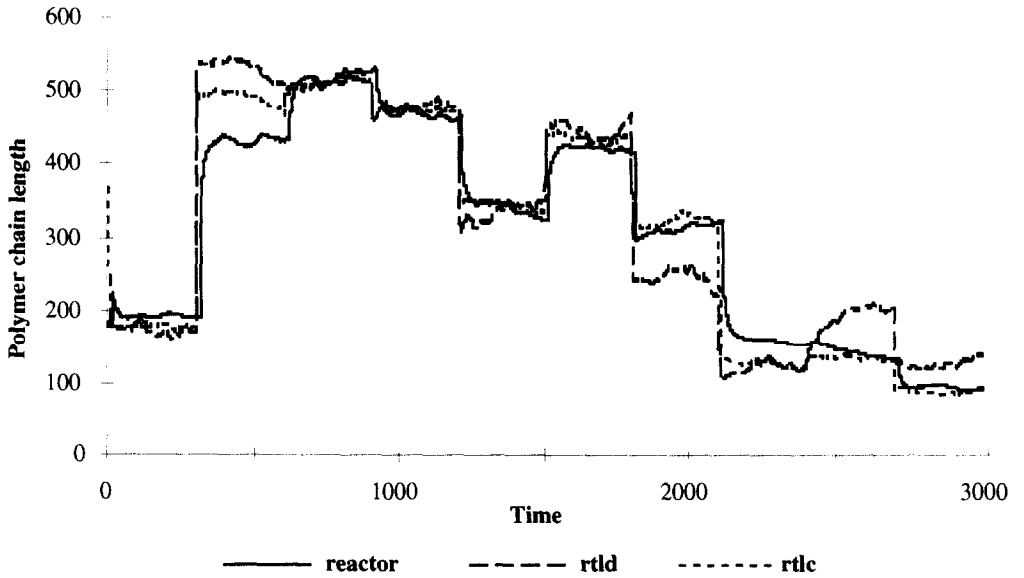

Figure 6. The evolution of the correct and predicted chain length of a single RTRL network (rtld) and of a modular RTRL network (rtlc).

The modular network outperforms the single network on the training set by 16% and on the test set by 41%. Figure 6 shows the evolution of the correct chain length and of its prediction by the modular as well as by the single network.

Conclusion

In this paper various techniques were compared to each other. For this purpose we applied them to a challenging benchmark problem, i.e. the modelling of a nonlinear dynamic polymer reactor. First five different network types and architectures, i.e. a BP, a RTRL and a MLRN network and the NMAX versions of these last two network types, were applied to the polymer reactor modelling problem. The more complex spatio-temporal network types outperformed the networks with reduced temporal capacity, i.e. the NMAX -RTRL and -MLRN networks.

Moreover to improve the modelling performance of these networks, isotime lines were used. The networks with limited dynamics benefited most of this additional information. Nevertheless the NMAX network structures remained the best performing networks.

Finally a modular network structure was used to model the polymer reactor solely based upon the controlling parameters and the intermediate prediction of the controlled parameters. The modular network structure outperforms the single network.

As a general conclusion we can state that the usage of a priori problem knowledge in combination with the right network type or structure results in an optimal model.

Notation

f = initiator efficiency
F_{II} = initiator input flow rate
F_{IM} = monomer input flow rate
F_O = output flow rate
I = initiator concentration in the reactor
I_i = initiator concentration in the input flow
k_i = reaction rate constant for the initiation reaction
k_{pr} = reaction rate constant for the propagation reaction
k_t = reaction rate constant for the termination reaction
M = monomer concentration in the reactor
M_i = monomer concentration in the input flow
$NN()$ = vector valued neural network function
$\overline{P}_N(t)$ = average chain length at time t
P_l = polymer with chain length l
T = reactor temperature
V = reactor volume

References

Chen S. and Billings S.A., 1992, Neural Networks for Nonlinear Dynamic System Modelling and Identification, *International Journal on Control*, **56**, 319-346.

Dimitratos J., Eliçabe G. and Georgakis C., 1994, Control of Emulsion Polymerization Reactors, *AIChE Journal*, **40**, 1993-2021.

Ellis M.F., Taylor T.W. and Jensen K.F., 1994, On-Line Molecular Weight Distribution Estimation and Control in Batch Polymerization, *Aiche Journal*, **40**, 445-462.

Elman J.L., 1990. Finding Structure in Time", *Cognitive Science*, **14**, 179-211.

Froment G.F. and Bischoff K.B., 1979, *Chemical Reactor Analysis and Design*, Wiley & Sons, New York.

Holland C.D. and Anthony R.G., 1979, *Fundamentals of Chemical Reaction Engineering*, Prentice-Hall, New Jersey.

Meert K. and Ludik J., 1997, A Multilayer Real-Time Recurrent Learning Algorithm for Improved Convergence", *proc. of the International Conference on Artificial Neural Networks '97*, Lausanne, Switzerland, 445-450.

Pearlmutter B.A., 1995, Gradient calculations for dynamic recurrent neural networks: a survey, *IEEE Transactions on Neural Networks*, **6**, 1212-1228.

Ponnuswamy S.R., Shah S.L. and Kiparissides C.A., 1987, Computer Optimal Control of Batch Polymerisation Reactors, *Industrial Engineering Chemical Research*, **26**, 2229-2236.

van Dootingh M., Viel F., Rakotopara D., Gauthier J.P. and Hobbes P., 1992, Nonlinear Deterministic Observer for State Estimation: Application to a Continuous Free Radical Polymerisation Reactor, *Computers and Chemical Engineering*, **16**, 777-791.

Wiabel A., Hanaza T., Hinton G., Shikano K. and Lang K., 1989, Phoneme Recognition using Time-delay Networks, *IEEE Transactions on Acoustics, Speech, Signal Processing*, **37**, 328-339.

Williams R.J. and Zipser D., 1989, Experimental Analysis of the Real-time Recurrent Learning Algorithm, *Connection Science*, **1**, 87-111.