Volterra Model Predictive Control of Lyophilization Plant: A Newton Optimization Method Approach

Y. Todorov, S. Ahmed, M. Petrov

Key words: *Fuzzy-neural models; Newton method; Predictive control; Lyophilization.*

Abstract. *The lyophilization process is widely used by pharmaceutical and food industries preparing stable dried medications and important biopreparations. Recent advances in lyophilization technology impose the application of innovative strategies for reliable determination of the current process conditions and control of the drying cycles. This paper describes a method for designing a nonlinear model predictive controller to be used in a lyophilization plant. The controller is based on a truncated fuzzy-neural Volterra predictive model and a simplified Newton method as an optimization algorithm. The proposed approach is studied to control the product temperature in a lyophilization plant. Several simulation experiments have been performed in to demonstrate the efficiency of the proposed approach. The obtained results are compared with the classical Gradient optimization procedure.*

1. Introduction

Nearly 70 years ago, lyophilization began to change the way scientists developed food products and drugs. Since then, the use of lyophilization has resolved several problems in food and pharmaceutical industries. However, from a process point of view, lyophilization also has created some challenges.

Early uses of freeze-drying involved the naturally occurring processes of freezing and dehydration. For example, residents of the Andes recognized the phenomenon and used it to preserve vegetables. Other references cite the industrial application of freeze-drying in the 1920s, forecasting it as a means of preserving grain crops and other foods on a large scale. The basic process has been used at least since the 1930s for commercial purposes. Several theoretical applications have also been recognized, including military purposes for the development of offensive weapons as an adjunct mechanism for delivering stable, viable microorganisms or chemicals as well as its use in the field of medical treatment [1].

Lyophilization is nearly always investigated as an alternative to a frozen product for extended clinical trials and for commercialization. The process can reduce or eliminate the need for difficult storage and handling arrangements and may provide a pathway to a drug product with favourable shelf life.

Lyophilization is a drying process in which the solvent and/or the suspension medium is crystallized at low temperatures and thereafter sublimed from the solid state directly into the vapour phase. Freeze-drying is mostly done with water as a solvent. From the phase diagram of water it can be seen the area in which this transfer from solid to vapour is possible. The drying transforms the ice or water in an amorphous phase into vapour. The goal of lyophilization is to produce a substance with good shelf stability unchanged after reconstitution [2].

On the other hand, the lyophilized products are very expensive due to the high energy demands to maintain vacuum and refrigeration processes and the latent heat for sublimation, as well. For this purpose, it is needed to be used improved control strategies based on intelligent control methods, such as Model Predictive Control.

The model Predictive Control (MPC) has received a strong position when it comes to industrially implemented advanced control methodologies [3-4]. The main reason for this is the intuitive way MPC incorporates the process model in the controller design. In many problems relevant to the process control field today, the plant under control shows a strongly nonlinear behaviour. As a means to handle this, the Nonlinear MPC (NMPC) is an often used method. The NMPC, simply put, is model predictive control, where a nonlinear process model is used for prediction purposes, as opposed to a linear model for basic MPC [5-7]. Recently, several researchers report different applications of NMPC on lyophilization plants [8-10].

Most of industrial processes are nonlinear and the system nonlinearity cannot be ignored in practice. This has stimulated work in synthesizing MPC for use with a nonlinear analytical Volterra model and in Volterra series modelling. The main criticism in using Volterra series as nonlinear models lies in its large number of parameters needed to represent the kernels [11]. For this reason, in most practical solutions some structural restrictions to Volterra type models are imposed in order to attend a better model accuracy using a small number of parameters and to facilitate the identification procedures in notion to the computational effort. It has been shown, that any time-invariant nonlinear system can be approximated by a finite Volterra series to an arbitrary precision. Volterra models have the property to be linear in their parameters, i.e. the coefficients of their kernels, so that standard parameter estimation methods can be used [12].

In this paper, the proposed Volterra Fuzzy-Neural (VFN) model is implemented in a MPC control scheme by using a simple fuzzy-neural approach and theNewton method as an optimization procedure. The efficiency of the presented approach is proved by simulation experiments in Matlab & Simulink environment to control the product temperature in a lyophilization plant. The results obtained during the simulation experiment are compared to the case when a standard gradient optimization procedure is used.

2. Design of Fuzzy-neural Volterra Model

Volterra models are widely used to model nonlinear processes. Since, with the increasing level of model nonlinearity, the number of its parameters increases sharply, in practice mostly used are truncated Volterra models [13]. In this approach, the fuzzy-neural implementation of a second order Volterra model is considered. As is well known, a wide class of nonlinear dynamic systems can be described in discrete time by the NARX (Nonlinear AutoregRessive model with eXogenous inputs) input-output model. The model used in this paper is also taken in NARX type:

(1) $y(k)=f_y(x(k))$ and $x(k)=[u(k\text{-}1),...,u(k\text{-}n_u), y(k\text{-}1),...,y(k\text{-}n_y)]$.

The unknown nonlinear functions f_y can be approximated by Takagi-Sugeno type fuzzy rules:

(2) $R^{(i)}$: *if* x_1 is $\tilde{A}_1^{(i)}$ and x_p is $\tilde{A}_p^{(i)}$ then $f_y^{(i)}(k)$

(3) $f_y^{(i)}(k) = a_1^{(i)}y(k\text{-}1) + a_2^{(i)}y(k\text{-}2) + ... + a_{ny}^{(i)}y(k\text{-}n_y) +$
$+ b_1^{(i)}u(k\text{-}1) + b_2^{(i)}u(k\text{-}2) + ... + b_{nu}^{(i)}u(k\text{-}n_u) +$
$+ b_0^{(i)} + v_{m1}^{(i)}(k) + v_{m2}^{(i)}(k)$

where the Volterra kernels are

(4) $$v_{mi} = \sum_{i=1}^{n_u}\sum_{j=1}^{i} c_{ij}\, u(k\text{-}i)u(k\text{-}j).$$

The upper index (i) =1,2,…,N represents the number of the fuzzy rules, $\tilde{A}_i$ is an activated Gaussian fuzzy set defined in the universe of discourse of the input vector x, the crisp coefficients $a_1, a_2,...,a_{ny}, b_1, b_2,...,b_{nu}, c_{1,1}, c_{2,1}, c_{i,j}$ are the coefficients into the Sugeno function f_y and n_y/n_u is the history dependence on the input/output.

Finally, the actual implementations of the relevant fuzzy predictions have been obtained by appropriately shifting the inputs of the model. Therefore, a sequential algorithm based on the knowledge of current values of the regression vector, along with the fuzzy inference, computes:

(5) $$\hat{y}(k+j) = \sum_{i=1}^{N} f_y^{(i)}(k+j)\bar{\mu}_{yi}^{(i)}(k+j)$$

where $\bar{\mu}_{yi}$ is the normalized value of the membership function degree μ_{yi} upon the i^{th} activated fuzzy rule which can be expressed as

(6) $$\bar{\mu}_{yi} = \mu_{yi} \Big/ \sum_{i=1}^{N}\mu_{yi}$$

where N is the number of the activated rules. Fuzzy implication in the i^{th} rule can be realized by means of product composition

(7) $\mu_{yi} = \mu_{i11}*\mu_{i12}*\mu_{i1j}*...*\mu_{ipj}$

2.1 Fuzzy-neural Model Identification

The identification procedure involves structure identification of the process and estimation of the unknown parameters. The structure of the neuro-fuzzy model depends on the number of membership functions, their shape and the coefficients into the functions f_y in the consequent part of the rules (2). The task of model identification is to determine both groups of parameters of the Gaussian membership functions in the rule premise part and the linear parameters in the rule consequent part of the local models. A simplified fuzzy-neural approach is applied in this work, because of its simplicity and recurrent implementation of a tuning procedure for on-line applications [14].

The learning algorithm for the fuzzy-neural model is based on minimization of an instant error measurement function between the real plant output and the process output, calculated by the fuzzy-neural model

(8) $$E(k) = (y(k) - \hat{y}(k))^2/2$$

where $y(k)$ denotes the measured real plant output and $\hat{y}(k)$ is calculated by the fuzzy-neural network. The algorithm performs two steps gradient learning procedure. Assuming that βij is an adjustable i^{th} coefficient for the Sugeno function f_y into the j^{th} activated rule (2) as a connection in the output neuron, the general parameter learning rule for the consequent parameters is

(9) $$\beta_{ij}(k+1) = \beta_{ij}(k) + \eta\left(-\partial E/\partial\beta_{ij}\right).$$

After calculating the partial derivatives, the final recurrent predictions for each adjustable coefficient $\beta ij(a^{(i)}, b^{(i)}$ or $c^{(i)})$ and the free coefficient are obtained by the following equations:

(10) $$\beta_{ij}(k+1) = \beta_{ij}(k) + \eta\varepsilon(k)\bar{\mu}^{(j)}(k)x_i(k)$$

(11) $$\beta_{0j}(k+1) = \beta_{0j}(k) + \eta\varepsilon(k)\bar{\mu}^{(j)}(k).$$

The output error E can be used back directly to the input layer, where there are the premise adjustable parameters (center - Ω_{ij} and the deviation - σ_{ij} of a Gaussian fuzzy set). The error E is propagated through the links composed by the corresponded membership degrees, where the link weights are the unit. Hence, the learning rule for the second group adjustable parameters in the input layer can be done by the same learning rule:

(12) $$\Omega_{ij}(k+1) = \Omega_{ij}(k) + \eta\varepsilon(k)\bar{\mu}_y^{(i)}(k)[f_y^{(i)} - \hat{y}(k)]\frac{[x_i(k) - \Omega_{ij}(k)]}{\Omega_{ij}^2(k)}$$

(13) $$\sigma_{ij}(k+1) = \sigma_{ij}(k) + \eta\varepsilon(k)\bar{\mu}_y^{(i)}(k)[f_y^{(i)} - \hat{y}(k)]\frac{[x_i(k) - \sigma_{ij}(k)]^2}{\sigma_{ij}^3(k)}.$$

3. Basics of the Applied Model Predictive Control Strategy

During the past years, the Model Predictive Control has received a lot of attention in the control theory and applications. A model of the controlled process provides the forecast of the process output signal and the control signal is calculated in every step in a way that the difference between the reference and the output signal is minimized. The good system performance depends on the model accuracy and parameters in the objective function. NMPC as it was applied with the VFN process model can be described in general by a block diagram, as it is depicted in *figure 1*.

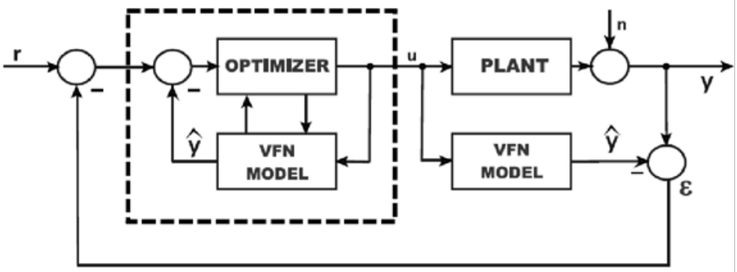


Figure 1. Block diagram of the proposed MPC system

The selection of a minimization algorithm is a crucial issue in MPC, since this feature affects the computational efficiency of the control loop. Using the Newton method as an optimization algorithm reduces the iterations to convergence in contrast to other techniques. The main cost of the Newton algorithm is the calculation of the Hessian matrix, but even with this overhead the low iteration numbers make the Newton algorithm faster for real time control [15]. As is well known, the Newton method is based on a quadratic approximation of an objective function as described:

(14) $$\widetilde{P}(x) = P(x^{(k)}) + \nabla^T P(x^{(k)})\Delta x^{(k)} + \frac{1}{2}(x^{(k)})^T \nabla^2 P(x^{(k)})\Delta x^{(k)}$$

This requires the evaluation of the Hessian and the gradient of the objective function. To implement the Newton method as an optimization algorithm the following recurrent equations are used:

(15) $$x^{(k+1)} = x^{(k)} - \left[\nabla^2 P(x^{(k)})\right]^{-1} \nabla P(x^{(k)})$$

(16) $$x^{(k+1)} = x^{(k)} - H^{-1}(x^{(k)})\nabla P(x^{(k)})$$

where H is the Hessian matrix with the second order partial derivatives as elements. An important principle in the Newton method is that the cost function must be quadratic one and the Hessian matrix must be positive by definite.

Using the VFN model, the Optimization Algorithm computes the future control actions at each sampling period, by minimizing the following cost function:

(17) $$J(k,u(k)) = \sum_{i=N_1}^{N_2} (r(k+i) - \hat{y}(k+i))^2 + \rho \sum_{i=N_1}^{Nu} \Delta u(k+i-1)^2$$

where $\hat{y}$ is the predicted model output, r is the reference and u is the control action. The tuning parameters of the predictive controller are: N_1, N_2, N_u and ρ. N_1 is the minimum prediction horizon, N_2 is the maximum prediction horizon, N_u is the control horizon and ρ is the weighting factor penalizing changes in the control actions. When the criterion function is a quadratic one and there are no constraints on the control action, the cost function can also be minimized analytically. If the criterion J is minimized with respect to the future control actions, then their optimal values can be calculated by applying the condition $\nabla J(k,U(k)) = 0$, where each element of the gradient vector can be calculated using the following equation:

(18) $$\frac{\partial J(k,U(k))}{\partial u(k)} = \left[-2\left[R(k) - \hat{Y}(k)\right]^T \frac{\partial \hat{Y}(k)}{\partial U(k)} + 2\rho U(k)^T \frac{\partial \hat{U}(k))}{\partial U(k)} \right]$$

where $R(k)$ is the system reference vector, is the vector of the predicted model output and $U(k)$ is the vector of the control actions.

Since, the VFN model consists of a set of local sub models an explicit analytic solution of the above optimization problem can be obtained. A simplified method for calculation of the elements of $\nabla J(k,U(k)$ based on the VFN model, is proposed here. Hence, according to f_y function (3) the unknown elements in (18) can be evaluated as follows:

(19) $$\frac{\partial \hat{y}(k+1)}{\partial u(k)} = \sum_{i=1}^{N} \begin{bmatrix} a_1^{(i)} \frac{\partial \hat{y}(k)}{\partial u(k)} + b_1^{(i)} \frac{\partial u(k)}{\partial u(k)} + \\ + c_{1,1}^{(i)} \frac{\partial u^2(k)}{\partial u(k)} + \\ + c_{2,1}^{(i)} \frac{\partial u(k-1)u(k)}{\partial u(k)} \end{bmatrix} \overline{\mu}_y^{(i)}(k+1)$$

(20) $$\frac{\partial \hat{y}(k+2)}{\partial u(k)} = \sum_{i=1}^{N} \begin{bmatrix} a_1^{(i)} \frac{\partial \hat{y}(k+1)}{\partial u(k)} + a_2^{(i)} \frac{\partial \hat{y}(k)}{\partial u(k)} + \\ + b_1^{(i)} \frac{\partial u(k)}{\partial u(k)} + c_{2,2}^{(i)} \frac{\partial u^2(k)}{\partial u(k)} + \\ + c_{2,1}^{(i)} \frac{\partial u(k)u(k+1)}{\partial u(k)} \end{bmatrix} \overline{\mu}_y^{(i)}(k+2)$$

(21) $$\frac{\partial \hat{y}(k+N_2)}{\partial u(k)} = \sum_{i=1}^{N} \begin{bmatrix} a_1^{(i)} \frac{\partial \hat{y}(k+N_2-1)}{\partial u(k)} + \\ + a_2^{(i)} \frac{\partial \hat{y}(k+N_2-2)}{\partial u(k)} \end{bmatrix} \overline{\mu}_y(k+N_2)$$

Since, $\Delta u(k) = u(k) - u(k\text{-}1)$ then $\partial\hat{U}/\partial U$ represents a matrix with zeroes and ones. As Newton method imposes the implementation of the second order derivative of the cost function we can rewrite

(22) $$\nabla^2 J(k,U(k)) = \left[\frac{\partial^2 J(k,U(k))}{\partial u^2(k)}, \ldots, \frac{\partial^2 J(k,U(k))}{\partial u^2(k+N_u-1)} \right]$$

(23) $$\frac{\partial^2 J(k,U(k))}{\partial u^2(k)} = \begin{bmatrix} 2\left(\frac{\partial \hat{Y}k)}{\partial U(k)}\right)^2 - 2\left[R(k) - Y(k)\right] \frac{\partial^2 \hat{Y}(k)}{\partial U^2(k)} + \\ 2\rho \left(\frac{\partial \hat{U}(k))}{\partial U(k)}\right)^2 + 2\rho\left(U(k)^T\right) \frac{\partial^2 \hat{U}(k))}{\partial U^2(k)} \end{bmatrix}$$

Since $\partial^2\hat{U}(k)/\partial U^2(k)$ always evaluates to zero, the second order derivatives

$$\frac{\partial^2 \hat{y}(k+1)}{\partial u^2(k)} = \sum_{i=1}^{N} \left[c_{1,1}^{(i)} \left(\frac{\partial u^2(k)}{\partial u(k)} \right) \right] \overline{\mu}_y^{(i)}(k+1)$$

(25) $$\frac{\partial^2 \hat{y}(k+2)}{\partial u^2(k)} = \sum_{i=1}^{N} \begin{bmatrix} a_1^{(i)} \frac{\partial^2 \hat{y}(k+1)}{\partial u^2(k)} + \\ + c_{2,2}^{(i)} \left(\frac{\partial u^2(k)}{\partial u(k)} \right) \end{bmatrix} \overline{\mu}_y^{(i)}(k+2)$$

(26) $$\frac{\partial^2 \hat{y}(k+N_p)}{\partial u^2(k)} = \sum_{i=1}^{N} \begin{bmatrix} a_1^{(i)} \frac{\partial^2 \hat{y}(k+N_2-1)}{\partial u^2(k)} + \\ + a_2^{(i)} \frac{\partial^2 \hat{y}(k+N_2-2)}{\partial u^2(k)} \end{bmatrix} \overline{\mu}_y(k+N_p).$$

The Newton algorithm then iterates using the following expression:

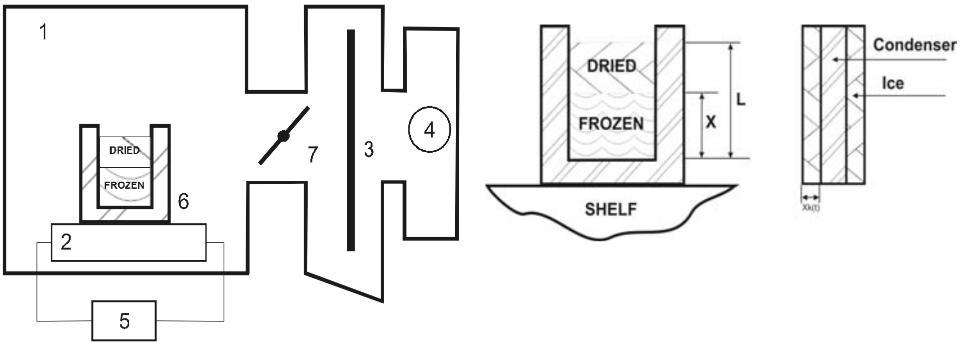


Figure 2. Schematic diagram of a simplified lyophilization plant

$$-\frac{\partial J(k,U(k))}{\partial U(k)}=\frac{\partial^2 J(k,U(k))}{\partial U^2(k)}\Delta U(k)\,. \tag{27}$$

Using the last notation a simple analytical solution of the above optimization problem can be found iterating along the control horizon:

$$\begin{aligned}&\Delta u(k+N_u-1)=\Delta u(k+N_u)+\\&+\rho^{-1}\left[\hat{e}(k+N_1)\frac{\partial\hat{y}(k+N_1)}{\partial u(k+N_u-1)}+...+\hat{e}(k+N_2)\frac{\partial\hat{y}(k+N_2)}{\partial u(k+N_u-1)}\right]+\\&+\alpha\left[\begin{aligned}&+\left(\frac{\partial\hat{y}(k+N_1)}{\partial u(k+N_u-1)}\right)^2-\left(\hat{e}(k+N_1)\frac{\partial^2\hat{y}(k+N_1)}{\partial u^2(k+N_u-1)}+...+\hat{e}(k+N_2)\frac{\partial^2\hat{y}(k+N_2)}{\partial u^2(k+N_u-1)}\right)+\\&+\left(\frac{\partial u(k+N_u-1)}{\partial u(k+N_u-1)}\right)^2\end{aligned}\right]\end{aligned} \tag{28}$$

$$\begin{aligned}&\Delta u(k+N_u-2)=\Delta u(k+N_u-1)+\\&+\rho^{-1}\left[\hat{e}(k+N_1)\frac{\partial\hat{y}(k+N_1)}{\partial u(k+N_u-2)}+...+\hat{e}(k+N_2)\frac{\partial\hat{y}(k+N_2)}{\partial u(k+N_u-2)}\right]+\\&+\alpha\left[\begin{aligned}&\left(\frac{\partial\hat{y}(k+N_1)}{\partial u(k+N_u-2)}\right)^2-\left(\hat{e}(k+N_1)\frac{\partial^2\hat{y}(k+N_1)}{\partial u^2(k+N_u-2)}+...\hat{e}(k+N_2)\frac{\partial^2\hat{y}(k+N_2)}{\partial u^2(k+N_u-2)}\right)+\\&\left(\frac{\partial u(k+N_u-2)}{\partial u(k+N_u-2)}\right)^2\end{aligned}\right]\end{aligned} \tag{29}$$

$$\begin{aligned}&\Delta u(k+1)=\Delta u(k+2)+\\&+\rho^{-1}\left[\hat{e}(k+N_1)\frac{\partial\hat{y}(k+N_1)}{\partial u(k+1)}+...+\hat{e}(k+N_2)\frac{\partial\hat{y}(k+N_2)}{\partial u(k+1)}\right]+\\&+\alpha\left[\begin{aligned}&+\left(\frac{\partial\hat{y}(k+N_1)}{\partial u(k+1)}\right)^2-\left(\hat{e}(k+N_1)\frac{\partial^2\hat{y}(k+N_1)}{\partial u^2(k+N_1)}+...+\hat{e}(k+N_2)\frac{\partial^2\hat{y}(k+N_2)}{\partial u^2(k+1)}\right)+\\&+\left(\frac{\partial u(k+1)}{\partial u(k+1)}\right)^2\end{aligned}\right].\end{aligned} \tag{30}$$

The whole control sequence is calculated consequently starting from the last element of the control horizon and then only the first element of the control vector is sent to the plant.

4. Simulation Experiments

4.1. Experimental plant description

During the last years, extensive efforts by industry and research have been made to predict and optimize the course of the lyophilization cycles in order to control the quality of the product and to minimize the costs [16-17]. In *figure 2* a simplified diagram of the main components of a lyophilization plant is shown. The plant consists particularly of a drying chamber (1); temperature controlled shelves (2), a condenser (3) and a vacuum pump (4). The major purposes of the shelves are to cool and freeze or to supply heat to the product. This is supported by the shelves heater and refrigeration system (5). On those shelves the product is placed (6). The chamber is isolated from the condenser by the valve (7). The vacuum system is placed after the condenser. When the product is entirely frozen, the chamber is evacuated in order to increase the partial vapour water pressure difference between the frozen ice zone and the chamber.

The shelf heating system starts to provide enthalpy for the sublimation process. The sublimation takes place at a moving ice front, which proceeds from the top of the frozen material downwards. The stage in which the remaining water content is further reduced is called secondary drying, which takes place at higher temperature. In this contribution only the first stage of the drying process called primary drying is assumed.

The considered plant is a small scale lyophilization apparatus, for drying of 50 vials filled with glycine in water adjusted to pH 3, with hydrochloric acid. The schematic diagram in *figure 2* depicts the sublimation process occurring at the interface which is located at a distance x from the vial bottom. During sublimation the interface moves in a negative direction, while the product height remains constant.

4.2. Simulation experiments

Simulation experiments in *Matlab & Simulink* environments to control the heating shelves temperature depending of the temperature inside the frozen product layer are made. According to this circumstance, the system is nonlinear and non stationary and this is because during the sublimation process the properties of the product are changed.

The following initial conditions for simulation experiments are assumed; $N_1=1$, $N_2=3$, $N_u=3$, system reference $r = 255\ K$, initial shelf temperature, $T_{S_{in}} = 228\ K$, initial thickness of the front $x = 0.0023\ m$, thickness of the product $L = 0.003\ m$. In the primary drying stage it is required to maintain the shelf temperature about $298\ K$, until the product is dried. This circumstance requires about 45 minutes of time for the primary drying stage of the process.

The aim of the control system is to reduce the system error between the reference product temperature and the current product temperature at each sampling period, by calculating an appropriate control action, which will drive the drying process as fast as possible. The physical explanation of this is minimizing the energy for the drying process, as computing the optimized

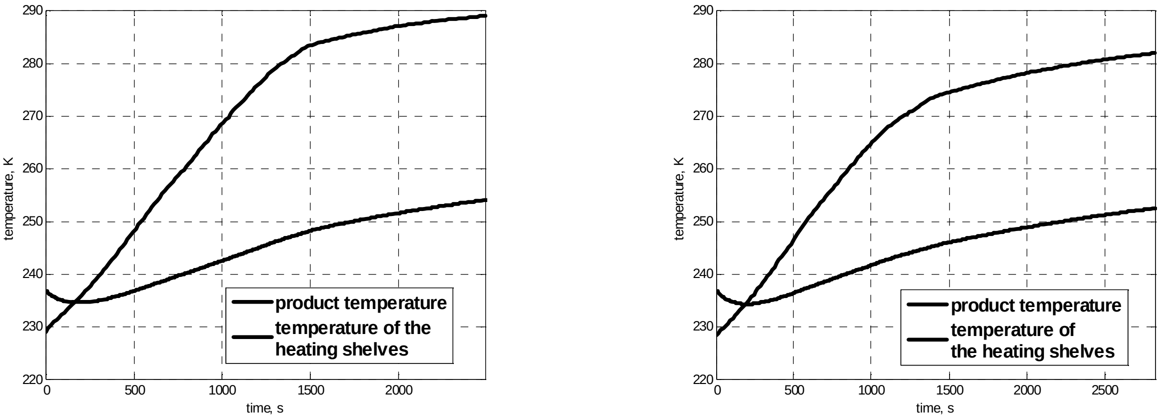


Figure 3. Product and shelf temperatures using the Newton and Gradient optimization methods

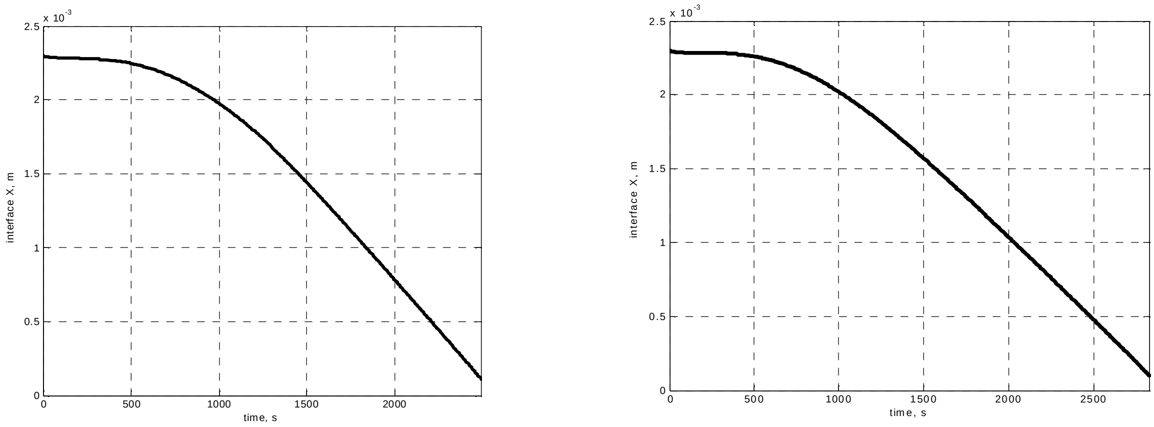


Figure 4. Interface position using the the Newton and Gradient optimization methods

values for the heating shelves temperatures. According to this a criterion is defined in which the efficiency parameter Eef represents a notion between the cumulative energy which is minimized and the energy provided for the heating process [14].

$$E_{ef} = \left(\int_0^t (T_r(t) - T_p(t))dt\right) / \left(\int_0^t T_s(t)dt\right) \quad (31)$$

where Ts - temperature of the heating shelves, Tr - reference product temperature, Tp - product temperature. As a reference criterion for the process it is also taken the settling time of the process (t_p). Evaluation of the model performance is demonstrated by the *Root Mean Squared Error* (RMSE) and the *Root Squared Error* (RSE) plots of the model.

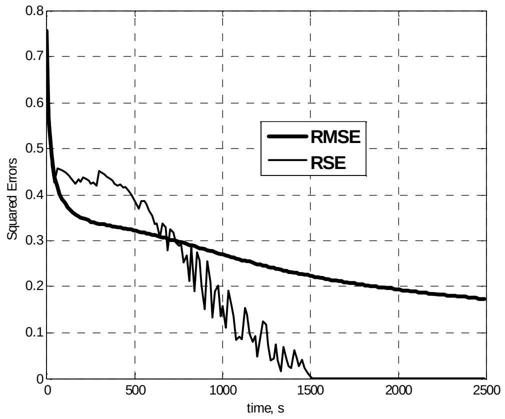
Comparative experiments with the proposed VFN model using the Newton method as optimization algorithm and the standard Gradient optimization algorithm as reference, for two different values of the penalty term ρ, are made. The validated plant model used as the plant process for simulation in this study was derived from the physical laws of heat and mass transfer for a typical laboratory plant. The temperature versus time profile for the product and heating shelf temperatures for the representative vial are presented in *figure 3*.

The primary drying phase for the cycle was started by increasing the shelf temperature from 228 K. The initial drop of the product temperature represents the sudden loss of heat due to sublimation and indicates the start of the primary drying, as well. After all of the unbound water has sublimated, the loss of heat due to sublimation vanishes and the enthalpy input from the shelf causes a sharp elevation of the product temperature. The VFN model responses of the RMSE and RSE are shown in *figure 5*. In *figure 4* the decrease of the frozen layer interface x is demonstrated.

Comparative results

Method	t_p	E_{ef}	*RMSE*
Gradient	2828	0.000660	0.1735
Newton	2490	0.000757	0.1575

As it can be seen from the presented results, the realization of a predictive controller on the basis of fuzzy-neural Volterra model requires the statement of considerable amount of fuzzy rules. This leads to improved dynamic qualities of the model, accurate identification and approximation of the nonlinear effects related to the occurring drying phenomena. On the other hand, the large amount of fuzzy rules increase the number of parameters under adaptation during the learning procedure, which



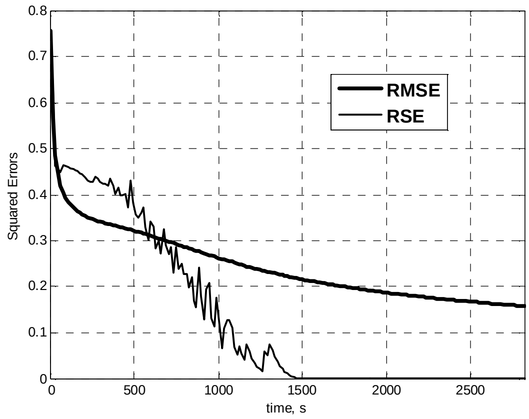


Figure 5. RMSE and RSE responses of the model using the Newton and Gradient optimization methods

affects the computational effort needed for identification and optimization purposes.

A major advantage of the proposed model is the ability to model nonlinear processes using nonlinear functions in contrast to the classical structure of many Takagi-Sugeno model representations. The structure of the model is flexible and the order of the model nonlinear kernels can be set of power n, depending on the complexity of the identification task.

Generally, the major industrial process can be represented by a model of $n=2$. For this purpose, in this contribution the designed model is truncated, which provides an acceptable trade off between the computational burden and the modelling accuracy. Using a model of higher order will significantly increase the number of the identified parameters in the consequent part of the rules.

The simulation experiments show the advantages of the proposed predictive controller based on an analytical Newton like optimization procedure. As expected the Newton procedure increases the convergence which minimizes the drying time in contrast to the Gradient method. Also, the major system parameters are under its maximum bounds and the process is driven into the acceptable region of operation. On the other hand this positive effect is compensated by slightly increased temperature of the heating shelves and the E_{ef} parameter.

Conclusions

A method for designing a nonlinear Model Predictive Controller was presented in this paper. The controller is based on a truncated Volterra fuzzy-neural model and Newton method as an optimization algorithm. The proposed approach was used to control the product temperature in a lyophilization plant. The simulation experiments show the efficiency of the proposed control strategy. The product temperature in the frozen region rises according to lyophilization cycle regime requirements and constraints. The proposed nonlinear Volterra fuzzy-neural model ensures an accurate identification of the nonlinear lyophilization plant and its application along with the Newton method as optimization procedure ensures the reduction of the the drying time in contrast to classical Gradient descent procedure.

References

1. Smith, T. A History of Lyophilization in Pharmaceutical Applications: Control and Monitoring, Accuracy, and Reproducibility Continue to Define the Lyophilization Process, thus Enhancing Product Aesthetics, Stability, and Reconstitution. Pharmaceutical Technology, 2004. http://www.findarticles.com/p/articles/mi_m0EEH/is_2_28/ai_113416745.

2. Oetjen, G., P. Hasley. Freeze-Drying 2-nded., WILEY, 2004.

3. Qin, J., T. Bagwell. A Survey of Industrial Model Predictive Control Technology. - C*ontrol Engineering Practice*, 11 (7), 2003, 733-764.

4. Takatsu, H. Advanced Process Control and Optimization Solutions. Yokogava Tech. Report. Ed., 43, 2007.

5. Camacho, E., C. Bordons. Model Predictive Control 2nd ed., Springer Verlag, 2004.

6. Rawlings, B. Tutorial Overview of Model Predictive Control. - *IEEE Control Systems Magazine*, 20(3), 2000, 38-52.

7. Allgöwer, F., R. Findeisen. An Introduction to Nonlinear Model Predictive Control. Proc. of 21st Benelux Meeting on Systems and Control, Veldhoven, 2003.

8. Daraoui, N., P. Dufour, H. Hammouri, A. Hottot. Model Predictive Control During the Primary Drying Stage of Lyophilisation. - *Control Engineering Practice*, 18 (5), 2010, 483-494.

9. Dufour, P. Optimal Operation of Sublimation Time of the Freeze Drying Process by Predictive Control: Application of the MPC@CB Software. Proceedings of the 18th EFCE Symposium, France, 25, 2008, 453-458.

10. Pisano, R., D. Fissore, S. Velardi, A. Barresi. In-line Optimization and Control of an Industrial Freeze-drying Process for Pharmaceuticals. - *Pharmaceutical sciences*, 99 (11), 2008, 4691-4709.

11. Favier, G., A. Kibangou, A. Khouaja. Nonlinear System Modeling by Means of Volterra Models: Approaches for Parametric Complexity Reduction. Tech. Sym. on Innovative Strategies in Modeling and Robust Control of Ind. Proc., 2004.

12. Zheng, Q., E. Zafirou. Nonlinear System Identification Using Volterra-Laguerre Expansion. - *ACC*, 3, 1995, 2195-2199.

13. Kozak, S., P. Taraba. MPC Using AR Volterra Models. The 11th Med. Conf. on Control and Automation, t5-002, 2003.

14. Todorov, Y., Ts. Tsvetkov. Volterra Model Predictive Control of a Lyophilization Plant. Int. In Proceedings of IEEE Conference Intelligent Systems, Bulgaria, 3, 2008, 20/13-18, 2008.

15. Soloway, D., P. Hasley. Neural Generalized Predictive Control: A Newton-Raphson Implementation. NASA Technical Memorandum,1997, 110244.

16. Shoen, M. A Simulation Model for Primary Drying Phase of Freeze-drying. - *International Journal of Pharmaceutics*, 114, 1995, 159-170.

17. Shoen, M., & R. Jefferis. Simulation of a Controlled Freeze-drying Process. Proc. of IASTED International Conference, 1993, 65-68.

Manuscript received on 28.10.2011

Yancho Todorov *was born in Plovidiv, Republic of Bulgaria in 1981. He graduated the master degree in Department of Control Systems, at the Faculty of Electronics and Automation, the Technical University - Sofia, branch Plovdiv. During his studies he specializes in intelligent control methods, theory and application. Currently, he is with the Institute of Cryobiology and Food Technologies part of Agricultural Academy of Bulgaria, as Assistant Professor and Ph.D. He continues his research works within the field of intelligent control systems design and application on specific cryogenic plants for production and preservation of foods and biocomponents.*

Contacts:
Institute of Cryobiology and Food Technologies, Agricultural Academy
53 Cherni Vrah Blvd., 1407 Sofia, Bulgaria
tel: +359-2-9628413
e-mail: yancho.todorov@ieee.org

Sevil Ahmed *was born in Kardzhali, Republic of Bulgaria in 1982. She graduated the Master degree in Automation, Information and Control Systems in the Department of Control Systems at the Faculty of Electronics and Automation, the Technical University - Sofia, branch Plovdiv, Bulgaria in 2006. She is currently working toward the Ph.D. degree in the Department of System and Control, the Technical University - Sofia. Her research interests include intelligent control, fuzzy logic systems and control, fuzzy-neural structures, and model predictive control.*

Contacts:
Technical University - Sofia, Branch Plovdiv
25 Tsanko Dustabanov St., 4000 Plovdiv, Bulgaria
tel: +359-32-659-555
e-mail: sevil.ahmed@gmail.com

Michail Petrov *was born in Pernik, Republic of Bulgaria in 1952. He received the master degree in electrical engineering in 1977 from Technical University in Prague, Czech Republic and the Ph.D. degree in control engineering from Mining University Sofia, Bulgaria. Currently, he is Full Professor at the Faculty of Electronics and Automation of the Technical University □ Sofia, branch Plovdiv, Bulgaria. He teaches Control Systems Design and his current research interests include fuzzy logic applications in control systems, applications of neural and fuzzy techniques to model predictive control, industrial process control and etc. He has (co)authored more than 50 papers in this area.*

Contacts:
Technical University - Sofia, Branch Plovdiv
25 Tsanko Dustabanov St., 4000 Plovdiv, Bulgaria
tel: +359-32-659-555
e-mail: mpetrov@tu-plovdiv.bg