Identification and Control of pH using Optimal Piecewise Linear Wiener Model

William Ipanaqué*. José Manrique*

*Piura University, PO Box 353 Piura Perú; e-mail: william.ipanaque@udep.pe; jose.manrique.s@gmail.com

Abstract: Wiener models consist of a linear dynamic element followed by a static non linear element. This paper shows a non linear model predictive control (NMPC) based on a piecewise linear; the Wiener model is applied on an experimental control of pH. The static nonlinear element of the Wiener model is approximated using piecewise linear function. Identification using optimal local linear model is applied and parameter estimation as well as partitioning of the local linear models is simultaneously obtained. The techniques are then applied to an experimental control of pH and the performance of NMPC is shown.

Keywords: Nonlinear model predictive control; Block-structured model; Wiener model, non linear identification, local linear model.

1. INTRODUCTION

Model Predictive Control (MPC) is one of the design techniques that can be proven to stabilize process in the presence of nonlinearities, Quin and Badgwell (1998). MPC allows to use the detailed knowledge of a process in the form of a dynamic model, as an aid to controlling that process. Linear MPC used models which have been linearised about an operating point as an aid to predict the responses of the controlled process.

Many processes are sufficiently nonlinear to preclude good applications of linear MPC technology. Such processes, as pH control, include highly non linear processes that operate near a fixed operating point. This has led to development of non linear model predictive control (NMPC) in which a more accurate model is used for process prediction and optimization.

Some papers have reported controllers incorporating nonlinear models such as neural networks, as Patie et al. (2010) and Saindonat (1998); Volterra series model, Genceli and Nikolau (1995). Wiener models are useful in representing the nonlinearities of a process without complications associated with general non linear operators. A Wiener nonlinear system is a cascade of a linear dynamic system followed by a static non linearity and is effective in representing a class of nonlinear systems. Identification of Wiener systems has been proposed in several papers: literature: Greblicki (1992), Gomez, et al. (2004), Kalafatis et. Al (1995), Kozek and Jovanovic (2003), Sung and Lee (2004).

The pH processes are frequently encountered in the chemical process industries and represent a significant challenge in terms of their identification and control. The pH process is considered in the literature to have structure of Wiener systems with linear element describing the mixing dynamic of the reagent streams in a stirred vessel and the nonlinear element representing a static nonlinear titration curve which gives pH as a function of the chemical species, Niemi and Julia (1977).

Piecewise local linear models is a class of Wiener models, where the parameters of a linear transfer function are valid for all segments, while each gain and bias parameter is only valid inside of one data segment e.g. Cervantes et al (2003), Shafiee et al (2006). Therefore, the non linear characteristic is given by a piecewise linear function.


In this paper we report the application to pH module, of an identification algorithm using optimal local linear models based on Kosek (2008). This algorithm automatically chooses the number of models and develops an optimal partitioning. Other approaches require a predefined number of local models for satisfying performance, e.g. Vörös (2003), Zhu (2002), thus the algorithm applied, on the basis of quadratic criteria, finds the number of optimal partitions. Wiener models and the applied identification algorithm are described in section 2. Section 3 illustrates the pH process and the experimental module where identification and predictive control is implemented. Section 4 shows the experimental results of the identification and predictive control based on the Wiener model. Finally, some conclusions are given.
2. WIENER MODEL

The Wiener model (Fig. 1) is composed of a linear dynamic element which is followed by a non-linear static function. The linear part can be modeled by a transfer function. The nonlinear element ranging from simple algebraic functions to complex neural network, for control purposes this function must have an inverse, Norquay et al, (1999); e.g., polynomials, splines, basis functions, radial basis functions, etc., Bai (2003), Celka et al (2001), Saintdomat et al, (1999).

\[ X'(k) = [-y(k-1) ... -y(k-n) u(k-1) ... u(k-m)] \]
\[ \theta_i = [\theta_1 ... \theta_n k_1 b_1 ... k_n b_n d_i (1 + \alpha_1 + \alpha_2 + ... + \alpha_n)] \]

Making:
\[ \begin{align*}
\theta_1 &= a_1 \\
\theta_n &= a_n \\
\theta_{n+1} &= k_1 b_1 \\
&\vdots \\
\theta_{n+m} &= k_i b_m \\
\theta_{n+m+1} &= d_i (1 + \alpha_1 + \alpha_2 + ... + \alpha_n)
\end{align*} \]

We can represent the vector \( \theta_i \) as:
\[ \theta_i = [\theta_1, ..., \theta_i, ..., \theta_{n+m}, \theta_{n+m+1}] \]
\( e_i(k) \): residual or equation error

In general the static gain \( K_{ne} \) of the linear transfer function is:
\[ k_n = \frac{b_1 + ... + b_m}{1 + \alpha_1 + ... + \alpha_n} \]

Since that gain \( k_n \) is different from one, in order to normalize, we use an intermediate variable \( \hat{v} \):
\[ \hat{v} = \frac{v}{k_n} \]

The output prediction is computed by the new equation:
\[ \hat{y} = k y + d \]
\[ \hat{y} = G \left( \frac{b_1 + ... + b_m}{1 + \alpha_1 + ... + \alpha_n} \right) v + d \]

Parameters of local linear model are determined by:
\[ \hat{y} = \frac{d}{v} Y = \frac{\theta_{n+1} + ... + \theta_{n+m}}{1 + \alpha_1 + ... + \alpha_n} \]
\[ \delta = \frac{\theta_{n+m+1}}{1 + \alpha_1 + ... + \alpha_n} \]

Transfer function for each model is redefined in the following way:
\[ G(z) = b_1 z^{-1} + ... b_m z^{-m} \]

Where:
\[ \theta = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_n \\ \theta_{n+1} \\ \theta_{n+2} \\ \vdots \\ \theta_{n+m} \\ \theta_{n+m+1} \end{bmatrix} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \\ k_1 b_1 \\ k_2 b_2 \\ \vdots \\ k_m b_m \end{bmatrix} \]
\[ d_i (1 + \alpha_1 + \alpha_2 + ... + \alpha_n) \]
**Criteria for optimizing the number of local models**

Fundamental idea of the identification algorithm is based on Kozek and Sinanovic (2008). Testing the hypothesis that the variance $\sigma_{e,i}^2$ of $e_i(k)$, defined by (19), for each local model is only caused by the normally distributed noise.

The residual $\sigma_{e,i}^2$ of each local model is composed of noise covariance $\sigma_{n,i}^2$ and bias covariance $\sigma_{b,i}^2$:

$$\sigma_{e,i}^2 = \sigma_{n,i}^2 + \sigma_{b,i}^2$$  \hspace{1cm} (15)

If the local model is valid, $\sigma_{b,i}^2 = 0$, then:

$$\sigma_{e,i}^2 = \sigma_{n,i}^2$$  \hspace{1cm} (16)

Alternatively, this leads to propose the following hypothesis:

$$\sigma_{e,i}^2 > \sigma_{n,i}^2$$  \hspace{1cm} (17)

With the scope of validating the proposed hypothesis, (16) and (17), critical values of probability $\chi^2$-square function is used; where: $a$: level of reliability, $v$: degree of freedom; where $v = S_i - n - m$ and $S_i$ is the number of samples for the current local model.

As long as:

$$\chi^2_{a,v} < \chi^2_{a,v}$$  \hspace{1cm} (18)

The proposed hypothesis will be satisfied, and therefore the local model will be valid. When this hypothesis failed, a new model must be initialized.

For a ramp like input signal with additional white noise for instance Fig. 3, and for a starting parameter of the model, $e_i(k)$ is computed:

$$e_i(k) = y_i(k) - \hat{y}(k)$$  \hspace{1cm} (19)

where: $\hat{y}(k)$: estimated output; $y(k)$: output value. Then the error’s variance is determined, $\chi^2$-square distribution values are obtained in statistic tables for a determined degree of freedom, and for certain level of significance.

If

$$\chi^2_{a,v} < \chi^2_{a,v}$$  \hspace{1cm} (20)

The model computed is valid and therefore a new set of data is taken $(u(k), y(k))$ and the previous procedure is repeated.

If

$$\chi^2_{a,v} > \chi^2_{a,v}$$: A new local model is started, saving the parameters of the previously estimated model thus successively until reach the entire set of data.

**Recursive Least - Squares**

For the estimating of the model’s parameters, minimal recursive squares are used,

$$\gamma(k) = \frac{P(k)x(k+1)}{1+x^T(k+1)P(k)x(k+1)}$$  \hspace{1cm} (21)

$$\theta_i(k+1) = \theta_i(k) + \gamma(k)[\gamma(k+1) - x^T(k+1)\theta_i(k)]$$  \hspace{1cm} (22)

$$P(k+1) = P(k) - x(k+1)^2P(k)x(k+1)$$  \hspace{1cm} (23)

Where:

$$x(k) = [-y(k-1) ... -y(k-n) u(k-1) - u(k-n)]$$  \hspace{1cm} (24)

$$\theta_i = [a_1, ....... , a_{m}, b_1, ...... , b_{m}, d_1(1+a_1 + a_2 + ... + a_m)]$$  \hspace{1cm} (25)

In order to starting the algorithm, the following assumptions could be considered:

- Values of $P(k)$ are initiated in a rather big value, e.g.,

$$P(0) = \begin{bmatrix} 1000000 & 0 & 0 \\ 0 & 0 & 1000000 \\ 0 & 0 & 1000000 \end{bmatrix}$$  \hspace{1cm} (26)

- The values of $\theta$ are initiated to zero

$$\theta(0) = 0$$  \hspace{1cm} (27)

- Once the first local model has been calculated, the new model are update with the values of $P(k)$ and $\theta(k)$:

$$P_{r,i}(0) = \begin{bmatrix} P_{n,i} & 0 \\ 0 & P_{r,i} \end{bmatrix}$$  \hspace{1cm} (28)

Where $P_{n,i}$ is a scalar value and $P_{r,i}$ a diagonal matrix $2\times2$:

$$\text{diag}(P_{n,i}) = \begin{bmatrix} \sigma_1^2 & \sigma_2^2 & \ldots & \sigma_{\infty}^2 \end{bmatrix}$$  \hspace{1cm} (29)

$$\text{diag}(P_{r,i}) = 1*10^6 \begin{bmatrix} 1 & 1 & ... & 1 \end{bmatrix}$$  \hspace{1cm} (30)

The value of $\theta_{r,i}$ is initialized with:

$$\theta_{r,i}(0) = [a_1, ...... , a_{m}, r_1b_1, ...... , r_mb_m, r_2]$$  \hspace{1cm} (31)

Where $r_1, r_2$ are random numbers in the interval $[-1,1]$.

3. pH PROCESS

A schematic of pH pilot plant which was used to evaluate the proposed identification and control procedure is shown in Fig. 5.

The mechanistic model can be described as:

$$F_S = F_A + F_B$$  \hspace{1cm} (32)

$$F^r \frac{dx_A}{dt} = F_{ACD} - F_SX_A$$  \hspace{1cm} (33)

$$F^r \frac{dx_B}{dt} = F_{ACD} - F_SX_B$$  \hspace{1cm} (34)

Where:

- $F_S$: reactor’s output flow. (l/s)
- $F_A$: HCl input flow. (l/s)
- $F_B$: NaOH input flow. (l/s)
- $X_A$: Concentration of HCl acid. (mol/l)
- $X_B$: Concentration of the NaOH. (mol/l)
$C_{OA}$ : Initial concentration of HCl (mol/l)  
$C_{OB}$ : Initial concentration of NaOH (mol/l)  
t : Time (s)  
$V$ : Reactor’s volume (l)  

Chemical equations

The system is formed by the NaOH as base, and HCl as acid. The neutralization process between a strong acid and a strong base are described by:

\[ \text{HCl}(ac) \rightarrow H^+(ac) + Cl^-(ac) \quad (35) \]
\[ \text{NaOH}(s) \rightarrow Na^+(ac) + OH^-(ac) \quad (36) \]
\[ H_2O \Leftrightarrow H^+ + OH^- \quad (37) \]
\[ Kw = [H^+][OH^-] \quad (38) \]

From electro neutrality:

\[ [Na^+] + [H^+] = [Cl^-] + [OH^-] \quad (39) \]
\[ Q = [H^+] - [OH^-] = [Cl^-] - [Na^-] \quad (40) \]
\[ [H^+] - [Cl^-] = [Na^+] \quad (41) \]
\[ [OH^-] = [NaOH] \quad (42) \]

From (32), (33), (34) and (40); we can determine the following equation:

\[ \frac{dQ}{dt} = \frac{1}{V} (F_s C_a - F_b C_b - (F_s + F_b) Q) \quad (43) \]

The relation between $Q$, $k_w$ and pH is:

\[ Q = 10^{-14} - \frac{k_w}{10^{14}} \quad (44) \]

Equations (43) and (44) describe the dynamic behavior of the reaction between a strong acid and a strong base.

**pH Module**

From Fig. 4; pump (A) is an on/off Walcham pump EZB30D2VC to dosifier HCl, flow max is 0.21 l/m. Pump (B) to dosifier NaOH flow, is a variable speed pump Walcham model EHE35E2-VC, max flow is 0.52 l/min. The magnet mixer is from Velp AGE model, able to provide speed until 1200 r.p.m. for 8 l. The pH probe located in the reactor is from Hanna Instrument model HI 1006. This probe is connected to the indicator Hanna Instrument model HI 8510E420 which provides signal output 4-20 mA. Substances to be mixed are $N_{NaOH}$ and $HCl$. Table 1 shows nominal values for experiments.

Fig. 4. Experimental Module

### 4. EXPERIMENTAL RESULTS

To identify the process, an ramp signal with added noise was applied to the pump (B), starting near to $2 \times 10^{-3}$ l/s until reach $6.5 \times 10^{-3}$ l/s. Fig. 5 shows input signal and output response of the pilot plant.

Models obtained applying the procedure described, are summarized in Table 2. Fig. 6 shows different local linear models. Defining $v_n$ as the noise variance we can see that for $v_n=0.5$ we obtain the optimal model (FIT 87.47 %) while for $v_n=50$ the model has a FIT of 58.1%.

<table>
<thead>
<tr>
<th>Table 1. Nominal values</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Volume</strong></td>
</tr>
<tr>
<td><strong>HCl concentration</strong></td>
</tr>
<tr>
<td><strong>NaOH concentration</strong></td>
</tr>
<tr>
<td><strong>HCl flow</strong></td>
</tr>
<tr>
<td><strong>NaOH flow</strong></td>
</tr>
<tr>
<td><strong>Control horizon</strong></td>
</tr>
<tr>
<td><strong>Prediction horizon</strong></td>
</tr>
<tr>
<td><strong>Input weighting</strong></td>
</tr>
</tbody>
</table>

Wiener model was incorporate into MPC Scheme and tuned using the simulation where the well known Generalized Predictive control, Clarke (1987), was used.

The predictive controller has been designed using only the parameters of the linear transfer function, it means that the regulator considered as output the variable $v(k)$. This implies that both the reference and the output variable must be converted into equivalent variable $v(k)$ using the inverse of the nonlinear function.

The control law has been estimated considering a horizon of prediction and control of 3. Moreover, the value that weights the input variable $u$ is equal to 16.

To evaluate the control, several pH setpoint changes were made. Fig. 7 shows the behavior of control with good performances.

<table>
<thead>
<tr>
<th>Table 2. Model parameters.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Range</strong></td>
</tr>
<tr>
<td>$k_i$</td>
</tr>
<tr>
<td>$M(z^{-1}) = 0.0479$</td>
</tr>
<tr>
<td>0.5</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>25</td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

To verify the behavior of the system when setpoint decreases, it was changed from 3.5 to 2.8 pH values. Results are shown in Fig. 8. Settling time was 17 minutes, the system does not present overshoots, despite the changes in set point was higher than before case. The control variable, shown in Fig. 8, changes between $4.4 \times 10^{-3}$ l/s to $2.4 \times 10^{-3}$ l/s of NaOH.

Behavior of control variables in the event of disturbances was evaluated: at instant time 3750 s, the amount of pumped HCl is diminished; when this disturbance occurs, the pH
immediately increases. The control loop brings it back to its setpoint decreasing the value of the control variable.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Volume</td>
<td>1.8</td>
<td>l</td>
</tr>
<tr>
<td>HCl conc.</td>
<td>0.005</td>
<td>Mol/l</td>
</tr>
<tr>
<td>NaOH conc.</td>
<td>0.005</td>
<td>Mol/l</td>
</tr>
<tr>
<td>HCl stream</td>
<td>0.0067</td>
<td>l/s</td>
</tr>
<tr>
<td>NaOH stream</td>
<td>0.002-0.007</td>
<td>l/s</td>
</tr>
<tr>
<td>Control horizon</td>
<td>3</td>
<td>---</td>
</tr>
<tr>
<td>Prediction horizon</td>
<td>3</td>
<td>---</td>
</tr>
</tbody>
</table>

Fig. 5. Input (up) and output (down) signal for identification of pH Process.

Fig. 6. Found models.

Fig. 7. pH (up) and NaOH flow (down) for setpoint 3.5

Fig. 8. pH (up) and NaOH flow (down) for setpoint 2.8

**Comparison between Predictive Regulator and PI**

A comparison using predictive regulators and using a PI controller to the same process have been made under the same nominal values of work.

The following figures show the results of using a PI regulator.

Fig. 9. PI control of pH

We can appreciate that the settling time of the process variable goes from 4 to 8 minutes with an overshoot of 30%.

The results of using a predictive regulator with control and predictive horizons of 3 and an input weighting of 16 are shown below.

Fig. 10. Using a predictive regulator

The settling times goes from 15 to 25 minutes but reducing the overshoot.

**5. CONCLUSIONS**

In this paper an identification algorithm using optimal local linear models was applied to pH process, in order to obtain Wiener model. The number of local linear models and parameter estimation is simultaneously obtained. The
results have demonstrated the benefits of MPC based on such Wiener model.

The model which is an important factor for prediction in MPC, is selected a Wiener model with local piecewise linear model. Wiener MPC was shown to posses good setpoint tracking and disturbance rejection features.

REFERENCES


