ICA2025OSLO
14th IWA Specialised Conference on Instrumentation, Control and Automation

# Overcoming Delays: Effective Control Strategies for Bipolar Membrane Electrodialysis (BMED) in Organic Acid Extraction

G. Herold[1,3,4,5], **, Q. Marien[2,4], K. Rabaey[3,4], M. Schneider[1,4,5], E. Torfs[1,5]

[1]Biomath, Ghent University, 9000 Ghent, Belgium, gaetan.herold@ugent.be

[2]CMET, Ghent University, 9000 Ghent, Belgium

[3]Traslab, Ghent University, 9000 Ghent, Belgium

[4]Centre for Advanced Process Technology for Urban Resource Recovery (CAPTURE), 9000 Ghent, Belgium

[5]modelEAU, Université Laval, QC G1V 0A6 Québec, Canada, elena.torfs@gci.ulaval.ca

**Abstract:** This study investigates control strategies for Bipolar Membrane Electrodialysis (BMED) systems used to recover succinic acid from organic waste fermentation. We compare the performance of various control strategies, including simple proportional (P) and proportional-integral-derivative (PID) controllers, with more sophisticated methods such as a Smith predictor. The predictor utilizes FOPDT (first-order plus dead-time) models to compensate for the process delay and improve control accuracy. Additionally, we explore an explicit mechanistic model that captures buffering dynamics and the effect of acidification and hydroxide ion addition to assist the Smith predictor.

Our findings demonstrate the potential of model-assisted Smith predictors for effective control in this self-regulated system. The final approach significantly reduces oscillations and achieves faster stabilization compared to simpler controllers, particularly when dealing with less buffered solutions.

**Keywords**: bioproduction, BMED, control, mechanistic model, Smith Predictor

The recovery of organic acids from fermentation processes represents a critical step in the transition to a bio-based economy by valorising organic waste into high-value biochemicals (Koutinas et al., 2014). Bipolar Membrane Electrodialysis (BMED) processes have emerged as an efficient technique to separate organic acids from fermentation broths and the impurities they contain (Handojo et al., 2019). In these applications, maintaining a stable pH in the fermenter remains crucial, as deviations can compromise bacterial viability and productivity (Chen et al., 2017; Tran et al., 2023).

By using BMED as an extraction technique, the dual challenges of acid recovery and pH regulation in the fermenter can be simultaneously addressed. By generating $H^+$ ions for succinic acid recovery and $OH^-$ ions for pH adjustment directly from water molecules, BMED can provide an energy-efficient solution, eliminating the need for external neutralizing agents such as NaOH (Wong et al., 2010). This self-regulation presents a unique challenge: $OH^-$ ions must counteract the continuous acidification in the reactor caused by bacterial metabolism but precise pH regulation is complicated by a delay introduced by the hydraulic residence time of the fermentation broth in the electrodialysis cell. This delay arises from

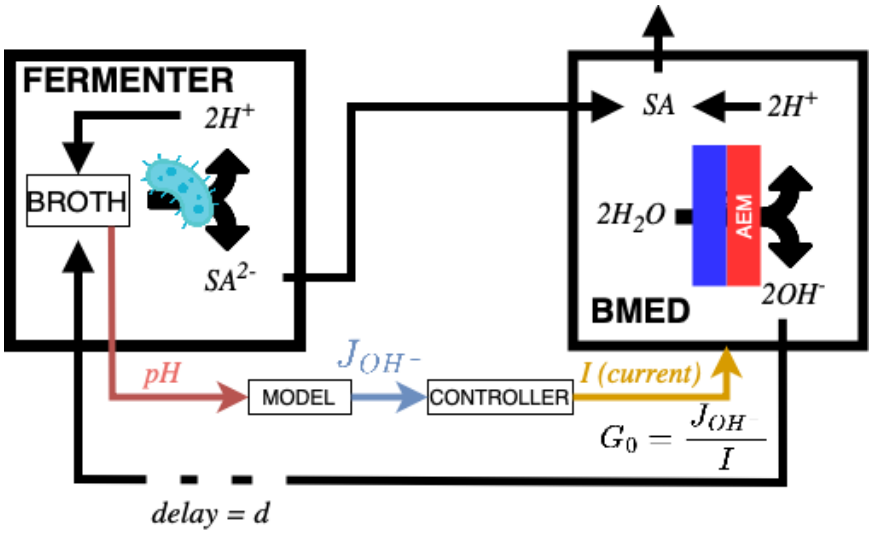


Figure 1 Process and control loop with model-based control

the volume of water circulated and the limited flow rate through the cross-flow filtration system that separates the BMED unit from the fermenter (Figure 1). Furthermore, the unidirectional nature of the control process—allowing only pH increases through $OH^-$ dosing—adds another layer of complexity to the problem (Michiels et al., 2020).

This study explores advanced control strategies for this self-regulated system with notable delay. We compare the performance of simple proportional (P) and proportional-integral-derivative (PID) controllers with more sophisticated approaches, including Smith predictors based on FOPDT functions derived through system identification. Furthermore, we incorporate mechanistic models accounting for buffering dynamics, bacterial acidification, and effect of $OH^-$ addition in order to improve the accuracy of the control and understand the added value of mechanistic-model based control.

Our findings provide valuable insights into the design of effective control strategies for fermentation-based organic acid production, offering a way to improve process stability and efficiency in industrial resource recovery applications.

**Methodological approach:**

The pH model for the broth comes from a broader model for BMED processes previously developed for similar applications (Herold et al., 2024). It is based on buffer equilibrium equations, the mass conservation equation for each buffer system and a proton equilibrium, capturing the dynamics of proton concentration in the broth.

We considered phosphate, succinate, and ammonium buffer systems—measured in the experimental broth environment—as well as the carbonate buffer system, which is known to be present. All these components are included for completeness, but at the targeted pH (~6.5), ammonium has minimal influence due to its high $pK_a$[1].

To test the control systems, we modelled the process by integrating a pH buffering model with a FOPDT system that accounts for the delay linking current to $OH^-$ addition in the fermentation broth. We tested four control strategies on the model: a simple proportional (P) controller, which had been experimentally already found to be insufficient for accurate control; a modified, enhanced proportional-integral-derivative (PID) controller with an asymmetric anti-windup for the integral term and a low-pass filter applied to the proportional term; and two Smith Predictors based on first-order systems with dead-time. The dead-time is identified through cross-correlation analysis. The first Smith Predictor assumes fermentation causes a constant pH decrease, meaning a constant change in $H^+$ concentration in the broth. This constant is determined when current is zero and subtracted from the whole pH data. The remaining dynamics are fitted with a first-order transfer function. The second Smith Predictor uses the full buffer model, which consider buffering effect of the weak acids in the solution. From the pH data, it deduces trough the model the hydroxyl flux in the broth $J_{OH^-}$[s] and determines a transfer function G[s] such that $J_{OH^-}$[s]=G[s]I[s], linking the current I[s] to the hydroxyl flux.

Transfer functions were fitted on a training set of 10 hours of extraction experiment carried out with a BMED system on a real fermentation broth, with a simple P controller. The acidification

[1] pKa of ammonium = 9.25 at 25°C

effect of fermentation has been replaced by the continuous addition of 0.85 g/L/h of succinic acid in the broth. Accuracy of the model has been controlled on a 2 hours validation set.

**Results and discussion:**

In regions where the pH is well-buffered, such as around pH 6.2 (near the first $pK_a$ of carbonate at 6.38), all controller options performed effectively with aggressive proportional control. In Figure 2, both Smith predictors were paired with a proportional controller $K_P = 7$ and compared to the proportional controller alone. The model-assisted Smith predictor significantly reduced oscillations and achieved faster stabilization, though it exhibited the same steady-state error (0.02) as a regular P controller. The simple Smith predictor also stabilized the system faster than the standalone proportional controller but resulted in a larger steady-state error (0.09).

When we set the control target at pH=6.7, where the solution is much less buffered, only the model assisted Smith predictor achieved a satisfactory performance.

In Figure 3, with a PID controller with a relatively aggressive proportional constant ($K_P = 4, K_I = 0.011, K_D = 0.1$), the simple Smith control presented a large stead-state error (0.29), and the PID failed to stabilise entirely as well as presented important oscillations (0.23).

In Figure 4, with a PID controller with a much less aggressive proportional constant ($K_P = 0.7, K_I = 0.007, K_D = 0.08$), the model assisted smith predictor stabilises faster but presents a larger maximal overshoot with respect to Figure 3 (0.51 vs 0.19). The other controllers fail to stabilise entirely and present large overshoots.

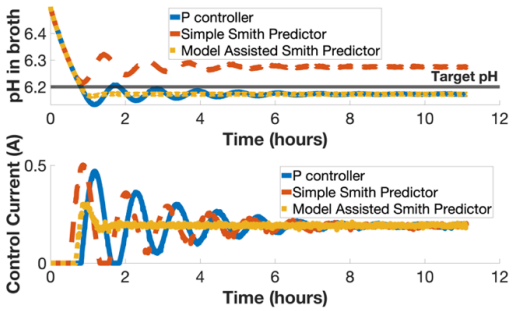


*Figure 2: pH start value = 6.5, pH control value = 6.2, $K_p$=7*

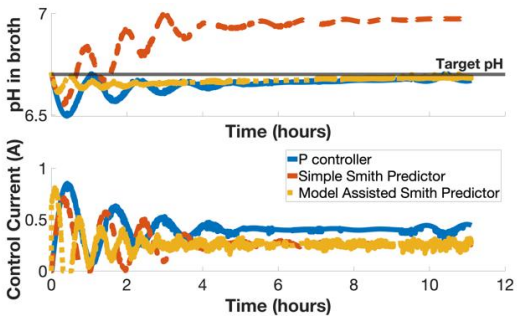


*Figure 3: pH start value = 6.71, pH control value = 6.7, $K_p$=4*

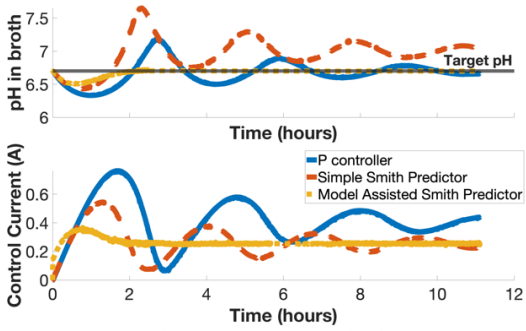


*Figure 4: pH start value = 6.71, pH control value = 6.7, Kp=0.7*

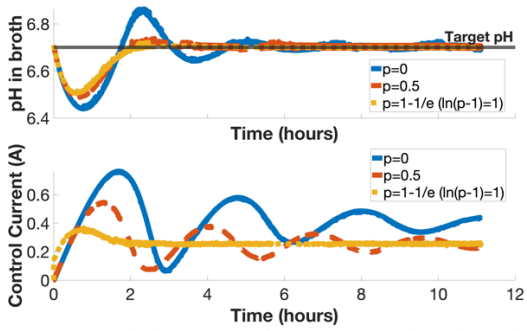


*Figure 5: Extended predictions for model assisted Smith predictor*

For a transfer function characterized by a relatively large time constant, the performance of the Smith predictor can be improved by extending the prediction beyond the dead time to capture a meaningful portion of the system's response (Figure 5). The extended prediction horizon is defined

as $d_e = d + \left(-ln(1-p)\right) * \tau$ , where $p$ represents the fraction of the step response captured, $d$ is the dead time, and $\tau$ is the time constant. This approach ensures that the prediction incorporates the system's gradual response following the dead time. $p$ should be tuned carefully as such an extension can amplify prediction errors, be more sensitive to noise, and increases computational demand.

The increased accuracy of the model assisted Smith predictor is coupled with increased computational complexity. Indeed, the P and PID controllers only need to calculate the control value at each step for a cost of $\mathcal{O}(1)$ [2]. The simple Smith predictor has to calculate the control value for the next $k = \frac{d_e}{\Delta t}$ steps, $\Delta t$ being the controller time step, at each step for a cost of $\mathcal{O}(k)$. The model assisted Smith predictor has to solve the linear equation system for the next $\frac{d_e}{\Delta t}$ steps at each step, resulting in a major overhead. The cost of solving such a system is in $\mathcal{O}\left(i(n^4 * m)\right)$, with *i* the number of iterations, *n* the number of variables (i.e. the number of ions considered), and *m* the number of equations (= *n* in our case of a square system) resulting in a total cost of $\mathcal{O}\left(k * i(n^4 * m)\right)$. The execution time for 100 steps with MATLAB 2024b (MathWorks, USA) on one core of an M3 Pro CPU (Apple, USA) is 0.0014 sec (SD=0.00082sec) for the simple Smith Predictor 0.0053sec (SD=0.00047sec) for the P/PID controller, and 23.218sec (SD=0.677sec) for the model assisted Smith Predictor (n=3).

## Conclusion

The results demonstrate the potential of model assisted Smith predictor for control of self-regulated BMED system for organic acid production and extraction. Over the next months, full experimental validation is planned as well as further control experiments including the full mechanistic model presented in (Herold et al., 2024) for predictive control purposes.

## REFERENCES

Chen, X., Wu, X., Jiang, S., & Li, X. (2017). Influence of pH and neutralizing agent on anaerobic succinic acid production by a Corynebacterium crenatum strain. *J Biosci Bioeng*, *124*(4), 439-444.

Handojo, L., Wardani, A. K., Regina, D., Bella, C., Kresnowati, M. T. A. P., & Wenten, I. G. (2019). Electro-membrane processes for organic acid recovery [10.1039/C8RA09227C]. *RSC advances*, *9*(14), 7854-7869.

Herold, G., Schneider, M. Y., Rabaey, K., & Torfs, E. (2024). Towards a Unified Framework for Modelling of Bipolar Membrane Electrodialysis for Resource Recovery Processes. Frontiers in Membrane Technology, Cham.

Koutinas, A. A., Vlysidis, A., Pleissner, D., et al. (2014). Valorization of industrial waste and by-product streams via fermentation for the production of chemicals and biopolymers [10.1039/C3CS60293A]. *Chemical Society Reviews*, *43*(8), 2587-2627.

Michiels, W., Baillieul, J., & Smad, T. (2020). Control of Linear Systems with Delay. In (pp. 1-7). Springer.

Tran, V. G., Mishra, S., Bhagwat, S. S., et al. (2023). An end-to-end pipeline for succinic acid production at an industrially relevant scale using Issatchenkia orientalis. *Nature Communications*, *14*(1), 6152.

Wong, M., Woodley, J. M., & Lye, G. J. (2010). Application of bipolar electrodialysis to E. coli fermentation for simultaneous acetate removal and pH control. *Biotechnol Lett*, *32*(8), 1053-1057.

[2] Landau Symbol, which describes the asymptotical behaviour of a function, here the computational cost.