Trajectory optimization of an oscillating industrial two-stage evaporator utilizing a Python-Aspen Plus Dynamics toolchain

Citation for the original published paper (version of record):
Nolin, M., Andersson, N., Nilsson, B. et al (2020). Trajectory optimization of an oscillating industrial two-stage evaporator utilizing a
Python-Aspen Plus Dynamics toolchain. Chemical Engineering Research and Design, 155: 12-17.
http://dx.doi.org/10.1016/j.cherd.2019.12.015

Contents lists available at ScienceDirect

Chemical Engineering Research and Design

journal homepage: www.elsevier.com/locate/cherd

Trajectory optimization of an oscillating industrial two-stage evaporator utilizing a Python-Aspen Plus Dynamics toolchain

Mikael Yamanee-Nolin[a,*], *Niklas Andersson*[a], *Bernt Nilsson*[a], *Mark Max-Hansen*[b], *Oleg Pajalic*[b]

[a] *Lund University, Faculty of Engineering, Department of Chemical Engineering, P.O. Box 124, SE-221 00 Lund, Sweden*
[b] *Perstorp AB, Industriparken, 284 80 Perstorp, Sweden*

ARTICLE INFO

Article history:
Received 20 September 2019
Received in revised form 18 November 2019
Accepted 14 December 2019
Available online 24 December 2019

Keywords:
Aspen Plus Dynamics
Python
Dynamic optimization
Derivative-free optimization
Evaporator system
Oscillations

ABSTRACT

Evaporators are integral parts of many separation processes across production industries, and they need to be well understood in order to be operated well, thereby enabling high resource-efficiency and productivity. In a previous investigation, the effects of disturbing oscillations in a two-stage evaporator system were quantified. In the current study, these oscillations were reduced through trajectory optimization using steam consumption as a temporally discretized decision variable, taking advantage of a dynamic process flowsheet model in Aspen Plus Dynamics (APD) employed as if it were a black-box model. The optimization was performed utilizing a Python-APD toolchain with the SciPy implementation of COBYLA. The optimal trajectory was able to successfully reduce the objective function value (including the product stream mass flow variance and a bang-bang penalty on the trajectory itself) to slightly less than 0.3 % of that of the nominal case, in which a time-invariant steam consumption was employed. This in turn grants opportunities to increase throughput of the process, leading to significant financial gains.

1. Introduction

Evaporator systems are key components of production processes across many different industries and disturbances (such as unexpected feed concentration perturbations) can become a real issue for process engineers. Considering concurrent demands on companies to increase productivity and resource-efficiency, and that evaporators are inherently energy intense, proper control and disturbance management is essential. In order to manage these demands better, model-based studies lend themselves as viable options.

Whilst there is plenty of published research on evaporator systems utilizing steady-state models, far fewer studies have been published involving dynamic simulation and control (Luyben, 2018); in addition to those listed by Luyben (2018), there are a few other studies of particular relevance to the current work, especially with regards to managing disturbances. For instance, Kumar et al. (2013) created a generalizable dynamic model that they used to gain insights into the behavior of a multi-effect evaporator system subjected to disturbances, whilst Adams et al. (2008) designed a control architecture to deal with oscillatory disturbances. Pitarch et al. (2017) implemented a real-time optimization strategy to boost the resource-efficiency of an evaporator system, showcasing the usefulness of non-linear programming (NLP) (Biegler, 2010). Moreover, disturbances of an oscillating nature can

* Corresponding author.
E-mail address: mikael.yamanee-nolin@chemeng.lth.se (M. Yamanee-Nolin).
https://doi.org/10.1016/j.cherd.2019.12.015

become especially problematic when they propagate through and decrease performance of an entire process (Yuan and Qin, 2012). This in turn prompted a previous in-silico investigation of an industrial evaporator system (Nolin et al., 2018), in which disturbing oscillations of the industrial system were analyzed and quantified for the first time. The oscillations were further found to be present throughout the whole system, effectively increasing steam consumption by 1 % and reducing productivity by 2 %, which is of economic concern due to the scales of the processing industries. Furthermore, since the evaporator system was bottlenecking the entire process, eliminating the oscillations at a low cost would translate into financial gains.

Considering the problem accounted for above, the aim of the current study was to eliminate the oscillations in the evaporator system reported by Nolin et al. (2018) through model-based dynamic NLP, taking advantage of the existing Aspen Plus Dynamics model used in that study. This optimization based solution would serve as an alternative to expensive retrofits or reconstruction. However, the built-in tools of the AspenTech software lack the necessary capabilities for advanced, model-based optimization studies. For instance, Aspen Plus does not include algorithms for multiobjective optimization, which severely limits the options for investigating a process (Muñoz López et al., 2018). Furthermore, it is unclear how to deal with dynamic optimization problems such as grade transitions in Aspen Plus Dynamics, which motivates the use of an external optimization architecture in conjunction with a dynamic flowsheet model (Negrellos-Ortiz et al., 2018). This way, it is possible to take advantage of the component and unit-operation libraries available in commercial flowsheeting software in combination with generalizable optimization tools. Therefore, the pre-existing Aspen Plus Dynamics model was employed as if it were a black box type of model in the sense that the input and output data of the model are the main concerns (in a similar fashion as Negrellos-Ortiz et al. (2018)). To perform the optimization, the model was coupled to a modified version of the COM-based Python Module Coupler (PyMoC) cosimulation tool, which was first presented in a cosimulation-based optimization study that included a version of the current evaporator system model (Yamanee-Nolin et al., 2019). In order to easily implement general inequality constraint equations as well as decision variable boundaries specifically, the SciPy implementation of the COBYLA algorithm (Powell, 1994) was used to perform the optimization.

The results show that it is possible to minimize the oscillations through trajectory optimization, leading to an increase in throughput, a reduction in steam consumption, while product purity was kept at a satisfying level. In addition to the performance improvements in the evaporator system, the minimization of the oscillations further means that throughput can be increased even further, and the downstream propagation of the oscillations will pose less of a problem, potentially leading to positive effects in the subsequent subsystems as well.

The remainder of this paper is structured as follows: section 2 introduces the process and the model that is used for the current study. This is followed by the presentation of the mathematical problem formulation along with a brief description of the modified PyMoC architecture with which the optimization problem is solved. Results are provided and discussed in section 3, with a summary of the major conclusions presented in the final section.

2. Materials and Methods

This section begins with a description of the investigated process along with the Aspen Plus Dynamics model used for the study. The formulation of the trajectory optimization problem then follows, along with a description of how it was solved by using the PyMoC tool to transfer data between the optimizer and the model.

2.1. *Process and model*

The studied evaporator system is the first stage of a series of separation processes for purifying a polyalcohol component at Perstorp AB and it is presented schematically in Fig. 1. The system starts with a balance tank working as a buffer vessel between the upstream batch-based system and the evaporator system; the upstream system intermittently feeds the balance tank with a solution that is highly concentrated with respect to the desired product (The 'Feed' stream in Fig. 1). The balance tank is also fed by a dilute recycling stream from a part of the downstream processing. The balance tank mixture feeds a stripping column, the purpose of which is to remove water. The bottom stream of the stripping column is then heated by a stream from a mechanical vapor recompression (MVR) circuit, which also contains some added steam and water (the latter for direct intercooling purposes), before it is fed to the first evaporator stage. The top stream of the first evaporator stage drives the stripping column, whereas the bottom is fed to the second stage. The bottom of the second stage is the product stream, named 'Prod' in Fig. 1, which is sent downstream for further purification.

The process was subject to oscillatory disturbances, which have been analyzed and quantified in-silico previously (Nolin et al., 2018). These disturbances are generated by a combination of how the process was originally constructed, then retrofitted and currently operated. The main cause is the fact that the concentrated feed stream is intermittent, whilst the diluted recycling stream and stripping column feed stream are both continuous. This will affect the liquid level in the tank, as well as how the concentration changes over time. Essentially, the feed from the batch system will make the concentration of product in the balance tank increase rapidly, and the material in the balance tank will subsequently be diluted by the recycling stream. As the stripping column feed stream is continuous, this diluting effect propagates into the whole system as the stripping column feed is successively diluted over the course of a cycle. This cycle restarts the next time the feed sequence is activated, which leads to the concentration increasing again, giving rise to performance-worsening oscillations that can propagate, leading to unnecessarily high steam consumption as well as production rate margin.

To study these dynamic disturbances, a dynamic model of the process was needed. Aspen Plus, along with its dynamic counterpart Aspen Plus Dynamics, were used as modeling and simulation tools for this purpose. This is a combination which has been used as an efficient computer-aided process engineering tool kit for many different studies concerned with the dynamics of vapor-liquid systems, with two good examples being the studies presented by Bildea and Kiss (2011) and Zhang et al. (2018). The current model was based on the one presented by Nolin et al. (2018), albeit run at a different nominal case setting. The model was constructed using the standard blocks available in Aspen Plus; *HeatX* and *Flash2* were

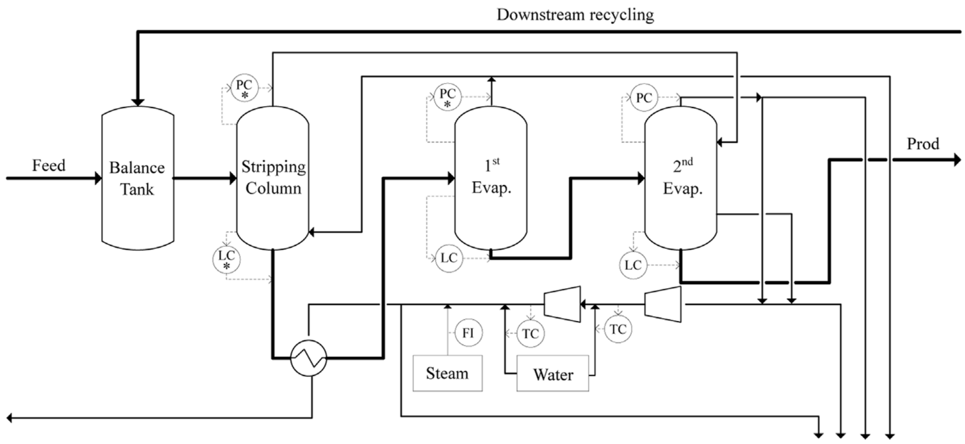

Fig. 1 – The modeled process.

used to model the evaporator stages, and a *RadFrac* column was used for the stripper. The MVR circuit was modeled using a combination of compressors, and mixers for adding water and steam. The steady state model built in Aspen Plus was then converted as a *flow-driven* simulation to Aspen Plus Dynamics using the built-in tool for that purpose. During conversion, controllers were automatically added and were decided to be kept and used at their default settings, as their positions for the most part reflected the real plant. The controllers that did not reflect real controller positions are marked by asterisk in Fig. 1; these are used in the model to mimic pressure-driven flow related phenomena, which otherwise would risk unrealistic behavior in terms of mass flow rates and pressure changes in the simulation. Furthermore, there was no boiling-point elevation control in place in the current model (as opposed to the previous study (Nolin et al., 2018)), as the use of steam consumption as the decision variable was desirable. Finally, the intermittent stream feeding the balance tank was modeled using a 'Task' in Aspen Plus Dynamics, which was programmed to mirror the real-life behavior where the balance tank is nearly filled at a specific low liquid level condition.

2.2. *Trajectory optimization using PyMoC*

The goal of the study was to minimize the product stream mass flow oscillations over eight hours by utilizing the steam consumption as a decision variable discretized in time. This was chosen as a decision variable due to expectations (based on a development of the conclusions in a previous study (Yamanee-Nolin et al., 2019)) that a form of counter-phase on the steam consumption could be found that would minimize the oscillations. The variance (σ^2) of the product stream mass flow was used as a measure of the oscillations for the objective function. Furthermore, a bang-bang trajectory (e.g. a trajectory consistently alternating between and only between its lower and upper bounds (Lang and Biegler, 2007)) could lead to long-term premature equipment fatigue from regular and significant changes in temperatures and/or pressures in the system. Therefore, an additive penalty term was introduced in the objective in order to make the resulting trajectory smoother. The term penalizes the square of the signal difference of two neighboring horizons. Finally, the sum of the variance and the penalty is normalized with respect to the nominal objective value, $\phi_{nom} = \sigma^2(w_{PROD}(t, u_{nom}))$.

To account for process requirements in place, inequality constraints were formulated for the averages of the mass flow and purity of the product stream. In order to give the optimizer some leniency whilst maintaining process demands, these averages were required to be greater or equal to 99.8 % of the average of the nominal run, giving $\beta = 0.998$. Furthermore, bounds were set on each of the discretized decision variables so that the steam consumption would be kept between half and twice the average steam consumption of the nominal case at all times.

The problem was solved utilizing a single-shooting approach and was mathematically formulated as follows:

$$\begin{aligned}
\underset{\mathbf{u}}{\text{minimize}} \quad & \phi(\mathbf{u}) = \frac{\sigma^2(w_{PROD}(t, \mathbf{u})) + \rho(\mathbf{u})}{\phi_{nom}} \\
\text{s.t.} \quad & \text{Flowsheet model} \\
& \bar{w}_{PROD} - \beta \cdot \bar{w}_{PROD,\ nom} \geq 0 \\
& \bar{P}_{PROD} - \beta \cdot \bar{P}_{PROD,\ nom} \geq 0 \\
& 0.5 \cdot u_{nom} \leq u \leq 2 \cdot u_{nom},\ \mathbf{u} = [u_1, \ldots, u_{N_u}]
\end{aligned}$$

where ϕ is the objective function and u the (discretized) decision variables; $\rho(\mathbf{u}) = \Sigma_{k=1}^{N_u - 1} \Delta u_k^2$ is the bang-bang penalty and R_u is a case-specific weight to scale it; w is mass flow rate; P is purity; $\bar{w}$ and $\bar{P}$ denote the average of those variables over the simulated time period; subscript *PROD* refers to the stream thus named in Fig. 1; and subscript *nom* refers to the nominal case.

To solve the problem, derivative-free optimization (DFO) was considered. DFO algorithms can be classified as *stochastic* or *deterministic* algorithms, depending on whether or not an algorithm takes random steps in its minimization procedure (Rios and Sahinidis, 2013). An example of deterministic DFO algorithms is the Nelder-Mead simplex algorithm, whereas stochastic examples include simulated annealing and particle swarm algorithms (Rios and Sahinidis, 2013). As presented by Negrellos-Ortiz et al. (2016), DFO has previously been suggested to serve as an effective alternative in solving dynamic optimization problems at the engineering-level (as defined by Biegler (2010)), where gradient information may be unavailable or unreliable in practice. Negrellos-Ortiz et al. furthermore showed the effectiveness of deterministic DFO techniques, as they utilized Powell's BOBYQA algorithm (Powell, 2009) to perform dynamic product transitions for a set of different reactors (Negrellos-Ortiz et al., 2016) as well as for an air separation unit (Negrellos-Ortiz et al., 2018). In both these studies, they utilized Aspen Plus Dynamics models as a type of black-box model, only concerned with inputs and outputs. Another successful use of DFO algorithms for trajectory optimization has been presented by Mohd Fuad et al. (2012), who employed Powell's COBYLA algorithm (Powell, 1994) to find the optimal trajectory for long-term catalyst deactivation. The deterministic DFO approach was thus chosen to solve the current optimization problem and stochastic approaches were left for future studies. Whilst both BOBYQA and COBYLA were considered as potential algorithms, COBYLA was eventually chosen in order to easily implement the constraints and bounds of the problem. The SciPy implementation of COBYLA was successfully used to solve the problem with the final accuracy tolerance left at the default value of 10^{-6}, the absolute constraint violation tolerance set to zero, and the nominal trajectory (i.e. flat) used as an initial trajectory.

In the current work, the PyMoC algorithm first presented in (Yamanee-Nolin et al., 2019) was modified in order to employ a direct, sequential approach to trajectory optimization (Biegler, 2010). In this approach, the originally infinite-dimensional problem is recast as a finite-dimensional problem through discretization of the decision variable along the temporal axis. For cosimulation purposes, PyMoC utilizes a zero-order hold analogy for data transfer during cosimulation. This data transfer strategy makes use of synchronization points (SPs) at which simulation is paused for data transfer; values in different modules are updated and then held constant for a horizon (τ). For trajectory optimization purposes, PyMoC and the direct, sequential approach were as such found to work nearly seamlessly together, as the SPs in PyMoC were possible to use for updating the decision variable signal for the next horizon instead of transferring data between modules. Due to this good fit between PyMoC and the chosen optimization approach, it was sufficient to add a trajectory optimization wrapper to PyMoC that accepts the decision variable vector from the optimizer and injects the relevant information at the correct SPs. Furthermore, the dynamic optimization problem in the current work was solved in an open-loop fashion similar to Sellberg et al. (2018), for eight simulation hours with ten decision horizons per simulation hour giving $N_u = 80$ and $\tau = 0.1$ h.

In order to provide better insight into how a user can establish the COM based link between Python and Aspen Plus Dynamics, a basic example is provided in Plate 1. This example includes how to set up such a connection (given some model), as well as reading data from and writing data to generically named parts (blocks/streams) of the model. These functionalities are essential and can be further built upon in order to carry out a customized optimization study, where an objective function can be composed and supplied to any of the algorithms that are freely available to all Python users. As for the current study, a schematic overview of the optimization process, starting at the optimizer call, is presented in Fig. 2. COBYLA is used for evaluating the objective function, which in turn utilizes PyMoC for transferring data between the optimizer and the model during trajectory optimization. PyMoC will take the full decision variable vector as an input, and it will then loop through the vector and advance the simulation one synchronization point and horizon at a time, until the set final time of eight hours is reached. The simulation output will then be sent back to the objective function for evaluation and updating of the decision variable vector until convergence is achieved.

```
import win32com.client

if __name__ == " __main__ ":
    path = ’C:\\filepath\\’
    model_name = ’filename.dynf’

    # Create the Python -Aspen connection via COM
    adyn = win32com.client.Dispatch(’AD Application’)

    # Make Aspen window visible, activate,
    # and load specified model
    adyn.Visible = True
    adyn.activate()
    adyn.OpenDocument(path+model_name)

    # Create model object handles
    sim     = adyn.Simulation
    fsheet  = sim.Flowsheet
    streams = fsheet.Streams
    blocks  = fsheet.Blocks

    # Enable recording of variable history
    sim.options.TimeSettings.RecordHistory = True

    # Set end time to 2 hours and run simulation
    sim.endtime = 2
    sim.run(1) # Synchronous (1), runs to end.

    # Retrieve temperature in Block B1
    Block1_T = blocks(’B1’).T.Value

    # Set Stream S1 mass flow rate to 100 kg/h
    streams (’S1 ’).FmR.Value = 100
```

Plate 1 – An example of how to establish the COM based connection between Python and Aspen Plus Dynamics, as well as for reading and writing data. This can be extended into a non-linear program by calling an available optimization algorithm and supplying a custom-made objective function to it.

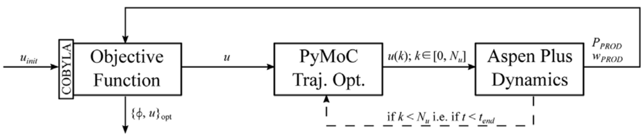

Fig. 2 – A schematic of the optimization process: COBYLA evaluates the objective function, which in turn makes use of PyMoC. PyMoC accepts the decision variable vector as a whole as input, and uses a zero-order hold strategy to implement the discretized decision variable at the correct synchronization points of the simulation, essentially through looping of the vector.

3. Results and discussion

In Fig. 3, the results from the nominal and the optimized runs are presented, normalized with respect to the mean value of the nominal run. The oscillations of the nominal operating conditions using a constant steam consumption are shown in Fig. 3a. For the current nominal operating point, the oscillations were quantified to be at around ±1%, thus having a negative effect on the production rate due to capacity constraints and safety margins of the process. However, as seen in Fig. 3b, the optimal trajectory is able to increase the average production rate slightly; it also brings the oscillations to a much lower level whilst lowering the average steam consumption to 69 % of the nominal operating point, which in part is made possible given that the average purity was allowed to decrease. That a reduction of the steam consumption is possible further indicates that the method for dealing with the

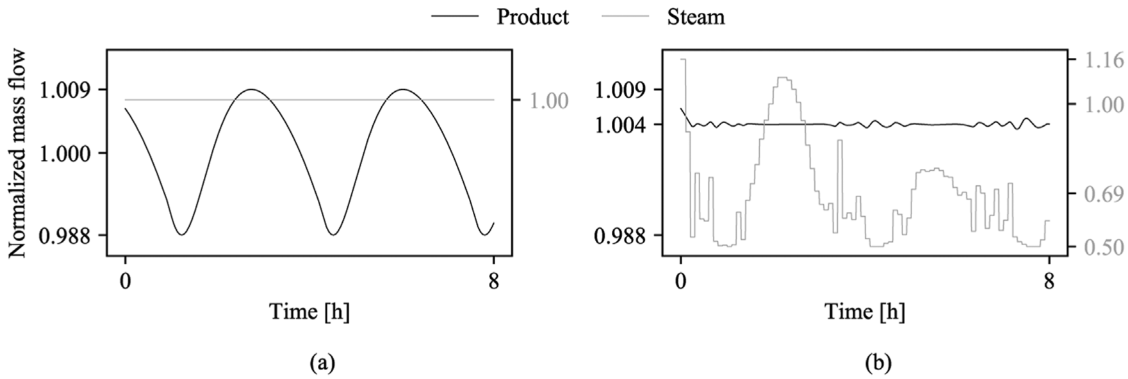

Fig. 3 – The stream results from the nominal and optimal runs. The black line is the product stream, and the gray line is the steam consumption (i.e. the discretized decision variable). With an optimized steam trajectory, the product mass flow oscillations have been dramatically reduced.

oscillations in the plant thus far has been to use an excessive amount of steam. This means that by using an optimal trajectory on the steam consumption in order to minimize the oscillations, the process can be improved in terms of both production rate as well as resource-efficiency whilst the product purity constraint is in place and satisfied. This in turn translates to significant financial and environmental gains.

However, even though the objective was formulated to minimize the occurrence of bang-bang optimization, we still see this at times, especially towards the end of the optimal trajectory. This is most probably due to the fact that the effect of the decision variables does not have time to propagate through the system before the final time of the whole optimization horizon, promoting relatively extreme behavior. The behavior where the signal approaches a constant value and at the end of the whole horizon makes short and sudden moves away from that value is called the turnpike effect, according to Faulwasser et al. (2017). Rawlings and Amrit (2009) also describe and provide an example of the turnpike effect. There have been many studies regarding how to avoid or remedy stability issues of the turnpike effect. Methods that have been suggested include, for instance, employing an infinite-horizon approach (Würth et al., 2009) or by formulating terminal conditions, e.g. equality constraints ensuring closed-loop stability, as reviewed by Mayne et al. (2000). Whilst it is an interesting issue, implementing the aforementioned methods for counteracting the turnpike effect was not performed in the current study.

Another interesting detail in the results is the amplitude reduction between the two major peaks in the optimal steam trajectory. The reduction indicates that at the start of the simulation, the process carries a lot of momentum that is counteracted by the first peak, which in turn reduces the workload required of the second peak. This difference in workload is due to an integrating effect of the product dilution happening as a result of a reduced steam consumption. There is thus potential for lowering the average steam consumption even more over time, which has to be studied further. However, it is important to note that this would still be subject to the product purity constraint.

The purity results are presented in Fig. 4, again normalized with respect to the mean value of the nominal run. The purity constraint based on downstream requirements is satisfied in the optimized scenario, with an average of 99.8 % of the nominal average. The purity decrease in the optimal case is to be expected as a result from reducing the average steam consumption, since using less steam will evaporate less water in the system. This decreasing trend is not expected

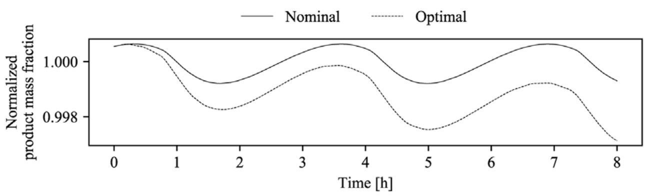

Fig. 4 – The product purity from the nominal and optimal runs.

to constitute a major issue, since the previously performed sensitivity analysis (Nolin et al., 2018) showed that the purity only changes very little with the different perturbation cases. However, since the results of the sensitivity analysis were assumed to depend to a certain extent on the tight boiling-point elevation control, which is overridden in the model of the current study, the trend should be further investigated in order to ensure the purity is kept at a required level. The downstream effect of the purity oscillations, which are present in the nominal case as well as in the optimal case, should also be investigated in order to ascertain to what extent the oscillations influence the rest of the purification process and to determine how to minimize any potential negative impact.

The nominal cases of the previous cosimulation-based study (Yamanee-Nolin et al., 2019) and the current work are not the same, but it is still both possible and interesting to compare the results between the current trajectory (time-variant) optimization and the previous time-invariant optimization. In the cosimulation study (Yamanee-Nolin et al., 2019), the max-min difference of the oscillations were reduced by a substantial 27 % simply by setting the steam consumption at a specific (and time-invariant) level, leading to possibilities for increasing production rate whilst satisfying all purity and production rate constraints. In the current work, the oscillations have been reduced even further by utilizing a trajectory, with the max-min difference having been reduced by 85 %. This reduction is actually in one sense slightly conservatively stated since the maximum value appears only once in the optimized case of the current work (at the start); this is in contrast to the nominal case of the current work, as well as the optimal case in the cosimulation study (Yamanee-Nolin et al., 2019), in both of which it appears in each oscillatory cycle. Furthermore, the objective function, i.e. the sum of the variance and the additive bang-bang penalties, was in the current work decreased by 99.7 % in the optimal case compared to the nominal case. It should here be duly noted that the nominal case essentially does not include the bang-bang penalty as it employs a time-invariant value on the steam consumption, whilst the optimal case actually does include the penalty.

4. Conclusions

The optimal trajectory is able to minimize the oscillations, reducing them by 99.7 % compared to the nominal case, whilst satisfying constraints. By using a steam trajectory, it is furthermore possible to reduce the average steam consumption by 31 % over the simulated period. This reduction is connected to less evaporated water in the system, but with satisfied constraints on both productivity and purity, the product requirements are met at this stage. As an alternative to reconstructing the evaporator system, implementing steam trajectories should therefore be regarded an idea to consider. Future studies should focus on implementation of the trajectories on a practical level, since the open-loop method applied in this contribution is not practically feasible for real-time operation. Finally, the results from this study show that PyMoC with a wrapper can be successfully used for trajectory optimization of an Aspen Plus Dynamic model employed as if it were a black box.

Acknowledgments

The authors gratefully acknowledge the financial support from VINNOVA [grant number 2015-02421].

References

Adams, G., Burke, B., Goodwin, G., Gravdahl, J., Peirce, R., Rojas, A., 2008. Managing steam and concentration disturbances in multi-effect evaporators via nonlinear modelling and control. IFAC Proceedings Volumes 41 (2), 13919–13924, http://dx.doi.org/10.3182/20080706-5-KR-1001.02356, 17th IFAC World Congress.

Biegler, L.T., 2010. Nonlinear Programming: Concepts, Algorithms, and Applications to Chemical Processes. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA.

Bildea, C.S., Kiss, A.A., 2011. Dynamics and control of a biodiesel process by reactive absorption. Chemical Engineering Research and Design 89 (2), 187–196, http://dx.doi.org/10.1016/j.cherd.2010.05.007.

Faulwasser, T., Korda, M., Jones, C.N., Bonvin, D., 2017. On turnpike and dissipativity properties of continuous-time optimal control problems. Automatica 81, 297–304, http://dx.doi.org/10.1016/j.automatica.2017.03.012.

Kumar, D., Kumar, V., Singh, V., 2013. Modeling and dynamic simulation of mixed feed multi-effect evaporators in paper industry. Applied Mathematical Modelling 37 (1), 384–397, http://dx.doi.org/10.1016/j.apm.2012.02.039.

Lang, Y.-D., Biegler, L., 2007. A software environment for simultaneous dynamic optimization. Computers & Chemical Engineering 31 (8), 931–942, http://dx.doi.org/10.1016/j.compchemeng.2006.10.017, 7th World Congress of Chemical Engineering.

Luyben, W.L., 2018. Dynamic simulation of multi-effect evaporators. Chemical Engineering and Processing - Process Intensification 131, 106–115, http://dx.doi.org/10.1016/j.cep.2018.07.005.

Mayne, D.Q., Rawlings, J.B., Rao, C.V., Scokaert, P.O.M., 2000. Survey constrained model predictive control: Stability and optimality. Automatica 36 (6), 789–814, http://dx.doi.org/10.1016/S0005-1098(99)00214-9.

Mohd Fuad, M.N., Hussain, M.A., Zakaria, A., 2012. Optimization strategy for long-term catalyst deactivation in a fixed-bed reactor for methanol synthesis process. Computers & Chemical Engineering 44, 104–126, http://dx.doi.org/10.1016/j.compchemeng.2012.05.003.

Muñoz López, C.A., Telen, D., Nimmegeers, P., Cabianca, L., Logist, F., Impe, J.V., 2018. A process simulator interface for multiobjective optimization of chemical processes. Computers and Chemical Engineering 109, 119–137, http://dx.doi.org/10.1016/j.compchemeng.2017.09.014.

Negrellos-Ortiz, I., Flores-Tlacuahuac, A., Gutiérrez-Limón, M.A., 2016. Product dynamic transitions using a derivative-free optimization trust-region approach. Industrial & Engineering Chemistry Research 55 (31), 8586–8601, http://dx.doi.org/10.1021/acs.iecr.6b00268.

Negrellos-Ortiz, I., Flores-Tlacuahuac, A., Gutiérrez-Limón, M.A., 2018. Dynamic optimization of a cryogenic air separation unit using a derivative-free optimization approach. Computers and Chemical Engineering 109, 1–8, http://dx.doi.org/10.1016/j.compchemeng.2017.10.020.

Nolin, M., Andersson, N., Nilsson, B., Max-Hansen, M., Pajalic, O., 2018. Analysis of an oscillating two-stage evaporator system through modelling and simulation: an industrial case study. Chemical Engineering Transactions 69, 481–486, http://dx.doi.org/10.3303/CET1869081.

Pitarch, J., Palacín, C., Prada, C.D., Voglauer, B., Seyfriedsberger, G., 2017. Optimisation of the resource efficiency in an industrial evaporation system. Journal of Process Control 56, 1–12, http://dx.doi.org/10.1016/j.jprocont.2017.04.002.

Powell, M.J.D., 1994. A Direct Search Optimization Method That Models the Objective and Constraint Functions by Linear Interpolation. Springer Netherlands, Dordrecht, pp. 51–67.

Powell, M.J.D., 2009. The BOBYQA algorithm for bound constrained optimization without derivatives. University of Cambridge, Cambridge, pp. 26–46, Cambridge NA Report NA2009/06.

Rawlings, J.B., Amrit, R., 2009. Optimizing Process Economic Performance Using Model Predictive Control. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 119–138.

Rios, L.M., Sahinidis, N.V., 2013. Derivative-free optimization: a review of algorithms and comparison of software implementations. Journal of Global Optimization 56 (3), 1247–1293, http://dx.doi.org/10.1007/s10898-012-9951-y.

Sellberg, A., Nolin, M., L“ofgren, A., Andersson, N., Nilsson, B., 2018. Multi-flowrate Optimization of the Loading Phase of a Preparative Chromatographic Separation. Vol. 43 of Computer Aided Chemical Engineering, http://dx.doi.org/10.1016/B978-0-444-64235-6.50282-5.

Würth, L., Rawlings, J.B., Marquardt, W., 2009. Economic dynamic real-time optimization and nonlinear model-predictive control on infinite horizons. IFAC Proceedings Volumes 42 (11), 219–224, http://dx.doi.org/10.3182/20090712-4-TR-2008.00033, 7th IFAC Symposium on Advanced Control of Chemical Processes.

Yamanee-Nolin, M., Löfgren, A., Andersson, N., Nilsson, B., Max-Hansen, M., Pajalic, O., 2019. Single-shooting optimization of an industrial process through co-simulation of a modularized aspen plus dynamics model. In: Kiss, A.A., Zondervan, E., Lakerveld, R., Özkan, L. (Eds.), 29th European Symposium on Computer Aided Process Engineering, Vol. 46 of Computer Aided Chemical Engineering. Elsevier, pp. 721–726, doi:doi.org/10.1016/B978-0-12-818634-3.50121-1.

Yuan, T., Qin, S.J., 2012. Root cause diagnosis of plant-wide oscillations using granger causality. IFAC Proceedings Volumes 45 (15), 160–165, http://dx.doi.org/10.3182/20120710-4-SG-2026.00172, 8th IFAC Symposium on Advanced Control of Chemical Processes.

Zhang, Z., Wu, Z., Durand, H., Albalawi, F., Christofides, P.D., 2018. On integration of feedback control and safety systems: Analyzing two chemical process applications. Chemical Engineering Research and Design 132, 616–626, http://dx.doi.org/10.1016/j.cherd.2018.02.009.