Application of mixed integer nonlinear programming for system identification

Cite as: AIP Conference Proceedings **2293**, 260003 (2020); https://doi.org/10.1063/5.0026410
Published Online: 25 November 2020

Natércia C. P. Fernandes, Florbela P. Fernandes and Andrey Romanenko

Application of Mixed Integer Nonlinear Programming for System Identification

Natércia C.P. Fernandes[1], Florbela P. Fernandes[3] and Andrey Romanenko[1,2,a)]

[1] *CIEPQPF, Department of Chemical Engineering, University of Coimbra, Portugal*

[2] *Ciengis, SA, Coimbra, Portugal*

[3] *Research Centre in Digitalization and Intelligent Robotics (CeDRI), Instituto Politécnico de Bragança, 5300 - 253 Bragança, Portugal*

[a)] Corresponding author: andrey.otw@gmail.com

Abstract. This work describes a method of deadtime approximation in dynamic systems, particularly in the context of nonlinear model predictive control based on mechanistic models where the differentiability of the equations must be ensured. The resulting system identification system is solved using the BBMCSFilter (Branch and Bound based on a Multistart Coordinate Search Filter) global optimization algorithm to determine the order and the parameters of the resulting model, taking into account not only the model-plant mismatch but also the model complexity and the resulting computation time. The application of the method is illustrated with a simulated example of a chemical process unit.

INTRODUCTION

System identification is a critical step in model based simulation and control where, in addition to the proper model structure choice, the resulting simulation quality and closed-loop performance is determined by the tradeoff between the accuracy and the complexity of the model. The present work deals with the use of the so-called first principle models in nonlinear model predictive control (NMPC). The NMPC problem is an optimal control problem that is solved online in a receding horizon framework, as for instance the work of Santos, Oliveira, and Biegler [1]. In this context, process models are nonlinear and of significant size. However, the nature of NMPC requires that the optimal control problem be solved fast to minimize the feedback delay. This calls for automatic differentiation that provides derivative information to the optimizer. Some recent formulations, such as CaSaDi [2] and Plantegrity [3] already include the automatic differentiation tools that drastically reduce the computational burden and eliminate, at the same time, the tedious and error prone manual development of the derivative matrices required by the numerical integrator and optimizer. While this increases the execution speed, it precludes the use of components with nondifferentiability. One of such commonly occuring phenomena is the deadtime that shows as the time difference between the control action and resulting response of the process, for instance, in the following discrete time system

$$\dot{x} = f(x, u_{t-\theta}, p), \tag{1}$$

where x is the state variable vector of the process, u is the control signal vector, and θ is the deadtime. Fig.2 illustrates the dynamic response that is clearly shifted in time from the control signal profile.

When linear multivariable models are used, each pairing of controls and states/outputs is independent. In such case, in the moving horizon formulation of MPC it is sufficient to shift the control profile in time in relation to the output. However, such an approach is not applicable in equation based nonlinear formulations of MPC. Indeed, a single control variable may affect one or more equations with a different associated deadtime.

International Conference of Numerical Analysis and Applied Mathematics ICNAAM 2019
AIP Conf. Proc. 2293, 260003-1–260003-4; https://doi.org/10.1063/5.0026410
Published by AIP Publishing. 978-0-7354-4025-8/$30.00

PROPOSED SOLUTION

MINLP problems combine the combinatorial difficulty of optimizing over discrete variable sets with the challenge of handling nonlinear functions. They are the most general optimization problems and they are, in general, NP-hard.

The underlying idea is to approximate deadtime with a linear system of variable order, for instance, the Padé approximation [4]. This means that for each equation of the original dynamic system there is a set of linear equations that introduce a delay simulating the deadtime for that particular control and output pair i

$$\begin{aligned} \dot{x}_i &= f_i(x_i, u_{j,0}, p) \\ \dot{u}_{j,0} &= A_{ij}(u_{j,0} + u_{j,1}) \\ \dot{u}_{j,1} &= A_{ij}(u_{j,1} + u_{j,2}) \\ \dot{u}_{j,2} &= A_{ij}(u_{j,2} + u_{j,3}) \\ &\ldots \\ \dot{u}_{j,n-1} &= A_{ij}(u_{j,n-1} + u_{j,n}) \end{aligned} \tag{2}$$

where $u_{j,n}$ is the undelayed input signal, and $u_{j,0}$ is the input signal propagated through n first-order systems and that affects the main nonlinear equation.

The resulting system identification problem may be posed in the following form

$$\begin{gathered} \min_{d} \sum_{l=1}^{m} (x_l - \hat{x}_l)^2 + w\,n \\ \text{s.t. (2) and } d_{\min} \le d \le d_{\max} \end{gathered} \tag{3}$$

where d is the vector of decision variables, x_l is the measurement l from the plant, $\hat{x}_l$ is the prediction of the model for the same time instance, and n is the number of the additional equations that approximate the deadtime. The first term of the objective function is the prediction error of the model in the least square sense, the second term penalizes the complexity of the model. The relative importance of the latter is set by the weight w.

The decision vector is

$$d = [n, A, p], \tag{4}$$

where A is the time constant of the additional equations and p is the parameter vector of the original dynamic system. It should be noted that n is an integer variable.

The approach works as follows. First, the experimental dynamic data is acquired from the process in the time window m. Second, the optimization problem is solved using BBMCSFilter algorithm that is able to obtain the global solution [5]. It is based on a branch and bound (BB) scheme to treat the integer variables and the NLP problems that appear in the BB tree search are solved to optimality by a derivative-free global method (MCSFilter) that is based on a multistart algorithm coupled with a coordinate search filter method. More details about these two methods can be found in [6], [5]. Each call of the objective function performs a numerical integration of the model with the current trial point d. The algorithm stops when the convergence criteria have been met.

The use of a global optimization algorithm is desired because the nonlinear nature of the original system and its interaction with the deadtime approximation subsystem may results in multiple minima. The computer implementation is a set of Matlab scripts that contain the optimization algorithm as well as the functions of the objective function and of the constraint violation.

APPLICATION AND RESULTS

The applicability of the proposed approach is illustrated using the so called Plug-Flow Reactor (Fig.1). For the sake of simplicity, only the mixing phenomenon is considered. It is clear that there is a time delay between a change in the flowrate F_1 and the response of the system measured by the concentration analyzer AI. The normalized dimensionless model of the process is

$$\dot{y} = ky + bu_{t-\theta}. \tag{5}$$

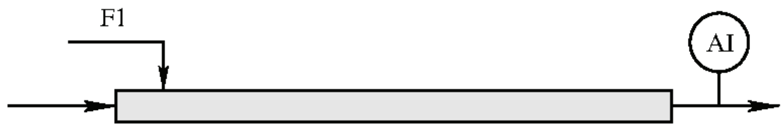

FIGURE 1. Plug-Flow Reactor

The “plant” data was generated using the following conditions inside a 100 point time window

$$
\begin{aligned}
y(0) &= 0,\\
k &= -1,\\
b &= 1,\\
\Delta t &= 1\text{s},\\
u &= \sin(t^2), \text{updated every 10 steps},\\
\theta &= 5\text{s}.
\end{aligned}
$$

The resulting profiles of the control and output variables are illustrated in Fig.2. The decision vector is $d = [n\ \ A\ \ k\ \ b]$

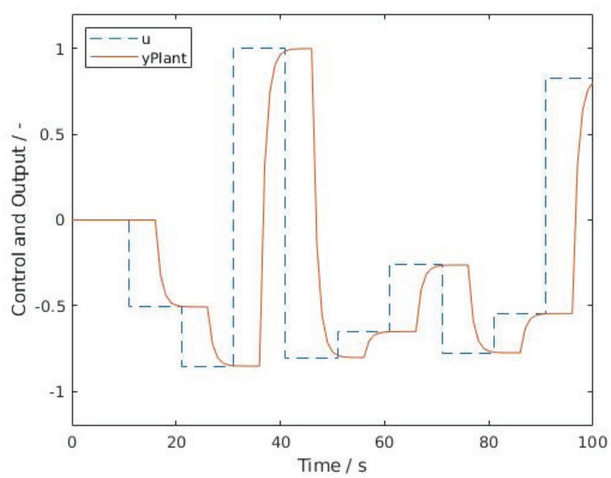

FIGURE 2. Dynamic response with deadtime

and the weight w is chosen to be 0.001. The bounds of the decision variables are

$$
\begin{aligned}
d_{\mathrm{L}} &= [1\ \ 1\ \ -10\ \ 0.1],\\
d_{\mathrm{U}} &= [100\ \ 10\ \ 0\ \ 10],
\end{aligned}
$$

BBMCSFilter’s parameters in this work are those used by Fernandes, Costa, and Fernandes [5]. The same input profile and other simulation settings are used in the system identification step. The initial values of the decision variables are selected randomly within the respective bounds.

The optimization results are represented in Fig.3. One can notice that the identified model captures well the deadtime and the gain of the process but that there is a slight discrepancy in the speed of response. The obtained minimizer is $d = [58\ \ 10\ \ -8.22\ \ 8.20]$ and the minimum value of the objective function is 0.2702. The quality of the obtained model is sufficiently good for control and simulation purposes and while the number of auxiliary equations is high, they are easily handled by the numeric integrators and optimizers due to the underlying linear nature. The obtained values of the parameter vector p do not match the original values because in the augmented system, the dynamics is approximated by the whole set of the equations (2).

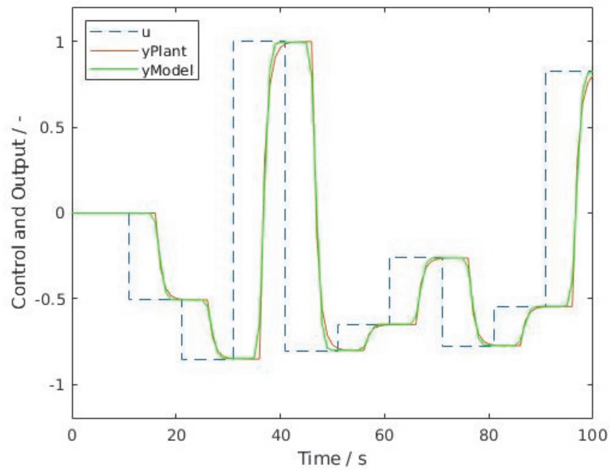

FIGURE 3. Optimization results

The algorithm required 26150 function evaluations and took 25 min to converge. While this is acceptable for offline operations, some algorithmic features that accelerate the convergence will be considered, along with a more sophisticated information based criteria for the choice of the penalization term w.

CONCLUSIONS

The proposed approach of approximating the deadtimes in dynamic nonlinear systems was presented and tested by simulation. The resulting augmented system identification problem was successfully solved using numerical integration and the BBMCSFilter algorithm.

ACKNOWLEDGMENTS

This work was supported by Fundação para a Ciência e a Tecnologia, UID/EQU/00102/2019. The first author also thanks project MATIS, ref. CENTRO-01-0145-FEDER-000014, with financial support of ERDF (CENTRO2020).

REFERENCES

[1] L. O. Santos, N. M. C. Oliveira, and L. T. Biegler, "Reliable and Efficient Optimization Strategies for Nonlinear Model Predictive Control," in *Proc. of DYCORD+'95, Helsingør, Denmark*, edited by J. B. Rawlings (Elsevier Science, Oxford, 1995), pp. 33–38.

[2] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, Mathematical Programming Computation (In Press, 2018).

[3] J. R. Leal, A. Romanenko, and L. O. Santos, Industrial & Engineering Chemistry Research **56**, 3332–3346 (2017).

[4] D. Seborg, T. Edgar, D. Mellichamp, and I. Francis J. Doyle, *Process Dynamics and Control* (John Wiley & Sons, 2010).

[5] F. P. Fernandes, M. F. P. Costa, and E. M. G. P. Fernandes, "Branch and bound based coordinate search filter algorithm for nonsmooth nonconvex mixed-integer nonlinear programming problems," in *Computational Science and Its Applications – ICCSA 2014* (Springer International Publishing, 2014), pp. 140–153.

[6] F. P. Fernandes, M. F. P. Costa, and E. M. G. P. Fernandes, "Multilocal programming: A derivative-free filter multistart algorithm," in *Computational Science and Its Applications – ICCSA 2013* (Springer Berlin Heidelberg, Berlin, Heidelberg, 2013), pp. 333–346.