Real-time optimization with machine learning models and distributed modifier adaptation applied to the MDI-process

Jens Ehlhardta, Inga Wolfb, Sebastian Engella

a TU Dortmund, Process Dynamics and Operations Group, Emil-Figge-Str. 70, 44227 Dortmund, Germany

bCovestro Deutschland AG, Process Technology, Kaiser-Wilhelm Allee 60, 51373 Leverkusen, Germany

jens.ehlhardt@tu-dortmund.de

Abstract

In this work, the allocation of steam in a part of the isocyanate (MDI) production process is optimized with the goal to reduce fouling of the heat exchangers. Real-time optimization (RTO) based upon a rigorous stationary model is applied to optimize the corresponding temperature set-points. The plant-model mismatch is handled by an iterative optimization using Modifier Adaptation with Quadratic Approximation (MAWQA). The rigorous models that are available in a proprietary flowsheet simulator are approximated by surrogate models. In order to reduce the number of input moves in MAWQA for this large-scale plant, concepts of distributed optimization are applied in a tailored fashion. The scheme is tested at the operator training simulator (OTS) of the production plant, and a fast convergence to optimal operating conditions is achieved.

**Keywords**: Isocyanate Production, Modifier Adaptation with Quadratic Approximation, Distributed Modifier Adaptation, Surrogate Models.

* 1. Introduction

The transition to a sustainable and green industry is currently the biggest challenge of the chemical industry. Towards this goal, an important factor is the optimal operation of the processes and plants in terms of material and energy efficiency. Real-time optimization (RTO) can be used to identify the optimal stationary operating conditions and set-points by model-based optimization. However, the inevitable mismatch between the optimization model and the plant may lead to suboptimal performance or even infeasible set-points. Iterative RTO with correction of the model gradients (Gao and Engell, 2005) is a powerful approach to resolve this problem. In this approach, later termed Modifier Adaptation (MA) (Marchetti et al. 2016), the cost and constraint functions of the optimization problem are adjusted based on measured data. In practice with noisy measurements, the determination of the “true” gradients of the cost function and of the constraints from measured responses is challenging. Often finite difference methods are used which are inaccurate for large step sizes and vulnerable to noise for small step sizes. Gao et. al (2016) proposed Modifier Adaptation with Quadratic Approximation (MAWQA) in which the cost and constraint functions are first approximated by quadratic functions and subsequently, the gradients are computed from the approximations as a solution to this problem.

An additional issue in this industrial application is that the stationary process model is only available in a proprietary software and the evaluation is too slow for real-time application. As reported by Ehlhardt et al. (2023), this problem can be overcome by computing surrogate models of the responses of the flowsheet simulator. The simulator is then used as a data source for the identification of artificial neural networks (ANNs) that represent the nominal behavior of the plant.

A bottleneck of MAWQA (and of MA in general) is the number of data points, i.e. setpoints for which the plant is probed, that are needed for the estimation of the gradients. In MAWQA the minimum number of experiments grows quadratically with the number of inputs. Distributed modifier adaptation schemes as proposed by Schneider et al. (2017) are a possible solution to this problem.

In this paper, we propose an efficient decomposition of the optimization problem based on the structure of the MDI-process in order to apply MAWQA to the large-scale system at hand. Compared to previous investigations of the straightforward application of the MAWQA scheme as described in Ehlhardt et al. (2023), less iterations are needed to converge to the optimal set-points. In section 2, MAWQA and its proposed decomposition is explained. Section 3 gives an overview about the investigated process and corresponding optimization problem. The results are discussed in section 4 and a conclusion is given in section 5.

* 1. Real-time optimization via modifier adaptation
		1. Modifier Adaptation with Quadratic Approximations

The general formulation of modifier adaptation is given in equation (1). In iteration , the next set-point is determined by the solution of the following optimization problem (Gao and Engell (2005)):

|  |  |
| --- | --- |
|   | (1) |
|  |   | (2) |
|  |   | (3) |

The adapted cost function and constraint functions  consist of the nominal plant functions () and affine corrections, the so-called modifiers. Thereby, the subscripts and denote the model and plant functions respectively. and are the upper and lower bounds of the set-points. depicts the gradients of the model and plant functions with respect to the set-points. The true plant functions however are usually not known but are approximated from observations of the plant. In MAWQA, quadratic approximations (QA) of the objective function and the constraint functions are identified and used to calculate the necessary plant gradients. For this, a minimum number of data points are required to determine the parameters of the quadratic models, where is the number of inputs for which the set-points are optimized.

The algorithm is as follows. In the first step, probing inputs are applied to the plant to compute the gradients by finite differences. Then the algorithm from (Gao and Engell, 2005) is applied to compute the next steps until enough information is available for the quadratic approximation. The MAWQA algorithm includes also other elements of derivative-free optimization. The optimization problem may be solved solely based on the fitted quadratic functions and instead of the adapted cost and constraint functions if the accuracy of the quadratic approximation was better in the last step. A trust region constraint is added to ensure the validity of the approximations. For details see Gao et al. (2016).

* + 1. Distributed MAWQA

For processes and plants in which multiple units are operated in parallel with input-dependent couplings, the number of required data points required to estimate the quadratic functions can be reduced. An illustrative example is the operation of parallel reactors with subsequent downstream processing of the combined output streams. The input to the separation section depends only on the total load and the concentrations are defined by the mixing rule. Therefore, they can be introduced as local coupling variables. After the introduction of the coupling variables , the description of the distributed optimization problem for the case of an additive cost function and additive global constraints as well as local constraints reads:

|  |  |
| --- | --- |
|   | (4) |
|   |   | (5) |
|  |   | (6) |
|  |   | (7) |
|  |   | (8) |

The global inequality constraints are defined as the sum of the contributions of the sub-models . Eq. (6) defines the local coupling variables in terms of the inputs of the other sub-models. is a known, continuously differentiable function. For each sub-system, the modifiers, quadratic approximations and required gradients can then computed only with respect to the local inputs and coupling variables. Hence, the adapted contributions of the sub-models to the additive cost function read as follows:

|  |  |
| --- | --- |
|   | (9) |

with

|  |  |
| --- | --- |
|   | (10) |

The variables consist of the local inputs and the coupling variables and indicates the gradient with respect to the individual input vector Their dimensions are often smaller than that of the full input vector .

|  |  |
| --- | --- |
|   | (11) |

The with the largest dimension determines the number of initial probing moves required. If (11) holds true, fewer data points are needed for the local quadratic approximations and gradient computations because the quadratic approximations are functions only of the individual input vectors. For more information on distributed optimization, the interested reader is referred to (Boyd 2010).

|  |  |
| --- | --- |
|  |  |

**Figure** **1**: Schematic representation of the considered stage of the isocyanate production process. Multiple reactors are operated in parallel, and their output streams are added up and fed into the train of separation units. and are the temperature set-points of the heat exchangers of the reactors and of the separators. is the temperature of the sum of the outgoing streams of the reactors. Each heat exchanger is equipped with a cascaded control structure, with a temperature controller in the outer loop and a pressure controller for the steam flow in the inner loop.

* 1. Application to the isocyanate production

In this case study, we consider a process stage of the production of diphenylmethane diisocyanate (MDI) as described by Ehlhardt et al. (2023). In this process stage, methylenedianiline (MDA) reacts to MDI in several reactors. The streams from all reactors are then mixed and MDI is separated from other gaseous reaction products. During both the reaction and the separation task, several heat exchangers provide heat to the different units via steam. The layout of the process stage is displayed in Figure 1. The goal is to operate the heat exchangers to reduce fouling. In the test of the algorithm, an operator training simulator (OTS) serves as a plant replacement. Operator training simulators are typically used to train new plant operators, to increase operator awareness for abnormal operating conditions, and to train operators to handle challenging situations. OTS simulation results can also be used to increase the acceptance of advanced process control methods among plant personnel.

* + 1. Modelling of the isocyanate production process

The iterative RTO scheme uses multiple artificial neural network models (ANNs) that were trained on simulation data from a first-principle steady-state model. Each heat exchanger is modeled by an individual ANN, and all ANNs are combined to represent the complete model in the optimization. The first-principles model consists about 160,000 equations and is implemented in Covestro’s in-house simulator. The manipulated variables were sampled on an equidistant grid and the outputs such as steam pressures and heat duties were calculated by the simulation program. Equations (12) and (13) show the modelled input-output relationships:

|  |  |
| --- | --- |
|   | (12) |
|   | (13) |

The operation of the heat exchangers of the reactors is determined by the corresponding set-points , and the operation of the heat exchangers of the separators is determined by the associated set-points  and the mixing temperature . The ANNs have one layer each with 8 to 32 neurons per layer. The -activation function is used, and all models were trained using the MATLAB deep learning toolbox. The models in the flowsheet simulator and in the OTS are different, so there is a significant plant-model mismatch.

* + 1. Optimal steam distribution for the isocyanate process

The goal of the case study is to optimally distribute the total amount of heat required in this process stage among the available heat exchangers in such a way that fouling processes are reduced and cleaning activities have to be performed as seldom as possible. The fouling processes are directly related to the vapour pressures and temperatures of the steam (Engell et al. 2022). It is preferable that the vapour pressures of all heat exchangers remain close to their values for a clean state. This mode of operation extends the time until the next cleaning activities are required. The associated optimization problem is given below:

|  |  |
| --- | --- |
|   | (14) |
|  |   | (15) |
|  |   | (16) |
|  |   | (17) |
|  |   | (18) |
|  | .  | (19) |

The objective penalizes the sum of the differences between the actual steam pressures and their clean values of the -reactors and the -separators. The manipulated variables are the temperature set-points of the heat exchangers of the reactors and of the separators . The coupling variable is the mixing temperature , which depends on all reactor set-points and a vector of constant weighting coefficients . The coefficients are calculated from the mass flows and heat capacities. In this case, the mappings are linear or identities. The first constraint (15) ensures that the heat exchangers are used for heating only. The total heat supplied is bounded by (16). The constraints (17) and (18) result from the maximum available steam pressure and the a priori given bounds on the temperature set-points.

* 1. Results

Distributed MAWQA, as presented in section 2, was applied to the OTS of the MDI-process shown in section 3. The resulting trajectories are shown in Figure 2. The algorithm starts with a predefined sequence of perturbations for iterations 1-3. Due to the reduced number of degrees of freedom of each sub-model, enough data is available to construct the quadratic approximations for all models in the reaction section in step 3. Starting with iteration 4, set-points are calculated by distributed MAWQA but for the separation section quadratic approximations can only be built after iteration 6. After a slight violation of the total heat flow constraint, the algorithm converges to an optimal combination of the temperature references in iteration 7. The value of the objective function is reduced to approximately 50% of the initial value.



**Figure** **2**: Results of the application of distributed MAWQA to the OTS of the MDI process. The second and third subplot display the trajectories of the total heat flow and the value of the objective function. The dashed lines represent the constraints. In all subplots, the iteration count is shown on the x-axis and the values on the y-axes are scaled.

* 1. Conclusion

In this work, distributed Modifier Adaptation with Quadratic Approximations was successfully applied to a stage of the isocyanate production process to optimize the distribution of steam between the heat exchangers of the process units. The optimization model consists of individual artificial neural networks for each sub-model that were trained on data provided by a flow-sheet simulator. The number of iterations performed before the quadratic approximations are available is significantly lower than when solving the full problem. Consequently, the time to convergence to optimal operating conditions is reduced by half compared to previous work (Ehlhardt et al. 2023). In future work, different scenarios such as load changes and load sharing will be investigated.

* 1. Acknowledgements

The German Ministry of Economic Affairs and Climate Action (BMWK) is acknowledged for funding the KEEN project under the grant number 01MK20014S.

References

Boyd, S., 2010, Distributed optimization and statistical learning via alternating direction method of multipliers. Found. Trends® Mach. Learn. 3 (1), 1-122

Ehlhardt, J., Ahmad, A., Wolf, I., Engell, S., 2023. Real‐Time Optimization Using Machine Learning Models Applied to the 4,4′‐Diphenylmethane Diisocyanate Production Process. Chemie Ing. Tech. 95, 1096–1103.

Engell, S., Ahmad, A., Ehlhardt, J., Wolf, I., Arras J., Hielscher A., 2022. Method for controlling a distributed control system. EP22209645.5.

Gao, W., Engell, S., 2005. Iterative set-point optimization of batch chromatography. Comput. Chem. Eng. 29, 1401–1409.

Gao, W., Wenzel, S., Engell, S., 2016. A reliable modifier-adaptation strategy for real-time optimization. Comput. Chem. Eng. 91, 318–328.

Marchetti, A.G., François, G., Faulwasser, T., Bonvin, D., 2016. Modifier adaptation for real-time optimization - Methods and applications. Processes 4, 1–35.

Schneider, R., Milosavljevic, P., Bonvin, D., 2019. Distributed modifier-adaptation schemes for the real-time optimisation of uncertain interconnected systems. Int. J. Control 92, 1123–1136.