Proceedings of the 14th International Symposium on Process Systems Engineering – PSE 2021+
June 19-23, 2022, Kyoto, Japan

Surrogate Modeling for Superstructure Optimization with Generalized Disjunctive Programming

H. A. Pedrozo[a,b]; S. B. Rodriguez Reartes[a,b]; D. E. Bernal[d]; A. R. Vecchietti[c],

M. S. Diaz[a,b,*], I. E. Grossmann[d]

[a]*Planta Piloto de Ingeniería Química (PLAPIQUI CONICET-UNS), Camino La Carrindanga km. 7, Bahía Blanca, Argentina*
[b]*Departamento de Ingeniería Química, Universidad Nacional del Sur (UNS), Bahía Blanca, Argentina*
[c]*Institute of Design and Development (INGAR CONICET-UTN), Avellaneda 3657, Santa Fe, Argentina*
[d]*Department of Chemical Engineering, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, USA*
sdiaz@plapiqui.edu.ar

Abstract

In this work, we propose an iterative framework to solve superstructure design problems, which includes surrogate models, with a custom implementation of the Logic-based Outer- Approximation algorithm (L-bOA). We build surrogate models (SM) using the machine learning software ALAMO exploiting its capability for selecting low-complexity basis functions to accurately fit sample data. To improve and validate the SM, we apply the Error Maximization Sampling (EMS) strategy in the exploration step. In this step, we formulate mathematical problems that are solved through Derivative Free Optimization (DFO) techniques. The following step applies the L-bOA algorithm to solve the GDP synthesis problem. As several NLP subproblems are solved to determine the optimal solution in L-bOA in the exploitation step, the corresponding optimal points are added to the SM training set. In case that an NLP subproblem turns out to be infeasible, we solve the Euclidean Distance Minimization (EDM) problem to find the closest feasible point to the former infeasible point. In this way, the entire information from NLP subproblems is exploited. As original model output variables are required, we solve EDM problems using DFO strategies. The proposed methodology is applied to a methanol synthesis problem, which shows robustness and efficiency to determine the correct optimal scheme and errors less than 0.2% in operating variables.

Keywords: superstructure optimization; surrogate models; disjunctive programming; derivative free optimization

1. Introduction

Advances in computers and mathematical modeling have enabled the detailed representation of process systems, and thus, the development of fundamental tools for decision making in process design. This scenario also presents new challenges. In mathematical programing, the standard method to formulate a problem is to declare all process unit equations to perform the optimization. However, when formulating highly

accurate models, some constraints or even the objective analytic function may not be available if they are evaluated through simulators or special programs. These functions that are not analytically available, are referred to as black-box models. When a mathematical problem includes both, explicit and black-box equations, it is referred to as hybrid or grey box model. A common approach to address this kind of problems includes building surrogate models (SMs) to replace the black-box models. SMs are simplified functions that can estimate output data from a set of input variables, requiring small CPU times.

When working with surrogate models, there is a trade-off between exploration and exploitation steps. Exploration strategies improve the global performance of the SM in the entire feasible region to reduce the probability of excluding the global optimum. On the other hand, exploitation-based methods refine the SM in regions where optima could be potentially found.

The interest of the Process Systems Engineering (PSE) community in developing efficient methods to address the formulation and solution of black/grey box problems has increased significantly in recent years (Bhosekar and Ierapetritou, 2018). Kim and Boukouvala (2020) developed a surrogate-based optimization procedure to solve mixed-integer nonlinear problems focused on avoiding the binary variable relaxation. Pedrozo et al. (2021a) proposed an iterative framework to address hybrid problems, replacing highly nonlinear equations for SM in order to reduce problem complexity. Thus, it was assumed that the analytic function was available for the exploration and exploitation steps.

In this work, we include Derivative Free Optimization (DFO) techniques (Zhao et al., 2021) in the exploration and exploitation steps to avoid using the analytic functions. Numerical results show that the strategy is efficient and accurate to address the synthesis problems and the generation and refinement of SMs.

2. Methodology

The proposed optimization framework is outlined in Fig. 1. Initially, lower and upper bounds are set for the input variables of each SM. The Latin Hypercube Sampling (LHS) technique is employed in MATLAB to generate sampling data. Output variables corresponding to each sampling point are obtained by performing simulations of the true or original model. When working with hybrid problems, a filtering step is required to discard infeasible sample points. Then, an initial SM is built in the machine learning software ALAMO (Wilson and Sahinidis, 2017) considering simple algebraic regression functions (SARFs).

Since the accuracy of this initial SM may not be good enough in all sampling points, we evaluate the corresponding relative errors, and we add Gaussians Radial Basis Functions (GRBFs) to represent those points whose errors are greater than a tolerance. In this way, we build the first SM based on both, SARFs and GRBFs, and then, we carry out the first exploration step. The Error Maximization Sampling (EMS) (Wilson and Sahinidis, 2017) strategy is applied in the exploration step. This method consists of maximizing the relative error of the SM in the feasible region. In this work, this optimization is performed through the DFO solver (Powell, 2009), which makes use of black-box simulation models. As a result, low-accuracy points of the domain are identified, and then interpolated by means of GRBF to improve the SM performance in that region, until the relative error is less than a tolerance or a maximum number of EMS problems is solved.

In the following step, we solve the hybrid model-based Generalized Disjunctive Programming (GDP) problem in GAMS. A custom implementation of the Logic-based Outer-Approximation (L-bOA) algorithm is employed (Pedrozo et al., 2021b, Pedrozo et

al., 2020). We exploit the information of the L-bOA subproblems to refine the SM in the exploitation step. The feasible NLP subproblem solutions are compared to the rigorous black-box simulations to assess the SMs accuracy in that region. As some NLP subproblems or black-box simulations might be infeasible due to the performance of the SMs, we formulate an optimization problem to determine the feasible sampling point that minimizes the Euclidean distance to the NLP subproblem solution, and this point is then added to the training set. This optimization problem is also solved using DFO solvers (Powell, 2009).

The iterative algorithm, which is shown in Fig. 1, stops when the specified convergence criterion is met. Otherwise, the exploration step is carried out again (the number of major iterations of the algorithm is equal to the times the GDP problem is solved).

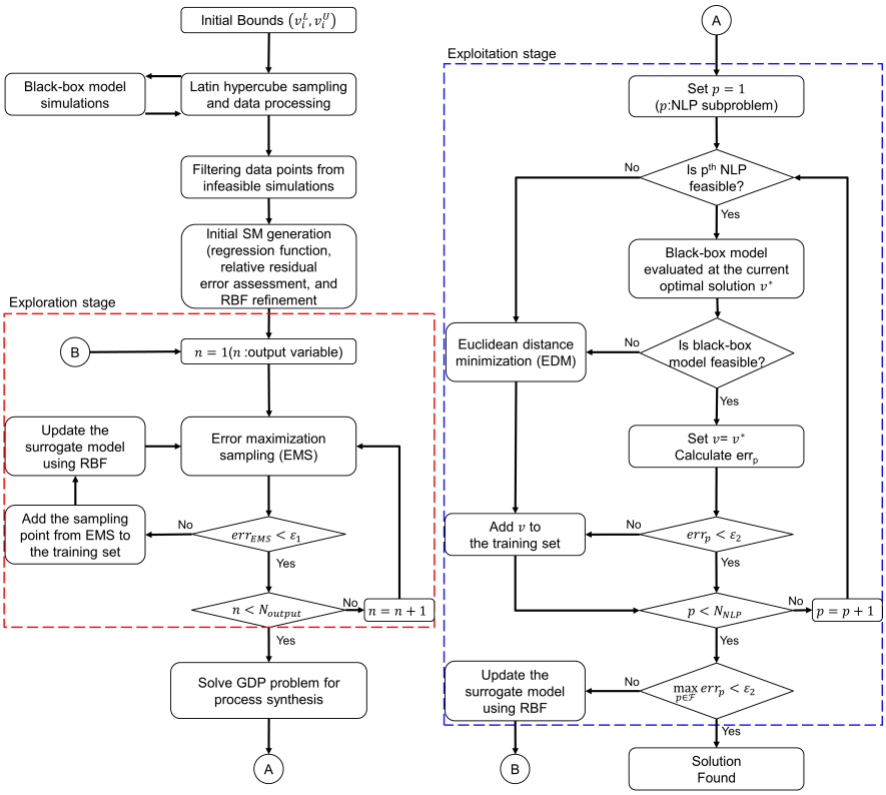

Figure 1: Iterative optimization framework

2.1. Software resources

The solution procedure is automated using MATLAB as a core for data transferring (see Fig. 2). In this way, ALAMO is run from MATLAB to generate the corresponding initial SMs. These functions and their derivatives are transferred to GAMS to formulate the hybrid GDP problem for process synthesis, and to solve it with the custom implementation of the L-bOA algorithm. To improve the SMs, we solve DFO problems for black-box models in the exploration and exploitation steps. In these cases, we employ the algorithms developed by Powell (2009), through the package provided by Ragonneau and Zhang (2021). Since these algorithms do not explicitly handle constraints, the Bound Optimization BY Quadratic Approximation (BOBYQA) algorithm is used.

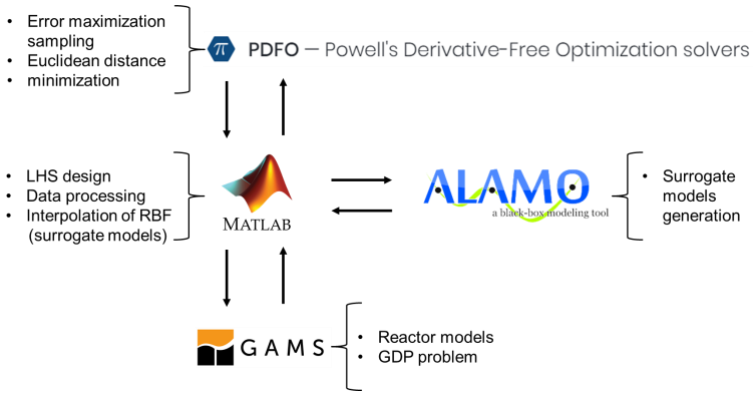

Figure 2: Software integration

3. Case Study

The methanol synthesis problem (Chen and Grossmann, 2019) is used as case study to test the proposed iterative algorithm. Figure 3 shows the process superstructure, where discrete decisions are represented using dashed lines for both, equipment and streams. The objective function is profit maximization.

In order to illustrate the algorithm, reactor models (units 9 and 10) are replaced by surrogate models to calculate conversion in each reactor. In this way, a hybrid formulation, which includes first principles and two SMs, is obtained.

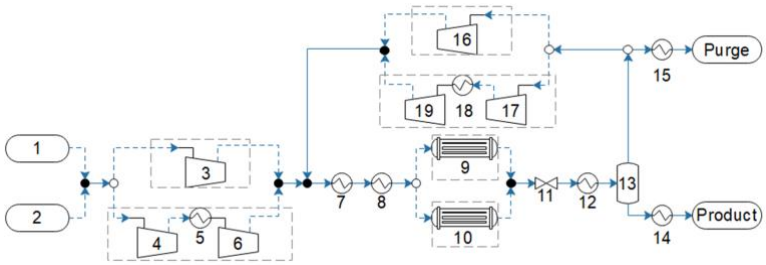

Figure 3: Superstructure for methanol synthesis (Chen and Grossmann, 2019)

4. Results

In order to show the robustness of the method and to consider the random component of the sampling technique, the problem is solved using ten different initial sampling data sets. In addition, we test 100 and 1,000 initial sampling points to assess the impact of the initial SM in the algorithm performance.

We observe that the iterative algorithm of Fig. 1 determines the optimal solution of the problem in each run (1,840 M$/y), and the error in the objective value is less than 0.2 % even in the worst case. Moreover, we observe that for the runs with 100 initial sampling points, the algorithm generally requires two major iterations (in 7 runs of 10, Fig.4) to satisfy the convergence criterion. Thus, these SMs are refined only one time in the neighbourhood of the optimal solution during the exploitation step to make them accurate enough. However, in the worst case, four major iterations are required to meet the convergence criterion. On the other hand, considering large initial sampling data (1,000 points), the proposed method generally converges in one iteration (in 5 runs of 10, Fig.4).

Accordingly, the initial SM after exploration step has enough accuracy, so no data points are included in the exploitation step. In the worst cases, four iterations are also required. This analysis indicates that we cannot guarantee the quality of the initial SM. Even working with a large initial sampling data set, SM refinement during the exploration and exploitation steps can be required to achieve the desired accuracy of the generated SMs.
Regarding the algorithm performance, Fig. 5 shows the corresponding CPU time distributions. On average, 11.9 and 2.5 minutes are the total CPU time for 100 and 1,000 initial sampling points, respectively. The exploration step is the most time consuming, followed by the exploitation step, while CPU times associated with the initial fit and the GDP problem solution are negligible. These results are related to the use of DFO strategies in the refinement steps. Solving either an optimization problem for the exploration or exploitation step with DFO methods, requires 40 s approximately. Thus, the quality of initial SMs strongly influences the method's performance. When the algorithm is run with a large initial sampling data set (1000 points), the SMs require less refinement, and consequently, fewer problems must be solved using DFO strategies, as compared to the case of using 100 initial sampling points. These results are in agreement with those from Wilson and Sahinidis (2017).
When comparing this strategy with the case of using NLP solvers (CONOPT) for the exploration and exploitation steps (Pedrozo et al. 2021a), there is a significant increase in CPU time, i.e., 15 s vs. 2.5 min for 1,000 initial sampling points on average.

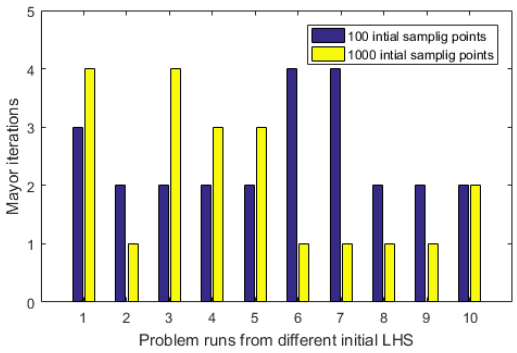

Figure 4: Major iterations of the iterative framework from different initial LHS sets

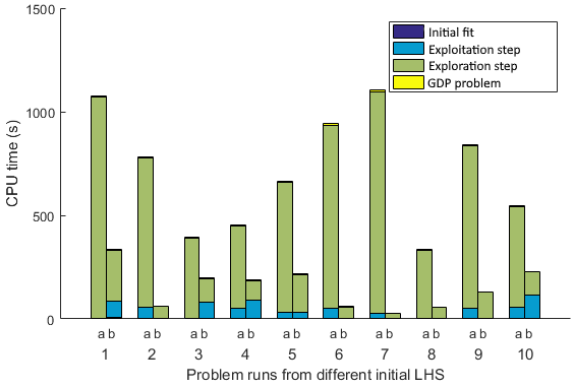

Figure 5: CPU time distribution. a) 100 initial sampling points. b) 1000 initial data sampling points. CPU times corresponding to initial fit and GDP problem solution are less than six seconds, so they are not easily distinguishable in the figure

5. Conclusions

In this work, we propose an algorithm for SMs generation and refinement using DFO strategies in the exploration and exploitation steps for the synthesis of process flowsheets using Generalized Disjunctive Programming with surrogate models. The algorithm has been tested with a methanol synthesis case study. The optimization tool has been proven to be robust and effective in generating solutions with relative errors lower than 0.2 % for the objective function in the worst cases, and obtaining the same optimal flowsheet as the rigorous model. The CPU time can be reduced by using a larger initial sampling point set. This strategy paves the way to efficiently refine SMs by the use of black-box models and DFO solvers.

References

Bhosekar, A., & Ierapetritou, M. (2018). Advances in surrogate based modeling, feasibility analysis, and optimization: A review. Comp. & Chem. Eng., 108, 250-267.

Chen, Q., & Grossmann, I. (2019). Modern modeling paradigms using generalized disjunctive programming. Processes, 7(11), 839.

Kim, S. H., & Boukouvala, F. (2020). Surrogate-based optimization for mixed-integer nonlinear problems. Comp. & Chem. Eng., 140, 106847.

M. J. D. Powell, The BOBYQA algorithm for bound constrained optimization without derivatives, Technical Report DAMTP 2009/NA06, Department of Applied Mathematics and Theoretical Physics, Cambridge University, Cambridge, UK, 2009

Pedrozo, H. A., Reartes, S. R., Bernal, D. E., Vecchietti, A. R., Díaz, M. S., & Grossmann, I. E. (2021a). Hybrid model generation for superstructure optimization with Generalized Disjunctive Programming. Comp. & Chem. Eng., 154, 107473.

Pedrozo, H. A., Reartes, S. R., Vecchietti, A. R., Díaz, M. S., & Grossmann, I. E. (2021b). Optimal design of ethylene and propylene coproduction plants with generalized disjunctive programming and state equipment network models. Comp. & Chem. Eng., 149, 107295.

Pedrozo, H. A., Rodriguez Reartes, S., Diaz, M. S., Vecchietti, A. R., Grossmann, I. E. (2020), Coproduction of ethylene and propylene based on ethane and propane feedstocks, Computer Aided Chemical Engineering, 48, 907-912. https://doi.org/10.1016/B978-0-12-823377-1.50152-X

T. M. Ragonneau and Z. Zhang, PDFO: Cross-Platform Interfaces for Powell's Derivative-Free Optimization Solvers (Version 1.1), available at https://www.pdfo.net

Wilson, Zachary T., and Nikolaos V. Sahinidis. "The ALAMO approach to machine learning." Comp. & Chem. Eng., 106 (2017): 785-795.

Zhao, F., Grossmann, I. E., García-Muñoz, S., & Stamatis, S. D. (2021). Flexibility index of black-box models with parameter uncertainty through derivative-free optimization. AIChE Journal, 67(5), e17189.