Contents lists available at ScienceDirect

Chemical Engineering Research and Design

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

Hybrid simulation-equation based synthesis of chemical processes

*Natalia Quirante, Juan Javaloyes-Antón, José A. Caballero**

Institute of Chemical Processes Engineering. University of Alicante, PO 99, E-03080 Alicante, Spain

ARTICLE INFO

Article history:
Received 18 May 2017
Received in revised form 23 September 2017
Accepted 20 February 2018
Available online 6 March 2018

Keywords:
Simulation
Optimization
MINLP
Kriging algorithm
Vinyl chloride monomer process

ABSTRACT

A challenging problem in the synthesis and design of chemical processes consists of dealing with hybrid models involving process simulators and explicit constraints. Some unit operations in modular process simulators are slightly noisy or require large CPU times to converge. In this work, this problem is addressed by combining process simulators and surrogate models. We have replaced some unit operations, which cannot be used directly with a gradient-based optimization, by surrogate models based on Kriging interpolation. To increase the robustness of the resulting optimization model, we perform a degree of freedom analysis and aggregate (or disaggregate) parts of the model to reduce the number of independent variables of the Kriging surrogate models (KSMs). Thus, the final model is composed of KSMs, unit operations (maintained in the process simulator) and also explicit equations.

The optimization of the well-known vinyl chloride monomer (VCM) production process is performed to test the proposed approach. The effect of the heat integration is also studied. In addition, the economic feasibility of the optimized process is calculated assuming uncertainty in raw material and product prices.

1. Introduction

Methodologies for the synthesis of chemical processes can be classified into two different categories: sequential-conceptual methods and superstructure optimization-based methods.

Sequential methods follow a natural hierarchy between the engineering decisions to obtain a chemical process flowsheet (Douglas, 1985, 1988). This approach is commonly used because the original problem is divided into a set of sub-problems that reduce the complexity of the initial problem. However, due to its sequential nature, this approach cannot guarantee an optimal solution since it ignores the different trade-offs between the various objectives of the prior stages.

Superstructure optimization-based methods consider the complete network which is composed of all the unit operations, their connections, and other constraints (Grossmann, 1985; Yeomans and Grossmann, 1999). The solution of the mathematical model specifies which of the initial units and connections are kept in the optimal structure. These methods are used because they offer a simultaneous optimization of the process structure and the operating conditions. However, superstructure optimization-based methods are complex to solve due to the resulting models, usually large-scale non-convex mixed-integer nonlinear programs (MINLP). The general algebraic form of these MINLP optimization problems is shown in Eq. (1).

$$\begin{aligned} min \quad & f(x,y) \\ s.t \quad & h(x,y) = 0 \\ & g(x,y) \leq 0 \\ & x \in \mathbb{R}^n,\ y \in \{0,1\}^t \end{aligned} \tag{1}$$

where $f(x,y)$ is the objective function (e.g., economic, environmental, etc.); $h(x,y)$ are the equations that describe the behavior of the system (e.g., mass and energy balances, reaction rates, etc.); and $g(x,y)$ are inequality constraints that define process specifications (e.g., product purities, maximum temperature allowed, etc.). The real n-vector x represents the continuous independent variables, and the t-vector y represents the discrete independent variables.

* *Corresponding author.*
E-mail address: caballer@ua.es (J.A. Caballero).
https://doi.org/10.1016/j.cherd.2018.02.032

When a sequential-modular process simulator is used as a black box to compute the objective function and/or the equations that describe the behavior of the system (equality constraints in the previous MINLP problem), different approaches can be employed to solve the resulting optimization problem. We can try to address the problem directly by using commercial derivative-based solvers (e.g., DICOPT, ALPHAECP, SBB...) or metaheuristic algorithms (e.g., genetic algorithms or particle swarm optimization algorithms). However, some important drawbacks arise with both paths.

On the one hand, when mathematical programming solvers are used, the following challenges arise. First, due to the nonlinearities and non-convexities inherent to some unit operations and thermodynamic models, these methods do not guarantee a global solution and can be easily stuck in local solutions. Moreover, the solution depends very sensitively on initial values. However, what is much more important is the fact that the objective function and/or the set of constraints are analytically intractable (discontinuous, non-differentiable, and inherently noisy). Hence, derivatives of the objective and/or constraint functions with respect to the independent variables must be calculated by numerical differentiation (which limits the accuracy and effectiveness of such solvers).

Derivatives calculated by perturbation can be very expensive to compute, and even in the case in which the CPU time is not excessive, some unit operations introduce numerical noise. Thus, an accurate derivative cannot be obtained. Of course, all these models are perfectly valid for simulation purposes, but even relatively small differences in two instances – completely negligible in any simulation – prevent the calculation of accurate derivatives (a detailed discussion can be found in Caballero and Grossmann, 2008).

On the other hand, metaheuristic techniques (which belong to a class of optimization strategies that does not require gradient information, i.e., derivative-free optimization (DFO) methods) seem to be well suited to simulation-based optimization when sequential-modular simulators are used. This is because they only require the values of the objective function. Of course, there are significant disadvantages of not having derivative information. We cannot expect that the performance of DFO methods matches those of derivative-based methods. In particular, the scale of the problems that can be efficiently solved by DFO algorithms does not exceed a few tens of variables (Conn et al., 2009). Besides, these techniques are not able to guarantee the optimality of the solution found, although they are designed to have the ability to escape from local optima. In addition, DFO algorithms normally require a large number of function evaluations, and perhaps, one of the most important disadvantages is that DFO algorithms exhibit poor performance in highly constrained systems. Generally, these algorithms handle the constraints by adding a penalty to the objective function to account for infeasibility.

An added difficulty regardless the optimization technique is related to the convergence of flowsheets with several recycle streams. As simulations become more complex, the robustness (in terms of convergence) decreases and the simulator becomes prone to errors.

These approaches for solving synthesis problems are not new. A considerable amount of literature supports the synergy achieved through the smart integration of chemical process simulators with an external optimizer based on gradient information (Balas, 1979; Brunet et al., 2012; Caballero et al., 2005; Díaz and Bandoni, 1996; Diwekar et al., 1992; Navarro-Amorós et al., 2014; Raman and Grossmann, 1994; Reneaume et al., 1995) and metaheuristic techniques (Aspelund et al., 2010; Bravo-Bravo et al., 2010; Chen et al., 2014; Dantus and High, 1999; Eslick and Miller, 2011; Gross and Roosen, 1998; Gutiérrez-Antonio and Briones-Ramírez, 2009; Javaloyes-Antón et al., 2013; Leboreiro and Acevedo, 2004; Odjo et al., 2011; Vazquez-Castillo et al., 2009).

Some researchers have proposed frameworks to reduce the complexity of the optimization models through the use of surrogate models (Jones et al., 1998; Shao et al., 2007; Won and Ray, 2005; Xiong et al., 2007). A surrogate model is a set of mathematical functions, based on data generated from the simulation. In this way, the optimization of an analytically tractable and computationally cheap surrogate model replaces the original black box process. Most often, for complex systems, it is recommended to disaggregate the whole process into smaller units and model each block separately, ensuring that all relevant connectivity variables have also been included (i.e., component flows, temperatures, and pressures of each stream).

The main novelties of the proposed approach are at the modeling stage and at solution stage.

At the modeling level, as far as we know, the deterministic optimization of problems related to large-scale superstructures with a non-fixed topology dealing with hybrid models involving process simulators, explicit constraints and surrogate models for dealing with noise units and third-party modules (i.e., non-numerical noisy proprietary models) has never been studied.

At the solution stage, we use a logic-based algorithm—the Logic-Based Outer Approximation algorithm (Turkay and Grossmann, 1996). Logic-Based Algorithms do not require the reformulation of the problem as an MINLP. The NLP sub-problems can, therefore, be efficiently solved. The numerical efficiency of the optimization is improved by using a distributed approach for surrogate models (we use small surrogate models with a reduced number of degrees of freedom instead of a single large model). The advantage is that we minimize the necessity of resampling during the optimization (Biegler et al., 2014; Quirante et al., 2015).

In this work, we have used the Kriging algorithm to build the surrogate models since they are computationally efficient and they need relatively small sampling data to be built. Several works have been carried out to overcome the challenges of simulation-based optimization using surrogate models based on Kriging interpolation. Caballero and Grossmann (2008) studied the optimization of a disaggregated flowsheet using Kriging models, obtaining unit operations with low-level noise. Henao and Maravelias (2011) used artificial neural networks for disaggregating models. Other works have modeled and optimized chemical processes using Kriging-based techniques (Davis and Ierapetritou, 2007; Palmer and Realff, 2002a,b; Quirante and Caballero, 2016; Quirante et al., 2015). A review of Kriging applications in simulation was made by Kleijnen (2009).

The objective of this work is to develop an optimization-based simulation tool that uses a process simulator as calculation engine, where surrogate models based on Kriging interpolation replace those unit operations that introduce numerical noise or need a large CPU time to converge. At the same time, it allows us to introduce explicit equations in such a way that the resulting model includes surrogate models, unit operations maintained in the process simulator and explicit equations. The rest of this article is organized as follows. In the next section, we discuss the main features of the proposed approach. Then, to illustrate this approach, we use the superstructure proposed by Turkay and Grossmann (1998) for the synthesis of the vinyl chloride monomer (VCM). To complete the study of the VCM process, the influence of the heat integration on the profit of the process is studied. Besides, the economic feasibility of the optimized VCM process is evaluated assuming uncertainty in raw material and product prices. Finally, the conclusions of this work are summarized at the end of the paper.

2. Methodology

In the proposed approach, simulation-based optimization of complex processes is performed using derivative-based solvers. The superstructure, which includes all the alternatives of interest of the process that we need to optimize, is implemented at the level of the process simulator, with the added feature that those unit operations that are inherently noisy and/or expensive to converge (in terms of CPU time) are replaced by surrogate models based on Kriging interpolation (e.g., distillation columns and reactors). The units that do not introduce numerical noise (such as mixers, splitters, coolers/heaters, compressors, valves or pumps) are maintained in the process simulator. The surrogate models are built in MATLAB from training data sets obtained from the process simulator. In addition, the equations related to capital and operating costs are implemented as explicit equations.