Computers and Chemical Engineering 180 (2024) 108501

Contents lists available at ScienceDirect

Computers and Chemical Engineering

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

A hybrid deterministic-stochastic algorithm for the optimal design of process flowsheets with ordered discrete decisions

David A. Liñán [a], Gabriel Contreras-Zarazúa [b,c], Eduardo Sánchez-Ramírez [b], Juan Gabriel Segovia-Hernández [b], Luis A. Ricardez-Sandoval [a,*]

[a] *Department of Chemical Engineering, University of Waterloo N2L 3G1, Canada*
[b] *Departamento de Ingeniería Química, Universidad de Guanajuato GTO 36050, Mexico*
[c] *Centro de Innovación Aplicada en Tecnologías Competitivas (CIATEC) GTO 37545, Mexico*

ARTICLE INFO

Keywords:
Discrete-steepest descent algorithm
Differential evolution with tabu list
Hybrid deterministic-stochastic optimization
Distillation flowsheet
Optimization

ABSTRACT

This work presents a hybrid stochastic-deterministic algorithm for optimal design of process flowsheets, i.e., finding the optimal design variables and operating conditions of multiple interconnected units using rigorous phenomenological chemical engineering models. Unlike previous studies that propose hybrid deterministic and stochastic algorithms in sequential and nested arrangements, the present work proposes a parallel configuration to perform the hybridization. The proposed hybrid algorithm combines a stochastic method (SM) with the deterministic Discrete-Steepest Descent Algorithm with Variable Bounding (DSDA-VB). The SM and DSDA-VB strategies interact in parallel by exchanging new feasible solutions identified by the SM and improved search bounds determined by the DSDA-VB. The proposed method is illustrated using a thermally coupled system and a sequence of reactive, extractive, and traditional distillation columns. The results indicate that the proposed algorithm outperforms the traditional Differential Evolution with Tabu List (DETL) algorithm, showing faster and improved convergence.

1. Introduction

Flowsheet optimization is typically regarded as a two-stage process involving the optimal synthesis problem and the optimal design problem. The former (i.e., flowsheet synthesis) determines the type of operations and how they are interconnected to achieve a desired production target from raw materials and energy inputs, e.g., see Cremaschi (2015), Gooty et al. (2022), Göttl et al. (2022), Ryu et al. (2020), and Tula et al. (2017). The latter (i.e., flowsheet design) aims to determine the optimal operating conditions and other operation/equipment related variables (e.g., discrete and continuous design variables) of the synthetized process with a fixed flowsheet, e.g., see Contreras-Zarazúa et al. (2019), Gómez et al. (2006), Hong et al. (2019), Jia et al. (2023), Peccini et al. (2023). The flowsheet synthesis problem often considers shortcut or simplified models that allow the screening of several flowsheet configurations, which are later evaluated and optimized in the flowsheet design stage using rigorous nonlinear models (Ramapriya et al., 2018). Alternatively, the simultaneous synthesis and design problem using rigorous nonlinear models has also been studied in the literature (Aliaga-Vicente et al., 2017; Ma et al., 2021; Ma and Li, 2022; Zhang et al., 2018). This work is focused on flowsheet design optimization problems, which are regarded as Mixed-Integer Nonlinear Programming (MINLP) problems. Formulating and initializing MINLP flowsheet design problems may require modeling expertise, and constructing accurate mechanistic models for systems with complex thermodynamic equilibrium and hydrodynamic relationships can be challenging and time-consuming. Typically, these problems are non-convex and challenging to solve due to the nonlinearity of non-ideal thermodynamic models, mass, and energy balances in chemical processes; the inherent non-convexity of discrete variables, and the nonlinearities that arise from the interactions of discrete and continuous variables. Hence, optimizing the design of rigorous flowsheet models can turn into a computationally expensive and time-consuming process. In this work, a hybrid algorithm that parallelizes stochastic and deterministic optimization methods is proposed to exploit the advantages of stochastic and deterministic techniques and to reduce the computational effort required to optimize the design of chemical process flowsheets using rigorous process models. Chemical engineering software with

* Corresponding author.
E-mail address: laricard@uwaterloo.ca (L.A. Ricardez-Sandoval).

https://doi.org/10.1016/j.compchemeng.2023.108501
Received 27 May 2023; Received in revised form 12 October 2023; Accepted 1 November 2023
Available online 4 November 2023

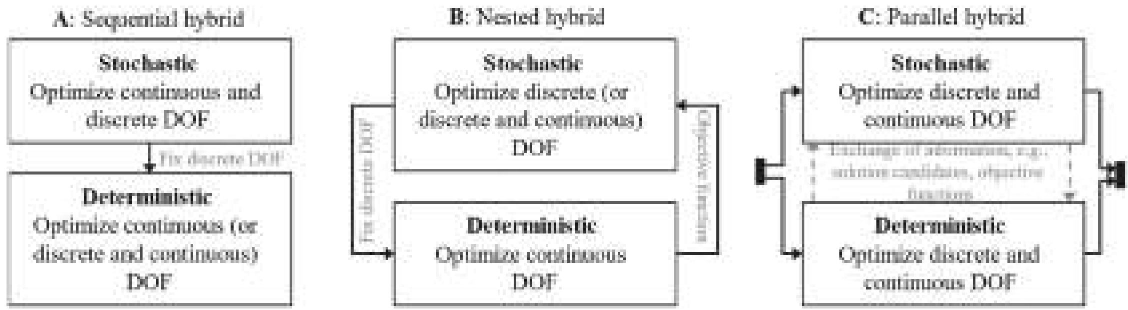

Fig. 1. Hybrid arrangements for optimal flowsheet design aided with simulation software. A: sequential hybrid, B: nested hybrid, and C: parallel hybrid. DOF means degrees of freedom.

simulation and Nonlinear Programming (NLP) optimization capabilities is considered for modeling purposes.

The available optimization strategies to solve the MINLP flowsheet design problem are generally classified into deterministic and stochastic (Segovia-Hernández et al., 2015). Both deterministic and stochastic optimization techniques exhibit limitations when solving flowsheet optimization problems. Local optimization strategies highly depend on the initialization of continuous and discrete decisions, due to "zero flow" numerical issues (Liñán and Ricardez-Sandoval, 2023). Also, finding a global solution to non-convex problems is not guaranteed with local solvers (Kronqvist et al., 2019). Conversely, traditional global deterministic MINLP methods may be computationally prohibitive when dealing with large-scale flowsheet design problems that consider continuous and discrete degrees of freedom (DOF) and rigorous nonlinear models for multiple processing units (Franke, 2017; Kruber et al., 2021). Furthermore, the performance of both local and global deterministic solvers may be impacted by the level of detail of the process models and the formulation selected to represent the interactions between continuous and discrete variables, e.g., Big-M, convex hull, etc (Liñán et al., 2020). In contrast, stochastic methods that use randomized search strategies and exploration techniques that are not tied to specific model assumptions or simplifications. This makes stochastic methods especially suitable to consider the simulation software as a black box. Despite their advantages, black box stochastic approaches have other limitations such as a large number of function evaluations, which may result in slow convergence, or refined parameter tuning to achieve optimal performance (Costa and Bagajewicz, 2019). To partially address these issues, previous works have suggested different arrangements of hybrid deterministic-stochastic algorithms that counterbalance the benefits and drawbacks of each strategy. The hybrid strategies that are currently available for optimal flowsheet design aided with simulation software are shown in Fig. 1A and B, as explained below.

The simplest hybrid deterministic-stochastic technique consists of applying these techniques sequentially (Fig. 1A), where stochastic optimization is used first, and the best solution found is further refined using a deterministic optimization solver, see e.g., Srinivas and Rangaiah (2006), Munawar and Gudi (2005), Staudt and Soares (2009), Chia et al. (2021), and Herrera Velázquez et al. (2022). Given that there is no iterative interaction between the deterministic and the stochastic step, the performance of neither of the algorithms improves. Moreover, this sequential methodology would fail at providing local optimality guarantees in flowsheet optimization problems implemented within a commercial simulator without MINLP capabilities. More advanced strategies hybridize stochastic and deterministic strategies in a nested fashion (Fig. 1B), by keeping the stochastic method in an outer loop and the deterministic strategy in an inner loop, see e.g., Urselmann et al. (2011a, 2011b, 2016) , Zhou et al. (2017), Gómez et al. (2006), and Holtbruegge et al. (2015). Most of those works implement a memetic algorithm, i.e., an evolutionary algorithm that optimizes the discrete (or a mix of discrete and continuous) DOF in an outer loop, while NLP optimization with discrete variables fixed is used for each solution candidate in an inner loop. Although these memetic algorithms have demonstrated improvements in solution quality and computational performance, they fail to guarantee local optimality for discrete DOF in the inner loop. To address this issue, Kruber et al. (2021) and Skiborowski et al. (2015) used GAMS to investigate the inclusion of both discrete and continuous DOF when evaluating solution candidates in the inner loop. To avoid the rigorous solution of an MINLP for each solution candidate, Skiborowski et al. (2015) recommend a continuous reformulation of discrete variables such as feed locations and number of stages, which approximates the solution of the MINLP as a series of NLPs.

Considering a simulation software to define the flowsheet design optimization problem exhibits a few attractive features. For instance, they do not require the explicit formulation of the model equations and they allow to modify the underlying property methods, e.g., test different thermodynamic packages. Also, models embedded in chemical engineering software can be applied to a wide range of flowsheets with multiple unit operations and degrees of freedom (DOF) while preserving the robustness of the models, thanks to their readily available unit operation, thermodynamic, and physical property models. Nevertheless, a practical limitation of a simulation tool is the lack of integration between mathematical programming techniques and process simulators. Chemical engineering simulation software are typically able to perform local NLP optimizations; nonetheless, most simulation tools do not have MINLP optimization capabilities (Franke, 2017; Hernández-Pérez et al., 2020; Javaloyes-Antón et al., 2022). This hinders the application of hybrid algorithms that rely on MINLP optimization to flowsheet design optimization with simulation software. For instance, the nested hybrid algorithm proposed by Skiborowski et al. (2015) relies on the continuous reformulation of a MINLP problem; however, traditional simulation tools do not allow the continuous reformulation of discrete decisions within their corresponding NLP optimizer. To the authors' knowledge, a hybrid deterministic-stochastic algorithm that can guarantee local optimality for both discrete and continuous DOF is not currently available to optimize the process design using chemical engineering simulation software.

This work aims to propose a new hybrid stochastic-deterministic algorithm to optimize the design of chemical process flowsheets involving a mixture of continuous and ordered discrete decisions[1], which is a common characteristic in chemical engineering applications. Examples of ordered discrete decisions include selecting the number of stages in multi-effect evaporator sequences (Hong et al., 2019), number of reactors connected in series or parallel (Zhang et al., 2018), or the number of trays and location of interconnecting streams in separation

[1] We refer to ordered discrete decisions instead of integer decisions, given that, depending on the problem formulation, a reformulation step may be needed to identify these ordered structures (Liñán and Ricardez-Sandoval, 2023).

sequences (Ma et al., 2021). While previous works hybridize deterministic and stochastic algorithms in sequential and nested arrangements (Fig. 1A and B), the present work proposes a parallel configuration to perform the hybridization, i.e., the stochastic algorithm runs in one processor whereas the deterministic strategy runs in an alternate processor (Fig. 1C). The proposed hybrid algorithm combines a stochastic method (SM) with a deterministic optimization algorithm that is based on the previously developed deterministic Discrete-Steepest Descent Algorithm (DSDA) (Liñán et al., 2021, 2020). A new version of the traditional DSDA that integrates this method with a variable bounding (VB) strategy that iteratively generates tighter search bounds and avoids convergence failures is presented herein. A key feature of this deterministic algorithm is its capability to optimize discrete DOF within flowsheet simulation software. This capability eliminates the need to rely solely on a SM for optimizing discrete decisions. Rather than focusing on the comparison of different hybrid strategies, this work highlights the performance of the proposed parallel-hybrid strategy against a pure SM. The utilization of pure SMs, specifically the Differential Evolution with Tabu List (DETL), has undergone extensive testing in solving highly nonlinear problems characterized by a large number of equations, which include features such as discontinuous and multi-objective functions (Contreras-Zarazúa et al., 2022; González-Navarrete et al., 2022; Rangaiah, 2008; Rangaiah et al., 2020; Romero-García et al., 2022; Sánchez-Ramírez et al., 2022; Wang et al., 2020; Wang and Rangaiah, 2017). SMs have proven to be efficient at finding good quality solutions, thanks to their convergence proofs to a global optimum in infinite time (Liberti and Kucherenko, 2005; Zielinski and Laur, 2008). Nonetheless, the computational effort required by these methods is often significant and is still an open challenge. This bottleneck has prompted the exploration of a parallel-hybrid methodology, conceived with the objective of tackling the existing computational time limitations. By reducing computational time, the proposed parallel-hybrid approach aims to make SMs more accessible and applicable in real-world scenarios. The key novelties of the proposed hybrid algorithm are summarized as follows:

- The proposed hybrid algorithm is the first methodology that executes a deterministic algorithm and a stochastic algorithm in parallel, which is a feature that has not been exploited by the available hybrid strategies for optimal design of process flowsheets that makes use of chemical process simulators. This ultimately leads to solutions that satisfy local optimality for both discrete and continuous DOF, i.e., concepts from discrete convex analysis (Murota, 2003) are used to verify the local optimality of discrete DOF, while the local optimality requirement of continuous variables is assessed by an NLP solver. These optimality criteria are not guaranteed by the available methodologies for flowsheet design available in the literature.
- The DSDA-VB is developed as a deterministic MINLP algorithm integrated within the proposed hybrid method. This results in the first application of a DSDA-based algorithm to optimize problems involving flowsheet simulators. The proposed DSDA-VB strategy allows to iteratively update the search region explored by the SM. The VB strategy also provides adequate variable bounds when the NLP solver used by DSDA-VB is prone to convergence failure, which is a common situation in process flowsheet simulation software.

This study is organized as follows: Section 2 presents a general description of the optimization problem addressed in this work. Section 3 describes the working principle of the SM and DSDA algorithms. This section also introduces the proposed hybrid algorithm and the DSDA-VB strategy. Section 4 illustrates the main features of the proposed method using a thermally coupled system and a sequence of reactive, extractive, and traditional distillation columns as case studies. Concluding remarks and future areas of research are outlined in Section 5.

2. Problem statement

This work considers the optimization problem presented in Eq. (1), which involves the minimization of an objective function (f), e.g., minimize capital and operating costs. Vector of variables $\boldsymbol{x} \in \mathbb{R}^{n_x}$. represents the continuous DOF of the problem, e.g., output flow specifications, equipment sizes, energy requirements, among others. Variables in $\boldsymbol{y} \in \mathbb{Z}^{n_y}$. are the discrete DOF of the problem, i.e., a given combination of y completely specifies structural decisions of the flowsheet such as number of sequential/parallel units, feed locations, location of interconnecting flows, etc. There are other continuous variables that are handled internally by the simulator. Among these, variables in vector $z \in \mathbb{R}^{n_z}$. are the outputs of interest among the internally managed variables, e.g., output flows and compositions that are not specified by the user.

$$
\begin{aligned}
&\min_{x,y,z} f(\boldsymbol{x},\boldsymbol{y},z) \\
&s.t. \\
&\boldsymbol{x}^L \le \boldsymbol{x} \le \boldsymbol{x}^U,\ \boldsymbol{y}^L \le \boldsymbol{y} \le \boldsymbol{y}^U,\ \boldsymbol{g}(\boldsymbol{x},\boldsymbol{y},z) \le 0,\ [\boldsymbol{x},\boldsymbol{y}] \in \Omega
\end{aligned} \tag{1}
$$

Different types of constraints are considered, including bounds on the decision variables ($\boldsymbol{x}^L \le \boldsymbol{x} \le \boldsymbol{x}^U$, $\boldsymbol{y}^L \le \boldsymbol{y} \le \boldsymbol{y}^U$), constraints with a known mathematical expression ($\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{y}, z) \le 0$) and the constraints handled internally by black-box simulations ($[\boldsymbol{x},\boldsymbol{y}] \in \Omega$). The upper (superscript U) and lower (superscript L) bounds for continuous and discrete DOF are denoted by $\boldsymbol{x}^L$,$\boldsymbol{x}^U$ and $\boldsymbol{y}^L$, $\boldsymbol{y}^U$ respectively. Given that one of the key features of the proposed hybrid algorithm is the iterative refinement of these bounds, their values will be updated throughout the iterative solution of optimization problems. Thus, bounds $\boldsymbol{x}^{L,0}, \boldsymbol{x}^{U,0}, \boldsymbol{y}^{L,0}$ and $\boldsymbol{y}^{U,0}$ will be referred to as the user-defined bounds of the problem, which impose limits over the iteratively refined algorithmic bounds as follows: $\boldsymbol{y}^{L,0} \le \boldsymbol{y}^L \le \boldsymbol{y} \le \boldsymbol{y}^U \le \boldsymbol{y}^{U,0}$ and $\boldsymbol{x}^{L,0} \le \boldsymbol{x}^L \le \boldsymbol{x} \le \boldsymbol{x}^U \le \boldsymbol{x}^{U,0}$. Vector of constraints $\boldsymbol{g}$ includes those design requirements that are specified by the user, e.g., product compositions. Other constraints are handled internally by the simulator, and these are represented through set Ω. These constraints include mass and energy balances, equilibrium relationships and thermodynamic models, reaction rates, etc. Set Ω is therefore defined as those combinations in x and y that guarantee convergence of all the constraints within the simulator. Throughout this work, it is assumed that objective function f is single-objective function that is bounded from below, and that the effective domain of f is nonempty. Also, functions f and $\boldsymbol{g}$ are assumed to be twice differentiable with respect to the continuous variables. Concerning the structure of the DOF, the only requirement is that variables $\boldsymbol{y}$ must be ordered, which is a coon feature in process flowsheets such as sequences of intensified distillation systems, reactor networks, multi-effect evaporator sequences, among others.

The solution strategy developed in this study aims to handle problems involving multiple interconnected units in a flowsheet, with high dimensionality in the discrete ($\boldsymbol{y}$) and continuous ($\boldsymbol{x}$) DOF. This work emphasizes the application of the proposed strategy to process optimization with flowsheet simulators using rigorous process models, thanks to the flexibility process simulators offer to include multiple operation units in a simulation, i.e., their readily available unit operations, nonideal thermodynamic and physical property models allow to simulate processes with multiple interconnected unit operations. Nevertheless, the hybrid algorithm proposed in this work could also be applied in the context of optimization modeling languages, e.g., GAMS or Pyomo, although these would require the users to build their own first-principles process models.

3. Mathematical framework

The approach proposed in this work to solve problem (1) is a hybrid deterministic-stochastic algorithm that combines an SM with the herein proposed DSDA-VB strategy. An overview of stochastic and deterministic optimization using the DSDA is provided in sections 3.1 followed by the proposed hybrid algorithm presented in Section 3.2.

3.1. Overview of stochastic methods and the DSDA

Stochastic methods (SMs) aim to find the optimal solution of a problem by iteratively generating and evaluating a set of candidate solutions based on random sampling. These algorithms use a probabilistic approach to explore the search space and escape local optima, rather than relying on deterministic approaches that often converge to a local solution. The operation of stochastic optimization algorithms typically involves four steps: initialization, generation, evaluation, and selection. These steps are repeated until a satisfactory solution is found or a stopping criterion is met. The initialization step involves generating an initial set of candidate solutions randomly. The generation step involves creating new candidate solutions by applying random perturbations or mutations to the current population. The evaluation step involves calculating the objective value of each candidate solution using the objective function and constraints defined for the problem. The objective value is used to assess the quality of the solution and compare it to other candidate solutions. The selection step involves choosing the best candidate solutions based on some criteria, such as the highest objective value, lowest cost, or shortest distance. Once the selection step is completed, the algorithm repeats the generation, evaluation, and selection steps until a satisfactory solution is found or a stopping criterion is met, e.g., the number of iterations, the quality of the best solution found, or a predefined threshold (Deb, 2009; Spall, 2003).

The DSDA is a deterministic decomposition algorithm originally developed for integer programming (Murota, 2003), and recently extended to MINLP optimization problems involving continuous and ordered discrete decisions such as those emerging in vector $\boldsymbol{y}$ (Liñán et al., 2020). To initialize the DSDA, the user must provide a feasible initialization for the variables in $\boldsymbol{y}$. At each iteration, the DSDA fixes the values of discrete decisions at their current trial value (denoted by y^*) in Problem (1) and solves for the remaining continuous variables $\boldsymbol{x}$ and $\boldsymbol{z}$ using NLP optimization to return an objective function value $f^*(y^*) = f(\boldsymbol{x}^*, \boldsymbol{y}^*, z^*)$ with optimized continuous variables $\boldsymbol{x}^*$ an z^*. The NLP optimization problem that results from fixing discrete decisions in Problem (1) is referred to as a subproblem. Discrete variables are then fixed at every discrete point within a neighborhood of $\boldsymbol{y}^*$ denoted as $N_\Theta(\boldsymbol{y}^*)$, where Θ indicates the neighborhood type. For example, this neighborhood typically considers those discrete points whose ℓ_∞ -norm (infinity norm) with respect to $\boldsymbol{y}^*$ is less or equal to l, and it is denoted by $N_\infty(\boldsymbol{y}^*)$. To reduce the computational effort, subsets of $N_\infty(\boldsymbol{y}^*)$ such as the N_2-neighborhood ($N_2(\boldsymbol{y}^*)$) with $2n_y + 1$ points within an ℓ_2 -norm of $\boldsymbol{y}^*$ are often considered in practice. Each neighbor $\boldsymbol{y} \in N_\Theta(\boldsymbol{y}^*)$. results in an NLP subproblem that returns a new objective function value $f^*(y) = f(\boldsymbol{x}^*(y), \boldsymbol{y}, z^*(y))$, where notation $\boldsymbol{x}^* = \boldsymbol{x}^*(y)$ and $z^* = z^*(y)$ represents the optimized values of $\boldsymbol{x}$ and $\boldsymbol{z}$ with discrete variables $\boldsymbol{y}$ fixed in Problem (1). Given that discrete variables in $\boldsymbol{y}$ are ordered, concepts from discrete convex analysis can be used to define local optimality, as stated in Definition 1.

Definition 1. Consider function $f^*(y) = f(\boldsymbol{x}^*(y), \boldsymbol{y}, z^*(y))$, where function $f^* : \mathbb{Z}^{n_y} \mapsto \mathbb{R} \cup \{+\infty\}$ takes the optimal objective function of the subproblem obtained after fixing $\boldsymbol{y}$ in Problem (1) for feasible values of $\boldsymbol{y}$ and $+\infty$ otherwise. A point y^* is a *local minimum* of f^*. if $f^*(y^*) \leq f^*(y)$ for every $\boldsymbol{y}$ that belongs to $N_\Theta(\boldsymbol{y}^*)$ (Murota, 2003).

Based on Definition 1, if $f(y^*)$ takes the minimum value within its neighborhood, then y^* is declared as locally optimal and the DSDA algorithm stops; otherwise, the combination of discrete decisions that minimize the objective function is used to define a steepest-descent direction. NLP problems with discrete variables fixed are then sequentially solved along the steepest-descent direction until no improvement is found. Then, a new iteration of the algorithm begins with a new neighborhood exploration. Previous works have demonstrated that the DSDA outperforms traditional MINLP and GDP optimization algorithms when optimizing systems involving binary or Boolean variabs defined over ordered discrete sets (Liñán et al., 2021, 2020; Liñán and Ricardez-Sandoval, 2023). Note that several discrete analogues of convexity such as integral-convexity, $M^\natural$-convexity and $L^\natural$-convexity have been studied in the literature to detect when a local optimum according to Definition 1 is a global solution (e.g., see Chen and Li (2021); Murota and Tamura (2023), and Zhang et al. (2022)); however, function $f^*(\boldsymbol{y})$. does not have an analytical representation because it is obtained via NLP optimization, i.e., the convexity properties of $f^*(y)$ are unknown a priori. For this reason, only local optimality can be guaranteed in this work.

Some of the limitations of stochastic and deterministic methods can be overcome by combining them into a single optimization framework when solving flowsheet design problems. Typically, exceeding a resource limit is considered as the stopping criterion for SMs, e.g., stop once a maximum number of objective function evaluations is attained (F_{max}). However, selecting the appropriate resource limit (e.g., F_{max} value) requires trial-and-error tests, given that the computational performance depends on the specific characteristics of each optimization problem, e.g., number of variables, non-linearities, and variable's bounds. Additionally, the stopping criterion may differ in different runs of the same problem due to the stochastic nature of SM algorithms (Zielinski and Laur, 2008). A deterministic optimization algorithm such as DSDA can help to alleviate these issues since it can be used to improve and assess the optimality characteristics of the current best solutions identified by the SM. Hence, if the current SM solution is close to a locally optimal solution, then the execution of the SM can stop. Nevertheless, the DSDA exhibits other limitations, such as the initialization requirement (i.e., a feasible combination of discrete decisions), or the exponential growth of discrete neighbors when using a ℓ_∞ -norm to define the vicinity of a discrete point (Liñán and Ricardez-Sandoval, 2023)The DSDA may benefit from SMs, which use randomness and probabilistic techniques to search for feasible and optimal solutions. These methods do not guarantee finding optimal solutions, but they can be efficient in exploring large search spaces and finding acceptable feasible solutions. In the case of constrained optimization problems, stochastic optimization methods typically use a penalty function approach, where the objective function is modified to include a term that penalizes infeasible solutions. This penalty term increases as the solution violates constraints, and the objective function is minimized while keeping the penalty as low as possible. Another approach used in stochastic optimization for constrained problems is the use of constraint handling techniques, such as the use of repair mechanisms to transform infeasible into feasible solutions. These mechanisms can modify the solution to satisfy the constraints, but they must be designed carefully to avoid introducing new infeasibilities. Concerning the size of the neighborhoods, subsets of $N_\infty(\boldsymbol{y}^*)$ such as $N_\Theta(\boldsymbol{y}^*) = N_2(\boldsymbol{y}^*)$. have been considered in previous DSDA works to decrease computational costs, at the expense of scarifying local optimality within the larger $N_\infty(\boldsymbol{y}^*)$ neighborhood. To alleviate this issue, both the SM and the DSDA can perform the neighborhood exploration, by exploring $N_\Theta(\boldsymbol{y}^*) \subset N_\infty(\boldsymbol{y}^*)$ neighborhoods with DSDA and heuristically using the SM to find candidate solutions within the larger $N_\infty(\boldsymbol{y}^*)$ neighborhood.

One problem that affects both the DSDA and SMs is the adequate selection of bounds to perform the search (i.e., $\boldsymbol{x}^{L,0}, \boldsymbol{x}^{U,0}, \boldsymbol{y}^{L,0}$ and $\boldsymbol{y}^{U,0}$). Bounds on optimization variables limit the search space of SMs, which can have a significant impact on the optimization process. While tight bounds can lead to premature convergence and limit the exploration of the search space, thereby leang to suboptimal solutions, overly loose bounds can lead to a broader search space, thus allowing the identification of attractive solutions, at the cost of lengthy (or even prohibitive) computational times. Selecting bounds that balance the trade-off between exploration and exploitation of the search space is thus key to find attractive feasible solutions in acceptable turnaround times. If bounds are not properly selected for SMs, slow convergence, algorithm stagnation or lack of convergence may arise. In the case of the DSDA, an

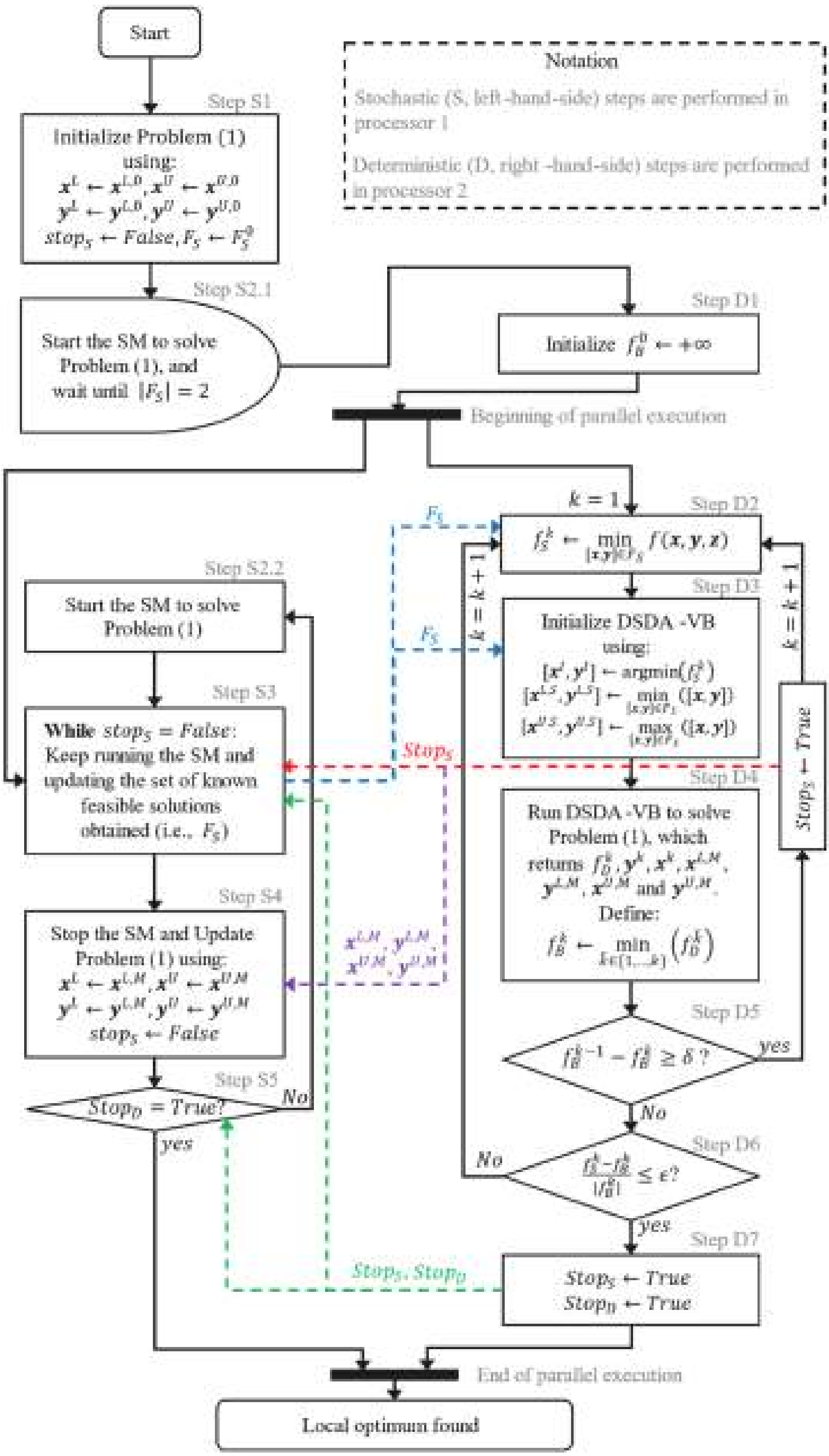

Fig. 2. Proposed deterministic-stochastic algorithm. Black rectangles at the top and the bottom represent the beginning and the end of the parallelization, respectively. Dashed lines represent the exchange of information.

inadequate selection of bounds for discrete variables (i.e., $\boldsymbol{y}^{L,0}$ and $\boldsymbol{y}^{U,0}$) does not represent a major concern since the algorithm is successively fixing discrete variables and discarding infeasible neighbors. Nevertheless, bounds for continuous variables (i.e., $\boldsymbol{x}^{L,0}$ and $\boldsymbol{x}^{U,0}$) must be carefully selected for the NLP solver used within DSDA, whose performance may be affected by these bounds. For instance, the NLP optimization performed by process flowsheet simulators such as Aspen Plus typically implements Sequential Quadratic Programming (SQP), which requires good initial estimations and suitable bounds for the continuous DOF. Overly restricted variable bounds may result in optimal solutions lying at their lower/upper bounds, while loose bounds may result in lack of convergence of the SQP algorithm.

3.2. Proposed hybrid deterministic-stochastic algorithm

The hybrid algorithm presented in this section addresses the limitations described above for SMs and the DSDA by iteratively refining variable bounds within a parallel-hybrid algorithm. In this work, the variable's bounds considered by the SM are updated by the deterministic algorithm. That is, iteratively refined variable bounds (i.e., modified algorithmic bounds $\boldsymbol{x}^{L,M}, \boldsymbol{x}^{U,M}, \boldsymbol{y}^{L,M}$ and $\boldsymbol{y}^{U,M}$) are used to iteratively update the bounds on the decision variables ($\boldsymbol{x}^L, \boldsymbol{x}^U, \boldsymbol{y}^L$ and $\boldsymbol{y}^U$), instead of using fixed user-defined bounds (i.e., $\boldsymbol{x}^L = \boldsymbol{x}^{L,0}, \boldsymbol{x}^U = \boldsymbol{x}^{U,0}, \boldsymbol{y}^L = \boldsymbol{y}^{L,0}$ and $\boldsymbol{y}^U = \boldsymbol{y}^{U,0}$) throughout the execution of the SM. For continuous variables ($\boldsymbol{x}$), this is achieved by embedding the DSDA within a variable bounding (VB) algorithm that iteratively refines bounds $\boldsymbol{x}^{L,M}$ and $\boldsymbol{x}^{U,M}$ based on the convergence status of the NLP optimizer, i.e., the DSDA-VB algorithm. For discrete variables ($\boldsymbol{y}$), the N_∞ neighborhood of each locally optimal solution found by DSDA-VB defines refined bounds $\boldsymbol{y}^{L,M}$ and $\boldsymbol{y}^{U,M}$.

Parallelizing the SM and the DSDA-VB offers different advantages. First, both the SM and DSDA-VB are being executed simultaneously, resulting in faster convergence compared to processing the SM and DSDA-VB optimization steps sequentially. Also, the proposed parallelization allows discrete and continuous DOF to be optimized by both the deterministic and stochastic techniques simultaneously, which leads to solutions that satisfy local optimality requirements for both discrete and continuous DOF. This feature would not be possible using the available hybrid stochastic-deterministic strategies for flowsheet optimization with process simulators. The parallel implementation presented in this work also offers advantages with respect to the independent execution of the DSDA and an SM, e.g., improvements in neighborhood exploration compared to the traditional DSDA, and improved convergence of the SM in situations where the traditional SMs may stagnate.

Based on the above, the SM and DSDA-VB algorithms benefit from each other by considering their parallel execution in an iterative manner. As shown in Fig. 2, the algorithm starts with an initialization that consists of sequential *Steps S1, S2.1* and *D1*. Then, the parallel algorithm begins by simultaneously executing *Step S3* of a stochastic algorithm that iteratively executes the SM (see Section 3.2.2), and *Step D2* of a deterministic algorithm that iteratively executes the DSDA-VB (see Section 3.2.3). The main iterations of the parallel-hybrid algorithm are denoted by superscript k. The overall idea of the proposed algorithm is to initialize and start the execution of the SM (*Steps S1* and *S2.1*) and have the SM running, while the deterministic algorithm is constantly retrieving a database consisting of all feasible solutions (F_S) detected by the SM. The deterministic algorithm waits until enough feasible solutions are available in F_S. Then the DSDA-VB is initialized (*Step D1*) by assigning an objective function value of $+\infty$ to the best objective function identified thus far by the DSDA-VB ($f_B^0 \leftarrow +\infty$). The DSDA-VB uses the solution with the best objective function value from F_S (i.e., f_S^k) as initialization (*Step D2*), and then performs a local optimization to generate modified bounds $\boldsymbol{x}^{L,M}, \boldsymbol{x}^{U,M}, \boldsymbol{y}^{L,M}$ and $\boldsymbol{y}^{U,M}$ (*Steps D3* and *D4*), which are returned to the SM when there is an improvement in f_B^k (see *Step D5* and purple dashed lines in Fig. 2). If f_B^k improved in *Step D5*, then the SM search is re-started using the modified bounds obtained by DSDA-VB (*Steps S4* and *S2.2*). This procedure is repeated until the stopping criterion in *Step D6* is satisfied, meaning that both the deterministic (*Step D7*) and stochastic (*Step S5*) algorithms are terminated. According to this stopping criterion, the parallel hybrid strategy stops when the relative difference between f_B^k and f_S^k is below an ϵ-tolerance. Note that iterations k are controlled by the deterministic algorithm i.e., *Steps D2-D6*. Iterations are managed by the deterministic steps because these define when the SM must be reinitialized (see *Step D5* and red dashed lines in Fig. 2), and when both deterministic and stochastic steps must stop (see *Steps D6* and *D7* and green dashed lines in Fig. 2). Fig. S1 in the supplementary material provides in-depth (high level) information about the algorithm presented in Fig. 2, including information related to solver interactions, variables and constraints considered at each step, and data exchange aspects.

3.2.1. Convergence analysis and summary

The proposed hybrid method enforces requirements over the stochastic and deterministic algorithms to successfully converge to a local optimum. In general, the hybrid stochastic-deterministic strategy introduced above is expected to converge and close the gap within an ε-tolerance (ε≥0) if the three conditions detailed in Theorem 1 are satisfied (a detailed proof this theorem is available in section S2 of the supplementary material).

Theorem 1. Assume that: 1) the first run of the SM in *Step S1* (iteration *k=0*) can identify at least two different feasible solutions which are added to F_S; 2) the DSDA-VB in *Step D4* can eventually find at least one locally optimal objective function (f_D^k) using feasible information in F_S, and 3) the objective function in optimization problem (1) is bounded, i.e., the objective function cannot be improved indefinitely without violating the constraints, and the SM in *Steps S2.2* and *S3* eventually converges towards the global optimum, subject to the current selection of bounds ($\boldsymbol{x}^{L,M}, \boldsymbol{x}^{U,M}, \boldsymbol{y}^{L,M}, \boldsymbol{y}^{U,M}$). Then, the proposed hybrid stochastic-deterministic algorithm will eventually terminate in a local optimum for any user-defined ε-tolerance greater or equal to 0.

The hybrid strategy discussed above has the potential to improve the performance of both the SM and the DSDA when solving flowsheet design problems. First, the hybrid algorithm in Fig. 2 proposes a parallel execution of a deterministic and a stochastic algorithm that not only ensures local optimality for continuous variables, but also guarantees local optimality for discrete decisions within N_2-neighbors. To the authors' knowledge, these optimality guarantees cannot be provided by other hybrid algorithms available in the literature, see e.g., Gómez et al. (2006), Staudt and Soares (2009), Zhou et al. (2017), Herrera Velázquez et al. (2022). This parallel execution strategy also addresses some of the limitations that SMs or deterministic algorithms such as the DSDA exhibit when executed separately. For instance, the inability of DSDA to handle infeasible initializations is addressed by retrieving feasible solutions from the SM through set F_S, and the prohibitive evaluation of N_∞-neighborhoods for problems with many discrete decisions is addressed by delegating the N_∞-neighborhood exploration to the SM. Also, the slow convergence of the SM with respect deterministic strategies is addressed by continuously generating updated search bounds with the DSDA-VB method, and by defining a stopping criterion that incorporates stochastic and deterministic objective function values. In summary, the hybrid method takes advantage of the SM to identify feasible solutions at early stages of the execution (iteration $k = 0$) and explore a neighborhood of the current solution candidate (iterations $k \geq 1$). The SM heuristically seeks the global optimum of the problem within the neighborhood it is exploring, but it does not provide optimality guarantees. The deterministic DSDA-VB strategy, on the other hand, uses information from the SM to update search limits and, under a deterministic approach, DSDA-VB satisfies the local optimality conditions in Definition 1 for the highly nonlinear model in Problem 1. This is expected to result in shorter computational times, as illustrated in

Section 4. It is also acknowledged that the proposed hybrid algorithm has limitations. The proposed hybrid strategy still needs improvements to handle multi-objective, discontinuous, and highly multimodal objective functions. Also, updating the search bounds of the SM using DSDA-VB help the algorithm to converge faster. However, the bounds obtained using the proposed VB strategy only guarantee feasibility and local optimality; hence, any set of bounds returned by DSDA-VB may miss the global optimum. Detailed descriptions of the steps involved in the proposed parallel-hybrid algorithm are discussed below.

3.2.2. Steps followed by the stochastic algorithm

Steps S1 and S2.1: In *Step* S1, the stochastic algorithm begins by setting up Problem (1) aided with a process simulator for flowsheet calculations and initializing the algorithm using: 1) the user-defined variable bounds (i.e., $x^{L,0}, x^{U,0}, y^{L,0}$ and $y^{U,0}$.) for Problem (1), 2) a value of *False* is selected for control flow parameter $stop_S$., which will be latter used to define when to execute or reinitialize the SM, and 3) known feasible DOF combinations for set F_S, e.g., $F_S^0 = \varnothing$, indicating that no feasible solution is known for the problem. After that, an iterative procedure starts with the first execution of the SM in *Step S2.1*.

Step S3 and S4: According to *Step S3*, the execution of the SM keeps updating set F_S with new feasible solutions until the deterministic strategy determines that the execution of the SM must be stopped, i.e., until $stop_S$ takes a value of *True*. As described below in *Step D5,* stopping the SM occurs whenever the deterministic algorithm identifies an improvement in the best locally optimal solution found. Once $stop_S =$ *True*, the current SM run is stopped as indicated in *Step S4*.

Steps S4 and S5: As indicated in *Step S4*, $stop_S$ is set back to *False* and variable bounds in Problem (1) are updated using the modified bounds returned by the deterministic algorithm, denoted by purple dashed lines in Fig. 2 ($x^{L,M}, x^{U,M}, y^{L,M}$ and $y^{U,M}$.). One of the purposes of these new bounds is to guide the SM towards the best locally optimal solution identified by the deterministic strategy thus far, e.g., by keeping $y^{L,M}$ and $y^{U,M}$ within a N_∞-neighborhood of the best local solution. Before restarting the execution of the SM using these new bounds, a global stopping criterion is verified in *Step S5*. This global stopping criterion uses control flow parameter $Stop_D$, which stops the execution of both algorithms based on the best objective functions found by both the deterministic and stochastic algorithms. Flow parameters $Stop_D$ and $Stop_s$ (red and green dashed lines in Fig. 2) are controlled by the deterministic algorithm (*Steps D1-D7*), which is explained next.

3.2.3. Steps followed by the deterministic algorithm

Step D1: The deterministic strategy is constantly retrieving the most updated version of set F_S from either *Step S2.1* or *Step S3* (see *Steps D1* and *D2*). In *Step D1*, the deterministic strategy must await until at least two different feasible solutions are available in F_S, which is a requirement to initialize the DSDA-VB strategy. These two feasible solutions from the stochastic technique are required to provide a feasible initialization ($[x^I, y^I]$) to *Step D3* and an initialization for continuous variables bounds ($[x^{L,S}, x^{U,S}]$) for the execution of DSDA-VB in *Step D4*. The deterministic strategy is then initialized in *Step D1* by assigning a value of $+\infty$ to the best locally optimal objective function known f_B^0, where subscript B stands for "best". Having $f_B^0 = +\infty$ means that no locally optimal solution has been detected yet. Then, the iteration count k. is set at $k = 1$ and the iterations of the parallel hybrid algorithm start.

Steps D2 and D3: In *Step D2*, the solution with the best objective function value is retrieved from F_S and assigned to f_S^k, i.e., f_S^k is the best solution reported by the stochastic algorithm at iteration k. The DSDA-VB algorithm is then initialized at *Step D3* using the DOF associated to f_S^k (i.e., $[x^I, y^I]$), and the maximum ($[x^{U,S}, y^{U,S}]$) and minimum ($[x^{L,S}, y^{L,S}]$) DOF values in F_S.

Step D4 (The DSDA-VB): Selecting adequate variable bounds for deterministic NLP optimization is critical to avoid execution errors and improve solver performance, e.g., good bounds m avoid variable saturation, moving away from an optimal solution, or moving into regions with large function values, large derivative values, or singularities. Despite their relevance, variable bounds typically remain fixed, and the task of defining appropriate bounds is typically delegated to the user. To alleviate this issue, the DSDA-VB algorithm has been developed as an optimization method that iteratively executes the DSDA and a VB strategy that recomputes bounds for continuous variables ($x^{L,M}, x^{U,M}$). The VB strategy includes three distinctive features. First, it helps to circumvent potential convergence issues of the NLP solver. Second, it allows to keep the continuous variables returned by DSDA-VB (i.e., x^k) away from their algorithmic bounds $x^{L,M}$ and $x^{U,M}$; otherwise, a solution may be erroneously declared as locally optimal. Third, the VB strategy helps to indirectly accelerate the SM, whose convergence rate may benefit from exploring a subset of the original search space that is known to contain at least one locally optimal solution. Note that bounds for discrete decisions do not need to be iteratively recalculated within DSDA-VB, because the DSDA can consider the user-defined bounds $y^{L,0}$ and $y^{U,0}$ whenever it is executed, without affecting its performance. Instead, updated bounds of discrete decisions ($y^{L,M}, y^{U,M}$) are defined as upper and lower bounds of a discrete neighborhood of the local optimum identified by DSDA-VB. Note that these updated bounds of discrete DOF define a N_∞-neighborhood consisting of 3^{n_y} combinations of discrete DOF, which are considered by the SM whenever it is reinitialized. In contrast, each execution of the DSDA within DSDA-VB considers the smaller N_2-neighborhood of size $2n_y + 1$, given that the evaluation of N_∞-neighborhoods with DSDA may be computationally prohibitive. This means that the SM is allowed to explore many discrete DOF combinations that may improve the objective function, but that may be ignored by the DSDA-VB. In summary, the DSDA-VB executed in *Step D4* of Fig. 2 is an algorithm that not only optimizes the discrete and continuous DOF of the problem locally, but also returns new bounds for these variables as part of the solution.

The DSDA-VB returns an objective function (f_D^k) and the locally optimal solution found ($[x^k, y^k]$), as well as new variable bounds ($x^{L,M}, x^{U,M}, y^{L,M}$ and $y^{U,M}$) that are tighter than those originally defined by the user, i.e., $x^{L,0} \le x^{L,M}, x^{U,M} \le x^{U,0}, y^{L,0} \le y^{L,M}$ and $y^{U,M} \le y^{U,0}$. On the one hand, continuous variables bounds ($x^{L,M}, x^{U,M}$) guarantee that the continuous variables reported by DSDA-VB ($[x^k, y^k]$) are not at their bounds and that NLP convergence issues are avoided, e.g., discontinuities in the constraints that cause convergence errors when bounds are inappropriately selected are avoided. On the other hand, updated discrete variable bounds ($y^{L,M}, y^{U,M}$) are restricted within an N_∞-neighborhood of the locally optimal solution y^k returned by DSDA-VB as indicated in Eqs. (2A) and (2B), e.g., for a problem with a single discrete DOF, the improved bounds for optimal solution $y^k = 3$ are $y^{L,M} = \max(y^{L,0}, 2)$ and $y^{U,M} = \min(y^{U,0}, 4)$. The best objective function found by the deterministic algorithm (f_B^k) is updated after the execution of the DSDA-VB in *Step D4* as $f_B^k = \min_{\tilde{k} \in \{1,\dots,k\}} (f_D^{\tilde{k}})$. A detailed diagram and description of the DSDA-VB algorithm is provided in the supplementary material (Section S1, Fig. S2).

$$y^{L,M} = \max\left(y^{L,0}, \min\left(N_\infty\left(y^k\right)\right)\right) \tag{2A}$$

$$y^{U,M} = \min\left(y^{U,0}, \max\left(N_\infty\left(y^k\right)\right)\right) \tag{2B}$$

Steps D5, D6 and D7: The remaining steps (*Steps D5, D6* and *D7*) of the deterministic algorithm shown in Fig. 2 decide when the stochastic algorithm must be reinitialized, and when both algorithms must be stopped through control flow parameters $stop_S$ and $stop_D$, respectively. *Step D5* decides if the SM must be stopped and reinitialized, based on an objective function improvement criterion that uses a small tolerance $\delta > 0$. If the best objective function found by the deterministic strategy in the current iteration (f_B^k) improved with respect to the best objective found in previous iterations (f_B^{k-1}), then control flow parameter $stop_S$ is

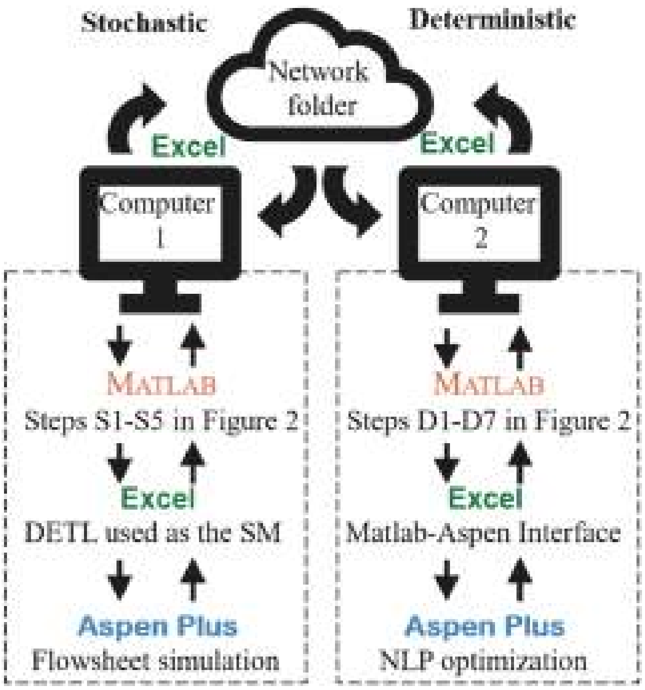

Fig. 3. Hybrid algorithm implementation and software considered in this work.

set to *True* indicating that the current SM run in the stochastic strategy must be stopped and reinitialized (see *Steps S3* and *S4* of the stochastic algorithm, which use the updated information from the red and purple dashed lines in Fig. 2). This allows the stochastic algorithm to update the search bounds of the SM and guide it towards the solution with objective f_B^k. If the stochastic technique does not require to be reinitialized, then the global stopping criterion of both algorithms is evaluated in *Step D6*. This global stopping criterion verifies if the relative difference between the best solutions found by the stochastic (f_S^k) and deterministic (f_B^k) algorithms is below an ϵ-tolerance. If the global stopping criterion is satisfied, the stochastic technique h thus converged near the best local solution found by the deterministic algorithm, hence both algorithms stop (*Step D7*). Otherwise, a new iteration of the deterministic strategy begins.

4. Computational experiments

This section presents two case studies that highlight the improvements in terms of solution quality and computational times when using the proposed hybrid algorithm to optimize the total annual cost (TAC in $/year) of intensified distillation flowsheets. In order to carry out the stochastic optimization process, we have selected the Differential Evolution with Tabu List (DETL) as the SM. This algorithm combines the advantages of Differential Evolution (DE), a powerful global search algorithm, with the concept of tabu search, which is a local search method that prevents the algorithm from revisiting previously explored solutions (Srinivas and Rangaiah, 2016). Fig. 3 illustrates the method used to implement the parallel algorithm proposed in this work. Two separate computers with a processor Intel® Core™ i7-3770 CPU @ 3.4 GHz and 16 GB of RAM were used to run the tasks related to the stochastic (computer 1) and deterministic (computer 2) algorithms. These computers were allowed to interact by exchanging information through a mappable network folder with 10 GB of storage capacity using Microsoft Excel. The main codes of the hybrid algorithm (Fig. 2) were implemented in Matlab. These codes were connected to an Excel spreadsheet that worked as an interface between Matlab and Aspen Plus, which was used for flowsheet simulation and NLP optimization with the SQP method. In the case of the stochastic algorithm, Excel was also used to

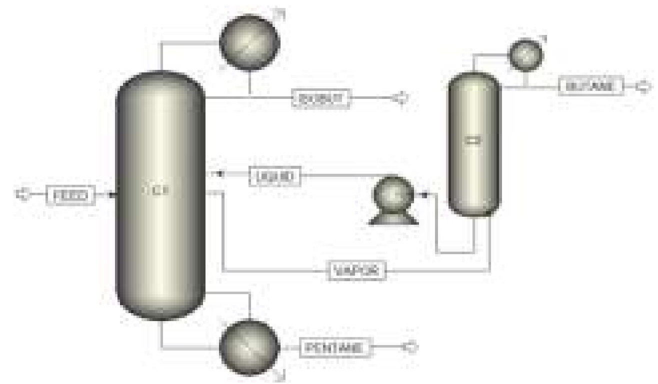

Fig. 4. Thermally coupled system that separates a mixture of isobutane, n-butane and n-pentane.

run an implementation of the DETL code that was developed in previous works (Sharma et al., 2012; Srinivas and Rangaiah, 2007).

The rigorous Mass-Equilibrium-Summation-entHalpy (MESH) model with non-ideal equilibrium relationships available in Aspen plus was considered. Vector of constraints *g* in Problem (1) was handled by DETL using the penalty function method, with a high penalty factor of $1 * 10^{10}$ that is likely to avoid infeasible solutions. The traditional DETL method was used as a benchmark to compare the performance of the proposed hybrid algorithm. To perform a fair comparison, the tuning parameters of DETL were kept fixed for all runs of the hybrid and traditional DETL method, respectively. That is, both algorithms were executed under the same conditions. We adopted standard tuning parameters available in the literature for DETL (Srinivas and Rangaiah, 2016; Storn, 1996): population size $NP = 10 * (n_x + n_y)$, crossover rate $CR{=}0.8$, mutation factor $F{=}0.7$, tabu list size $TLS = NP/2$, and tabu radius $TR = 1 * 10^{-6}$. For the DSDA exploration, N_2-neighborhoods were considered.

The objective function considered for both case studies is shown in Eq. (3A), which was obtained using the estimation method presented by Douglas (Douglas, 1988) and cost data available from previous works (Bernal et al., 2018; Ciric and Gu, 1994; Gómez et al., 2006). This function represents the addition of the annualized investment cost ($Investment_c$.) and the operating costs associated with utilities ($Operating_c$) for each column ($c \in C$) in the flowsheet. The investment cost in Eq. (3B) considers the column diameter in meters (d_c), the total column height in meters (h_c.), and the number of catalytic stages (nc_c) if any. The operating cost in Eq. (3c) represents the cost of the duties of the reboiler (Qr_c) and the condenser (Qc_c) in kilowatt. This objective function assumes 330 days of operation per year, an annual interest rate of 5%, and payback period of 5 years. Note that search bounds are imposed over d_c, h_c and nc_c whereas $|Qr_c|$. and $|Qc_c|$. are bounded below by 0, i.e., the objective function is bounded.

$$f = TAC = \sum_{c \in C} (Invesment_c + Operating_c) \tag{3A}$$

$$Invesment_c = 10000 + 292.67(3.28d_c)^{1.066}(3.77h_c)^{0.802} + 15.29(3.28d_c)^{1.55}h_c + 131.74(d_c)^2 nc_c \tag{3B}$$

$$Operating_c = 146.8|Qr_c| + 24.5|Qc_c| \tag{3C}$$

The distillation column models considered in this work involve multiple discrete DOF, continuous DOF, and constraints resulting from multiple interconnected units in a process flowsheet. Rigorous stage-by-stage nonlinear models combining mass/energy balances and non-ideal equilibrium relationships for each unit are considered. The problems considered in this work are non-convex due to the nonlinear relationships between the input variables (such as feed flow rate, reflux ratio, and tray temperatures) and the output variables (such as product purity

Table 1
Variables and their bounds for the first case study, and best solution found.

Continuous DOF (x)				
Variable	Units	Lower bounds ($x^{L,0}$)	Upper bounds ($x^{U,0}$)	Best solution found (Fig. 5A)
Reflux ratio (C1)	-	0.5	10	7.62
Distillate flow (C1)	kmol/h	17.29	19.11	18.18
Distillate flow (C2)	kmol/h	8.69	9.6	8.92
Vapor flow (from C1 to C2)	kmol/h	11.34	34.02	18.84
Diameter (C1)	m	0.1	6	1.68
Diameter (C2)	m	0.1	6	0.37
Discrete DOF (y)				
Variable	Units	Lower bounds ($y^{L,0}$)	Upper bounds ($y^{U,0}$)	Best solution found (Fig. 5A)
Number of stages (C1)	-	5	70	55
Number of stages (C2)	-	5	70	12
Feed stage (C1)	-	2	69	24
Liquid feed stage (from C2 to C1)	-	2	69	46

and recovery). Additionally, the large search spaces considered for discrete decisions and the nonlinear interactions between discrete and continuous variables make the problems described above challenging non-convex combinatorial optimization problems.

4.1. Case study 1: A thermally coupled system

The optimization of the thermally coupled system shown in Fig. 4 is considered. In this work, stages are numbered from top to bottom, i.e., the condenser corresponds to stage 1. The feed to the first column (C1) has a total molar flow rate of $\mathscr{F}^{FEED}_{total} = 45\ kmol/h$, with mole fractions of $\chi^{FEED}_{isobutane} = 0.4$, $\chi^{FEED}_{n-butane} = 0.2$ and $\chi^{FEED}_{n-pentane} = 0.4$. The first column aims to produce nearly pure isobutane at the top and nearly pure n-pentane at the bottoms. The separation of n-butane from the mixture is performed in another column (C2) that is thermally coupled to C1 through interconnecting vapor and liquid flows that are connected to the last stage of C2. These interconnecting flows avoid the need of a reboiler for the second column, decreasing energy consumption with respect to a traditional distillation sequence (Contreras-Zarazúa et al., 2019, 2021). The packages available in Aspen Plus were used for thermodynamic calculations. The Chao-Seader property method with Lee-Kesler enthalpy was considered, which uses the Redlich-Kwong equation of state for vapor phase properties, and the Scatchard-Hildebrand model for liquid activity coefficients (Carlson, 1996).

The DOF of this problem ($\boldsymbol{x}$ and $\boldsymbol{y}$), and their corresponding bounds

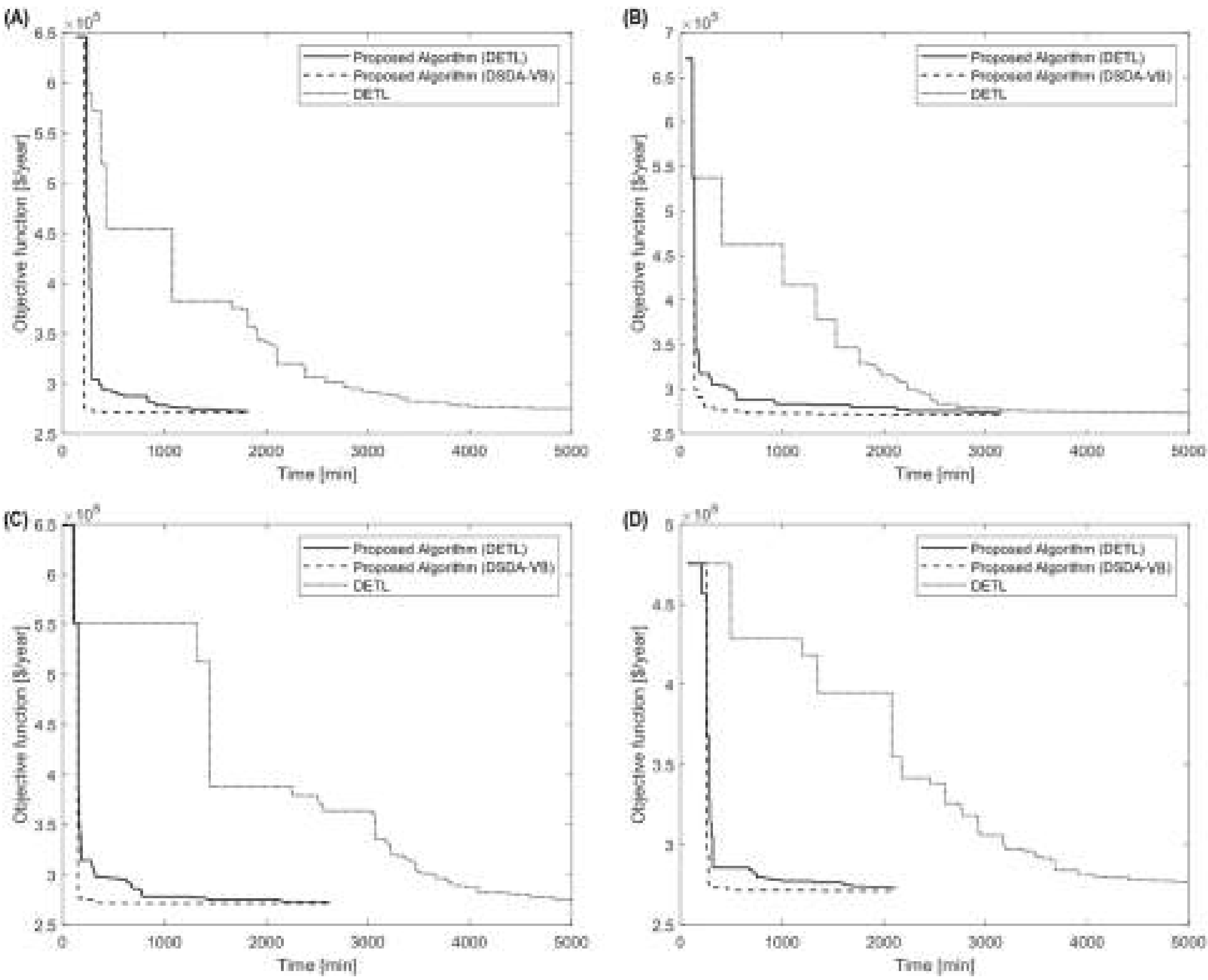

Fig. 5. Convergence plots for the optimal design of the thermally coupled system.

Table 2
Best solution found by each run of the hybrid algorithm (Fig. 5B-C).

Continuous DOF ($\boldsymbol{x}$)				
Variable	Units	Fig. 5B	Fig. 5C	Fig. 5D
Reflux ratio (C1)	-	7.40	7.64	7.61
Distillate flow (C1)	kmol/h	18.18	18.19	18.18
Distillate flow (C2)	kmol/h	8.92	8.92	8.92
Vapor flow (from C1 to C2)	kmol/h	18.99	19.29	18.95
Diameter (C1)	m	1.71	1.68	1.68
Diameter (C2	m	0.37	0.34	0.37
Discrete DOF ($\boldsymbol{y}$) Variable	Units	Fig. 5B	Fig. 5C	Fig. 5D
Number of stages (C1)	-	56	55	55
Number of stages (C2)	-	12	11	12
Feed stage (C1)	-	25	25	25
Liquid feed stage (from C2 to C1)	-	47	46	46
Objective function	\$/year	271519	271339	271281

($\boldsymbol{x}^{L,0}, \boldsymbol{x}^{U,0}, \boldsymbol{y}^{L,0}$ and $\boldsymbol{y}^{U,0}$) are summarized in Table 1. Note that the vapor flow that goes from C1 to C2 is assumed to be located one stage below the liquid flow that goes from C2 to C1. Other variables of interest that are retrieved from the optimization (i.e., those in vector $\boldsymbol{z}$) are the energy duties, which are required to calculate the objective function. Vector $\boldsymbol{z}$ also contains those variables required to specify the problem's constraints. These variables include the molar fraction and flow of the desired component for each output stream, i.e., molar flow ($\mathscr{F}^{ISOBUT}_{isobutane}$) and mole fraction ($\chi^{ISOBUT}_{isobutane}$) of isobutane at the top of C1; molar flow ($\mathscr{F}^{PENTANE}_{n-pentane}$.) and mole fraction ($\chi^{PENTANE}_{n-pentane}$) of n-pentane at the bottoms of C1, and molar flow ($\mathscr{F}^{BUTANE}_{n-butane}$) and mole fraction ($\chi^{BUTANE}_{n-butane}$) of n-butane at the top of C2. The constraints considered for this problem ($\boldsymbol{g}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$) are shown in Eqs. (4A)–(4C), which establish that: 1) a minimum molar recovery of 94% must be attained for all components, 2) a minimum molar product purity of 98%, 99% and 96% is enforced for isobutane, n-pentane and n-butane respectively, and 3) the heigh to diameter ratio of the columns must be less or equal to 20 (Barker, 2018).

$$\mathscr{F}^{ISOBUT}_{isobutane} \geq 0.94\,\mathscr{F}^{FEED}_{isobutane},\quad \mathscr{F}^{PENTANE}_{n-pentane} \geq 0.94\,\mathscr{F}^{FEED}_{n-pentane},\quad \mathscr{F}^{BUTANE}_{n-butane} \geq 0.94\,\mathscr{F}^{FEED}_{n-butane} \tag{4A}$$

$$\chi^{ISOBUT}_{isobutane} \geq 0.98,\quad \chi^{PENTANE}_{n-pentane} \geq 0.99,\ \chi^{BUTANE}_{n-butane} \geq 0.96 \tag{4B}$$

$$h_c/d_c \leq 20,\quad \forall c\ \in \{C1, C2\} \tag{4C}$$

This case study was used to test the proposed hybrid algorithm and to compare the performance of the proposed algorithm and the traditional DETL method. Given that both methods rely on random exploration, different solutions may be obtained from different runs of the same optimization problem. Thus, this case study was solved four times under the same conditions using both the proposed method and DETL. A relative optimality gap of ε=0.05 was established for the hybrid algorithm; also, δ was set to 0.05, and a CPU time limit of 5000 min was considered for both strategies. More details about the implementation and execution of the hybrid algorithm are available in sections S1 and S3 of the supplementary material.

Despite their inherent random variability, the four trials showed a similar qualitative performance, i.e., the parallel-hybrid method showed faster convergence and better objective function values than the traditional DETL. The evolution of the best objective function over time for these experiments is shown in Fig. 5A–D. Dotted lines represent the traditional DETL method, while solid and dashed lines represent the proposed DETL-based stochastic method and the proposed DSDA-VB deterministic method, respectively. The best solution identified by the hybrid algorithm is a local optimum and was obtained from the first trial (Fig. 5A). This solution corresponds to a design with the DOF reported in

Table 3
First iteration of the hybrid algorithm.

Run	Time [min]	Iteration (k)	f_B^k (DSDA-VB)	f_S^k (DETL)
Fig. 5A	209	1	275351	645529
Fig. 5B	141	1	300204.4	537369.4
Fig. 5C	150	1	276903.6	551489.4
Fig. 5D	252	1	297621.2	475507.7

the last column of Table 1, and it is compared with the best (local) solutions from the other runs (Fig. 5B–D) depicted in Table 2. The results from the different runs in Table 2 vary slightly, showing average percentage differences between the best-known solution (Fig. 5A) and the remaining solutions (Fig. 5B–D) below 1% for all continuous DOF, except for the vapor flow from C1 to C2 and the diameter of C2 which show average percent differences of 1.3% and 2.7%, respectively. The discrete DOF also exhibits similar values between runs, differing only by increments or decrements of 1 stage with respect to the best-known solution. As a result, the resulting objective function values in Table 2 only differ by 0.05% on average compared to the best objective (271229 \$/year).

Based on Fig. 5, the proposed hybrid algorithm has a higher convergence speed than the traditional DETL for this case study. The hybrid strategy closed the gap after 2435 minutes on average, while the traditional DETL could not find the best solution reported by the hybrid algorithm after 5000 minutes in any of the trials. For instance, the DETL objective found after 5000 minutes in Fig. 5A (275018 \$/year) is still 1.4% larger than the solution obtained at 1825 minutes by the hybrid algorithm (271229 \$/year). Also, when the hybrid algorithm stops in Fig. 5A–C, the objective function reported by the traditional DETL is on average 26% worse than the hybrid's strategy objective. These improvements in convergence speed are mainly attributed to the "jump" that the hybrid algorithm performs once the first execution of the DSDA-VB algorithm ($k = 1$) is performed and the first group of improved variable bounds is returned to the SM. This is illustrated in Table 3, where after 209, 141, 150, and 252 minutes of execution of the hybrid algorithm, the DSDA-VB improves the DETL objective by 134%, 79%, 99% and 59% for the trials shown in Fig. 5A, B, C, and D, respectively. Note that while the hybrid algorithm requires $\sim 10^2$ minutes to obtain the objective functions in column f_B^k shown in Table 3, the traditional DETL needs $\sim 10^3$ minutes to obtain solutions of this quality. Although the first iteration of the hybrid algorithm leads to a solution for which DOF are locally optimal, the objective function is further improved in subsequent iterations of the hybrid algorithm due to the continuous exploration of the DETL algorithm within the continuously refined bounds obtained from the DSDA-VB strategy. Additional information about the iterations of the hybrid algorithm for optimization runs in Fig. 5A–D is available in section S5 of the supplementary material.

The proposed hybrid algorithm is able to continuously identify new locally optimal solutions because the DETL explores solution candidates within N_∞-neighborhoods of discrete decisions of size $|N_\infty| = 3^4 = 81$, while DSDA-VB only guarantees local optimality within N_2-neighborhoods of size $|N_2| = 2(4) + 1 = 9$. N_2-neighbors consider the effect of varying one discrete decision at a time, while N_∞-neighbors consider interactions between variables, e.g., modifying both the number of stages and feed location of a column simultaneously. As a result, DETL can guide the local search towards solution candidates that are near the current DSDA-VB solution but cannot be identified by this deterministic algorithm. For instance, Fig. 6A illustrates the convergence plot of a discrete DOF: the number of stages N of column C1. The variables and bounds plotted in Fig. 6 correspond to the best objective found by each algorithm at each iteration. The bounds for the number of stages in Fig. 6A start with the user-defined bounds (i.e., $N^{L,0} = 5$ and $N^{U,0} = 70$), but these are updated to $N^{L,M} = 53$ and $N^{U,M} = 54$ after the first local solution with $N = 54$ is identified by DSDA-VB at 209 minutes (see the first row in Table 3). After 529 minutes of execution of the hybrid

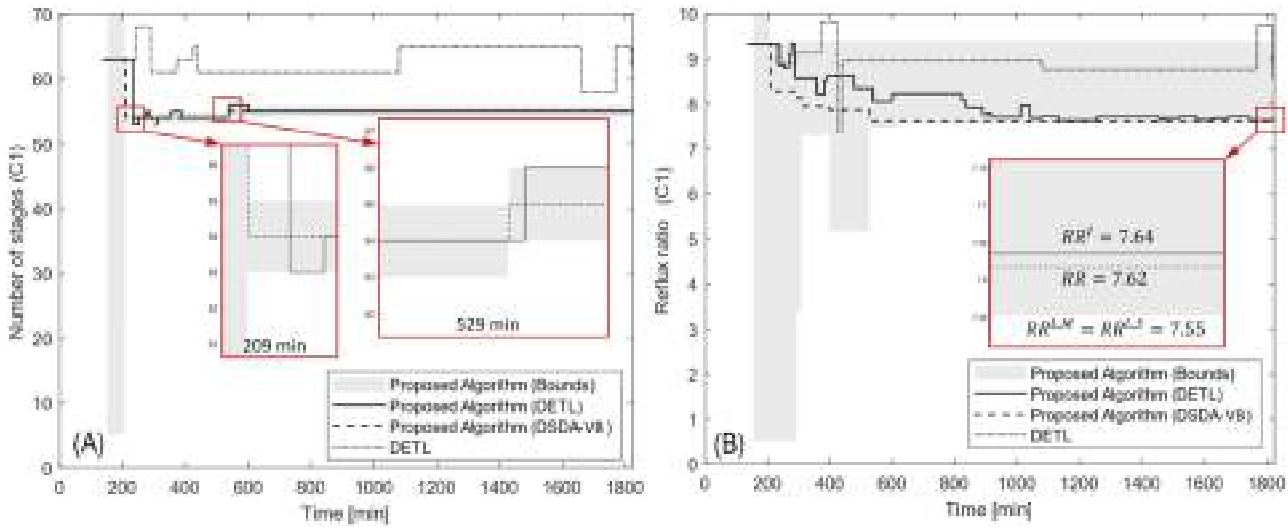

Fig. 6. Illustrative convergence plots for the number of stages of C1 (A) and the reflux ratio of C1 (B). These plots correspond to the optimization run in Fig. 5A.

algorithm, the bounds over N are recalculated again; this time around $N = 55$, with $N^{L,M} = 55$ and $N^{U,M} = 56$. This change coincides with increments of the number of stages of C2 and the location of the liquid interconnection stage, i.e., multiple discrete decisions changed at 529 minutes, which is something that could not be achieved by the DSDA-VB alone, which explores the search space using N_2-neighborhoods ($|N_2| = 11$). This indicates that this new solution identified at 529 minutes was attained thanks to the exploration performed by DETL in the larger N_∞-neighborhood ($|N_\infty| = 243$), which allowed varying multiple discrete DOF simultaneously. Note that although DSDA-VB only performs local optimality verifications within N_2-neighborhoods, it is allowed to explore the user-defined search region of discrete decisions ($\sim 10^7$ discrete solution candidates), while DETL seeks improved solutions within an N_∞-neighborhood of the current local DSDA-VB solution (81 discrete solution candidates).

In contrast to the discrete decisions, updating the bounds of the continuous variables does not necessarily limit the search of continuous variables within a vicinity of the current best solution. Fig. 6B shows the behavior of the reflux ratio of unit C1. The upper bound of the reflux ratio begins at $RR^{U,0}$, and then it its updated and kept at $RR^{U,S}$, which is the maximum feasible reflux ratio stored in F_S. Contrarily, the VB strategy varied the lower bound of the reflux ratio ($RR^{L,M}$) during the first 32 iterations of the algorithm (i.e., the first 615 minutes) and then $RR^{L,M}$ remained constant at $RR^{L,S} = 7.55$. This avoided binding RR at its lower bound in the locally optimal solutions reported by DSDA, i.e., after 615 minutes, $RR^{L,S} = 7.55$ guarantees that the best reflux ratio (7.62) is locally optimal, given that the optimized reflux ratio is no longer reaching its lower bound. Overall, updating bounds as illustrated in Fig. 6 allows to guide the hybrid algorithm towards locally optimal solutions faster than the traditional DETL, which requires more time to converge for this case study.

Additional computational experiments showcasing the performance of the proposed parallel-hybrid algorithm against a nested-hybrid algorithm are presented in Fig. S5 of the supplementary material. The nested-hybrid strategy optimizes discrete DOF using the DETL, which iteratively fixes those discrete DOF and optimizes continuous DOF through the NLP solver available in Aspen. As shown in Fig. S5 of the supplementary material, the nested-hybrid approach was 464 minutes on average slower than the parallel-hybrid method at finding the best solution reported by each algorithm. Also, the best parallel-hybrid solution (271229 \$/year) is 0.2% better than the best nested-hybrid solution (271669 \$/year) in terms of its objective function value. However, the nested-hybrid exhibits two key limitations: this method cannot refine search bounds for continuous DOF and cannot guarantee local optimality in the solutions. In contrast, the proposed parallel-hybrid approach addresses these issues offering both optimality guarantees and convergence improvements over the nested-hybrid methodology. More details about this comparison are provided in section S4 of the supplementary material.

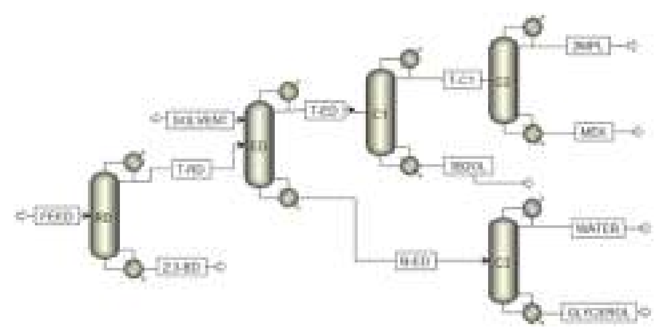

Fig. 7. Sequence of distillation processes designed for the production of MEK from 2,3-BD, and the separation of by-products.

4.2. Case study 2: A sequence of reactive, extractive and conventional distillation columns

This case study aims to demonstrate the applicability of the proposed algorithm to a more challenging process flowsheet optimization problem that involves a large number of DOF and constraints, i.e., 13 discrete DOF and 15 continuous DOF. This case study was adapted from the work by Torres-Vinces et al. (2020). The sequence of distillation systems presented in Fig. 7 consists of a reactive distillation (RD) column that has an input of pure 2,3-butanediol (2,3-BD) equal to 1000 kg/h. The pure 2,3-BD follows a series of dehydration reactions in the RD column, which simultaneously generates Methyl Ethyl Ketone (MEK) as the main product in the top stream (T-RD) and separates the unreacted 2,3-BD as a bottom's product. The dehydration reactions in the RD system also generate by-products in the T-RD stream such as 2-methylpropanal (2MPL), 3-buten-2-ol (3B2OL), water, and some 1,3-butadiene (1,3-BD). These by-products are separated from the MEK in subsequent separation stages. The first stage is an extractive distillation (ED) column that uses glycerol as solvent to extract water from stream T-RD. The solvent mixed with water is removed from the bottom of the ED column (steam B-ED), while the remaining components in the mixture leave the system from the top (steam T-ED). The latter stream (T-ED) is further

Table 4
Variables and their bounds for the second case study, and best solution found.

Continuous DOF ($\boldsymbol{x}$)				
Variable	Units	Lower bounds ($\boldsymbol{x}^{L,0}$)	Upper bounds ($\boldsymbol{x}^{U,0}$)	Best solution found
Reflux ratio (RD)	-	1	3.2	3.016
Reflux ratio (ED)	-	0.1	0.5	0.489
Reflux ratio (C1)	-	1	14.7	1.306
Reflux ratio (C2)	-	1	5.8	5.607
Reflux ratio (C3)	-	1	1.6	1.000
Distillate flow (from RD to ED)	kg/h	994.01	1004.00	999.258
Distillate flow (from ED to C1)	kg/h	792.57	800.53	797.241
Bottoms flow (C1)	kg/h	89.55	90.45	90.344
Bottoms flow (C2)	kg/h	434.82	439.19	437.230
Bottoms flow (C3)	kg/h	397.01	401.00	399.989
Diameter (RD)	m	0.5	3.5	2.408
Diameter (ED)	m	0.5	3.5	1.372
Diameter (C1)	m	0.5	3.5	2.286
Diameter (C2)	m	0.5	3.5	1.372
Diameter (C3)	m	0.5	3.5	0.500
Discrete DOF ($\boldsymbol{y}$)				
Variable	Units	Lower bounds ($\boldsymbol{y}^{L,0}$)	Upper bounds ($\boldsymbol{y}^{U,0}$)	Best solution found
Number of stages (RD)	-	76	86	79
Number of stages (ED)	-	45	55	45
Number of stages (C1)	-	75	85	75
Number of stages (C2)	-	45	55	45
Number of stages (C3)	-	13	23	13
Feed stage (RD)	-	40	50	49
T-RD feed stage (ED)	-	35	44	38
Solvent feed stage (ED)	-	2	10	2
Feed stage (C1)	-	45	55	53
Feed stage (C2)	-	20	30	20
Feed stage (C3)	-	9	12	10
Location of first reactive stage (RD)	-	2	7	5
Location of last reactive stage (RD)	-	75	85	78

purified in distillation columns C1 and C2, where column C1 generates a bottom product rich in 3B2OL, and C2 generates a top product with 2MPL and some 1,3-BD, and a bottom product with purified MEK. Moreover, stream B-ED is separated in distillation column C3, which recovers the glycerol (bottom product) and produces water as the top product. The non-random two-liquid (NRTL) thermodynamic model was used for the liquid phase, while the Redlich-Kwong equation of state was used for the vapor phase (Torres-Vinces et al., 2020).

The DOF of this problem ($\boldsymbol{x}$ and $\boldsymbol{y}$), and their corresponding bounds ($\boldsymbol{x}^{L,0}, \boldsymbol{x}^{U,0}, \boldsymbol{y}^{L,0}$ and $\boldsymbol{y}^{U,0}$) are summarized in Table 4. Variables in vector $\boldsymbol{z}$ include the reboiler and condenser duties for all columns, as well as those variables required to specify the problem's constraints, i.e., the mass fraction (ω_c^s) and mass flow ($\mathscr{M}_c^s$) of the desired component c for each output stream s. The user-defined constraints that are considered for this problem ($\boldsymbol{g}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$) include those proposed by Torres-Vinces et al. (2020): a purity constraint of at least 99.5% (wt) is enforced over the MEK output (Eq. (5A)); purity constraints of at least 99% (wt) are considered for the 2,3-BD, 3B2OL, and 2MPL streams (Eq. (5B)); purity constraints of at least 98% and 99.99% (wt) are imposed over the water and glycerol output streams, respectively (Eq. (5C)), and a recovery of at least 98% (wt) is enforced for all products (Eq. (5D)). Also, the heigh to diameter ratio of the columns is constrained to be less or equal to 20, as stated in Eq. (5E) (Barker, 2018). In addition, hydrodynamic constraints that aim to maintain proper column hydrodynamic operation (e.g., enough contact between vapor and liquid phases) such as entrainment and downcomer flooding are imposed for all columns. These constraints are handled internally by Aspen Plus, i.e., these constraints are included

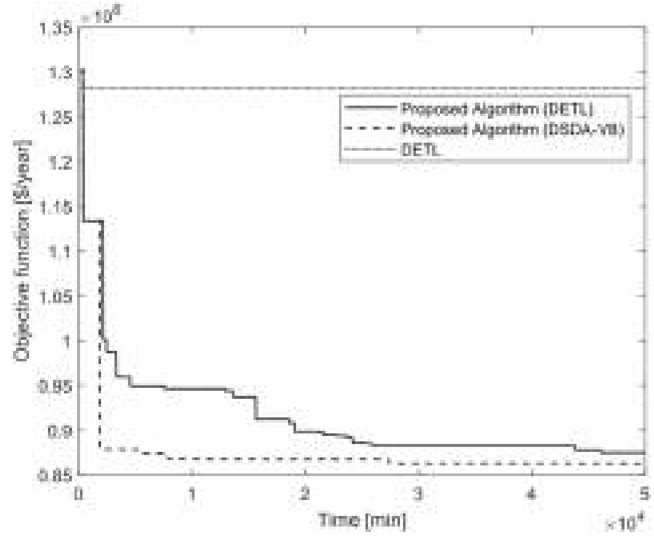

Fig. 8. Convergence plot for the MEK production process.

in the black-box constraints Ω that appear in problem (1). Note that this case study is more challenging than the problems usually considered to test the hybrid algorithms for flowsheet optimization, which usually consider flowsheets involving one or two processing units as in case study 1, e.g., see (Gómez et al., 2006; Skiborowski et al., 2015; Urselmann et al., 2016).

$$\omega_{MEK}^{MEK} \geq 0.995 \tag{5A}$$

$$\omega_c^s \geq 0.99, \ \forall (c,s) \in \{(2,3-BD,2,3-BD),(3B2OL,3B2OL),(2MPL,2MPL)\} \tag{5B}$$

$$\omega_{Water}^{WATER} \geq 0.98, \ \omega_{Glycerol}^{GLYCEROL} \geq 0.9999 \tag{5C}$$

$$\mathscr{M}_c^s \geq 0.98 \mathscr{M}_c^{T-RD}, \ \forall (c,s) \in \left\{ \begin{array}{c} (MEK,MEK),(2MPL,2MPL),(3B2OL,3B2OL), \\ (Water,WATER),(Glycerol,GLYCEROL) \end{array} \right\} \tag{5D}$$

$$h_c/d_c \leq 20, \ \forall c \in \{\text{RD},\text{ED},\text{C1},\text{C2},\text{C3}\} \tag{5E}$$

Due to the high number of variables and constraints involved, this problem requires much more time to converge than the previous case study. In the present case study, the proposed hybrid algorithm and the DETL strategy were left to run until a time limit of 5*10^4 minutes was reached with δ=0.05. The convergence plot of both algorithms is shown in Fig. 8. At 5*10^4 minutes, the relative gap between the traditional DETL and hybrid algorithms objectives is approximately 48%, while the gap between the DETL and DSDA-VB algorithms within the proposed hybrid approach is around 1%. The best solution reported by the hybrid algorithm has an objective function of 863423 $/year the DOF obtained for this solution is depicted in the last column of Table 4. In contrast, the traditional DETL found a much more expensive design with an objective function of 128280 $/year.

Due to the problem size and the search bounds selected, the performance of the traditional DETL deteriorated significantly for this case study (see the dotted line in Fig. 8). Although DETL was able to identify feasible solutions during the early stages of its execution, it was unable to improve the objective function within the specified time limit. This performance may be attributed to the low ratio of individuals evaluations that converged to feasible solutions, which in this case resulted in stagnation of the traditional DETL algorithm, i.e., at $5 * 10^4$ minutes the DETL population still remains diverse and does not seem to be progressing towards a particular solution. In contrast, the hybrid algorithm managed to handle this convergence problem by automatically

Table 5
Convergence of objective functions f_B^k and f_S^k, number of stages (RD) and reflux ratio (RD) for iterations in which variable bounds are updated.

			Number of stages (RD)			Reflux ratio (RD)			
Time [min]	k	f_B^k	$N^{L,M}$	$N^{U,M}$	N^k	$RR^{L,M}$	$RR^{U,M}$	RR^k	$N^{L,M}$
1833	1	878540.4	1134177	80	82	81	2.96	3.04	3.01
3692	49	878356.7	987397	80	82	81	2.93	3.03	2.97
5473	56	874157.8	959781.5	80	82	81	2.47	3.04	2.93
7826	161	868193.2	946637	79	81	80	2.89	3.04	2.95
10452	168	867731.8	946637	79	81	80	2.90	3.03	2.97
27378	272	863423.2	882641.1	78	80	79	2.89	3.04	3.02
50000	442	863423.2	875273.2	78	80	79	2.89	3.04	3.02

identifying variable bounds that, for this case study, allowed the DETL algorithm to improve its objective function and satisfy local optimality requirements for discrete and continuous variables (see solid line in Fig. 8).

Table 5 summarizes some of the iterations performed by the hybrid algorithm for this case study. These iterations correspond to those where variable bounds are updated. Each iteration shows the locally optimal objective function obtained by DSDA-VB (f_B^k), the best objective function identified by DETL (f_S^k), and the bounds and optimal solution for the number of stages (N) and reflux ratio (RR) of the RD. According to Table 5, the frequency at which variable bounds are updated increases as iterations progress, e.g., the first group of updated bounds is obtained after 1833 minutes at $k=1$, while the last group of updated bounds is obtained after 16926 minutes at $k=272$. Note that the hybrid algorithm spent 45% of the total execution time between iterations 272 and 442 (i. e., 22622 minutes). During this time interval, DSDA-VB could not identify better solutions, thus bounds were not updated, i.e., between iterations 272 and 442 DETL is aiming to improve the objective function without recdback from DSDA-VB. This suggests that, although the DETL performance improves with respect to its naïve execution, DETL is still the time-limiting factor when integrated with DSDA-VB. Concerning the convergence of variables and their bounds, the behavior obtained is similar to that obtained for the first case study. For example, the lower bounds are updated from the user-defined values (76 and 1) to 80 and 2.96 at $k=1$, for N and RR, respectively. After the first iteration, variable bounds and values do not change significantly, but they still have an impact on the objective function, e.g., f_B^k and f_S^k improve by 1.8% and 30% between iteration 1 and 442. Detailed data showing the evolution of the objective functions f_B^k and f_S^k is provided in section S5 of the supplementary material.

5. Conclusions

A hybrid stochastic-deterministic algorithm for optimal flowsheet design involving ordered discrete decisions, e.g., the number of stages/reactors in a superstructure, is presented in this work. This hybrid algorithm relies on the parallel execution of a deterministic algorithm that incorporates a Discrete-Steepest Descent Algorithm with Variable Bounding (DSDA-VB) and a stochastic algorithm that implements a Stochastic Method (SM), e.g., differential Evolution with Tabu List (DETL). The key feature of the proposed hybrid method is the parallel interaction between DSDA-VB and the SM, through continuous exchange of information. The SM is constantly generating new solution candidates that are optimized locally by the deterministic DSDA-VB method. In turn, the DSDA-VB generates new bounds for discrete and continuous variables that guide the SM toward a local optimum, while still allowing exploration by the SM within a neighborhood of the local optimum. The above procedure is executed iteratively, which allows the identification of different local optima during the search. Contrary to other hybrid methods, the proposed approach combines the advantages of a pure SM with the streamlined efficiency of the DSDA-VB, resulting in a practical and computationally efficient algorithm. Two case studies involving intensified distillation units were considered to test the performance of this hybrid algorithm. The results show that the proposed hybrid algorithm quickly identifies a good quality local solution in the early stages of its execution. After that, the hybrid method relies on the exploratory capabilities of the SM method to identify new feasible solutions that serve as starting points to find better local optima. Overall, the parallel hybridization of the DSDA-VB and an SM such as DETL results in an enhanced convergence performance of both algorithms toward improved solutions, with respect to the naïve execution of the SM. In contrast to traditional hybrid approaches (e.g., memetic algorithms), the discrete and continuous variables obtained from the present approach satisfy local optimality. Future work is needed to make the proposed strategy accessible to a broader audience, through the development of a toolbox that readily incorporates the different steps needed to execute the proposed hybrid algorithm. In terms of applications, further research is also needed to improve the algorithm's performance in handling multi-objective, and multimodal objective functions. The algorithm must also be expanded to consider decisions that typically appear in process synthesis problems, such as deciding between two separation or reaction technologies, e.g., molecular sieve membrane separation against distillation. Also, the proposed approach should be tested in the future with integrated design and control applications (Rafiei and Ricardez-Sandoval, 2020).

Funding

This work was supported by The Natural Sciences and Engineering Research Council of Canada (NSERC).

CRediT authorship contribution statement

David A. Liñán: Conceptualization, Methodology, Software, Validation, Visualization, Investigation, Writing – original draft. **Gabriel Contreras-Zarazúa:** Methodology, Software, Validation, Investigation, Formal analysis, Writing – review & editing. **Eduardo Sánchez-Ramírez:** Methodology, Software, Validation, Investigation, Formal analysis, Writing – review & editing. **Juan Gabriel Segovia-Hernández:** Methodology, Investigation, Formal analysis, Writing – review & editing. **Luis A. Ricardez-Sandoval:** Funding acquisition, Supervision, Project administration, Formal analysis, Writing – review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

The financial support provided by the Natural Sciences and Engineering Research Council of Canada (NSERC) is gratefully acknowledged.

Supplementary materials

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.compchemeng.2023.108501.

References

Aliaga-Vicente, A., Caballero, J.A., Fernández-Torres, M.J., 2017. Synthesis and optimization of membrane cascade for gas separation via mixed-integer nonlinear programming. AIChE J. 63, 1989–2006. https://doi.org/10.1002/aic.15631.

Barker, G., 2018. The engineer's guide to plant layout and piping design for the oil and gas industries. Chapter 11 - Towers 285–308. https://doi.org/10.1016/B978-0-12-814653-8.00011-4. Barker, G. (Ed.).

Bernal, D.E., Carrillo-Diaz, C., Gómez, J.M., Ricardez-Sandoval, L.A., 2018. Simultaneous design and control of catalytic distillation columns using comprehensive rigorous dynamic models. Ind. Eng. Chem. Res. 57, 2587–2608. https://doi.org/10.1021/acs.iecr.7b04205.

Carlson, E.C., 1996. Don't gamble with physical properties for simulations. Chem. Eng. Prog. 92, 35–46.

Chen, X., Li, M., 2021. M♮-convexity and its applications in operations. Oper. Res. 69, 1396–1408. https://doi.org/10.1287/opre.2020.2070.

Chia, D.N., Duanmu, F., Sorensen, E., 2021. Optimal Design of Distillation Columns Using a Combined Optimisation Approach. Elsevier, pp. 153–158. https://doi.org/10.1016/B978-0-323-88506-5.50025-5.

Ciric, A.R., Gu, D., 1994. Synthesis of nonequilibrium reactive distillation processes by MINLP optimization. AIChE J. 40, 1479–1487. https://doi.org/10.1002/aic.690400907.

Contreras-Zarazúa, G., Jasso-Villegas, M.E., Ramírez-Márquez, C., Sánchez-Ramírez, E., Vázquez-Castillo, J.A., Segovia-Hernandez, J.G., 2021. Design and intensification of distillation processes for furfural and co-products purification considering economic, environmental, safety and control issues. Chem. Eng. Process. Process Intensif. 159, 108218 https://doi.org/10.1016/j.cep.2020.108218.

Contreras-Zarazúa, G., Martin-Martin, M., Sánchez-Ramirez, E., Segovia-Hernández, J. G., 2022. Furfural production from agricultural residues using different intensified separation and pretreatment alternatives. Economic and environmental assessment. Chem. Eng. Process. Process Intensif. 171, 108569 https://doi.org/10.1016/j.cep.2021.108569.

Contreras-Zarazúa, G., Sánchez-Ramírez, E., Vázquez-Castillo, J.A., Ponce-Ortega, J.M., Errico, M., Kiss, A.A., Segovia-Hernández, J.G., 2019. Inherently safer design and optimization of intensified separation processes for furfural production. Ind. Eng. Chem. Res. 58, 6105–6120. https://doi.org/10.1021/acs.iecr.8b03646.

Costa, A.L.H., Bagajewicz, M.J., 2019. 110th anniversary: on the departure from heuristics and simplified models toward globally optimal design of process equipment. Ind. Eng. Chem. Res. 58, 18684–18702. https://doi.org/10.1021/acs.iecr.9b02611.

Cremaschi, S., 2015. A perspective on process synthesis: challenges and prospects. Comput. Chem. Eng. 81, 130–137. https://doi.org/10.1016/j.compchemeng.2015.05.007. Special Issue: Selected papers from the 8th International Symposium on the Foundations of Computer-Aided Process Design (FOCAPD 2014), July 13-17, 2014, Cle Elum, Washington, USA.

Deb, K., 2009. Multi-Objective Optimization Using Evolutionary Algorithms. Wiley, Chichester Weinheim, 1st edition. ed.

Douglas, J., 1988. Conceptual Design of Chemical Processes. McGraw-Hill Science/Engineering/Math, New York, 1st edition. ed.

Franke, M.B., 2017. Design of dividing-wall columns by mixed-integer nonlinear programming optimization. Chem. Ing. Tech. 89, 582–597. https://doi.org/10.1002/cite.201700005.

Gómez, J.M., Reneaume, J.M., Roques, M., Meyer, M., Meyer, X., 2006. A mixed integer nonlinear programming formulation for optimal design of a catalytic distillation column based on a generic nonequilibrium model. Ind. Eng. Chem. Res. 45, 1373–1388. https://doi.org/10.1021/ie0504506.

González-Navarrete, C., Sánchez-Ramírez, E., Ramírez-Márquez, C., Hernández, S., Cossío-Vargas, E., Segovia-Hernández, J.G., 2022. Innovative reactive distillation process for the sustainable purification of lactic acid. Ind. Eng. Chem. Res. 61, 621–637. https://doi.org/10.1021/acs.iecr.1c04050.

Gooty, R.T., Agrawal, R., Tawarmalani, M., 2022. Advances in MINLP to identify energy-efficient distillation configurations. Oper. Res. https://doi.org/10.1287/opre.2022.2340.

Göttl, Q., Grimm, D.G., Burger, J., 2022. Automated synthesis of steady-state continuous processes using reinforcement learning. Front. Chem. Sci. Eng. 16, 288–302. https://doi.org/10.1007/s11705-021-2055-9.

Hernández-Pérez, L.G., Ramírez-Márquez, C., Segovia-Hernández, J.G., Ponce-Ortega, J. M., 2020. Simultaneous structural and operating optimization of process flowsheets combining process simulators and metaheuristic techniques: the case of solar-grade silicon process. Comput. Chem. Eng. 140, 106946. https://doi.org/10.1016/j.compchemeng.2020.106946.

Herrera Velázquez, J.J., Zavala Durán, F.M., Chávez Díaz, L.A., Cabrera Ruiz, J., Alcántara Avila, J.R., 2022. Hybrid two-step optimization of internally heat-integrated distillation columns. Journal of the Taiwan Institute of Chemical Engineers 130, 103967. https://doi.org/10.1016/j.jtice.2021.06.061. J. Taiwan Inst. Chem. Eng.

Holtbruegge, J., Kuhlmann, H., Lutze, P., 2015. Process analysis and economic optimization of intensified process alternatives for simultaneous industrial scale production of dimethyl carbonate and propylene glycol. Chem. Eng. Res. Des. 93, 411–431. https://doi.org/10.1016/j.cherd.2014.05.002.

Hong, J., Van Duc Long, N., Harvianto, G.R., Haider, J., Lee, M., 2019. Design and optimization of multi-effect-evaporation-assisted distillation configuration for recovery of 2,3-butanediol from fermentation broth. Chem. Eng. Process. Process Intensif. 136, 107–115. https://doi.org/10.1016/j.cep.2019.01.002.

Javaloyes-Antón, J., Kronqvist, J., Caballero, J.A., 2022. Simulation-based optimization of distillation processes using an extended cutting plane algorithm. Comput. Chem. Eng. 159, 107655. https://doi.org/10.1016/j.compchemeng.2021.107655.

Jia, S., Cao, X., Qian, X., Liu, X., Luo, Y., Yuan, X., 2023. Boundary function method for stage number optimization for multi-stage distillation process design. Chem. Eng. Sci. 276, 118773 https://doi.org/10.1016/j.ces.2023.118773.

Kronqvist, J., Bernal, D.E., Lundell, A., Grossmann, I.E., 2019. A review and comparison of solvers for convex MINLP. Optim. Eng. 20, 397–455. https://doi.org/10.1007/s11081-018-9411-8.

Kruber, K.F., Grueters, T., Skiborowski, M., 2021. Advanced hybrid optimization methods for the design of complex separation processes. Comput. Chem. Eng. 147, 107257 https://doi.org/10.1016/j.compchemeng.2021.107257.

Liberti, L., Kucherenko, S., 2005. Comparison of deterministic and stochastic approaches to global optimization. Int. Trans. Oper. Res. 12, 263–285. https://doi.org/10.1111/j.1475-3995.2005.00503.x.

Liñán, D.A., Bernal, D.E., Gómez, J.M., Ricardez-Sandoval, L.A., 2021. Optimal synthesis and design of catalytic distillation columns: a rate-based modeling approach. Chem. Eng. Sci. 231, 116294 https://doi.org/10.1016/j.ces.2020.116294.

Liñán, D.A., Bernal, D.E., Ricardez-Sandoval, L.A., Gómez, J.M., 2020. Optimal design of superstructures for placing units and streams with multiple and ordered available locations. Part II: rigorous design of catalytic distillation columns. Comput. Chem. Eng. 139, 106845 https://doi.org/10.1016/j.compchemeng.2020.106845.

Liñán, D.A., Ricardez-Sandoval, L.A., 2023. A Benders decomposition framework for the optimization of disjunctive superstructures with ordered discrete decisions. AIChE J. 69, e18008. https://doi.org/10.1002/aic.18008.

Ma, Y., Li, J., 2022. Homotopy continuation enhanced branch and bound algorithms for strongly nonconvex mixed-integer nonlinear optimization. AIChE J. 68, e17629. https://doi.org/10.1002/aic.17629.

Ma, Y., Yang, Z., El-Khoruy, A., Zhang, N., Li, J., Zhang, B., Sun, L., 2021. Simultaneous synthesis and design of reaction–separation–recycle processes using rigorous models. Ind. Eng. Chem. Res. 60, 7275–7290. https://doi.org/10.1021/acs.iecr.1c00250.

Munawar, S.A., Gudi, R.D., 2005. A nonlinear transformation based hybrid evolutionary method for MINLP solution. Chem. Eng. Res. Des. 83, 1218–1236. https://doi.org/10.1205/cherd.04286.

Murota, K., 2003. Discrete convex analysis, discrete mathematics and applications. Soc. Ind. Appl. Math. https://doi.org/10.1137/1.9780898718508.

Murota, K., Tamura, A., 2023. Recent progress on integrally convex functions. Jpn. J. Indust. Appl. Math. https://doi.org/10.1007/s13160-023-00589-4.

Peccini, A., Jesus, L.F.S., Secchi, A.R., Bagajewicz, M.J., Costa, A.L.H., 2023. Globally optimal distillation column design using set trimming and enumeration techniques. Comput. Chem. Eng. 174, 108254 https://doi.org/10.1016/j.compchemeng.2023.108254.

Rafiei, M., Ricardez-Sandoval, L.A., 2020. New frontiers, challenges, and opportunities in integration of design and control for enterprise-wide sustainability. Comput. Chem. Eng. 132, 106610 https://doi.org/10.1016/j.compchemeng.2019.106610.

Ramapriya, G.M., Selvarajah, A., Jimenez Cucaita, L.E., Huff, J., Tawarmalani, M., Agrawal, R., 2018. Short-cut methods versus rigorous methods for performance-evaluation of distillation configurations. Ind. Eng. Chem. Res. 57, 7726–7731. https://doi.org/10.1021/acs.iecr.7b05214.

Rangaiah, G.P., 2008. Multi-Objective Optimization: Techniques and Applications in Chemical Engineering, Har/Cdr edition. World Scientific Publishing Company, Hackensack, N.J ed.

Rangaiah G.P., Feng Z., Hoadley A.F., 2020. Multi-Objective Optimization Applications in Chemical Process Engineering: Tutorial and Review. Processes 8, 508. doi:10.3390/pr8050508.

Romero-García, A.G., Mora-Morales, C., Chargoy-Amador, J.P., Ramírez-Corona, N., Sánchez-Ramírez, E., Segovia-Hernández, J.G., 2022. Implementing CO2 capture process in power plants: Optimization procedure and environmental impact. Chem. Eng. Res. Des. 180, 232–242. https://doi.org/10.1016/j.cherd.2022.02.023.

Ryu, J., Kong, L., Pastore de Lima, A.E., Maravelias, C.T., 2020. A generalized superstructure-based framework for process synthesis. Comput. Chem. Eng. 133, 106653 https://doi.org/10.1016/j.compchemeng.2019.106653.

Sánchez-Ramírez, E., Hernández, S., Romero-García, A.G., Alcocer-García, H., Segovia – Hernández, J.G., 2022. Synthesis and optimization of sustainable processes based on liquid-liquid extraction to purify methyl ethyl ketone. Chem. Eng. Process. Process Intensif. 171, 108522 https://doi.org/10.1016/j.cep.2021.108522.

Segovia-Hernández, J.G., Hernández, S., Bonilla Petriciolet, A., 2015. Reactive distillation: a review of optimal design using deterministic and stochastic techniques. Chem. Eng. Process. 97, 134–143. https://doi.org/10.1016/j.cep.2015.09.004.

Sharma S., Rangaiah G.P., Cheah K.S., 2012. Multi-objective optimization using MS Excel with an application to design of a falling-film evaporator system. Food Bioprod. Process. 90, 123–134. doi:10.1016/j.fbp.2011.02.005.

Skiborowski, M., Rautenberg, M., Marquardt, W., 2015. A Hybrid Evolutionary–Deterministic Optimization Approach for Conceptual Design. Ind. Eng. Chem. Res. 54, 10054–10072. https://doi.org/10.1021/acs.iecr.5b01995.
Spall, J.C., 2003. Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control, 1st edition. Wiley-Interscience, Hoboken, N.J. ed.
Srinivas, M., Rangaiah, G.P., 2016. Differential evolution with tabu list for global optimization: evaluation of two versions on benchmark and phase stability problems, in: differential evolution in chemical engineering. Advances in Process Systems Engineering. World Scientific, pp. 91–127. https://doi.org/10.1142/9789813207523_0004.
Srinivas, M., Rangaiah, G.P., 2007. Differential evolution with tabu list for solving nonlinear and mixed-integer nonlinear programming problems. Ind. Eng. Chem. Res. 46, 7126–7135. https://doi.org/10.1021/ie070007q.
Srinivas, M., Rangaiah, G.P., 2006. An integrated stochastic method for global optimization of continuous functions. Marquardt, W., Pantelides, C. (Eds.). Computer Aided Chemical Engineering, 16th European Symposium on Computer Aided Process Engineering and 9th International Symposium on Process Systems Engineering. Elsevier, pp. 439–444. https://doi.org/10.1016/S1570-7946(06)80085-4.
Staudt, P.B., Soares, R., 2009. Reliability vs. efficiency when solving multiphase equilibrium problems with hybrid optimization codes. Computer Aided Chemical Engineering. In: Proceedings of the 10th International Symposium on Process Systems Engineering: Part A. Elsevier, pp. 585–590. https://doi.org/10.1016/S1570-7946(09)70318-9.
Storn, R., 1996. On the usage of differential evolution for function optimization. In: Proceedings of the North American Fuzzy Information Processing. Presented at the Proceedings of North American Fuzzy Information Processing, pp. 519–523. https://doi.org/10.1109/NAFIPS.1996.534789.
Torres-Vinces, L., Contreras-Zarazua, G., Huerta-Rosas, B., Sánchez-Ramírez, E., Segovia-Hernández, J.G., 2020. Methyl ethyl ketone production through an intensified process. Chem. Eng. Technol. 43, 1433–1441. https://doi.org/10.1002/ceat.201900664.
Tula, A.K., Babi, D.K., Bottlaender, J., Eden, M.R., Gani, R., 2017. A computer-aided software-tool for sustainable process synthesis-intensification. Comput. Chem. Eng. 105, 74–95. https://doi.org/10.1016/j.compchemeng.2017.01.001. Process Intensification.
Urselmann, M., Barkmann, S., Sand, G., Engell, S., 2011. A memetic algorithm for global optimization in chemical process synthesis problems. IEEE Trans. Evol. Comput. 15, 659–683. https://doi.org/10.1109/TEVC.2011.2150753.
Urselmann, M., Barkmann, S., Sand, G., Engell, S., 2011. Optimization-based design of reactive distillation columns using a memetic algorithm. Comput. Chem. Eng. 35, 787–805. https://doi.org/10.1016/j.compchemeng.2011.01.038. Selected Papers from ESCAPE-20 (European Symposium of Computer Aided Process Engineering - 20), 6-9 June 2010, Ischia, Italy.
Urselmann, M., Janus, T., Foussette, C., Tlatlik, S., Gottschalk, A., Emmerich, M.T.M., Bäck, T., Engell, S., 2016. Derivative-free chemical process synthesis by memetic algorithms coupled to aspen plus process models. Kravanja, Z., Bogataj, M. (Eds.). Comput. Aided Chem. Eng. 187–192. https://doi.org/10.1016/B978-0-444-63428-3.50036-9, 26 European Symposium on Computer Aided Process Engineering.
Wang, Z., Parhi, S.S., Rangaiah, G.P., Jana, A.K., 2020. Analysis of weighting and selection methods for pareto-optimal solutions of multiobjective optimization in chemical engineering applications. Ind. Eng. Chem. Res. 59, 14850–14867. https://doi.org/10.1021/acs.iecr.0c00969.
Wang, Z., Rangaiah, G.P., 2017. Application and analysis of methods for selecting an optimal solution from the pareto-optimal front obtained by multiobjective optimization. Ind. Eng. Chem. Res. 56, 560–574. https://doi.org/10.1021/acs.iecr.6b03453.
Zhang, H., Zheng, Z., Lavaei, J., 2022. Gradient-based algorithms for convex discrete optimization via simulation. Oper. Res. https://doi.org/10.1287/opre.2022.2295.
Zhang, X., Song, Z., Zhou, T., 2018. Rigorous design of reaction-separation processes using disjunctive programming models. Comput. Chem. Eng. 111, 16–26. https://doi.org/10.1016/j.compchemeng.2017.12.013.
Zhou, T., Zhou, Y., Sundmacher, K., 2017. A hybrid stochastic–deterministic optimization approach for integrated solvent and process design. Chem. Eng. Sci. 159, 207–216. https://doi.org/10.1016/j.ces.2016.03.011 iCAMD – Integrating Computer-Aided Molecular Design into Product and Process Design.
Zielinski, K., Laur, R., 2008. Stopping criteria for differential evolution in constrained single-objective optimization. Chakraborty, U.K. (Ed.). Advances in Differential Evolution, Studies in Computational Intelligence. Springer, Berlin, Heidelberg, pp. 111–138. https://doi.org/10.1007/978-3-540-68830-3_4.