Systems & Control Transactions

Research Article
Foundations of Computer Aided Process Design (FOCAPD 2024)
Breckenridge, Colorado, USA. July 14-18, 2024
Peer Reviewed Conference Proceeding

PSE PRESS

Graph-Based Representations and Applications to Process Simulation

Yoel R. Cortés-Peña[a] and Victor M. Zavala[a*]

[a] University of Wisconsin Madison, Department of Chemical and Biomolecular Engineering, Madison, Wisconsin, United States
* Corresponding Author: zavalatejeda@wisc.edu.

ABSTRACT

Rapid and robust convergence of a process flowsheet is critical to enable large-scale simulations that address core scientific questions related to process design, optimization, and sustainability. However, due to the highly coupled and nonlinear nature of chemical processes, efficiently solving a flowsheet remains a challenge. In this work, we show that graph representations of the underlying physical phenomena in unit operations may help identify potential avenues to systematically reformulate the network of equations and enable more robust topology-based convergence of flowsheets. To this end, we developed graph abstractions of the governing equations of vapor-liquid and liquid-liquid equilibrium separation equipment. These graph abstractions consist of a mesh of interconnected variable nodes and equation nodes that are systematically generated through PhenomeNode, a new open-source library in Python developed in this study. We show that partitioning the graph into separate mass, energy, and equilibrium subgraphs can help decouple nonlinearities and guide decomposition algorithms. By employing the graph abstraction on an industrial separation process for separating glacial acetic acid from water, we implemented a new block decomposition scheme in BioSTEAM and demonstrated that this can accelerate convergence over a traditional sequential modular approach.

Keywords: Process simulation, Graph-Theory, Flowsheet Convergence, Distillation, Liquid Extraction

INTRODUCTION

The evaluation of thousands of potential scenarios in a chemical process enables researchers to navigate uncertainties in market conditions and technological performance to create optimized designs, form conclusions on sustainability, and chart development pathways for new processing technologies. Automating the evaluation of such a large number of simulations is limited by computational challenges in rapid and robust flowsheet convergence. While many algorithmic paradigms exist (e.g., classical sequential modular simulation [1], parallel modular simulation [2], equation-based simulation [3], dynamic numerical methods [4], and the design of surrogate models [5,6]) only a limited set of approaches exist that are able to leverage the mathematical topology of the underlying phenomenological equations across the flowsheet [7,8].

In classical sequential-modular simulation, each unit operation is treated as a separate model with only material streams as inputs and outputs. All mass, energy, and thermodynamic equations are formulated and solved independently within each unit operation. While specialized convergence strategies are employed to converge individual units, the convergence of recycle systems can be challenging due to nonlinear coupling between unit operations. Equation-based modeling leverages the sparsity of the full set of equations and employs algebraic differentiation to guide convergence of the entire system. Equation-based modeling may be faster than the sequential modular approach particularly when the initial guess is close to the steady state solution. However, highly coupled and complex networks of equations introduce instabilities and can lead to convergence failure [3].

If a process flowsheet could become aware of the connectivity of the governing phenomenological equations, robust solution strategies could be formulated that aggregate linear relationships and decouple nonlinearities. With the wealth of decomposition algorithms and computing architectures that enable the solution of

complex and large-scale problems, it may be feasible to screen for alternative problem formulations of a production process that result in more rapid and robust convergence. The problem formulation and the selection of a suitable solution method, however, is not a simple task. To address this challenge, we propose the use of graph-theoretic representations of the process equations and variables to better understand the topology of a chemical process at the *phenomenological level*. Leveraging graph abstractions in this manner can provide an avenue to systematically reformulate the equations and enable advanced, topology-aware decomposition schemes of flowsheets.

In classical representations of a chemical process, units are represented as nodes and streams as edges. Fundamentally, each unit encapsulates a set of equations with internal variables and each stream carries a set of variables with a unique connection. At the phenomenological level, however, each variable may describe multiple equations and may not have a unique connection. For example, temperature may play multiple roles in thermodynamic phase equilibrium, energy balances, and reaction rates. Graph abstractions of process phenomena may unfold the sets of equations and variables present in unit operations to capture how mass, energy, equilibrium, reaction, and transport equations are related through common variables.

A potential use of graph abstractions is the identification and development of robust decomposition algorithms whereby a subset of variables is decoupled to solve for sets of equations iteratively. Distillation column models are a classical chemical engineering example where decomposition algorithms are commonly employed. For example, the Wang-Henke bubble point method converges all stages by iteratively solving mass, equilibrium, summation and enthalpy (MESH) equations [9]. In fact, MESH partitioning within an equation-oriented approach has already been integrated to solve distillation trains within an equation-oriented approach [7]. At the flowsheet level, however, no unified approach for decomposition exists yet.

In this study, we developed a graph abstraction framework for process phenomena that can be used to systematically represent, analyze, and visualize the structure of a chemical process. To automate the generation of phenomenological graphs of complex chemical processes, we developed PhenomeNode, an open-source library in Python. We leverage PhenomeNode to form graph representations of an industrial separation process for separating glacial acetic acid from water and develop a preliminary convergence algorithm that leverages the mathematical topology. We benchmark this new algorithm against classical sequential modular simulation to understand its benefits and limitations.

MATERIALS & METHODS

Benchmark Acetic Acid Purification Process

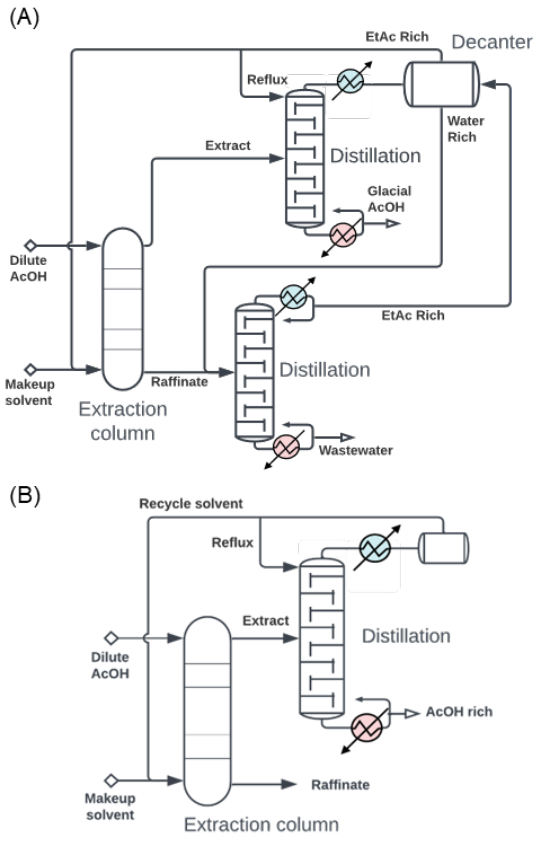

Figure 1. Process flowsheet for the industrial separation of acetic acid from water.

The industrial separation of acetic acid from water (Figure 1A) was chosen as a representative case to benchmark the new convergence algorithm against sequential modular simulation. The system includes highly coupled vapor-liquid equilibrium (VLE) and liquid-liquid equilibrium (LLE) stages and recycle loops that connect the end of the process with the start. In this process, ethyl acetate (EtOH) is used to extract acetic acid from a dilute aqueous mixture [10]. The extract is distilled to recycle the solvent and recover glacial acetic acid. The raffinate is also distilled to recover EtOH from the wastewater. Both distillates are sent to a decanter to separate the aqueous phase that forms after condensation. Additionally, a smaller system composed of a liquid extraction column and a distillation column for solvent recovery (Figure 1B) is used as a benchmark for small, highly coupled systems.

The phenomena graph includes only the equations pertinent to flowsheet convergence, including duties and

material flows across unit operations and stages. Equations for sizing and costing of unit operations are not included as they can be directly calculated based on the steady state results. Process simulations were performed 10 times for each configuration to accurately measure convergence errors and material and energy balance errors as a function of computation time. The convergence error for a given variable at iteration k is defined as the absolute difference between the current value and the last value (i.e., $X_{error} = |X_k - X_{k-1}|$). The material balance error is defined as the difference between outlet flow rates and inlet flow rates (i.e., $F_{error} = \sum_c |\sum_o F_{c,o} - \sum_i F_{c,i}|$). The energy balance error is defined as the difference between outlet enthalpy and inlet enthalpy divided by the outlet heat capacity flow (i.e., $T_{error} = |\sum_o H_o - \sum_i H_i| \cdot (\sum_o C_o)^{-1}$).

Phenomenological Graph Architecture

The graph abstractions in this study represent both variables and equations as nodes. Edges between variables and equations provide information on the variables present in each equation. Variables in the nodes are labeled by a shorthand name and their context. The context differentiates variables by information such as the phase and the parent unit operation. For example, the vapor mol fraction within a vapor-liquid equilibrium (VLE) stage may be labeled as $z_{gas,\ s=0}$, where "z" denotes the mol fraction, "gas" denotes the phase, and "s" denotes the stage number. Table 1 lists the subscript contexts and Table 2 lists the variable definitions used in this study. The "inlet source" refers to a neighboring stage that feeds the reference stage. Similarly, the "outlet sink" refers to a neighboring stage that is fed by the reference stage.It is possible to obtain different graph representations by reformulating the equations. For example, variable nodes can be eliminated via substitution of equations and new edges may be formed to reflect these changes. Aggregating equations in this manner may be helpful to represent a particular convergence algorithm. The graph representations in this study employ as many variables as possible to provide a starting point for future studies to strategically collapse nodes and/or partition the graph.

Table 1. Contexts of variable subscripts.

Subscript	Context	Subscript	Context
c	Chemical	gas	Gas
p	Phase	liq	Liquid
i	Inlet source	ext	Extract
o	Outlet sink	raf	Raffinate

Drawing, positioning, and labeling nodes for a large network of unit operations is a time-consuming task. To automate the systematic generation of graphical abstractions for complex chemical processes, we developed a new open-source Python library called PhenomeNode [11] that leverages the Graphviz software [12] to generate graphical abstractions from the phenomenological equations within unit operations. In the spirit of expanding the use of graph representations at the phenomena level —not just at level of unit operations or superstructures of units— the PhenomeNode software developed in this study is made open-source and readily available for others to leverage. The roadmap for PhenomeNode includes expanding the built-in equation library, refining core features for aggregation and decoupling of equations, and creating educational tools for student learning of the underlying phenomena behind production processes.

Table 2. Variable definitions.

Variable	Definition
F	Molar flow rate
H	Enthalpy flow rate
Q	Duty
V	Molar fraction of vapor phase
Φ	Ratio of top and bottom phase flow rates
L	Molar fraction of extract or liquid phase
T	Temperature
P	Pressure
Z	Molar composition
K	Partition coefficient
C	Heat capacity rate
t	Top phase split fraction
b	Bottom phase split fraction
h	Specific molar enthalpy
f	Fugacity

Mathematical Relationships within Graphs

The equations describing the physical phenomena within the graph abstractions are agnostic to the thermodynamic property package. Equations for pure component and mixture properties are abstracted by black-box function calls. For example, the VLE criteria is represented by equation 1, where $f_{c,gas}$ and $f_{c,liq}$ are function calls to calculate the gas and liquid fugacities, respectively, for an arbitrary chemical. Other key equations include the Rachford-Rice, material balance, energy balance, and pressure drop (eqs 2–5, respectively).

$$f_{c,gas}(z_{c,gas}, T, P) = f_{c,liq}(z_{c,liq}, T, P) \quad (1)$$

$$\sum_c \frac{z_c(K_{c,gas}-1)}{1+V*(K_{c,gas}V-1)} = 0 \quad (2)$$

$$F_{c,liq}\left(1+\frac{V}{1-V}K_{c,gas}\right) - F_c = 0 \quad (3)$$

$$F_{liq}\left(\frac{V}{1-V}h_{gas} + h_{liq}\right) - H = 0 \quad (4)$$

$$P_{o=0} = P_{o=1} - \Delta P \quad (5)$$

A liquid-liquid equilibrium (LLE) stage has two liquid phases, "ext" and "raf", which correspond to the extract

and the raffinate phases, respectively. The material and energy balance equations for an LLE stage are analogous to a VLE stage. The pressure drop across LLE stages is neglected assuming that liquid fugacities are a weak function of pressure. LLE criteria (eq 6) requires that the total Gibb's free energy of all phases is minimized.

$$\min_{z_{c,raf}} G\left(F_c, z_{c,raf}, , z_{c,ext}, T, P\right) = z_{c,raf} \quad (6)$$

These are the fundamental equations for distillation and liquid-liquid extraction algorithms. However, they can be reformulated to fit a specific solution scheme. For example, in the case of LLE, the enthalpy is a stronger function of temperature than the phase fraction and the energy balance is formulated in terms of temperature (assuming an average heat capacity flow for each phase and a reference temperature and enthalpy):

$$\left(H_{ext}^{ref} + H_{raf}^{ref}\right) + \left(T - T^{ref}\right)\left(\bar{C}_{ext} + \bar{C}_{raf}\right) - H = 0 \quad (7)$$

Phenomena-Oriented Simulation Algorithm

We developed a new flowsheet convergence algorithm that expands and integrates MESH-based multistage equilibrium algorithms together with the sequential modular approach. Additional details on the algorithm architecture and performance are discussed in the RESULTS & DISCUSSION section. The material and energy balance equations are used to tie in all the phenomenological equations together. A stage includes all mixing and splitting of inlet and outlet streams, respectively, without limitation on the number of connected stages. For a given stage, F_c for all streams can be computed through the material balance equations (8–11):

$$\Sigma_o F_{c,o} - \Sigma_i F_{c,i} = 0 \quad (8)$$

$$\Phi K_{c,gas} \Sigma_o F_{c,liq,o} - \Sigma_o F_{c,gas,o} = 0 \quad (9)$$

$$F_{c,gas,o} - t_o \Sigma_o F_{c,gas,o} = 0 \quad (10)$$

$$F_{c,liq,o} - b_o \Sigma_o F_{c,liq,o} = 0 \quad (11)$$

The mass balance equations apply to both LLE and VLE stages, with either extract and raffinate or gas and liquid phases. Note that subscripts i and o refer to upstream and downstream stages, respectively, which are directly connected to the reference stage. For computational efficiency, the energy balance equation solves for the change in linearized variables (i.e., T for a LLE stage and Φ for a VLE stage) in each iteration (e.g., we solve for ΔT where $T_{i+1} = T_i + \Delta T$). Combining equations 4 and 7 (and taking ΔT and ΔΦ as the linearized variables) the energy balance for an adiabatic VLE stage becomes:

$$\Delta\Phi h_{gas} \Sigma_o F_{liq,o} - \Sigma_i C_{liq,i} \Delta T_i - \Sigma_i \Delta\Phi_i h_{gas,i} F_{liq,i} = \Sigma_o H_o - \Sigma_i H_i \quad (12)$$

Similarly, the energy balance for an adiabatic LLE stage becomes:

$$\Delta T \Sigma_o C_{liq,o} - \Sigma_i C_{liq,i} \Delta T_i - \Sigma_i \Delta\Phi_i h_{gas,i} F_{liq,i} = \Sigma_o H_o - \Sigma_i H_i \quad (13)$$

Given the coupling between the material and energy balances through F_c and Φ, solving for T at each stage may help close the energy balance for large systems with strong compositional dependencies on specific enthalpies:

$$\Delta T \Sigma_o C_o - \Sigma_i C_i \Delta T_i = \Sigma_o H_o - \Sigma_i H_i \quad (14)$$

The VLE criteria (equation 1) is used to solve for K and T. The equilibrium temperature of an LLE stage, however, is not unique for a bulk composition. Instead, the LLE criteria (equation 6) is used to solve for K and Φ. In the special case of multistage LLE, where equilibrium is highly sensitive to bulk compositions, the pseudo-equilibrium concept is employed for more rapid and robust convergence of the equilibrium criteria across all stages [13]. Given the equilibrium criteria variables and initial guesses for Φ and T, component flow rates across all stages can be linearly solved by the material balance equations (8–11), and Φ of VLE stages and T of LLE stages can be linearly solved by the energy balance equations (12–13). However, an algorithm employing solely equilibrium, material, and energy equations may fail due to poor conditioning of the initial guess. For this reason, we integrate sequential modular simulation —a robust approach used by leading process simulators— within the simulation algorithm for more robust convergence. All together, we propose the following algorithm as a preliminary architecture:

Phenomena-oriented simulation algorithm

1. In the absence of an initial guess for F_c, K, T, and Φ for each stage, run each unit operation sequentially to find initial guesses.
2. For each unit operation:
   - 2.1 Solve the unit operation and update variables K, T, and Φ.
   - 2.2 Solve for F_c across all unit operations as a system of linear equations (eqs 8–11) and update.
   - 2.3 Solve ΔΦ for each VLE stage and ΔT for each LLE stage as a system of linear equations (eqs 12–13) and update Φ and T.
3. Solve for the equilibrium criteria variables at each stage. For multistage LLE, solve for all stages simultaneously using the pseudo equilibrium approach.
4. Run steps 2.1, 2.2, then 2.1 again to close the material balance.
5. Solve for ΔT as a system of linear equations (eq 14) and update T to close the energy balance.
6. If partition coefficients, phase ratios, temperatures, and flow rates have not converged under a specified tolerance, repeat steps 2–6.

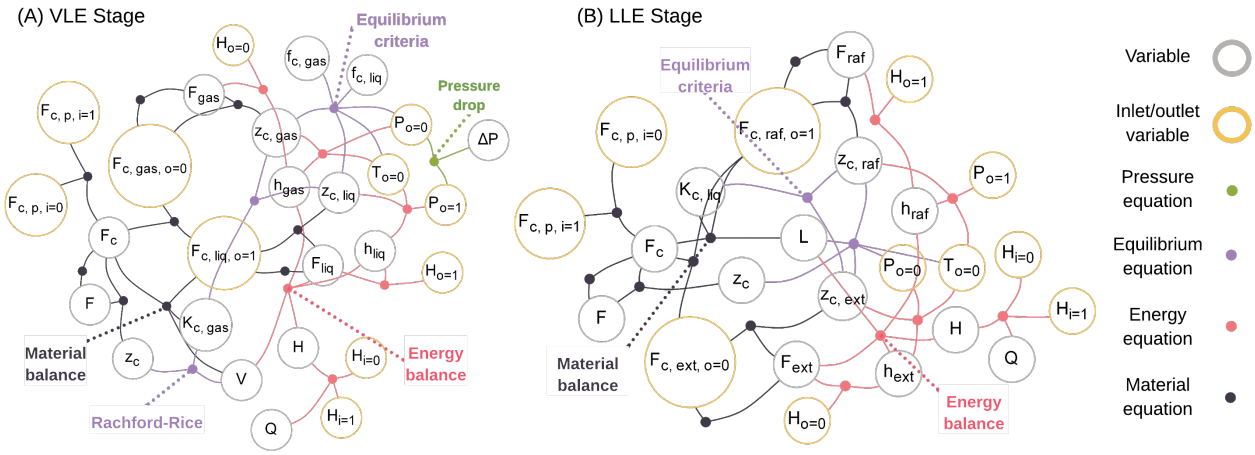

Figure 3. A phenomena graph representation of (A) a VLE stage and (B) a LLE stage. Equations related to material, energy, equilibrium, and pressure are colored black, red, purple, and green, respectively. Each grey ring node and its label represents a variable. Edges denote the variables present equations.

This algorithm was implemented in BioSTEAM —an open-source process simulation software [14]—and benchmarked against sequential modular simulation using a flowsheet for the industrial separation of acetic acid (also modeled in BioSTEAM). The thermodynamic property package estimates phase equilibrium using modified Raoult's law with activity coefficients estimated through Dortmund UNIFAC interaction parameters [15,16]. Pure component properties (e.g., heat capacity) of fluids are estimated using higher order polynomial fits to the fundamental Helmholtz equation of state (a state-of-the-art property prediction model)[17,18] and recommended correlations from a critical review on thermodynamic properties [19]. Mixture properties are estimated using a molar weighted average of the pure chemical properties.

RESULTS & DISCUSSION

Single and Multistage Equilibrium Graph

Material, energy, equilibrium, and summation equations are tightly coupled for both a VLE stage and a LLE stage (Figures 3 A and B, respectively). Note that the designation of what constitutes an equilibrium, material balance, or energy balance equation is based on the unit dimensionality of the equation as well as the decomposition scheme requirements. For example, the Rashford-Rice equation may be regarded as a material balance equation, but it is denoted here as an equilibrium relationship to maintain linear relationships within the material balance (assuming partition coefficients and phase fractions are decoupled).

A classical chemical engineering method to solve an adiabatic flash vessel consists of a 3-step fixed-point iteration: (1) compute partition coefficients and temperature assuming the vapor and liquid compositions (e.g., by computing the bubble point), (2) estimate the vapor and liquid compositions through material balances, and (3) solve for the vapor fraction using an energy balance [20]. In this iterative method, the equilibrium, energy, and material balances are decoupled. Given that non-linearities stem from the equilibrium criteria and the enthalpy of streams, this decomposition algorithm may offer greater stability than derivative-based numerical methods when employed for larger systems.

Partitioning the equations by equilibrium, energy, and material phenomena results in a more manageable problem formulation that is often used by published algorithms for multistage VLE and LLE. Assuming the temperature, pressure, and phase fraction (i.e., vapor or extract fraction) are held constant, the material balance for an arbitrary component becomes a linear combination of liquid (or vapor) flow rates and can be represented as a tridiagonal matrix that is convenient to solve. Similarly, if bulk liquid flow rates, compositions, and temperature are held constant, the energy balance becomes a linear combination of the boil-up ratio. The new phenomena-oriented simulation algorithm proposed in this study leverages this partitioning scheme towards the complete flowsheet to accelerate convergence.

Block convergence methods for distillation (e.g., Wang-Henke's bubble point method) and liquid-liquid extraction —which iteratively solve equilibrium, material, and energy equations— are still subject to instabilities and may not converge. Russell's inside-out method leverages approximate models for stage temperature and specific enthalpy to converge an inner loop of stage temperatures and phase flow rates [21]. Employing approximate models for highly coupled non-linear variables (i.e., partition coefficients, enthalpy, and temperature) allows for more robust convergence if the approximate form is better behaved than the strict form. Future work may

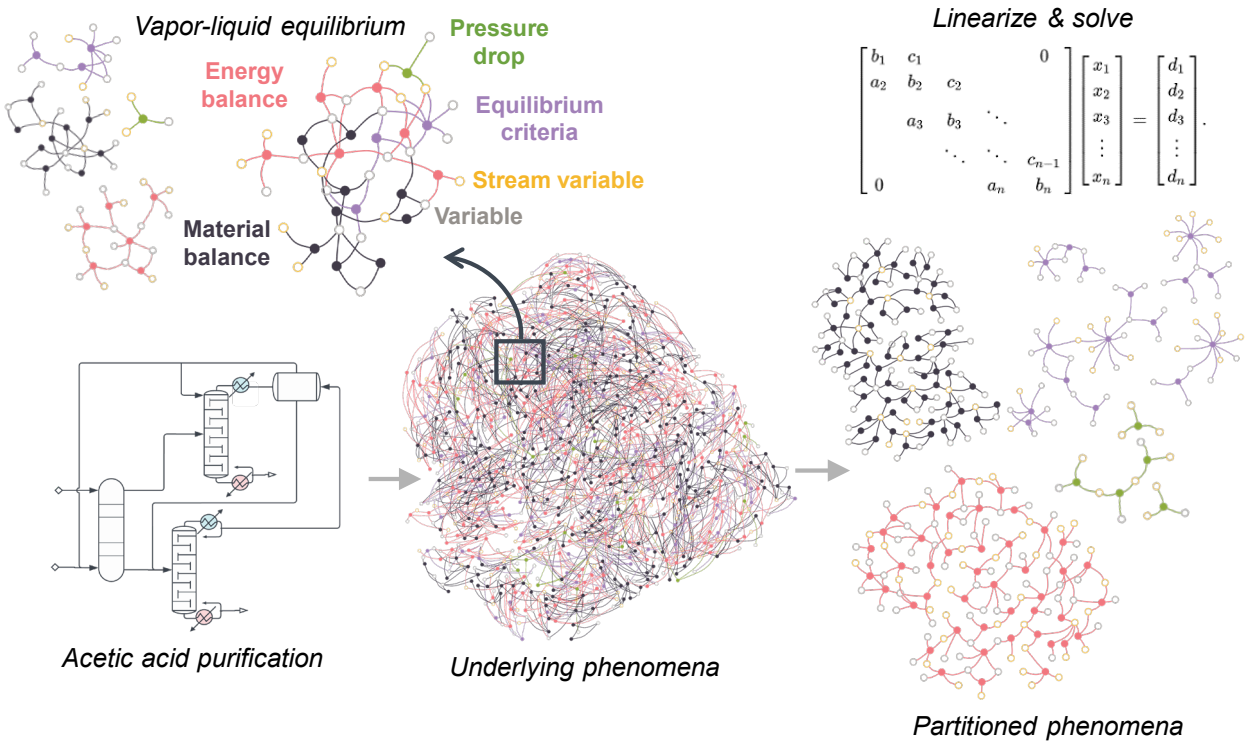

Figure 5. A phenomena graph representation of an industrial process for the separation of acetic acid from a dilute aqueous mixture to produce glacial acetic acid depicts how decoupling material (black), energy (red), and equilibrium (purple) equations may result in a more maganeable formulation even for complex recycle systems. Equation nodes for repeated adiabatic stages in the partitioned phenomena are aggregated for clarity. Variable nodes are shaped as silver and gold rings to denote intermediate variables and stream variables.

seek to reformulate the system of equations with approximate models that can describe the complete system and propose more advanced numerical methods to accelerate flowsheet convergence.

Acetic Acid Separation Graph

A graph of an industrial separation process for separating glacial acetic acid from water suggests that it may be feasible to extend MESH equations as a decoupling strategy for integrated convergence of the complete flowsheet (Figure 5). While the graphical representation may seem more coupled due to the recycle loops, the resulting problem is essentially the same as in multistage equilibrium. In the proposed phenomena-oriented algorithm, equilibrium variables of each stage are calculated in parallel while energy and material balances result in sparse matrices that are linearly solved (steps 3–5). It is possible recycle loops between distillation columns and the liquid-liquid extraction destabilizes stage-wise material and energy balances and lead to ill-conditioned matrices and infeasible results (e.g., negative flows) which would break the model. To resolve such issues, we integrate sequential modular simulation as a robust method to rigorously estimate equilibrium criteria variables (step 2.1). The system-wide mass and energy balances (steps 2.2 and 2.3) helps propagate the information of how an individual unit simulation impacts the flowsheet, enforce mass and energy balances, and accelerate simulations.

Phenomena-Oriented Simulation Speed

In both the benchmark cases, the preliminary phenomena-oriented simulation algorithm converged faster than the sequential modular approach (Figures 6 A, B, C, and D). The error in the mass and energy balance per stage was quickly minimized in the phenomena-oriented algorithm (Figures 6 E, F, G, and H) due to how mass and energy balances of the complete flowsheet are consolidated at the end of each iteration (steps 4 and 5). In comparison, sequential modular simulation has difficulties in closing the mass balance for the complex system even after 60 seconds of simulations (Figure 6H). By speeding up flowsheet convergence, phenomena-oriented simulation may enable large-scale simulations necessary for robust optimization and rigorous uncertainty and sensitivity analyses. It may be possible, however, that the proposed phenomena-oriented algorithm may not be as robust as sequential modular in other systems and more cases should be evaluated for more generalizable insight. Additionally, only one initial condition (starting with empty

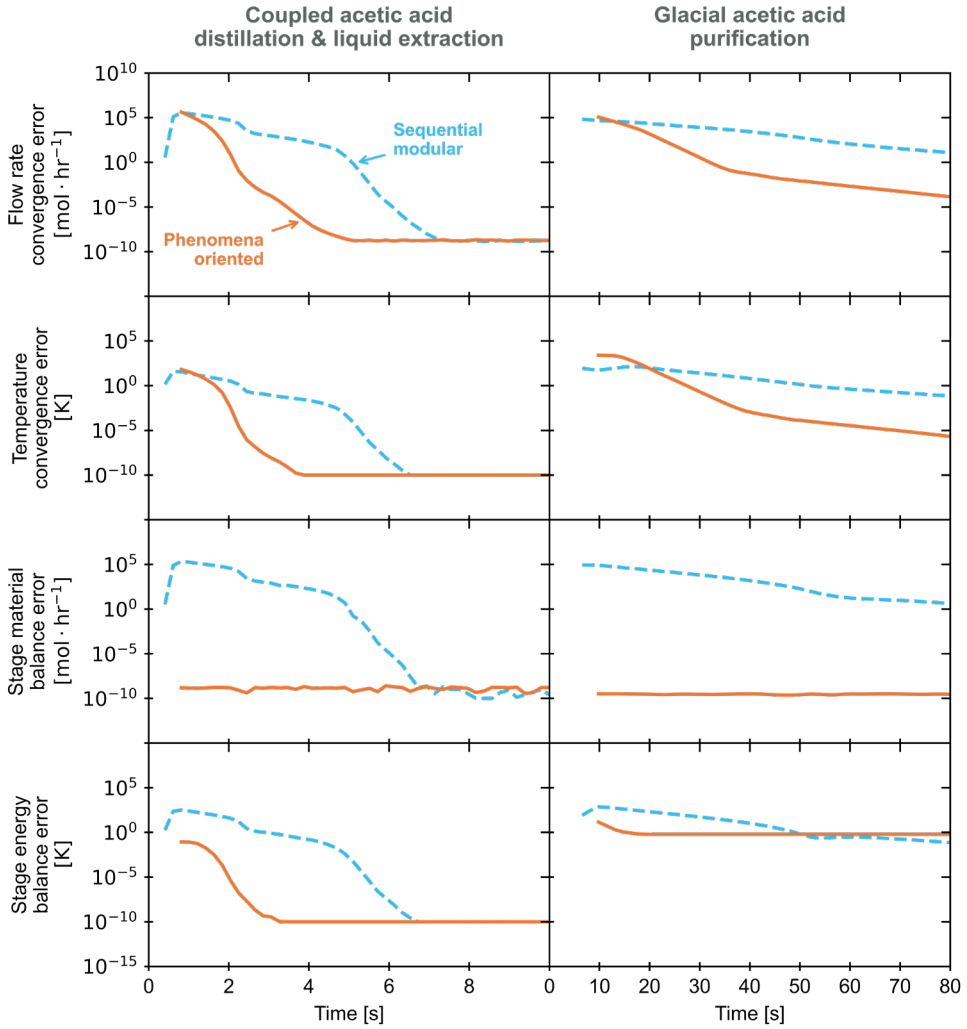

Figure 6. The convergence error of the flow rate (A, B), temperature (C, D), as well as the error in the material balance (E, F), and the energy balance (G, H) was evaluated across computation time for a simple configuration with liquid-liquid extraction and distillation and a more complex system for acetic acid purification (left and right columns, respectively). The blue curve represents values using the sequential modular approach and the orange curve represents values using the new phenomena-oriented approach.

streams) was tested yet simulations used in optimization and uncertainty analyses may leverage more informed guesses. Future studies may seek to analyze the robustness of this new convergence algorithm more rigorously through Monte Carlo methods. It is possible certain algorithms scale better with large, sparse systems than with small, highly coupled systems.

It is important to note that this separation process is limited to only phase-based separations and does not encapsulate reactions and mass and energy transfer-based unit operations that are common in a production process. Still, the equations and decoupling strategies leveraged here can be extended to include other physical phenomena. For example, equilibrium approaches to reactive distillation extend Russell's inside-out method to include reaction terms and decouples the kinetic terms from the equilibrium criteria [22]. We envision a family of phenomena-oriented simulation architectures that are optimized for different problems based on the relevant phenomena of the system, level of connectivity between unit operations, and chemical interactions that drive coupling between phenomenological equations. Ultimately, the phenomena graph representations introduced in this study may aid in the development of unified decoupling and linearization strategies that can allow for more rapid and robust flowsheet convergence.

CONCLUSIONS

Efficiently solving a flowsheet remains a challenge due to the tightly coupled, non-linear nature of chemical processes. While a variety of methods exist for

converging a flowsheet, there is no clear winner in terms of speed, flexibility, and robustness. The phenomena graph representations developed in this study helped identify how decoupling equations into material, energy, and equilibrium blocks (and potentially other phenomena such as reactions) can be used to formulate a strategy to converge the complete flowsheet. While the proposed algorithm showed promising advantages in convergence speed composed to sequential modular, further analysis is needed to fully characterize the speed, robustness, and applicability to a broader set of chemical processes. The PhenomeNode library in Python can be used as an educational tool to help students visualize a chemical process not just as a flowsheet of unit operations, but also as a coupled network of physical phenomena.

ACKNOWLEDGEMENTS

This material is based upon work supported by the U.S. Department of Energy, Office of Energy Efficiency and Renewable Energy, Bioenergy Technologies Office under Award Number DE-EE0009285.

REFERENCES

1. Motard, R. L., Shacham, M. & Rosen, E. M. Steady state chemical process simulation. *AIChE Journal* **21**, 417–436 (1975).
2. Mahalec, V., Kluzik, H. & Evans, L. B. Simultaneous modular algorithm for steady-state flowsheet simulation and design. *Computers & Chemical Engineering* **3**, 373 (1979).
3. Bogle, I. D. L. & Perkins, J. D. Sparse newton-like methods in equation oriented flowsheeting. *Computers & Chemical Engineering* **12**, 791–805 (1988).
4. Tsay, C. & Baldea, M. Fast and efficient chemical process flowsheet simulation by pseudo-transient continuation on inertial manifolds. *Computer Methods in Applied Mechanics and Engineering* **348**, 935–953 (2019).
5. McBride, K. & Sundmacher, K. Overview of Surrogate Modeling in Chemical Process Engineering. *Chemie Ingenieur Technik* **91**, 228–239 (2019).
6. Quirante, N., Javaloyes-Antón, J. & Caballero, J. A. Hybrid simulation-equation based synthesis of chemical processes. *Chemical Engineering Research and Design* **132**, 766–784 (2018).
7. Ishii, Y. & Otto, F. D. Novel and fundamental strategies for equation-oriented process flowsheeting. *Computers & Chemical Engineering* **32**, 1842–1860 (2008).
8. Ishii, Y. & Otto, F. D. An alternate computational architecture for advanced process engineering. *Computers & Chemical Engineering* **35**, 575–594 (2011).
9. Monroy-Loperena, R. Simulation of Multicomponent Multistage Vapor–Liquid Separations. An Improved Algorithm Using the Wang–Henke Tridiagonal Matrix Method. *Ind. Eng. Chem. Res.* **42**, 175–182 (2003).
10. Seader, J. D., Henley, E. J. & Roper, D. K. *Separation Process Principles, 3rd Edition*. (John Wiley & Sons, Inc., 2011).
11. Cortes-Pena, Y. PhenomeNode: Graphical Representations of Process Phenomena.
12. Ellson, J., Gansner, E., Koutsofios, L., North, S. C. & Woodhull, G. Graphviz— Open Source Graph Drawing Tools. in *Graph Drawing* (eds. Mutzel, P., Jünger, M. & Leipert, S.) vol. 2265 483–484 (Springer Berlin Heidelberg, Berlin, Heidelberg, 2002).
13. TSUBOKA, T. & KATAYAMA, T. General design algorithm based on pseudo-equilibrium concept for multistage multi-component liquid-liquid separation processes. *Journal of Chemical Engineering of Japan* **9**, 40–45 (1976).
14. Cortes-Peña, Y., Kumar, D., Singh, V. & Guest, J. S. BioSTEAM: A Fast and Flexible Platform for the Design, Simulation, and Techno-Economic Analysis of Biorefineries under Uncertainty. *ACS Sustainable Chem. Eng.* **8**, 3302–3310 (2020).
15. Cortés-Peña, Y. Thermosteam: BioSTEAM's Premier Thermodynamic Engine. *JOSS* **5**, 2814 (2020).
16. Gmehling, J., Kleiber, M., Kolbe, B., Rarey, J., & WILEY-VCH. *Chemical Thermodynamics for Process Simulation*. (2019).
17. Lemmon, E. W. & Tillner-Roth, R. A Helmholtz energy equation of state for calculating the thermodynamic properties of fluid mixtures. *Fluid Phase Equilibria* **165**, 1–21 (1999).
18. Bell, I. H., Wronski, J., Quoilin, S. & Lemort, V. Pure and Pseudo-pure Fluid Thermophysical Property Evaluation and the Open-Source Thermophysical Property Library CoolProp. *Ind. Eng. Chem. Res.* **53**, 2498–2508 (2014).
19. Zábranský, M., Kolská, Z., Růžička, V. & Domalski, E. S. Heat Capacity of Liquids: Critical Review and Recommended Values. Supplement II. *Journal of Physical and Chemical Reference Data* **39**, 013103 (2010).
20. Boston, J. F. & Britt, H. I. A radically different formulation and solution of the single-stage flash problem. *Computers & Chemical Engineering* **2**, 109–122 (1978).
21. Russel, R. A flexible and reliable method solves single-tower and crude-distillation-column problems. *Chem. Eng.* **90**, 53–59 (1983).
22. Wang, L., Sun, X., Xia, L., Wang, J. & Xiang, S. Inside–Out Method for Simulating a Reactive Distillation Process. *Processes* **8**, 604 (2020).