Digital Discovery

Accepted Manuscript

This article can be cited before page numbers have been issued, to do this please use: J. Laub, L. Bosetti and A. Bardow, *Digital Discovery*, 2026, DOI: 10.1039/D6DD00060F.

ROYAL SOCIETY OF CHEMISTRY

rsc.li/digitaldiscovery

View Article Online
DOI: 10.1039/D6DD00060F

Journal Name

ARTICLE TYPE

Cite this: DOI: 00.0000/xxxxxxxxxx

Text-to-Flowsheet: An LLM-Assisted Pipeline for Expert-Level Digitization and Automated Simulation of Chemical Processes†

Jan-Frederic Laub,[a,b] Luca Bosetti,[a] and André Bardow[a,b,*]

Received Date
Accepted Date

DOI: 00.0000/xxxxxxxxxx

Converting unstructured natural language descriptions into structured process flowsheets is a fundamental bottleneck in chemical engineering, traditionally requiring years of expert training. While large language models (LLMs) show promise in text comprehension, their ability to match human expertise in modeling complex chemical process flowsheets remains unproven. Here, we present a rigorous benchmark comparing a fully automated LLM-powered digitization pipeline against the collective performance of 50 chemical engineering experts. Our pipeline leverages LLMs to extract process structures from text and formalize them as flowsheet graphs. To handle the inherent ambiguities of natural language, we utilize constrained, step-by-step prompting augmented with thermodynamic property calculations. Subsequently, the digitized flowsheet graphs are automatically translated into the flowsheeting software Aspen Plus to compute rigorous mass and energy balances. Black-box optimization on subprocess structures is used to estimate unknown parameters and ensure simulation convergence, completing the pipeline from text to converged process simulation. For the first time, we demonstrate that an automated pipeline can achieve expert-level accuracy in process topology digitization. Using a unique, newly-generated dataset of 101 expert-drawn flowsheets, we show that our LLM-assisted approach faithfully captures process topology and operating conditions even in the face of incomplete information. This work provides a robust, validated framework for the large-scale digitization of chemical production literature, contributing a transformative tool and dataset for the community to accelerate automated process design and assessment.

1 Introduction

Process flow diagrams (PFDs) are the primary medium for organizing and communicating information about chemical production processes. Flowsheets are used at all stages of the process life cycle, from process development to operation. In the absence of real-world plant measurements, flowsheets are also highly relevant sources of data for assessing the economic and environmental impacts of chemical production, often serving as the best available data for predicting life cycle inventories.[1]

Setting up a flowsheet for process assessment typically involves expert-driven information search, manual digitization, and labor-intensive simulation. Recently, machine learning and other stochastic methods for automated process generation have also begun to utilize large flowsheet databases.[2,3] However, no large repository of industrially relevant flowsheets is publicly available yet. Thus, a workflow is needed to digitally collect, standardize, and utilize flowsheets efficiently.

In a pioneering work to generate such a repository, Schweidtmann and colleagues have conceptualized and partially implemented a mining and digitalization framework for process information and flowsheets.[4] This framework consists of four steps: publication mining, flowsheet digitalization, process description extraction, and semantic database synthesis. Within this framework, the flowsheets are digitized from images using a visual recognition algorithm.[5]

[a]Energy and Process Systems Engineering, ETH Zurich, Switzerland.
[b]NCCR Catalysis, Switzerland.
*Corresponding author. E-mail: abardow@ethz.ch

† Electronic supplementary information (ESI) available. See DOI: TBD before publication.

View Article Online
DOI: 10.1039/D6DD00060F

However, rich process information is already present in natural language descriptions themselves, e.g., from encyclopedias, academic literature, patents, commercial communications, and engineering-adjacent repositories, such as sustainability or cost databases. These text descriptions are sometimes, but not always, accompanied by a graphical flowsheet and usually contain more detailed information on chemical and operational conditions. In contrast to digitizing images of flowsheets, digitizing natural language descriptions lacks a direct topological correspondence between source inputs and digitized outputs; therefore, robust translation techniques are required to produce accurate and meaningful representations of flowsheets.

Recently, Gowaikar *et al.* (2025)[6] have developed an agentic workflow using a large language model (LLM) to translate natural language descriptions of subsystems of piping and instrumentation diagrams (P&IDs) into the XML-based DEXPI[7] format. The flowsheet information in the generated DEXPI file is then visualized in commercial software. The translation procedure is conversation-based and iterates on the subsystem-level over the different to-be-visualized sections of the P&ID with frequent user input. Therefore, the approach by Gowaikar *et al.* (2025)[6] serves as a co-pilot for creating DEXPI-compliant P&IDs rather than a comprehensive pipeline for process digitization from real-world literature.

Furthermore, an open challenge remains to extend beyond flowsheet visualization by automatically computing mass and energy balances for flowsheets digitized from natural language sources. First approaches rely on customized simulation software, which may limit general applicability, and are tailored to process optimization rather than digitization.[8] Even extensive, multi-agent AI systems for the extraction, organization, and synthesis of chemical process descriptions from literature sources still manually construct process simulations to validate their text-based PFDs.[9] Approaches to automatically generating process models exist, for example, at the scale of individual reactor models,[10] for low-fidelity validation of control functions,[11] or for industry-specific flowsheets without reactions or phase separations.[12] However, no holistic workflow exists that transforms process text descriptions into established simulation software, accounting for all relevant phenomena of chemical production, rigorous unit models, and the automated handling of missing information.

In this work, we describe and implement an LLM-assisted data pipeline that converts real-world, natural language text descriptions of chemical production into, first, machine-readable PFD-level flowsheet graphs and, second, converged simulations in commercial flowsheeting software. Our approach combines the language processing capabilities of LLMs with rigorous thermodynamic computations and rule-based flowsheet construction to provide physicochemical context to the automated flowsheeting. The resulting digitized flowsheets are automatically translated into converged Aspen Plus[13] simulations by augmenting missing information with black-box optimization. The result is a comprehensive, robust, and scalable digitization procedure for flowsheets from text sources, contributing to ongoing community efforts to collect and standardize chemical engineering knowledge.

To assess the validity of the extracted flowsheet structures from unstructured text, we introduce a validation method and dataset that compares expert- and computer-generated flowsheets. To that end, we have hand-collected a total of 101 expert-drawn flowsheets for 30 chemical production processes, enabling us to assess flowsheet similarity across expert interpretations and to derive validation targets for automatically digitized flowsheets. The comparison shows that our LLM-assisted pipeline achieves topological accuracy on par with the experts.

2 Automated pipeline from texts to process simulations

The methods developed in this work convert a natural language description of a chemical process into a machine-readable flowsheet graph, which is subsequently translated into an Aspen Plus simulation file. The conversion of the text into a graph is a fully automated, LLM-assisted data pipeline that creates a process topology and augments it with additional chemical and operational information extracted from the text ("text2flowsheet"). In Section 2.1, the data pipeline is described, and in Section 2.2, validation procedures at the flowsheet topology level are discussed. In Section 2.3, the automated translation of the flowsheet graphs into Aspen Plus simulations is explained ("graph2simulation"). Figure 1 shows the processing sequence of the implemented digitization methods from text description, via flowsheet graph, to converged Aspen simulation.

2.1 Converting texts into flowsheet graphs ("text2flowsheet")

2.1.1 Constructing the process topology

The data pipeline for generating graph-based flowsheet topologies employs an LLM to comprehend and systematize information from natural language descriptions. To execute this task, the LLM is given a process description and is systematically prompted multiple times to sequentially build a graph representation of the process. After each step, the LLM outputs are standardized and converted to machine-readable formats, then re-entered into subsequent prompts to progressively construct the flowsheet in a context-rich environment. In contrast to prior work on co-pilot applications,[6] our overall processing sequence is deterministic and not planned by the LLM itself to ensure robust adherence to the overarching digitization task. Furthermore, we include additional steps between LLM calls, based on thermodynamic calculations and flowsheeting logic, to contribute to the contextual framework of the prompts. All prompts are provided in the supplementary information.[†]

Figure 2 shows the overall processing step sequence. The core step in composing a process's topology is a depth-first, iterative graph construction in which the LLM determines the placement

View Article Online
DOI: 10.1039/D6DD00060F

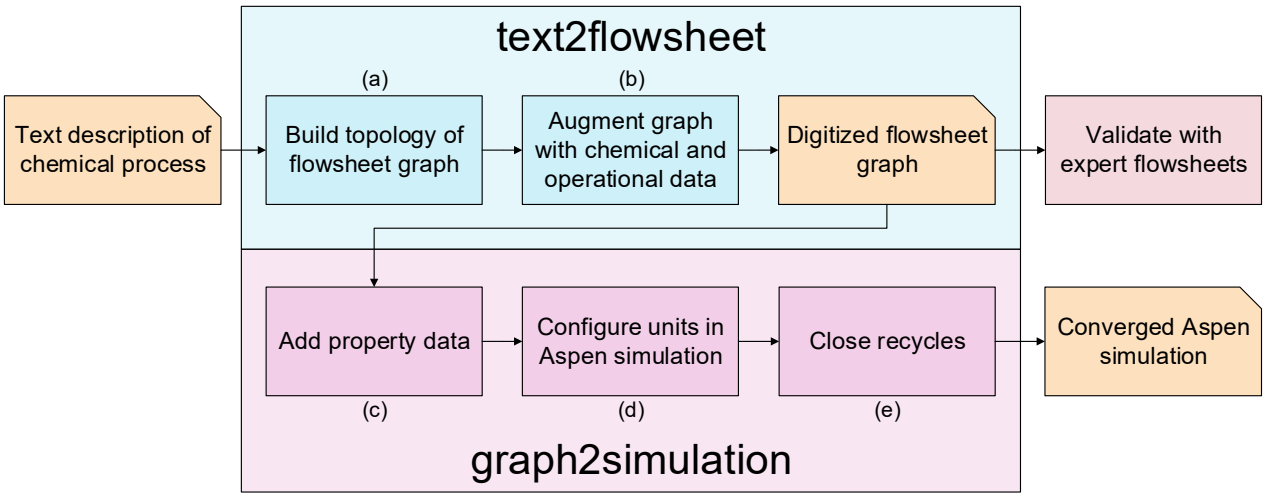

Fig. 1 High-level program flowchart for automatically digitizing and simulating chemical process flow diagrams from natural language descriptions.

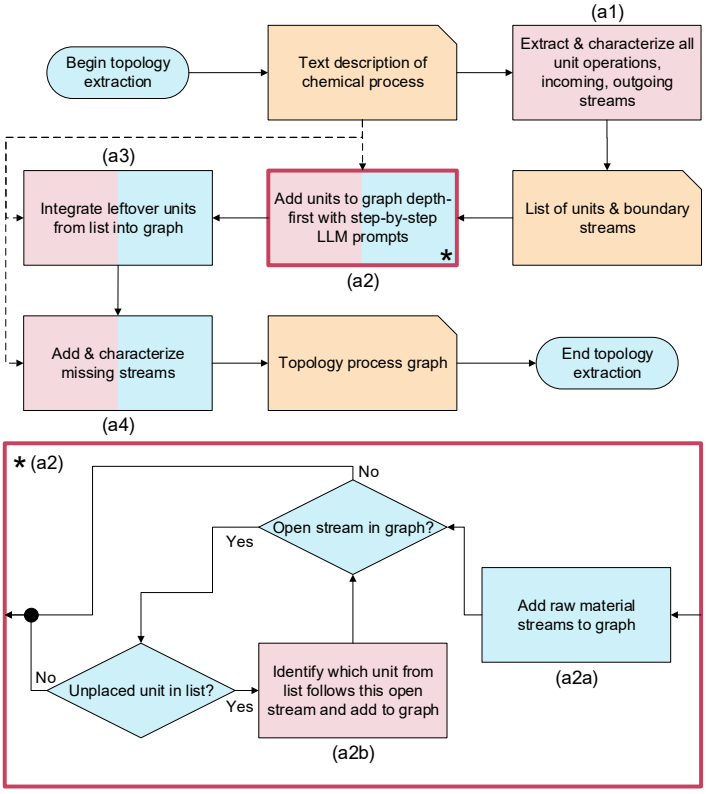

Fig. 2 High-level program flowchart of the "text2flowsheet" digitization pipeline. Light-red steps involve structured LLM prompts, while blue steps rely on logic and rule-based operations. Yellow boxes describe input, output, and intermediate data formats. The dashed lines indicate that the text description is provided for most of the subsequent LLM prompts as well. The step-by-step addition of units to the topology graph is further detailed in the box outlined in red.

of a subsequent unit node at each open stream (Figure 2, a2). The flowsheet graph is initialized with all raw material streams (defined as nodes) extracted in the previous step (Figure 2, a2a). Then, the LLM is queried to determine the next unit in the process sequence from the candidate pool of unit operations and product streams (Figure 2, a2b). The preliminary topology is complete once depth-first graph construction is exhausted, i.e., every node has a successor or is itself a product stream.

After undergoing further checks for logical consistency with the text and process design conventions, the topology graph closely reflects the structural information in the process description while meeting the basic topological requirements of a process flowsheet. From this point on, the topology is not further manipulated by the LLM and serves as the basis for subsequent information extraction steps that augment it with chemical and operational data.

2.1.2 Augmenting the topology with chemical and operational information

After completing the topology pipeline described in the previous section, the process graph describes the material flow through the process and contains the type of each unit operation. Usually, process descriptions include additional information, such as the names of produced and consumed chemicals, stream compositions, temperatures, and pressures. This data is now extracted from the text by the LLM and assigned to the relevant unit nodes in the graph representation.

The core element of process data augmentation is the analysis of the process's separation steps. For the eventual simulation, it is critical to determine which component-wise splits are targeted by each separation unit, as specified in the process description. As this information is often absent or only implied in natural language sources, we trace the component splits from products to reactants in the opposite direction to the process's material flow: First, all recycle streams are temporarily torn to convert the flowsheet graph into a tree. Then, a hierarchy of

View Article Online
DOI: 10.1039/D6DD00060F

separation nodes is determined based on each separator unit's distance to the last reactor node. The components of each stream are traced back through the flowsheet level by level, starting from the outlet streams. At each separation node, the outgoing components are aggregated into a list, and the component-wise split at that node is recorded and categorized by outlet stream type (e.g., top/bottom or vapor/liquid). When one of the separator's outlets is not connected to a product node because it is part of the recycle in the original cyclic graph, the LLM is queried to determine the recycled compounds using the process description as context.

Identifying the key components in each stream characterizes the separation tasks. In the case of distillation-like units, the boiling temperature of each compound is computed with FeOs[14] using the experimentally fitted PC-SAFT parameters from Esper *et al.* (2023)[15] if available, or, if not, the predicted PC-SAFT parameters from Winter *et al.* (2025).[16] If the separation pressure or temperature is given, the boiling points are calculated at that condition; otherwise, they are computed at atmospheric conditions. The compounds in the distillate and bottom streams are sorted by relative volatility, and the heavy and light key components are identified. If the determined heavy key has higher volatility than the light key, this check indicates that the LLM incorrectly interpreted the topology of the unit's outgoing streams. Consequently, the positions of the outlet streams are corrected in the graph.

The determined process topology, together with the augmented chemical and operational data, is further processed to enable its automatic translation into Aspen Plus: First, any graph structure is deleted that is not connected to the main process topology, i.e., the largest subgraph that involves a product stream of the processes' main product. Then, a mixing node is inserted before any unit with more than one incoming stream. Finally, paths that start from a raw material node and involve ill-defined chemical species, e.g., due to failed name lookup and standardization, are excluded.

2.1.3 Implementation of "text2flowsheet"

The data pipeline described in the preceding sections is implemented in Python, using DSPy[17] as a programmatic framework to configure, store, and execute LLM prompts. Converting natural language text into machine-readable, standardized data primarily leverages LLMs' structured-output capabilities. The LLMs' hyperparameters were chosen to yield deterministic inference (temperature $= 0$), which is generally recommended for information extraction tasks that rely on structured output, and has been shown not to negatively influence the extraction quality.[18] The desired output format is prescribed using Pydantic[19] typing for every LLM prompt. For example, when determining the type of a unit operation, the only allowed outputs are strings that are part of the union of all SFILES[20] unit abbreviations. The extracted information is collected in a JSON-like format and then parsed deterministically into a networkx[21] graph.

With the rapid advancements of LLMs, the question arises whether a multi-step inference pipeline like the one presented herein is actually necessary or if the LLM could generate the full flowsheet in suitable fidelity in a single prompt alone. However, we have found that a "single-prompt" request challenges the language model substantially. When instructed by a single prompt, the LLM does not produce the desired structured flowsheet output for 22 out of 30 test cases.[†] This structured output is the prerequisite to parse, compare, and further process the flowsheets. Even in the cases where structurally valid output is produced, the similarity of the flowsheets to the expert benchmark is on average lower than the flowsheets from the multi-step pipeline.

Different LLMs have been explored for processing the texts using the described pipeline. The final decision in the LLM selection was made in favor of OpenAI's GPT-5-mini[22], which is a model of adequate size, low cost, and straightforward API integration to empower users to deploy the tool. GPT-5-mini has consistently shown accurate results and reliably adheres to the structured output formats. A local GPT-OSS[23] instance with 120 billion parameters was used when license restrictions prevented the transmission of a process description to external servers. We have found that GPT-OSS performs slightly better on average across the examined test cases, as detailed in the supplementary information.[†] Nevertheless, the examples detailed in the results sections are drawn from GPT-5-mini to provide an adequate representation of what a user can expect when deploying the data pipelines without access to substantial local GPU resources.

We have also obtained adequate results with smaller local models in the 70B-parameter range. Currently, we do not recommend using models smaller than that, as the quality of understanding and adherence to output instructions was observed to deteriorate substantially. We show an example of a consumer-grade 8B-parameter model in the supplementary information.[†] The smaller model fails to produce adequate structured output in 19 out of 30 cases. In the cases where valid output is produced, the flowsheets are less similar to the expert benchmark than the flowsheets of the larger model. Nevertheless, with the rapid progress in local LLMs, we expect the pipeline to work reliably for smaller models in the near future.

The identified, augmented, and pre-processed flowsheets can be visualized using the Python packages SFILES2[20] and pyflowsheet.[24] The SFILES2 package also determines the SFILES string code for each flowsheet. With the SFILES string as an intermediate representation, the extracted flowsheets can be stored and shared in other standardized formats, such as DEXPI Process.[25]

2.2 Quantifying digitization success by comparison to expert flowsheets

The rapid adoption of LLMs in all fields of science, including chemistry and process engineering, necessitates thorough valida-

View Article Online
DOI: 10.1039/D6DD00060F

tion procedures to ensure reliable, accurate, and safe results.[26] Comparing LLM-generated answers with expert responses to domain-specific questions is an established method to assess an LLM's capabilities and highlight areas of further improvement.[27] For information extraction tasks in particular, expert scores and feedback have been used to develop training and validation methods when the generated answers are complex and fuzzy composites of entities and relations.[28] Thus, to quantify the success of extracting a flowsheet from a text description, we want to compare LLM- and expert-generated flowsheets.

Assessing the similarity of flowsheets in a meaningful way is not a trivial task. To provide a detailed assessment, we have selected two graph-based metrics: Commenge and Piña-Martinez (2026)[2] define a metric of dissimilarity between two flowsheets based on evolutionary graph manipulations. Although this metric does not account for the inner connectivity of the graphs, it provides a meaningful perspective, as it is rooted in domain knowledge by drawing on common graph manipulations in evolutionary process design (see, for example, Neveux (2018)).[29] To mitigate the shortcomings of the evolutionary metric, we additionally employ a graph kernel based on the Weisfeiler–Lehman (WL) subtree framework to obtain a topology-sensitive yet computationally tractable similarity score.[30] WL-type graph kernels are typically recommended for sparse graphs,[31] like our flowsheets. We discuss both metrics in more detail in the supplementary information.†

As a lower reference point, we compute the average similarity between a randomly generated graph with a similar size and edge density to the automatically generated one (see red bars in Figure 3). Any score near or below this bound would indicate a digitization performance that is no better than randomly assembling a similar-sized flowsheet from the given list of unit operation types.

The LLM-generated flowsheets are compared with the full benchmark set of expert and random similarities in Section 3.1.

2.3 Parsing the digitized flowsheets to Aspen Plus ("graph2simulation")

The machine-readable flowsheet graphs created with the "text2flowsheet" pipeline (Section 2.1) are already effective formats for assessing, storing, and optimizing chemical processes.[32,33] However, our primary goal of gathering and evaluating large amounts of process information is to determine mass and energy balances that can inform and improve process analysis and design. For example, using established simulation frameworks such as Aspen Plus is the preferred method for generating data for life cycle assessments when real-life plant data is unavailable.[34] Therefore, we now complete the digitization pipeline by automatically parsing the flowsheet graphs into the software Aspen Plus to obtain converged flowsheet simulations. A reverse approach is presented in Martinez-Hernandez (2023),[32] which converts Aspen Plus simulations into graphs for further analysis using graph-theoretic methods. However, in our case, the graphs

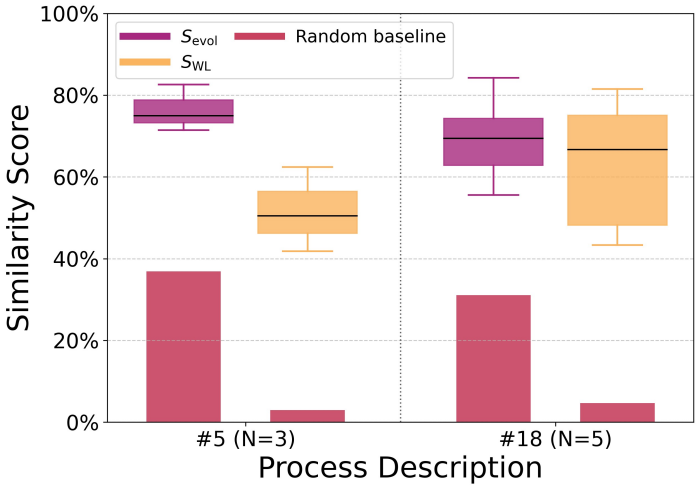

Fig. 3 Distribution of pairwise expert similarities for two exemplary processes from the test set. The red bars show the median similarity of a digitized flowsheet graph to 10000 random, flowsheet-like graphs. N denotes the number of expert-drawn flowsheets for each test process. All distributions and more information on the test set processes are provided in Figure 6 and in the supplementary information.†

generally contain less information than necessary for simulation, requiring knowledge-driven rules and optimization to address the missing degrees of freedom.

2.3.1 Translation of the digitized flowsheet into an Aspen simulation

Similar to the sequential approach for digitizing process information in Section 2.1.1, we build the Aspen simulation unit-by-unit to ensure topological consistency and convergence at each step (Figure 4). First, the recycle streams of the flowsheet graph are temporarily torn to create a tree structure. Then, the units and streams of the flowsheet are inserted into Aspen in topological order: starting with the raw material streams (Figure 4, d1), a new unit is added to the simulation once all its upstream units and streams are present (Figure 4, d2). Every unit operation type has been assigned a preferred Aspen Plus model (see supplementary information†) with which a corresponding unit is initialized in the simulation (Figure 4, d2a).

Whenever a new unit is added to the Aspen simulation, it is equipped with the operational parameters extracted from the text description, e.g., temperature and pressure. The unit's remaining degrees of freedom are determined by stochastically optimizing the Aspen simulation as a black-box (Figure 4, d2b). If the flowsheet converges, the reward is determined by a unit-specific objective function (see supplementary information†). Parameter values that lead to unconverged flowsheets are heavily penalized, so that the black-box optimization moves toward configurations that yield a converged flowsheet in which every unit performs its assigned role as described in the text.

The parameters of the unit operations are additionally co-optimized for energy or solvent consumption in a multi-objective manner. The energy or solvent demand is minimized, preventing,

View Article Online
DOI: 10.1039/D6DD00060F

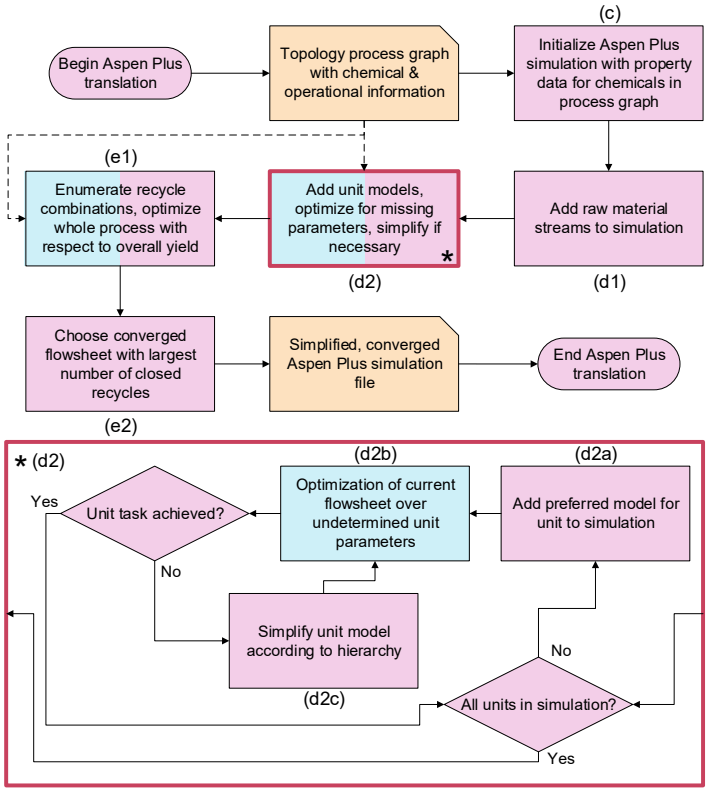

Fig. 4 High-level program flowchart of the graph-to-aspen translation pipeline. Blue steps involve stochastic optimization, while purple steps rely on logic and rule-based operations. Yellow boxes describe input, output, and intermediate data formats. The dashed lines indicate that the topology graph provides information to the simulation at several stages during translation and optimization. The step-by-step addition, optimization, and simplification of units in the simulation are detailed in the box outlined in red.

for example, evaporators from generating a vapor phase through excessive heating. This multi-objective optimization leads to more realistic values for operational parameters, as real-life processes tend to operate near economic and, therefore, energetic optima.

Suppose that, after adding a new unit, the black-box optimization does not yield a converged flowsheet or a unit that fulfills its task in a meaningful way, as defined by thresholds on the objective functions. In that case, the most recently added unit is systematically simplified, and the black-box optimization is re-run (Figure 4, d2c). The hierarchy of unit model simplifications is visualized in Figure 5. The lowest level of simplification leads to highly simplified models that always converge.

A separation unit is eventually simplified from rigorous models into short-cut "Sep" models, which severely underestimate the required energy demands but at least enable a reasonable material flow to downstream stages of the process. Note that the simplification steps occur outside the black-box optimization loops, which means that the optimizer cannot choose to simplify a model as a way to artificially reduce its energy demand. The optimizer operates solely on the model's individual parameters,

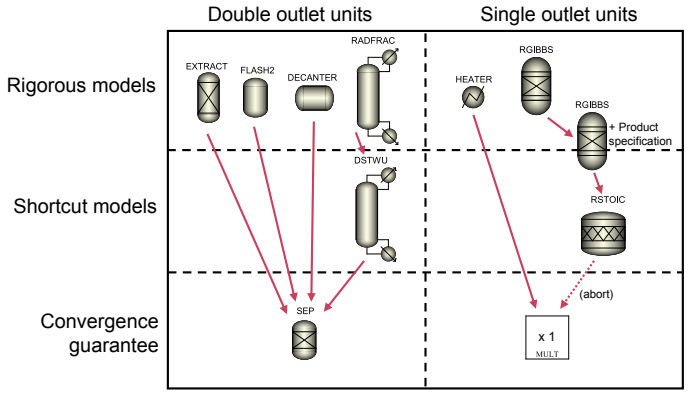

Fig. 5 Simplification hierarchy of Aspen Plus models. Every unit is initialized with its most rigorous model. If the separation or reaction task cannot be achieved, it is simplified to the next simpler unit. The "MULT" block is a dummy block whose outlet stream is identical to its inlet stream.

not on the model selection itself.

Finally, all possible combinations of recycle closures are enumerated, and corresponding Aspen files are generated. For each case, the resulting simulation is optimized to maximize the main product's yield with respect to the inlet flows of raw materials (Figure 4, e1). A recycle variant is rejected if it fails to converge during this optimization or if it results in substantially worse product yields. Ultimately, the converged flowsheet is accepted that possesses the most successfully closed recycles (Figure 4, e2). Heaters, coolers, pumps, and compressors are added at points where temperature or pressure changes between unit operations, making the resulting flowsheet the final product of the digitization pipeline.

The simulation results for the final flowsheet are automatically extracted from Aspen Plus and saved along with the flowsheet topology and associated chemical information. Even if simplifications of units or recycles were necessary to produce a converged simulation, the resulting file is still a highly informative starting point for expert-driven development or can yield preliminary estimates of mass and energy flows. All simplifications at both the unit and recycle levels are recorded and stored alongside the topological and chemical information. The transparency of flagging all performed simplifications enables an expert to conduct deeper analyses and deploy the simulation, while remaining aware of potential limitations and areas for improvement.

2.3.2 Implementation of "graph2simulation"

The procedures for obtaining a converged Aspen flowsheet from the digitized process information described in Section 2.3 are implemented in Python using the Aspen Plus V11 Automation Server Windows COM interface. The multi-objective black-box optimization is performed using the two-point crossover differential evolution algorithm ("TwoPointsDE") in Meta's Nevergrad package.[35] The solver always operates on five Aspen Plus instances in par-

View Article Online
DOI: 10.1039/D6DD00060F

allel with a budget of 400 objective evaluations for reactors and columns, and a budget of 200 evaluations for all other units. If suitable information is available, optimization variables such as temperature and pressure are initialized with sensible values from already configured upstream units. PC-SAFT is used as the property model, and the necessary parameters for the chemical species in each process are automatically added to the corresponding Aspen simulation file before flowsheet construction (Figure 4, c).

3 Results and Discussion

The methods described in Section 2 generate flowsheets from text descriptions in two successive output formats: machine-readable process topology graphs and converged Aspen Plus simulation files. To validate that the output graphs accurately represent the underlying process description, in Section 3.1 we compare the automatically generated flowsheets with expert-generated flowsheets using the similarity metrics and empirically collected benchmark data, as described in Section 2.2. From the systematic translation of the flowsheets into Aspen Plus, we further analyze when significant simplifications are necessary to achieve convergence in Section 3.2. Based on the results, we reflect on the current limitations of the methodology and potential avenues for extension in Section 3.3.

3.1 Similarity of expert- and LLM-generated flowsheets

To evaluate the accuracy of automated digitization procedures, we have selected 30 natural language descriptions of chemical processes from Ullmann's Encyclopedia of Industrial Chemistry[36] and the IHS Markit Process Economics Program Yearbook.[37] The "text2flowsheet" data pipeline processed each description (see Section 2.1), yielding one digitized graph representation per process.

For each process, several expert-drawn flowsheets were collected. Each expert was given a process description, a list of standard unit operation types (the same list used by the LLM), and 15 minutes to draw a flowsheet. The experts were instructed to draw flowsheets at the level of detail typically found in a process flow diagram, rather than a block flow diagram or a P&ID, and to focus on topological correctness, rather than complete chemical information. The majority of the experts who contributed drawings were external to the authors' research group and volunteered during the DECHEMA Annual Meeting of Process Engineering and Materials Technology 2025 in Frankfurt am Main, Germany. The drawn flowsheets were digitized by hand into the same graph format as the automatically digitized ones (see supplementary information†).

Thus, for each process, we have obtained an automatically digitized flowsheet and multiple expert-drawn ones. For each process, pairwise similarities between the automated flowsheet and the expert-drawn flowsheets, as well as between the expert-drawn flowsheets themselves, were computed using the evolutionary and WL similarity metrics. Figure 6 shows the aggregated similarity scores. Overall, the digitized flowsheets are both quantitatively and qualitatively similar to the expert-drawn test set. Analyzing the similarity scores further helps in understanding the decisions made by experts and the LLM in representing the textual process data.

By exceeding the random similarities (see red bars in Figure 6), our methodology consistently extracts a high degree of meaningful structural information from process descriptions. For most processes, the median similarity of LLM-generated flowsheets to expert flowsheets (see blue bars in Figure 6) lies within the distribution of pairwise expert similarities, indicating that the digitized flowsheets are virtually indistinguishable from the expert-generated ones. A meta-analysis of statistical permutation tests over every process (see supplementary information†) yields the same strong indication that a null hypothesis of indistinguishability cannot be rejected. However, more expert data would be necessary to draw definitive statistical conclusions on an individual process basis. Outliers, such as #26, can be explained by their low complexity and small size (3-5 nodes), so that even minor deviations between experts have a significant impact on the normalized similarity scores.

The underlying text descriptions were extracted from real-world sources and are therefore imperfect. In particular, the texts often lack information about several separation steps. Hence, the surveyed experts sometimes expressed difficulties in following the description and drawing correct flowsheets. Interestingly, this shortcoming does not noticeably affect the experts' pairwise similarities, indicating that the experts handled the uncertainty in similar ways. However, uncertainties were often abstracted away by using placeholder "X" or "Sep" units. On average, an expert graph contains 0.47 "X" and 0.76 "Sep" units. In comparison, the LLM determines the type of units more confidently, placing only 0.13 "X" and 0.33 "Sep" units on average in its flowsheets. The low selection of placeholder units could indicate that the LLM draws on prior knowledge acquired during training to inform specific decisions, for example, by leveraging physicochemical insights that experts did not have access to during flowsheet drawing. Apparently, experts adhered to the instruction of basing their interpretation strictly on the given process description more so than the LLM.

By examining two archetypal processes, we further explore the uncertainties in flowsheet drawing and exemplify the overall high quality of the digitization procedures.

3.1.1 Example 1: LLM and experts largely agree on flowsheet

A dichloromethane production process (#19 in Figure 6) was digitized to a high level of expert similarity considering both metrics. Indeed, we observe that the topology and unit distribution of this complex process align closely with the experts' interpretations. The process description from Rossberg *et al.* (2011)[38] is reproduced in the supplementary information.†

Figure 7 compares the automatically digitized flowsheet to

View Article Online
DOI: 10.1039/D6DD00060F

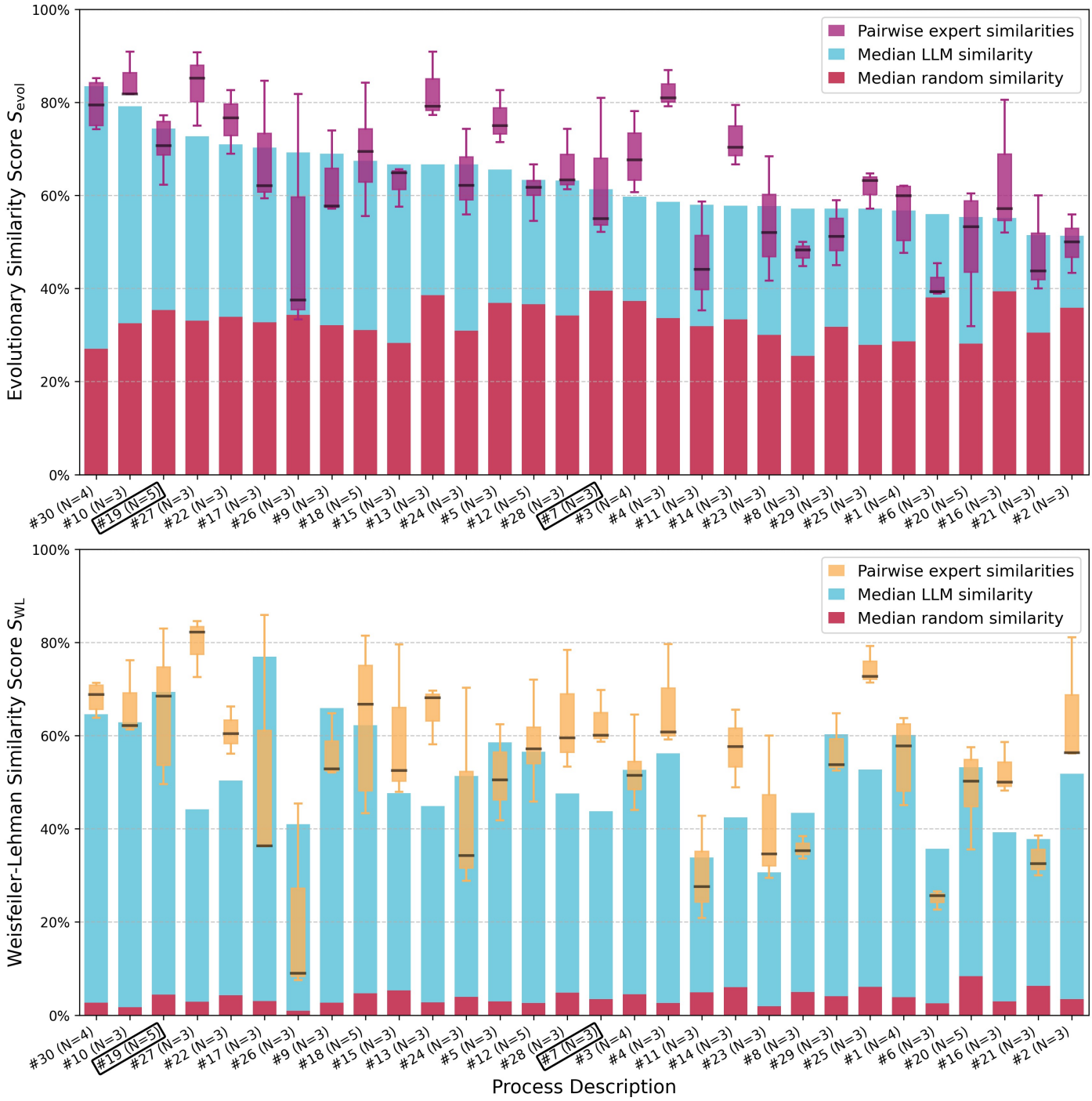

Fig. 6 Aggregated similarity scores for each process description in the test set; evolutionary similarity on top and Weisfeiler-Lehman (WL) similarity on the bottom. Each column represents one process description. The red sections of the columns indicate the median similarity of the digitized flowsheet to 10000 randomly generated flowsheet-like graphs of similar size and edge density. The height of the blue column shows the median similarity between the automatically digitized flowsheet and the expert-drawn ones. The overlaid boxplots on each column illustrate the distribution of pairwise similarities among experts, providing context for the performance of the automated digitization method described in this study. Boxed labels on the x-axis indicate the examples that are further detailed in Sections 3.1.1 and 3.1.2. Details of the textual process descriptions are provided in the supplementary information.†

View Article Online
DOI: 10.1039/D6DD00060F

two expert interpretations, focusing on the separation section. We see that all three representations fully agree on the central topology, including two absorption/scrubbing columns, a flash producing a recycle stream, and final purification by distillation. The text does not provide details on the number or arrangement of distillation columns. We observe that expert 2 applied their expertise by including three columns corresponding to the four components that are supposed to be separated, as stated in the text. The LLM did not expand on the distillation section because it is highly incentivized to stick to the explicit details of the text basis. Similarly, the LLM is constrained to adding distillation columns with exactly two outlets and cannot add additional product streams in a more unstructured manner, as expert 1 does.

The dichloromethane example shows that the LLM can faithfully digitize the information from the source material. By adhering to the steps described in the text, the LLM's decisions are readily interpretable. However, it would undoubtedly be advantageous if the digitization could fill in missing information, as expert 2 did. To remain interpretable and digitize with a core of domain knowledge and mechanistic understanding, future extensions could include process design heuristics and more physicochemical data to supplement the given text descriptions.

3.1.2 Example 2: LLM infers more from text than experts

For some descriptions, the median similarity of the LLM-generated flowsheet falls below the range of expert similarities. Upon inspection, these digitized flowsheets still accurately represent the underlying process information, but show notable differences to the experts' interpretations. Analyzing one of these cases, the production of aniline from the hydrogenation of nitrobenzene (#7 in Figure 6), yields insights into how the LLM and the experts structure and interpret the same information in different yet equally valid ways. The process description from Kahl *et al.* (2011)[39] is reproduced in the supplementary information.[†]

Figure 8 shows the three expert-drawn and the LLM-generated flowsheet for the aniline from nitrobenzene process. All flowsheets generally follow the text description: the reactants are mixed and reacted, steam is produced to cool down the reactor outlet, and the reaction mixture is separated. However, they differ in the details of the recycle stream, catalyst streams, and the number of separation steps.

Expert 3 provides the most detail on the separation sequence among the experts, displaying both a flash drum and a distillation column. In contrast, the LLM produces an even more detailed sequence, interpreting the described process as first removing the volatile hydrogen by flashing, then settling a two-phase liquid mixture of crude aniline and wastewater to remove the water, and then purifying the crude aniline by distillation. The inclusion of the settler extends beyond what is explicitly described in the text, suggesting that the LLM may draw on prior information from its training to augment the process description. Indeed, aniline and water possess a significant miscibility gap, thus confirming that the LLM's interpretation is reasonable and consistent. Similarly, the LLM inserts a compressor and a pump into reasonable yet not explicitly specified streams and combines two related waste streams. The aniline example shows that the automatically digitized flowsheet can be a valid representation of the process description, even when differences from the experts' interpretation lead to a below-average WL score. Interestingly, the evolutionary similarity lies within the range of expert values and therefore more closely matches the apparent similarity observed upon visual inspection of Figure 8. This similarity could motivate further exploration of adjusting and tuning a topology-aware similarity score to better reflect the particularities of flowsheet drawings. Furthermore, it would be interesting to explore how the LLM's training and prior knowledge in chemical engineering affect flowsheet generation.

Overall, we observe that the introduced digitization methods produce flowsheets that align with the underlying description and experts' interpretations. Therefore, the requirements are met for automatically inferring simulation models from the digitized data.

3.2 Quality of automatically generated Aspen Plus simulations

The test set's digitized flowsheets were automatically translated into converged Aspen Plus simulations using the "graph2simulation" methods described in Section 2.3. In the following, two representative examples are detailed, which exemplify the methodology, the necessary simplifications, and the conclusions that can be drawn from the digitization pipeline.

3.2.1 Example 3: Tracing the Aspen translation unit-by-unit

The flowsheet digitization and automated simulation were performed for the aniline production process from phenol feedstock based on the following process description:

"In the process phenol and fresh and recycle ammonia are vaporized separately (to prevent yield losses) and combined in the fixed bed amination reactor containing the silica – alumina catalyst. After the reaction at 370 °C and 1.7 MPa, the gas is cooled, partly condensed and the excess ammonia is recovered in a separation column, compressed and recycled. The condensation product is passed through a drying column to remove water and then through a finishing column to separate aniline from residual phenol and impurities in vacuum (less than 80 kPa). The phenol, containing some aniline (azeotropic mixture) is recycled." Kahl *et al.* (2011)[39]

Figure 9 shows the automatically generated Aspen Plus flowsheet by the "text2flowsheet" and "graph2simulation" pipelines for the aniline production process from phenol. Through tracing the text description, it can be determined that all essential processing steps have been successfully translated into the simulation.

First, the incoming feed streams of ammonia and phenol are mixed with their respective recycle streams and vaporized

View Article Online
DOI: 10.1039/D6DD00060F

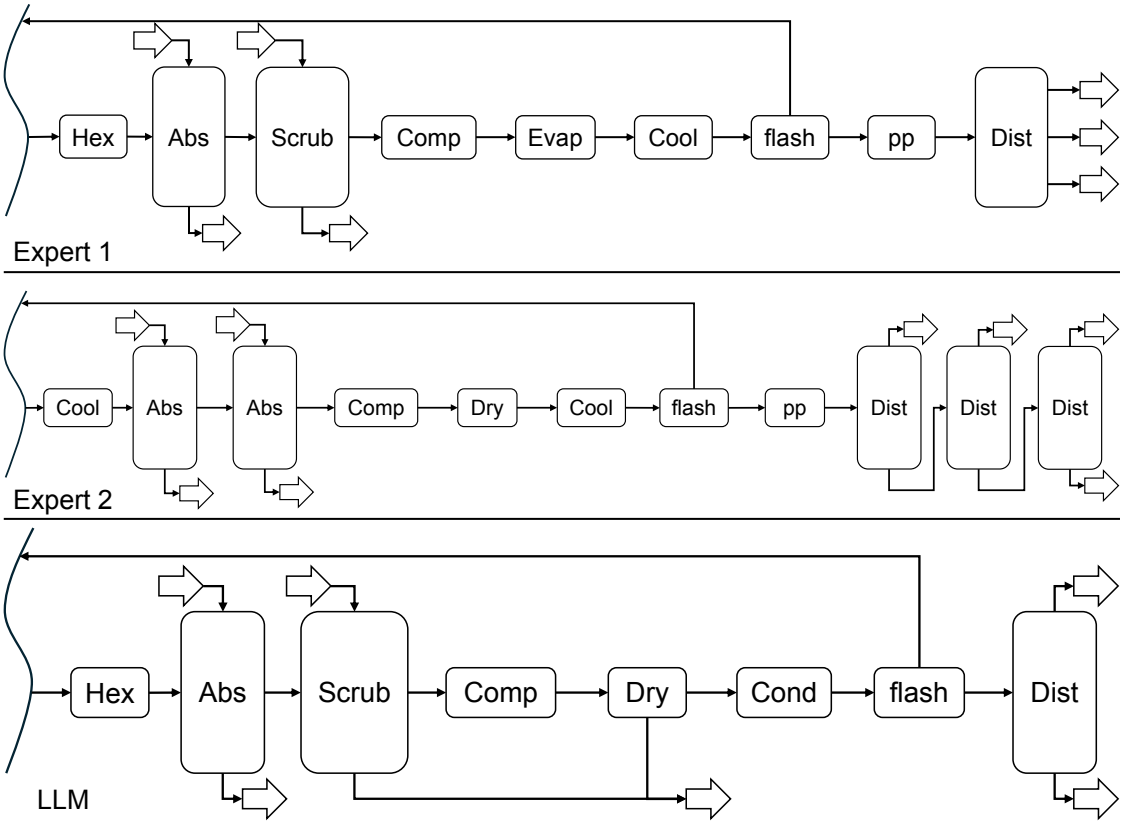

Fig. 7 Expert-drawn and LLM-generated separation sections of the dichloromethane production process based on the same text description. See supplementary information for unit abbreviations. Thick arrow symbols represent raw material and product streams, including auxiliaries.

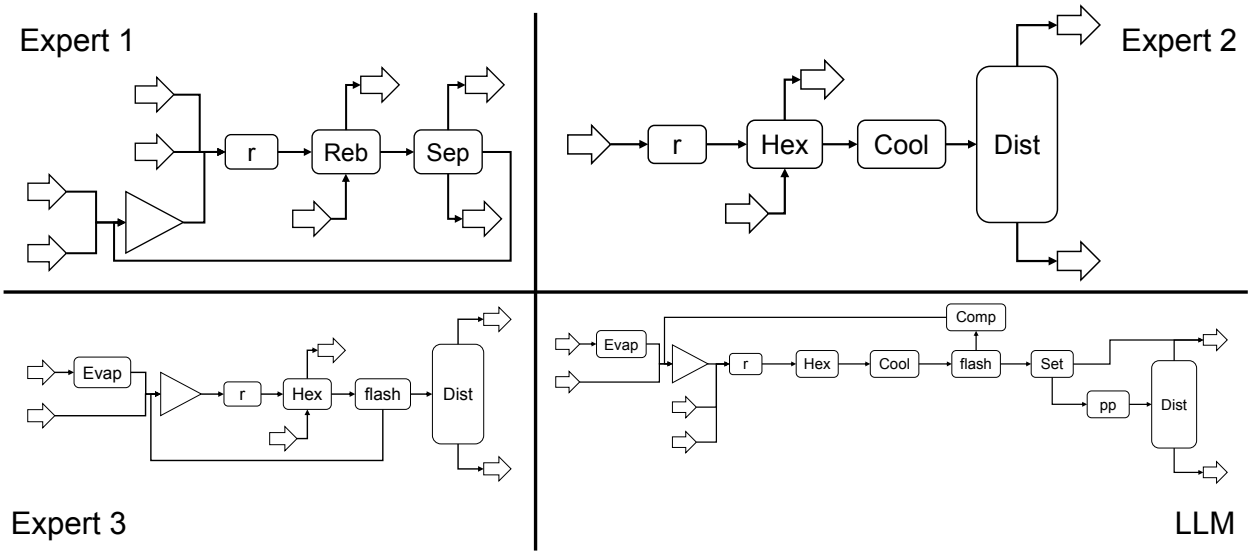

Fig. 8 Expert-drawn and LLM-generated flowsheets for the aniline from nitrobenzene process based on the same text description. See supplementary information for unit abbreviations. Broad arrow symbols represent raw material and product streams, while triangle symbols denote mixing units.

View Article Online
DOI: 10.1039/D6DD00060F

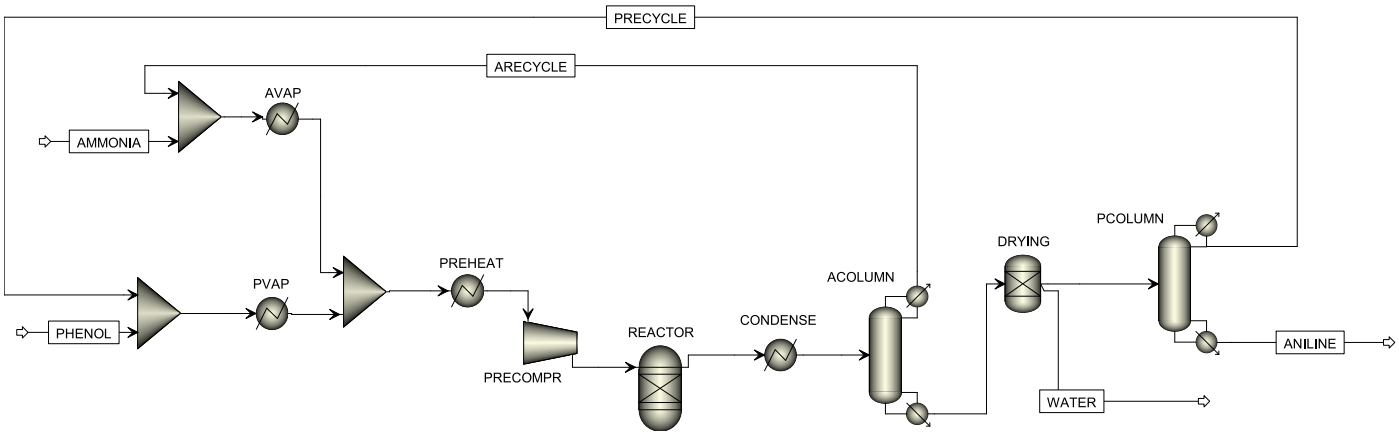

Fig. 9 Automatically generated Aspen flowsheet from the digitized process description of aniline production from phenol. Slightly cleaned and labeled for presentation purposes without changing unit or stream logic.

separately. The reactor operates above ambient temperature and pressure, so a preheater and compressor have been integrated rule-based to reach reaction conditions. The reactor is automatically modeled as a Gibbs' equilibrium reactor ("RGibbs") with the extracted reaction conditions from the process description and is therefore fully determined.

The separation sequence is modeled in the same three steps as outlined in the description. Ammonia is removed from the condensed reaction mixture using the rigorous "Radfrac" distillation model and recycled. In the standard configuration, "Radfrac" has five degrees of freedom: the number of stages, the position of the feed stage, the condenser pressure, and two out of nine possible operating specifications, from which we always choose the reflux and boilup ratios. Additionally, the condenser can be operated to condense only partially, so that the top product exiting the condenser remains in the vapor phase. For the feed-stage position, the middle stage is always selected, an artificial constraint that could be relaxed in future implementations. Black-box optimization is performed after the "ACOLUMN" is added to the simulation, because the text description does not provide sufficient information for any of the model parameters. For the ammonia separation column, 6 stages, a reflux ratio of 7.9, a boilup ratio of 2.7, and a condenser pressure of 5.5 bar were determined. The top ammonia product exits as vapor from a partial condenser. The determined column configuration mirrors design decisions made for comparable, expert-generated processes, e.g., regarding the elevated pressure level.[40,41]

The final separation step to purify aniline is modeled as a "Radfrac" column as well. The black-box optimization is run with one fewer degree of freedom, because the text specifies a value for the column pressure. As the text indicates, some aniline remains in the phenol recycling stream due to azeotropic behavior. However, the aniline product stream of the simulation also contains a significant amount of phenol. Although no expected purity is given in the process description, this simulation result could indicate an underlying error.

Indeed, if we examine the vapor-liquid equilibrium of phenol and aniline at 80 kPa with the Aspen analysis tools, the described azeotrope cannot be found. The analysis of the VLE indicates a mismatch between the property model and reality and could explain the divergence between the process description and the simulation. Compared to similar expert-generated process designs, the distillation column should be configured to yield a pure aniline stream on top and an azeotropic mixture at the bottom for recycling.[42] This discrepancy illustrates that grounding the LLM-assisted digitization in mechanistic frameworks is fundamentally limited by the fidelity of the underlying physical models. In future work, discrepancies between text descriptions and rigorous property models could be systematically analyzed using the LLM. Furthermore, the LLM's broad internalized black-box knowledge could be leveraged to identify other inaccuracies in the property models.

This example of aniline production from phenol demonstrates that our methodology can transfer knowledge from the process description, via a digitized graph, into an Aspen Plus simulation. The transfer of the process topology is successful, and our heuristics and black-box optimization determine reasonable operating conditions for the unit operations, even if details warrant further expert attention or corrections to underlying property data.

3.2.2 Example 4: Systematic simplifications on complex flowsheet

We now present an example of the carbonylation of methanol to produce acetic acid, further illustrating the methodology and demonstrating how inaccuracies are systematically addressed in an automated manner. The underlying process description from Le Berre *et al.* (2014)[43] is reproduced in the supplementary information.†

With 2500 characters, the acetic acid process description is among the longest and most detailed of the test set. Figure 10 shows the automatically generated, converged Aspen simulation.

View Article Online
DOI: 10.1039/D6DD00060F

The simulation is successfully generated with the majority of central processing steps modeled rigorously, and three out of five described recycles connected without convergence issues. The reaction pressure determined by the black-box optimization is in line with industrially reported design decisions, while the reaction temperature is below the commonly established range, which could motivate the integration of more detailed reactor models that consider kinetics.[44]

The acetic acid example yields three noteworthy observations about the LLM's interpretation of the flowsheet topology. First, the expansion chamber is missing, because the LLM classifies it as "Expand", which is ontologically linked to a single-outlet pressure changer and not a flash-type operation. Integrating few-shot examples to clearly differentiate unit types with ambigious common naming could mitigate issues like this in the future. Secondly, the catalyst recycle is misconnected to the washing column as the text lacks detail on the catalyst treatment. Currently, the chemical augmentation procedures in the pipeline lack specific catalyst considerations, which could help alleviate catalyst-related misinterpretations in the future. Finally, two of the described recycles have not been closed, as marked in red in Figure 10. Of the 32 possible combinations of recycle closures across the five recycles in the process, the simulation shown in Figure 10 had the most closed recycles and converged without error. A further examination of increasing convergence success with even more recycling streams could be conducted by an expert or with the help of emerging simulation co-pilots.[45]

Some separation units were hierarchically simplified (see Figure 5) to ensure they accurately reflect the material splits described in the text. The central reason why some separation units were modeled as "Sep" units instead of flashes or distillation columns was that the inlet stream did not contain at least one of the key components to be separated. Indeed, the reactor model produced fewer side products than mentioned in the process description, which, in most cases, does not explicitly name them. Due to this insufficient information, the LLM could not identify the corresponding chemicals, so the desired splits could not be included. The simplified separation steps can thus only be corrected by providing external information to the LLM, e.g., by including more details of the catalyst, by-products, and waste streams.

The acetic acid example shows that more complex systems can also be successfully translated into Aspen Plus simulations. The remaining simplifications are primarily due to insufficient background information. Therefore, the simplifications could be addressed by supplementing the data with external sources, as we discuss in the following section. Overall, the simulation accurately captures all essential reaction and separation steps while transparently flagging potential errors, thereby laying the foundation for further analysis and optimization of the process.

3.3 Discussion of current limitations and potential extensions

The current implementation of the presented "text2flowsheet" and "graph2simulation" workflows yields accurate digital flowsheet representations of chemical process descriptions and successfully translates them into Aspen Plus simulations. Its breadth of application with respect to the most commonly employed unit operations is extensive, and 30 process descriptions of varying length and level of detail have already been digitized. While mismatches and challenges in setting up the simulation can arise, they are addressed systematically to ensure a working, useful output. The discussed examples from the test set show that the digitization and simulation procedures are topologically accurate and provide converged simulations as starting points for further process analysis and expert-driven optimization.

A significant challenge throughout the workflow is the propagation of errors from LLM-based digitization to the final Aspen simulation. If unit operations, their interconnections, or their chemicals are digitized incorrectly or not at all, simulations need to resort to broad simplifications. The workflow already incorporates logic-based and thermodynamic rules that aim to prevent or mitigate errors, e.g., by prompting the LLM step-by-step with concrete tasks and background knowledge. This approach could be further systematized by applying flowsheet construction rules derived more generally from domain knowledge, such as in Schulze Balhorn *et al.* (2025).[46] Another promising approach would be to automatically extract these rules from validated digitized flowsheets, e.g., along the lines of graph grammar induction.[47]

The sequential design of the overall workflow, i.e., first digitizing the flowsheet and then simulating it, makes it particularly difficult to address issues that only become apparent after simulation, such as when the calculated reactor outlet does not match expectations from the process description. An integrated digitization procedure that includes simulation at every step of information extraction could be envisaged, but should be weighed against computational demands and complexity. Further uncertainty quantification could be achieved by simulating digitization errors and formally propagating their effects through the simulation results.

Extensions to this workflow in different directions can be conceptualized. Regarding process types, the "text2flowsheet" digitization should be applicable to batch processes without major additions. In contrast, the automated simulation pipeline ("graph2simulation") would require substantial adjustments to accommodate the significant modeling differences between continuous and batch processes. Furthermore, the 30 test processes all involved high-volume organic chemicals. Extensions to inorganic and fine chemicals should consider different heuristics that guide the automated process design. The modularity of the presented workflows would enable the integration of simulation environments tailored to different process modes,

View Article Online
DOI: 10.1039/D6DD00060F

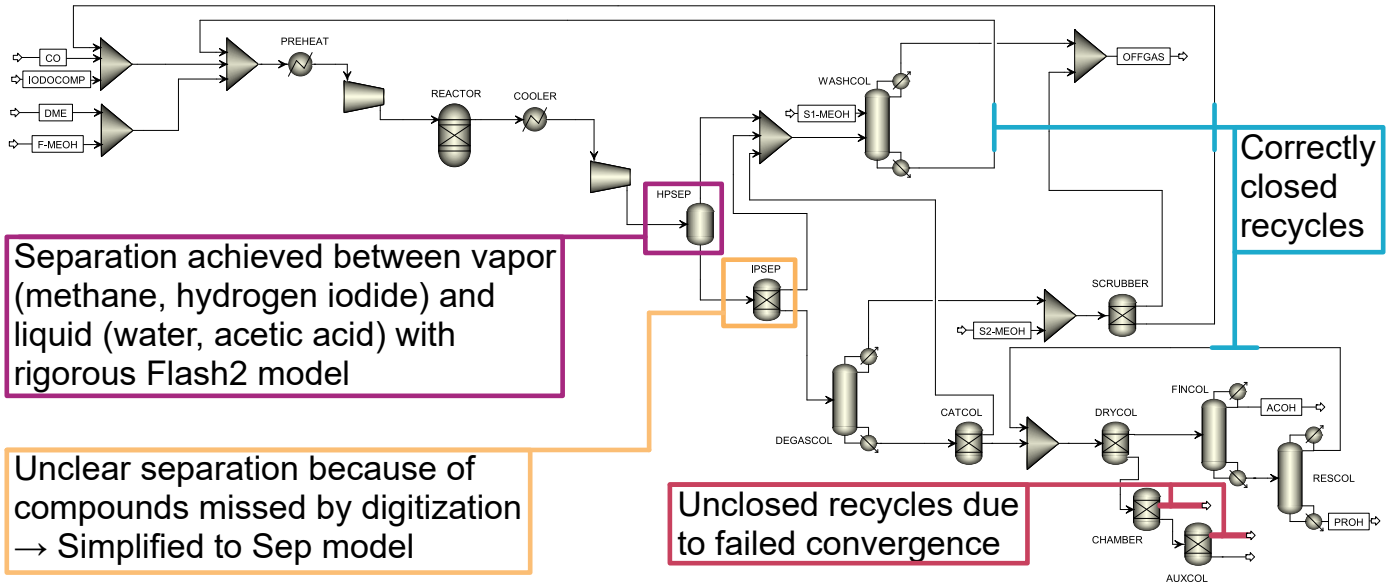

Fig. 10 Automatically generated Aspen flowsheet from the digitized process description of acetic acid production from methanol. Slightly cleaned, annotated, and labeled for presentation purposes without changing unit or stream logic.

industry-specific practices, and chemical families. Regarding further process design tasks, applying heat integration should be straightforward, and established methods for automatically generating preliminary heat-integrated designs could be used, for example, from Aguitoni *et al.* (2018)[48] or Aspen's built-in methods. Additionally, the automatic addition of control structures from PFDs to P&IDs has been addressed in the literature[49] and could be appended to our workflows.

The enumeration of recycle closures represents a significant combinatorial bottleneck of our methodology, as the number of recycle combinations grows exponentially with the number of recycles. Thus, for highly integrated processes, the automated closing of recycles can represent substantial or even intractable computational effort. Although the "graph2simulation" pipeline can evaluate multiple combinations in parallel, further research is required to achieve more efficient recycle closure. Recently, substantial advancements have been made with ML-assisted recycling-closure algorithms,[45] which we aim to integrate in future work.

Further improvements of the methodology should consider two perspectives: Firstly, the LLM could be provided with substantially more background knowledge about process design. Currently, the guiding design principle of this workflow is to focus LLM usage on tasks where LLMs are known to perform very well out of the box, i.e., text understanding and information extraction, while processing physics-guided heuristics and process design rules outside the LLM. With more background knowledge, the LLM's output freedom could be expanded from the very restricted, step-by-step procedure described in this work to a more direct translation into digitized formats. This extension would likely require larger LLMs with more advanced reasoning capabilities, thereby incurring additional costs. Nevertheless, we recommend computing the underlying thermodynamics separately from the LLM whenever reliable physics models are available, given the variable and black-box nature of LLMs.

Secondly, the black-box optimization with respect to uncertain parameters can be revisited by formulating more specific objective functions for each unit or by optimizing the flowsheet in a simultaneous rather than a sequential-modular mode. All possible extensions based on simulating the digitized flowsheet depend on the choice of simulation software. We chose Aspen Plus for its widespread trust and adoption within the chemical engineering community, as well as its comprehensive suite of unit models. However, other simulation software could be explored. The focus should lie on software that offers better integration with standardized flowsheet formats, external flowsheet manipulation and optimization, and open-source standards. To provide a more user-friendly interface, recent advances in LLM-based interaction with flowsheet simulation software could be integrated.[50]

The main limitation of digitizing flowsheets from textual descriptions is the incomplete information about the chemicals involved, processing steps, and operating conditions. We are already augmenting the information to generate a meaningful Aspen simulation using heuristics, thermodynamic calculations, and stochastic optimization. However, the highly case-dependent uncertainty of the text descriptions remains the most significant source of errors. From the test set of process descriptions, we observe that descriptions lacking information about separation steps and their sequence lead experts to employ subjective strategies to fill the gaps. Similarly, the LLM sometimes expands on information using prior training knowledge, thereby chal-

View Article Online
DOI: 10.1039/D6DD00060F

lenging the interpretability of the digitization procedure and the resulting models. Therefore, it would be sensible to first explicitly augment the process descriptions using the LLM and have possible additions checked by an expert-in-the-loop before deploying the digitization pipeline. Following our principle of using established knowledge whenever possible, any text augmentation should be rooted in mechanistic models. From surveying our test set of descriptions, we suggest the following sources as starting points:

- more physicochemical data, in particular liquid-liquid and solid-liquid phase equilibria,
- identification of heteroazeotropes and corresponding application of established separation strategies,
- identification of possible side reactions and products not (explicitly) mentioned in the text descriptions, through the use of retrosynthesis tools,[51]
- hazard information to determine if toxic or corrosive chemicals should be treated in special ways, and
- cost data to determine if recycles and heat integration are required for realistic process performance.

Integrating this knowledge through established design heuristics[52] could enhance the body of knowledge in the process descriptions and make subsequent digitization and simulation both easier and more accurate. Vice versa, existing process design methodologies could be enhanced with the digitized process knowledge from text.

Conclusions

Our results show that natural language descriptions of chemical processes can be faithfully digitized using medium-sized LLMs, yielding flowsheets that are topologically indistinguishable from expert-drawn ones. Automatically assembling and configuring corresponding Aspen Plus instances can provide practitioners with converged flowsheet simulations that serve as information-rich, digital objects for further process analysis and optimization.

By equipping the LLM with a list of standardized unit operations and domain-knowledge-based instructions, it extracts relevant and accurate information, enabling a meaningful model of the described process. We organized the LLM's tasks into a step-by-step algorithm and systematically restricted its output format. These constraints allow a sequential, structured, and accurate build-up of the process topology. Furthermore, the topological arrangement and augmentation with chemical and operational information are enhanced by performing thermodynamic property calculations external to the LLM. Nonetheless, the successful digitization of process descriptions into flowsheet graphs depends on the availability of suitable information in the descriptions, which, in future work, could be augmented by adding more chemical, thermodynamic, and economic information and applying process design heuristics. In the absence of concrete details, the LLM sometimes adds reasonable processing steps beyond the source material. Further investigation is warranted concerning the potential adversarial effects of such hallucinations.

We have validated the automatically digitized flowsheets by systematically comparing them to 101 flowsheets drawn by real-world experts. From the pairwise similarity among experts interpreting the same process text, we have derived process-specific target ranges of digitization success. Across 30 test descriptions, we have found that most LLM-generated flowsheets are highly similar to their expert-drawn counterparts. Remaining differences usually arise from ambiguous source information and individual mitigation strategies of experts and LLMs based on their respective expertise or prior training data. Generally, we have found the LLM to make more confident choices compared to experts with respect to unclear unit operations and connections, which could be fine-tuned by revising the underlying prompts.

All essential design and operating information from process digitization is automatically carried over to an Aspen simulation. Unspecified design and operational parameters are estimated using black-box optimization, successively aligning the behavior of the individual unit operations with the reaction and separation tasks implied by the process description. Performing this optimization in a multi-objective manner yields reasonable process specifications through simultaneously optimizing for energy or solvent demands. However, minor errors introduced during digitization can necessitate significant simplifications to achieve convergence in the Aspen simulation. These simplifications are automatically applied in a transparent and hierarchical manner, specific to each unit operation type.

Overall, the introduced methods enable large-scale automation of the digitization of chemical process information from natural language sources. Future extensions could address the simulation of non-continuous processes, the interpretable integration of additional chemical and process engineering knowledge, and the use of multi-modal source data. With this development, we aim to contribute to the collection and standardization of chemical engineering knowledge, thereby enabling interactive, machine-readable repositories for the economic and environmental analysis of chemical production processes.

Author contributions

Jan-Frederic Laub: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data Curation, Writing - Original Draft, Writing - Review & Editing, Visualization; **Luca Bosetti**: Conceptualization, Methodology, Writing - Review & Editing, Supervision; **André Bardow**: Conceptualization, Methodology, Resources, Writing - Review & Editing, Supervision, Project administration, Funding acquisition

Conflicts of interest

The authors declare the following financial interests and personal relationships that could be considered as potential competing in-

View Article Online
DOI: 10.1039/D6DD00060F

terests: A. B. has served on review committees for research and development at ExxonMobil and TotalEnergies, companies active in both oil and gas and chemical production. A. B. holds ownership interests in firms that provide services to industry, some of which may operate in the chemical industry. The remaining authors declare no known competing financial interests or personal relationships that could have influenced the work reported in this paper.

Data availability

Data and code for this article, including the digitization pipelines "text2flowsheet" and "graph2simulation", the validation set of expert-drawn flowsheets, and a demo process along with its LLM logs, are available at Zenodo at `https://doi.org/10.5281/zenodo.19910216`.

Acknowledgements

J.-F. L. and A. B. acknowledge funding by NCCR Catalysis, a National Centre of Competence in Research funded by the Swiss National Science Foundation, grant number 225147.

The authors thank all experts who kindly contributed flowsheet drawings.

The authors thank the scientific and organizational committees of the DECHEMA Annual Meeting of Process Engineering and Materials Technology 2025 for their support in collecting expert-drawn flowsheets.

References

1 A. G. Parvatker and M. J. Eckelman, *ACS Sustainable Chem. Eng.*, 2018, **7**, 350–367.
2 J.-M. Commenge and A. Piña-Martinez, *Comput. Chem. Eng.*, 2026, **204**, 109416.
3 Y. W. Son, J. H. Pak, C. Kim and J. M. Lee, *Comput. Chem. Eng.*, 2026, **205**, 109431.
4 A. M. Schweidtmann, *Sys. Cont. Trans.*, 2024, **3**, 84–91.
5 M. F. Theisen, K. N. Flores, L. Schulze Balhorn and A. M. Schweidtmann, *Digital Chem. Eng.*, 2023, **6**, 100072.
6 S. Gowaikar, S. Iyengar, S. Segal and S. Kalyanaraman, AAAI 2025 Workshop on AI to Accelerate Science and Engineering, Philadelphia, USA, 2025.
7 G. Tolksdorf, D. B. Cameron and M. Theißen, *Chem. Ing. Tech.*, 2025, **97**, 1065–1069.
8 X. Tian, W. Du, S. Yang, H. Hu, H. Xin, S. Qu and K. Ye, *Preprint on arXiv*, 2026.
9 S. S. Srinivas, S. Gupta and V. Runkana, *Preprint on arXiv*, 2025.
10 S. Rupprecht, Y. Hounat, M. Kumar, G. Lastrucci and A. M. Schweidtmann, *Sys. Cont. Trans.*, 2025, **4**, 1706–1711.
11 E. Arroyo, M. Hoernicke, P. Rodríguez and A. Fay, *Comput. Chem. Eng.*, 2016, **92**, 112–132.
12 S. Sierla, L. Sorsamäki, M. Azangoo, A. Villberg, E. Hytönen and V. Vyatkin, *Appl. Sci.*, 2020, **10**, 6959.
13 *Aspen Plus Version 11*, Aspen Technology, Inc., Bedford, USA, 2019.
14 P. Rehner, G. Bauer and J. Gross, *Ind. Eng. Chem. Res.*, 2023, **62**, 5347–5357.
15 T. Esper, G. Bauer, P. Rehner and J. Gross, *Ind. Eng. Chem. Res.*, 2023, **62**, 15300–15310.
16 B. Winter, P. Rehner, T. Esper, J. Schilling and A. Bardow, *Digital Discovery*, 2025, **4**, 1142–1157.
17 O. Khattab, A. Singhvi, P. Maheshwari, Z. Zhang, K. Santhanam, S. Vardhamanan, S. Haq, A. Sharma, T. T. Joshi, H. Moazam, H. Miller, M. Zaharia and C. Potts, *DSPy: compiling declarative language model calls into self-improving pipelines*, 2024.
18 W. Ning, M. Li, J. R. Reimers and R. Kobayashi, *Digital Discovery*, 2026, **5**, 698–715.
19 S. Colvin, *Pydantic: Data validation and settings management using Python type annotations*, 2023.
20 G. Vogel, E. Hirtreiter, L. Schulze Balhorn and A. M. Schweidtmann, *Opt. Eng.*, 2023, **24**, 2911–2933.
21 A. A. Hagberg, D. A. Schult and P. J. Swart, in *Proceedings of the 7th Python in Science Conference*, 2008.
22 *GPT-5-mini*, OpenAI, Inc., San Francisco, USA, 2025.
23 *GPT-OSS*, OpenAI, Inc., San Francisco, USA, 2025.
24 J. Steimel, *pyflowsheet: A Python package for drawing process flow diagrams*, 2020.
25 D. B. Cameron, W. Otten, H. Temmen, M. Hole and G. Tolksdorf, *Comput. Chem. Eng.*, 2024, **182**, 108564.
26 Y. Zhang, S. A. Khan, A. Mahmud, H. Yang, A. Lavin, M. Levin, J. Frey, J. Dunnmon, J. Evans, A. Bundy, S. Dzeroski, J. Tegner and H. Zenil, *npj Artif. Intell.*, 2025, **1**, 14.
27 A. Mirza, N. Alampara, S. Kunchapu, M. Ríos-García, B. Emoekabu, A. Krishnan, T. Gupta, M. Schilling-Wilhelmi, M. Okereke, A. Aneesh, M. Asgari, J. Eberhardt, A. M. Elahi, H. M. Elbeheiry, M. V. Gil, C. Glaubitz, M. Greiner, C. T. Holick, T. Hoffmann, A. Ibrahim, L. C. Klepsch, Y. Köster, F. A. Kreth, J. Meyer, S. Miret, J. M. Peschel, M. Ringleb, N. C. Roesner, J. Schreiber, U. S. Schubert, L. M. Stafast, A. D. D. Wonanke, M. Pieler, P. Schwaller and K. M. Jablonka, *Nat. Chem.*, 2025, **17**, 1027–1034.
28 J. Dagdelen, A. Dunn, S. Lee, N. Walker, A. S. Rosen, G. Ceder, K. A. Persson and A. Jain, *Nat. Commun.*, 2024, **15**, 1418.
29 T. Neveux, *Chem. Eng. Sci.*, 2018, **185**, 209–221.
30 N. Shervashidze, P. Schweitzer, E. J. van Leeuwen, K. Mehlhorn and K. M. Borgwardt, *J. Mach. Learn. Res.*, 2011, **12**, 2539–2561.
31 K. Borgwardt, E. Ghisu, F. Llinares-López, L. O'Bray and B. Rieck, *Found. Trends Mach. Learn.*, 2020, **13**, 531–712.
32 E. Martinez-Hernandez, *Digital Chem. Eng.*, 2023, **6**, 100075.
33 L. Stops, R. Leenhouts, Q. Gao and A. M. Schweidtmann, *AIChE J.*, 2022, **69**, e17938.

View Article Online
DOI: 10.1039/D6DD00060F

34 A. G. Parvatker and M. J. Eckelman, *ACS Sustainable Chem. Eng.*, 2020, **8**, 8519–8536.
35 P. Bennet, C. Doerr, A. Moreau, J. Rapin, F. Teytaud and O. Teytaud, *ACM SIGEVOlution*, 2021, **14**, 8–15.
36 *Ullmann's Encyclopedia of Industrial Chemistry*, John Wiley & Sons, Ltd, Hoboken, USA.
37 *Process Economics Program (PEP) Yearbook*, IHS Markit, London, UK.
38 M. Rossberg, W. Lendle, G. Pfleiderer, A. Tögel, T. R. Torkelson and K. K. Beutel, in *Ullmann's Encyclopedia of Industrial Chemistry*, John Wiley & Sons, Ltd, Hoboken, USA, 2011.
39 T. Kahl, K.-W. Schröder, F. R. Lawrence, W. J. Marshall, H. Höke and R. Jäckh, in *Ullmann's Encyclopedia of Industrial Chemistry*, John Wiley & Sons, Ltd, Hoboken, USA, 2011.
40 S. Bugosen, I. D. Mantilla and F. Tarazona-Vasquez, *Heliyon*, 2020, **6**, e05778.
41 Q. Ye, J. Zeng, Y. Li, P. Yuan and F. Wang, *Energies*, 2022, **15**, 9258.
42 C. Y. Choo, *Reduced pressure distillation process for recovering aniline from phenolaniline mixtures*, US Patent 3682782A, 1972.
43 C. Le Berre, P. Serp, P. Kalck and G. P. Torrence, in *Ullmann's Encyclopedia of Industrial Chemistry*, John Wiley & Sons, Ltd, Hoboken, USA, 2014.
44 G. J. Sunley and D. J. Watson, *Catalysis Today*, 2000, **58**, 293–307.
45 D. Jakobs, L. F. dos Santos and G. Guillén-Gosálbez, *Preprint on Research Square*, 2025.
46 L. Schulze Balhorn, N. Seijsener, K. Dao, M. Kim, D. P. Goldstein, G. H. M. Driessen and A. M. Schweidtmann, *Preprint on arXiv*, 2025.
47 I. Jonyer, L. B. Holder and D. J. Cook, *Int. J. Artif. Intell. Tools*, 2004, **13**, 65–79.
48 M. C. Aguitoni, L. V. Pavão, P. H. Siqueira, L. Jiménez and M. A. d. S. S. Ravagnani, *Comput. Chem. Eng.*, 2018, **117**, 82–96.
49 L. Schulze Balhorn, K. Degens and A. M. Schweidtmann, *Comput. Chem. Eng.*, 2025, **199**, 109121.
50 J. Liang, N. Groll and G. Sin, *Preprint on arXiv*, 2026.
51 Y. Sun and N. V. Sahinidis, *Curr. Opin. Chem. Eng.*, 2022, **35**, 100721.
52 A. C. Dimian, C. S. Bildea and A. A. Kiss, in *Integrated Design and Simulation of Chemical Processes*, Elsevier, Amsterdam, Netherlands, 2014.

View Article Online
DOI: 10.1039/D6DD00060F

Data and code for this article, including the digitization pipelines "text2flowsheet" and "graph2simulation", the validation set of expert-drawn flowsheets, and a demo process along with its LLM logs, are available at Zenodo at https://doi.org/10.5281/zenodo.19910216.