Adaptive Optimization for Disassembly Sequence Planning under Uncertain Component Conditions

Wael El Morabit[a], Mohammed-Amine Abdous[a,b], Flavien Lucas[a]

[a] *IMT Nord Europe, Institut Mines-Télécom, Univ. Lille, Center for Digital Systems, F-59000 Lille, France*
wael.elmorabit@imt-nord-europe.fr
[b] *Presenting author*

Abstract

Although disassembly supports circular production, planning is difficult because the true state of the parts is often unknown, and the precedence rules create a large search space. Unknown states include hidden wear, seized or missing fasteners, misalignment, and small damage. We build an adaptive optimization framework for disassembly sequence planning. Products are represented as directed acyclic graphs with cost, profit, and time at the component level for both complete and selective disassembly. A mixed integer linear program serves as a reference. A hybrid method that combines Greedy Randomized Adaptive Search and Variable Neighborhood Descent then produces sequences at scale. On 246 benchmark instances with 10 to 1,000 components, our method remains within 3% of the mixed-integer reference while reducing compute time by more than 90%. To cope with uncertainty during execution, we add a fuzzy logic layer that reads process signals such as force or torque spikes, repeated vision failure, timeouts, and slip or no grip, and then chooses one of four actions. It can bypass a noncritical node, change the tool or method, and retry, apply controlled destruction when the loss in value is acceptable, or trigger local re-planning of the remaining feasible successors. The chosen response updates the graph and adapts the sequence without re-solving the full model. For production planning, the framework can raise net recovery profit, lower operating cost through less rework and downtime, and improve plan reliability. It also gives a quantitative basis to add AI-based perception and inspection and assisted decision support, and translates into actionable policies for managers that balance cost against recovery profit.

Keywords: Disassembly sequence planning, Adaptive re-planning, Hybrid metaheuristics, Circular production and remanufacturing, Cost–profit optimization.

---

1. Introduction

Circular production relies on the ability to recover components and materials from returned products. Disassembly enables this recovery by making reuse, repair, remanufacturing, and recycling feasible (Sundin et al., 2012). From a production economics viewpoint, the main challenge is to secure a positive net recovery profit in spite of uncertainty. In many products, the true condition of joints and fasteners is not fully observable before execution. Hidden wear, missing or seized fasteners, misalignment, and minor damage are often discovered only when an operation is attempted. These events create delays, retries, and occasional loss of recoverable value. They also increase variability in cycle time and introduce disruptions that threaten flow in disassembly cells.

Disassembly sequence planning addresses the selection of an operation order that respects precedence constraints while balancing value recovery, cost, and time (Lambert, 2003). When the objective is economic, a sequence is attractive if it recovers high value components early, avoids unnecessary operations, and limits exposure to costly steps. This is particularly important in selective disassembly, where only a target set of components must be recovered and other steps are performed only if they are required to access the targets. However, sequences computed under nominal assumptions can become suboptimal or infeasible when execution reveals unexpected component states. In practice, operators respond through a small set of interventions, such as skipping a noncritical step when feasible, changing the tool or method and retrying, applying controlled destruction to remove a blockage at an acceptable loss of value, or replanning the remaining operations. Each intervention has a direct economic impact, through added time, added cost, and potential loss of recovery value.

This paper studies disassembly sequence planning with reactive decision support that preserves economic performance under disruptions. We represent a product as a directed acyclic graph in which nodes are disassembly operations and arcs encode precedence constraints. Each operation carries processing time, operating cost, and recovery value parameters. The planning task is to compute a precedence feasible sequence that maximises net recovery profit for either complete disassembly or selective recovery of a target set. To support execution, we combine offline optimisation with an online response policy that uses simple shop floor symptoms, such as abnormal force or torque, repeated perception failures, timeouts, and loss of grip, to recommend an intervention that keeps the plan feasible and protects value.

The contribution is threefold. First, we provide an economic problem formulation for complete and selective disassembly with component level time, cost, and recovery value parameters. Second, we propose a scalable optimisation approach where a mixed integer linear program serves as a reference on moderate instances and a hybrid planner based on Greedy Randomized Adaptive Search Procedure with variable neighbourhood descent produces high quality sequences for large graphs (Bentaha et al., 2015a; Guo et al., 2021). Third, we introduce a compact reactive response layer, implemented with an interpretable fuzzy inference mechanism (Mamdani, 1977), that selects among bypass, tool change with retry, controlled destruction, and local replanning on the residual graph. Computational experiments quantify both nominal planning performance and the economic protection offered by reactive responses under simulated disruptions.

The remainder of the paper is organised as follows. Section 2 positions the work in the literature on disassembly planning, uncertainty handling, and reactive decision support. Section **??** defines the planning model and the economic objective. Section **??** presents the reference model, the scalable hybrid planner, and the reactive response policy. Section 4 reports benchmark results and disruption simulations. Section 5 summarises managerial insights and practical tuning guidance. Section 6 concludes and outlines directions for future research.

2. Related work and positioning

Disassembly sequence planning has long been studied as a combinatorial decision problem where precedence constraints restrict feasible operation orders and the objective reflects time, cost, and value recovery (Lambert, 2003). For production economics, the practical question is not only which sequence is feasible, but which sequence maximises net recovery profit when product condition is uncertain and disruptions occur during execution. Recent reviews summarise this shift toward economically grounded objectives, scalable solution methods, and uncertainty aware decision support (Guo et al., 2021; Streibel et al., 2025).

A first modelling stream focuses on how to represent product structure and disassembly alternatives. Graph models remain the dominant choice because they encode feasibility and accessibility in a compact form. Directed acyclic graphs are widely used when precedence relations are sufficient, while richer representations such as AND OR graphs appear when optional paths and alternative removals must be represented (Streibel et al., 2025). These representations lead to large search spaces and make optimal sequence selection computationally difficult for realistic products.

A second stream focuses on optimisation and scalability. Exact approaches, including mixed integer formulations, are often used as reference models for small and moderate instances, but they face rapid growth in computation time and memory demands as product size increases (Bentaha et al., 2015a). For larger graphs, practitioners and researchers typically rely on heuristics and metaheuristics that combine a constructive phase with local improvement. Hybrid methods are particularly attractive because they can exploit problem structure, preserve feasibility efficiently, and deliver strong solutions with short run time, which matters when planning must be repeated across many product variants (Guo et al., 2021).

A third stream addresses uncertainty. In disassembly, uncertainty affects processing times, success probabilities, and realised recovery value, since component condition becomes observable only through action. Stochastic formulations, including chance constrained models, aim to ensure feasibility or performance with a prescribed probability when key parameters are random (Tian et al., 2013). Fuzzy approaches capture qualitative knowledge, for example difficulty, condition, or risk levels, and are useful when data is limited or when linguistic assessments are common on the shop floor (Tian et al., 2018). Selective disassembly strengthens the need for uncertainty aware planning because the value of information and the value of flexibility depend on which targets are requested and which operations can be avoided (Kim et al., 2018). Related work on disassembly line planning under uncertain task times also highlights the importance of robustness and coordination across operations when variability is high (Bentaha et al., 2013, 2015b).

A fourth stream focuses on reactive decision support during execution. Reactive disassembly planning considers how to adapt sequences and local operations when a planned step becomes difficult or fails. Data driven approaches have been proposed for this dynamic setting, especially when sensor signals can indicate emerging problems (Streibel et al., 2024). In parallel, fuzzy inference systems have been used in manufacturing to convert symptoms and context into interpretable actions when crisp thresholds are hard to calibrate (Mamdani, 1977). For disassembly, this is consistent with industrial practice, where interventions are typically limited to a small set of actions such as retrying with a different tool, skipping noncritical steps when feasible, applying controlled destruction for low value obstacles, or replanning the remaining tasks.

This paper connects these streams through an economic decision support view. We use a directed acyclic graph representation with operation level time, cost, and recovery value

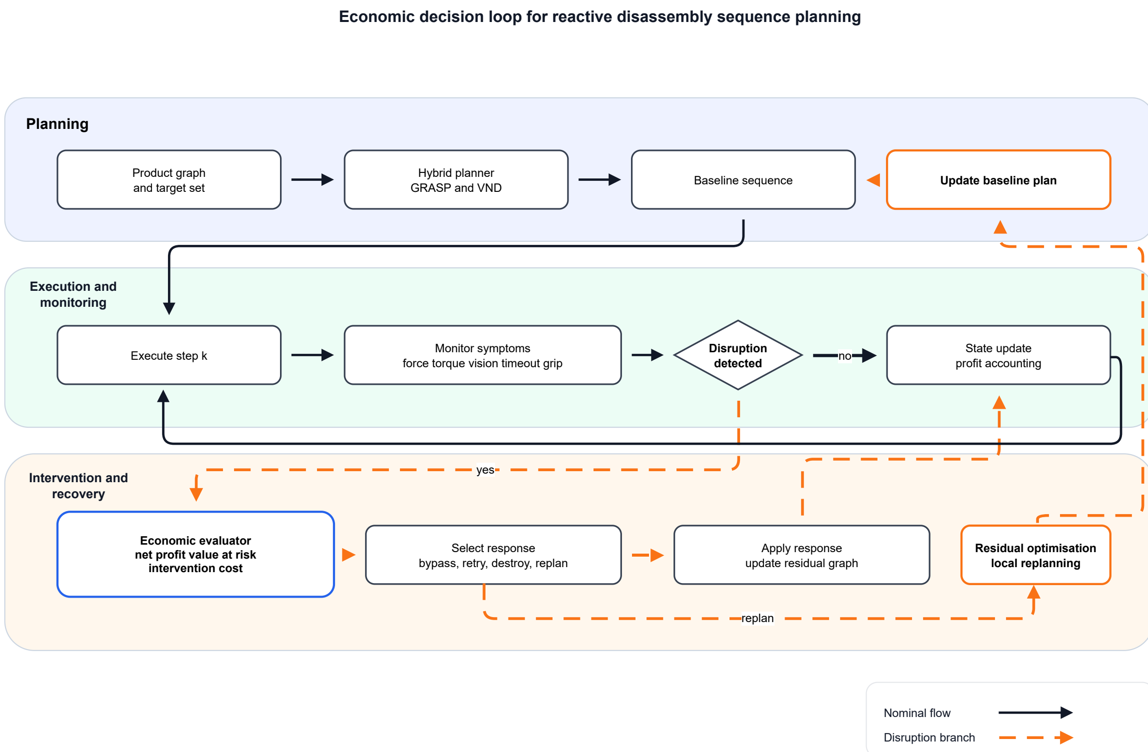

Figure 1: Economic decision loop for reactive disassembly sequence planning. A baseline sequence is computed offline using the hybrid planner. During execution, process symptoms such as force, torque, vision failures, timeouts, and loss of grip trigger a disruption branch. An economic evaluator weighs value at risk against intervention cost to select a response among bypass, retry after tool or method change, controlled destruction, and local replanning on the residual graph. The selected response updates the residual graph and, when needed, refreshes the remaining sequence.

parameters for both complete and selective disassembly. We combine a reference optimisation model for moderate instances with a scalable hybrid planner for large graphs, and we add a compact reactive policy that selects among bypass, tool change with retry, controlled destruction, and local replanning on the remaining graph. The aim is to quantify not only nominal solution quality, but also how reactive actions protect feasibility and net recovery profit under execution disruptions.

3. Reactive execution and decision loop

Figure 1 summarises the execution logic that connects the baseline sequence with online adaptation. The purpose is to keep target recovery feasible while limiting avoidable cost and value loss under uncertain component states. The controller maintains a residual precedence representation that reflects which operations are completed and which operations remain executable.

3.1. Baseline execution

Before execution, the hybrid planner computes a precedence-feasible sequence σ for the selected mode, either complete disassembly or selective recovery. During execution, the controller attempts operations in the order prescribed by σ. After each successful step, the residual state is updated and the controller advances to the next operation in σ that is currently executable. When several candidates become executable simultaneously, preserving the original order in σ reduces plan instability and limits unnecessary changes for operators. If the next operations in σ are no longer executable while targets are still pending, the controller triggers residual replanning as described in Section 3.4.

3.2. From station signals to a compact symptom description

Uncertainty becomes observable through station behaviour. In the experiments, we consider force and torque measurements, repeated perception failures, timeouts, and loss of grip. These raw signals are converted into a compact symptom description that indicates whether the current situation resembles overload, blocked access, perception instability, or grasp instability. This abstraction is consistent with the need for structured inputs in reactive disassembly decision support (Streibel et al., 2024) and avoids hard-coded, product-specific thresholds.

3.3. Fuzzy response selection

When a symptom is detected, the controller enters the intervention branch in Figure 1. The decision is driven by three inputs: the estimated failure probability p_{fail}, the measured force and torque level, and a context score that captures the availability of alternatives such as tools, methods, or feasible paths. We implement the mapping from these inputs to actions using a Mamdani fuzzy inference system (Mamdani, 1977). The output universe corresponds to four actions: bypass, change tool and retry, controlled destruction, and residual replanning.

The rule base contains nine rules that cover the operational patterns we expect in a disassembly cell. When alternatives are available, the controller prefers a tool change and retry (R0). When the estimated failure probability is low and loads are low, a tool change remains the preferred choice because it preserves recovery value at limited effort (R1–R2). When no alternative is available and the failure probability is medium under low or medium loads, bypass is preferred if the current operation is not required for target reachability (R3–R4). When loads are high but the failure probability is low, bypass is again preferred because the situation often corresponds to a locally difficult operation where progress can be maintained by taking another feasible step (R5). When the failure probability becomes high, the controller triggers residual replanning to recompute the best remaining path (R6). When failure probability is high and loads are high, controlled destruction is activated if it is economically acceptable, as it may be the only way to restore reachability (R7). Finally, when the failure probability is medium under high loads, destruction can also be selected as a preventive action to avoid repeated costly attempts (R8).

The fuzzy decision pipeline follows the standard sequence of steps: input collection, fuzzification, rule activation, aggregation of action preferences, defuzzification, and selection of a single action at each decision point (Mamdani, 1977). This design keeps the policy interpretable and tunable, which supports managerial acceptance and iterative calibration from operational data.

3.4. Residual update and local replanning

After an action is applied, the controller updates the residual state and the remaining plan. Bypass removes the operation from the remaining plan and recomputes accessibility. Change tool and retry performs one additional attempt and updates the state only if the operation succeeds. Controlled destruction records completion with the corresponding destruction cost and with zero recovery value for the destroyed component. If residual replanning is selected, the planner is executed on the residual instance induced by the remaining operations and precedence constraints, producing a new tail sequence that replaces the unused portion of σ.

Restricting replanning to the residual instance ensures that completed operations are never revisited and that computation remains local, which limits plan volatility for operators and keeps response times compatible with cell-level decision support.

4. Computational study

This section evaluates the baseline planner and the reactive decision loop shown in Figure 1. We report nominal optimisation results, behaviour under disruptions, and an illustrative selective recovery case based on a gear pump instance.

4.1. Nominal planning benchmark

The baseline planner is evaluated on 246 generated precedence graphs. Instance size ranges from 10 to 1000 operations, and we consider both complete disassembly and selective recovery. Each operation carries a duration, a direct operating cost, and an expected recovery value, and the objective is net recovery profit as defined in Section **??**. For each instance, the hybrid planner is run to produce a precedence feasible sequence. On the subset of instances where the mixed integer reference can be solved to optimality, we also compute the relative profit gap and compare solution times.

4.2. Disruption simulations

To evaluate the decision loop in Figure 1, we run disruption simulations on 30 instances sampled from the benchmark, with sizes from 7 to 148 operations. For each instance, we define three scenarios, giving 90 runs. Scenarios are grouped by how strongly the disruption restricts the remaining feasible choices. In low severity scenarios, the affected operation is not required for target access, so bypass can preserve reachability. In medium severity scenarios, the disruption affects a useful operation and forces a change in the remaining path while keeping targets reachable. In high severity scenarios, the disruption affects a critical operation on every access path to at least one target, so recovery may require controlled destruction or residual replanning.

4.3. Illustrative case: selective recovery on a gear pump

We illustrate the mechanism on a selective recovery instance derived from a gear pump product. The instance contains 22 operations, denoted $G01$ to $G22$, and the target set is $\{G20\}$. The initial sequence returned by the baseline planner is

$$\sigma = (G03, G06, G04, G02, G05, G01, G08, G10, G14, G12, G16, G07, G09, G11, G13, G15, G17, G19, G18, G2$$

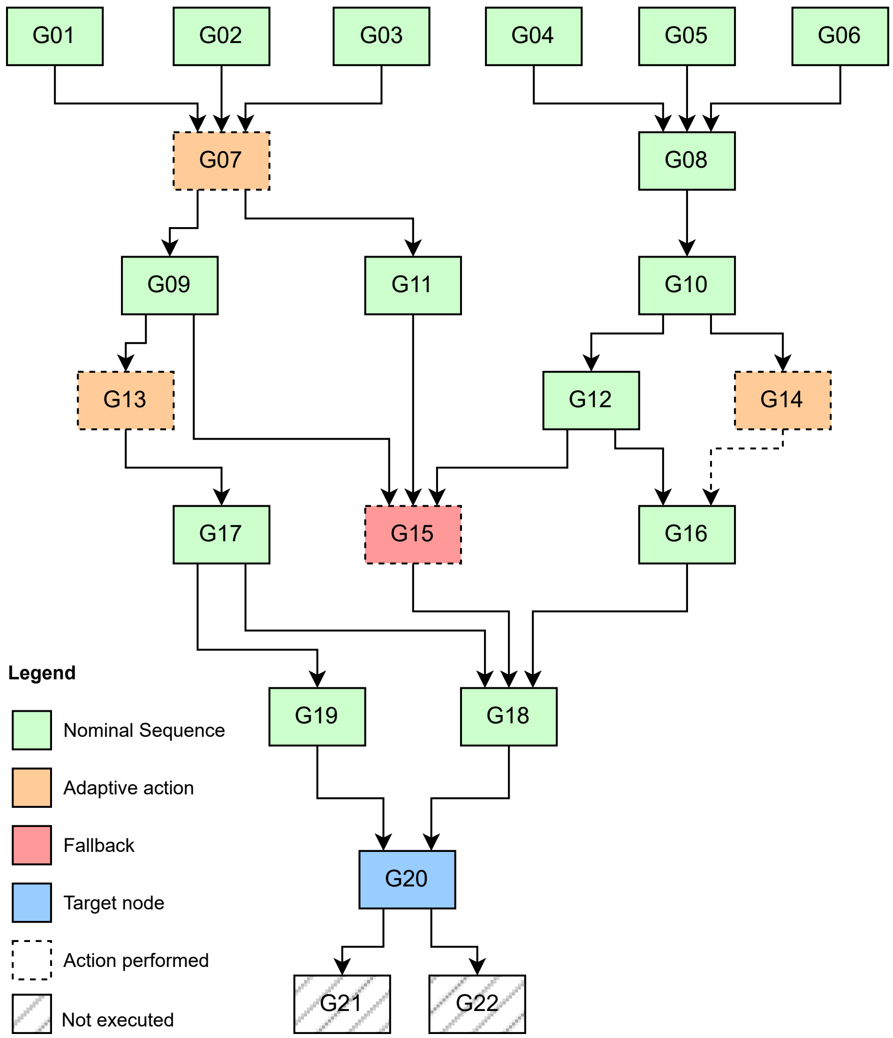

Figure 2: Gear pump illustrative instance and execution outcome. Nodes represent disassembly operations and arcs represent precedence constraints. Green nodes are executed along the nominal sequence. Orange nodes indicate operations where an adaptive action was triggered. The red node indicates a fallback with residual replanning. The blue node is the selective recovery target. Grey nodes are not executed.

Figure 2 visualises the precedence structure of the gear pump instance and summarises the realised execution. The colour coding distinguishes the nominal path from intervention steps and from fallback replanning, making it possible to relate each disruption event to the part of the residual graph that remains accessible.

Execution follows σ until symptoms indicate that the current step is at risk. The policy then selects one of the responses in Figure 1 and updates the remaining plan when needed. Table 1 summarises the observed events and the chosen responses.

The first event occurs at $G14$. Since $G14$ is not required to preserve an access path to the target, the policy bypasses it and continues along the remaining feasible operations. A

Table 1: Events and responses in the gear pump case

Operation	Force	Torque	p_{fail}	Symptom and response
G14	48.0	52.0	0.60	Access blocked, bypass
G07	30.0	35.0	0.65	Over torque without progress, tool change then retry
G13	75.0	80.0	0.40	Low value bottleneck, controlled destruction
G15	50.0	55.0	0.50	Access blocked, bypass then destruction via fallback

second event appears at $G07$ with elevated failure risk and an overload pattern. The policy recommends a tool change and the operation succeeds on the subsequent attempt. Later, $G13$ becomes a low value bottleneck. Controlled destruction is selected to avoid repeated attempts with limited expected economic upside. After this action, residual replanning is triggered on the remaining feasible graph, producing a new tail sequence $(G17, G15, G19, G18, G20)$.

A further blockage appears at $G15$. The policy first attempts bypass, then escalates to destruction through a fallback decision, followed by a second residual replanning step. The final tail sequence becomes $(G19, G18, G20)$ and the target $G20$ is recovered. This case shows how local actions and residual replanning interact in the loop of Figure 1 to preserve target reachability while controlling disruption cost.

5. Managerial implications

The execution loop in Figure 1 translates reactive disassembly planning into a small set of shop floor responses that can be configured with economic parameters. For managers, the main benefit is that interventions are no longer ad hoc. They follow a consistent logic that weighs value at risk against the cost of disruption, while keeping the number of possible actions limited.

A first practical step is parameterisation. The overhead rate h converts time into money and should reflect the effective cost of occupying the cell, including labour and capacity loss. The expected recovery value $r(i)$ should represent the conservative value that can realistically be captured when the operation succeeds, based on reuse price, remanufacturing margin, or scrap value. The direct operating cost $c(i)$ should include consumables and tool wear when these are not already covered by h. With these quantities, the planning objective and the reactive choices can be interpreted directly as profit protection decisions.

The four responses in Figure 1 can be managed as an escalation ladder. Bypass is appropriate when the current operation is not mandatory and when omitting it preserves at least one feasible access path to each remaining target. This is particularly valuable in selective recovery, where bypass avoids low value operations that can create disproportionate delays. Tool change with retry is appropriate when the blocked component has high recovery value and the expected additional effort of another attempt remains limited. In practice, a simple cap on retries per operation reduces hidden time losses. Controlled destruction is appropriate when an operation blocks progress and its recovery value is low compared with the expected disruption cost of prolonged attempts or manual intervention. This can be governed by a transparent threshold θ, where destruction is allowed if $r(i) \leq \theta$. A practical initial choice is $\theta = h \cdot \Delta t + \kappa$, where Δt is the downtime window considered unacceptable for one step and κ captures consumables and clean up costs. Residual replanning is appropriate when local actions are unlikely to restore progress, or when several alternatives exist and the priority is to protect net recovery profit rather than to follow the nominal sequence.

The gear pump illustration in Section 4.3 highlights two operational lessons. First, bypass is most effective when selective recovery is used and the residual structure offers alternative access paths, because it reduces exposure to difficult low value steps without sacrificing reachability. Second, when destruction is necessary, combining it with residual replanning stabilises execution by quickly recomputing the best remaining path and limiting cascades of retries on subsequent operations.

Deployment requires only a minimal monitoring set. Overload patterns in force or torque, repeated perception failure, timeouts, and loss of grip are sufficient to trigger the intervention branch. Converting raw signals into a small event set improves explainability and supports continuous improvement, since managers can tune h, $r(i)$, $c(i)$, and θ as labour cost, product mix, and resale prices evolve. Overall, the framework provides a compact, auditable basis for linking profitability targets to shop floor decisions under uncertainty.

6. Conclusion

This paper presented an economic decision support approach for reactive disassembly sequence planning when the condition of parts becomes clear only during execution. The planning task is modelled on a precedence graph with operation level duration, direct cost, and expected recovery value, and it covers both complete disassembly and selective recovery driven by a target set. For optimisation, a mixed integer formulation is used as a reference on moderate instances, while a fast hybrid planner generates high quality sequences for large instances.

The computational study indicates that the hybrid planner stays close to the reference objective on solvable instances while keeping short run times on large graphs. Under disruption scenarios, the response loop in Figure 1 preserves target reachability in most runs and limits avoidable loss of net recovery profit. The gear pump illustration shows how bypass, retry with an alternative tool, residual replanning, and controlled destruction can be combined to manage bottlenecks and maintain progress when individual operations become difficult.

Future work can extend the framework in three directions. First, richer product representations that capture optionality, such as AND OR structures, can improve realism and widen selective recovery choices. Second, economic parameters and decision thresholds can be estimated from operational data to support stable tuning across product families. Third, the decision loop can be strengthened by linking symptom interpretation with data driven decision support for reactive disassembly (Streibel et al., 2024) and with knowledge modelling approaches that structure heterogeneous disassembly information (Streibel et al., 2025; Bluvstein and Daub, 2026). Learning based assistance for collaborative stations and control can also complement this direction (Bauer et al., 2023; Abdous et al., 2025).

Acknowledgements

This work was carried out within the LAsSy project, supported by the German French Academy for the Industry of the Future.

Funding

The authors acknowledge financial support from the German French Academy for the Industry of the Future through the LAsSy project.

Acknowledgements (Optional)

AThis research was conducted within the LAsSy project (Learning Assistance Systems for Efficient Disassembly) and was supported by the German French Academy for the Industry of the Future. The authors thank the Academy for its financial support.

References

Abdous, M.A., Delorme, X., Battini, D., Sgarbossa, F., 2025. Scenario-based optimization and simulation framework for human-centered assembly line balancing. International Journal of Production Economics 282, 109513.

Bauer, J.C., Abdous, M.A., Kurscheid, S., Lucas, F., Lozenguez, G., Daub, R., 2023. Towards enabling human-robot collaboration in industry: Identification of current implementation barriers, in: 5th Conference on Production Systems and Logistics.

Bentaha, M.L., Battaïa, O., Dolgui, A., 2013. A stochastic formulation of the disassembly line balancing problem, in: Advances in Production Management Systems. Competitive Manufacturing for Innovative Products and Services (APMS 2012). Springer, Berlin, Heidelberg. volume 397 of *IFIP Advances in Information and Communication Technology*, pp. 397–404.

Bentaha, M.L., Battaïa, O., Dolgui, A., 2015a. An exact solution approach for disassembly line balancing problem under uncertainty of the task processing times. International Journal of Production Research 53, 1807–1818.

Bentaha, M.L., Dolgui, A., Battaïa, O., 2015b. A bibliographic review of production line design and balancing under uncertainty. IFAC-PapersOnLine 48, 70–75.

Bluvstein, G., Daub, R., 2026. Information extraction and knowledge modeling for disassembly processes, in: Procedia Computer Science (7th International Conference on Industry of the Future and Smart Manufacturing (ISM 2025)). In press.

Guo, X., Zhou, M., Abusorrah, A., Alsokhiry, F., Sedraoui, K., 2021. Disassembly sequence planning: A survey. IEEE/CAA Journal of Automatica Sinica 8, 1308–1324.

Kim, H.W., Park, C., Lee, D.H., 2018. Selective disassembly sequencing with random operation times in a parallel disassembly environment. International Journal of Production Research 56, 7243–7257.

Lambert, A.J.D., 2003. Disassembly sequencing: A survey. International Journal of Production Research 41, 3721–3759.

Mamdani, E., 1977. Application of fuzzy logic to approximate reasoning using linguistic synthesis. IEEE Transactions on Computers C-26, 1182–1191.

Streibel, L., Brunnenkant, F.A., Reinbold, N., Lindholm, N., Jordan, P., Bluvstein, G., Zaeh, M.F., Reuter, C., 2025. Systematic review of data modelling methods for system-level disassembly planning and control. International Journal of Sustainable Engineering 18, 2563271.

Streibel, L., Jordan, P., Zaeh, M.F., 2024. Data-driven approach for decision-making in reactive disassembly planning to enable case-based reasoning. Procedia CIRP 130, 1117–1123. 57th CIRP Conference on Manufacturing Systems 2024 (CMS 2024).

Sundin, E., Elo, K., Mien Lee, H., 2012. Design for automatic end–of–life processes. Assembly Automation 32, 389–398. doi:10.1108/01445151211262447.

Tian, G., Zhou, M., Chu, J., 2013. A chance constrained programming approach to determine the optimal disassembly sequence. IEEE Transactions on Automation Science and Engineering 10, 1004–1013.

Tian, G., Zhou, M., Li, P., 2018. Disassembly sequence planning considering fuzzy component quality and varying operational cost. IEEE Transactions on Automation Science and Engineering 15, 748–760.