Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 142 (2026) 986–991

www.elsevier.com/locate/procedia

36th CIRP Design Conference (CIRP Design 2026)

Disassembly sequence automation for discrete event simulation of flexible, hybrid remanufacturing systems

Julian Haller[a,*], Junjie Liang[a], Amon Göppert[a], Robert H. Schmitt[a,b]

[a]*Laboratory for Machine Tools and Production Engineering (WZL) of RWTH Aachen University, Campus-Boulevard 30, 52074 Aachen, Germany*
[b]*Fraunhofer Institute for Production Technology IPT, Steinbachstr. 17, 52074 Aachen, Germany*

* Corresponding author. Tel.: +49-241-80-25787 . *E-mail address:* julian.haller@wzl-iqs.rwth-aachen.de

Abstract

The finite nature of the Earth's resources and the need to reduce CO2 emissions require a shift from a linear to a circular production. Remanufacturing is a circular economy strategy that involves inspecting, disassembling, and reassembling of returned products to ensure they have equal or higher functionality before being dispatched again. However, the quantity and quality of returned products are uncertain, meaning production systems must be flexible enough to adapt their assembly and disassembly processes accordingly. Furthermore, products are designed for efficient assembly, but not for the flexible assembly and disassembly required. Currently, planning assembly and disassembly sequences is a manual and tedious task, and the resulting sequences are inflexible. We reviewed current approaches to derive assembly sequences automatically from CAD files. Based on this, we developed an approach that derives an AND-OR graph from STEP files. We then fed the resulting flexible assembly sequences into a discrete event simulation of flexible, hybrid assembly systems. The simulation results show diverse production KPIs based on different scenarios, which product designers can use to compare the impact of different design decisions on flexible, hybrid disassembly and assembly. In the future, we plan to deploy reinforcement learning to further automate design decisions.

© 2026 The Authors. Published by Elsevier B.V.
This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/)
Peer review under the responsibility of the scientific committee of 36th CIRP Design Conference (CIRP Design 2026)

Keywords: Circular Production; Remanufacturing; Disassembly Sequence Planning; Flexible Assembly Systems; Assembly Simulation; Discrete Event Simulation

1. Introduction

Nomenclature

AOG	AND-OR graph
ASP	Assembly sequence planning
CAD	Computer-aided design
DES	Discrete event simulation
DSP	Disassembly sequence planning
FAS	Flexible assembly systems
OCCT	Open CASCADE Technology

The necessary reduction in CO_2 emissions and the depletion of resources make the transition to a circular production economy essential [1]. To achieve this goal, nine Re:X strategies have been established as a framework for a circular economy spanning the entire life cycle of a product: Refuse, Rethink, Reduce, Reuse, Repair, Refurbish, Remanufacture, Repurpose, and Recycle [2]. Using refurbished or remanufactured products can result in up to 53 percent fewer CO_2 emissions and consume up to 88 percent fewer primary resource consumption than new goods [3]. However, implementing these two strategies poses major challenges for manufacturing companies: the quality and number of products that are returned to the cycle are unknown [4]. Therefore, all returned products must be

2212-8271 © 2026 The Authors. Published by Elsevier B.V.
This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/)
Peer review under the responsibility of the scientific committee of 36th CIRP Design Conference (CIRP Design 2026)
10.1016/j.procir.2026.05.380

manually inspected, which requires specialized expertise [5]. This additional uncertainty thus results in higher planning costs for production systems that simultaneously manufacture new goods and remanufacture returned products [6]. To meet these challenges, future assembly systems for remanufacturing must be designed to be flexible and reconfigurable [7].

However, current planning methods for assembly are not designed to assemble and disassemble products in flexible sequences and thus take benefit from flexible assembly systems. (Dis-)assembly sequence planning (DSP/ASP) has been a manual and time-consuming task to date [8]. In order to evaluate the suitability of a product design for future disassembly, a cost-intensive manual test using a physical prototype is still necessary [9]. Additionally, feedback from the assembly system on the disassembly capability of products is insufficient [10]. Consequently, an approach is needed that can evaluate the suitability of a product design for future disassembly during the development phase.

In [11] we proposed a framework for a design-based remanufacturing assessment. One core process of this framework is the automated deconstruction of CAD files into AND-OR (dis-)assembly sequence graphs. The resulting graphs can be fed into a discrete event simulation of flexible, hybrid (dis-)assembly systems to obtain production system feedback on the current design for remanufacturing. This publication explores the suitability of current DSP automation approaches to generate sequences for flexible (dis-)assembly.

The approach of automating disassembly sequences itself is not new. However, most publications focus on *optimizing* a set of sequences algorithmically, which are gathered in reviews like [12]. Neb provides an overview of automated assembly sequence extraction approaches, which was not conducted systematically [13]. Most approaches employ commercial closed-source software such as Autodesk's Inventor or Fusion360, Dassault Systèmes' CATIA or Solidworks, Siemens NX, or PTC Creo [14]. This entails not only the risk of a future vendor lock-in but also reduces solutions' applicability to proprietary data formats. Additionally, source code or algorithms are rarely made publicly available.

One solution is to build approaches around open exchange formats like STEP. This has already been done by semantic approaches like OntoSTEP [15] or by employing CAD software kernels like Open CASCADE Technology (OCCT) [16]. Few publications employed an OCCT Python wrapper pythonocc [17], but still did not disclose their source code [18–20]. The only publication known to us that pursued an open source approach and published the code is [14].

Therefore, we decided to conduct a systematic literature review to identify approaches that automate disassembly sequence planning and make it publicly available.

The remainder of the publication is structured as follows: Section 2 provides an overview of the methodology to gather and analyze relevant literature. Section 3 presents the results of the systematic literature review and our evaluation. Section 4 comprises our implementation of a STEP-to-AOG conversion by means of pythonocc followed by a discrete event simulation (DES) comparison of flexible versus fixed assembly systems in Section 5. Section 6 closes with a discussion of results and outlook.

2. Search methodology

The methodology to compile relevant literature adhered to the Preferred Reporting Items for Systematic reviews and Meta-Analyses (PRISMA) 2020 statement [21,22]. We selected the *Scopus* and *Web of Science Core Collection* (WOS) databases according to the criteria of [23]. We executed the search on 27th July 2025; English journal articles and conference papers published after 2011 were eligible. No other database filters were used. We analyzed keywords from an exploratory search and defined the search strategy as follows:

TITLE-ABS-KEY ((CAD OR "digital model" OR "3D model" OR "Computer-Aided Design") AND ("assembly" OR "disassembly") AND (sequence OR process OR path OR planning OR strategy) AND (generat* OR extract* OR automat* OR identif* OR sequenc*))

Wildcards (*) ensured that alternative spellings were fetched. The identified records were scanned for duplicates and then screened by two reviewers (dual review) in two rounds. First, only titles and abstracts were screened (abstract screening) and subsequently eligible and retrievable reports were screened completely (full text). Only reports that fully disclose their approach (e.g., via public repository) and are thus reproducible were deemed eligible. Pseudocode alone or the like was not considered eligible. Open source was no requirement at this point, leaving commercial systems solutions eligible if developed scripts, macros, or else were fully disclosed. The selection process was supported by PICO portal, New York, NY United States, available at www.picoportal.org. We only included reports that could be retrieved through open access or subscriptions held by RWTH Aachen University.

We evaluated compiled publications according to the following criteria:

- Ease of installation
- Ease of use
- Capability to create AOGs
- Computational effort (time)
- Compatibility (input data formats)

3. Search results

Following the PRISMA guidelines, the identification process began with importing 2417 records from the databases Scopus and WOS. Initially, 503 records were removed due to duplicates and supplemental records, leaving 1914 records for abstract screening. During abstract screening, 1750 records were excluded through dual independent review, resulting in 164 reports sought for retrieval. In the full text review stage, 157 reports were excluded due to inaccessibility, irrelevance, or mostly not providing the solution approach publicly. Lastly, 7 reports were considered eligible for the final review, as shown in Table 1.

Table 1. Resulting reports of the systematic literature review in chronological order.

Report		Repository	Comment
Münker and Schmitt (2022)	[24]	Github	CATIA macro
Gonnermann et al. (2023)	[25]	Suppl. Mat.	PyOCC
Schirmer et al. (2023)	[26]	Github	pythonocc
Cramer et al. (2024)	[14]	Github	FreeCAD/pythonocc
Gaarsdal et al. (2024)	[27]	Github	Prediction model
Vergez et al. (2024, 2025)	[28,29]	zenodo	Prediction model

[24] present an approach to create AOGs from STEP files. The published repository only provides the capability to visualize AOGs from input csv files that contain liaison and moving wedge data. The liaison data itself is produced with Visual Basic and CATIA, as the authors presented in [30]. However, this approach was not made public which restricts [24] functionality.

[25] provide an excerpt of their pythonocc-based script, which unfortunately did not produce any results on its own.

[26] made their *CAD Data Extraction Framework* publicly available on Github. Unfortunately, the tool is barely documented. It relies on *TopoDS* imports that are not part of pythonocc' current version. Furthermore, the tool has issues with color- and name extraction from STEP files. After some tinkering, we managed to run the tool and obtain the desired *topology* and *relationship matrix* information. However, it seems the relationship matrix function can only extract STEP-encoded sub-assemblies, no liaisons. Consequently, with our understanding we judge the tool unsuitable for creating AOGs.

[14] developed an add-on for the open-source CAD application *FreeCAD,* called *APLAN*. It is not available from the official FreeCAD add-on repository and thus requires compiling FreeCAD from scratch with APLAN modifications. It did not compile on our Windows 11 systems, only on Ubuntu 24.04 LTS. However, it worked only with a previous FreeCAD version 0.20, not the current version 1.0.1 (at the time of writing). Still, some debugging was required to run it, but we deem it capable to produce AOGs from STEP files and manually adjust edges as well.

[27] built the dataset *AssemblyNet* and associated functions to apply onto the dataset. It is functional in our environment and serves as an explosion view direction predictor trained on real assemblies. However, it does not generate an AOG by itself; that would require further logic such as collision/feasibility checks, choosing which part to remove next, and recording edges. AssemblyNet could be a component in that pipeline but does not work standalone for our purpose.

[28,29] present a comprehensive approach to predict assembly constraints using graph convolutional neural networks. The preprocessing STEP involves creating various graphs for possible assembly sequences and would thus be a suitable approach for generating AOGs. However, the authors only disclose their in- and output data in a zenodo dataset and not their algorithms.

Furthermore, both [27] and [28] utilize models to predict assembly movements or the next assembly STEP based on dataset training. This could support product designers during design process decision-making, but we estimate that (dis-) assembly sequences will have more constraints than a model could predict, due to unique (dis-)assembly system circumstances. Therefore, an ideal AOG generation tool will have the ability for manual intervention.

Beyond the results of the systematic literature review, three publications are known to us that could create AOGs from CAD files [8,31,32]. All authors belong to the Autodesk AI Lab (Source) that also published the *Fusion360Gallery* dataset and associated tools [33]. The tools of the Fusion360Gallery provide a script to create assembly graphs but take only data model of the assembly (json) as input that were created from the assembly.step with Fusion360 macros.

[31] present a prediction model, that predicts assembly joints based on boundary representations (BRep) of CAD files, which like [27–29] could be a powerful pre-processing STEP in the AOG generation pipeline but does not provide the capability out of the box.

The tools *Assemble Them All* [8] and *Automated Sequence Planning for Complex Robotic Assembly with Physical Feasibility* (ASAP) [32] provide the capability to derive assembly sequences. However, they were not trivial to install and only worked in our Ubuntu 24.04 LTS environment. Furthermore, both tools search for an optimal assembly sequence according to some criteria and output only one sequence as a result. Though, we estimate that it is feasible to modify the code to stop optimization and output all possible precedence graphs. The physics-based assessment what (dis-) assembly sequences are feasible looks powerful nonetheless and should provide more accurate results than only OCCT-based assessments.

Overall, we judge only [14] as a suitable, publicly available tool to create AOGs from STEP files. However, the installation method is cumbersome, and it is unclear how it will develop with future FreeCAD versions if the add-on is not maintained. Therefore, we conclude that it is necessary to develop an open-source tool to generate AOGs from STEP files, that provides a lightweight and reproducible installation.

4. Liaison matrix generation

To derive AOGs for discrete event simulation of flexible, hybrid remanufacturing systems, we developed a Python toolchain that generates liaison matrices and graphs for STEP files, based on contacts between parts. Heart of the tool are functions of the pythonocc-core library.

First, the system ingests 3D CAD data from STEP files using BRep topology parsing. TopExp_Explorer traverses the geometric hierarchy to identify and extract individual solid entities (TopoDS_Shape objects) representing discrete mechanical components within the assembly. The algorithm then extracts component identifiers from STEPs product entity declarations. This process associates semantic labels with each geometric solid, enabling part identification.

For each unique pair of components (i,j) where $i < j$, the system performs exact minimum distance computation using pythonocc' BRepExtrema_DistShapeShape class. This process calculates the Euclidean distance between surface boundaries of the 3D shapes.

A binary classification is applied to each pairwise distance measurement using a predefined tolerance threshold (we chose $\varepsilon = 1\times10^{-6}$ meters as default). Components are classified as "in contact" when their distance is smaller than the tolerance. The contact relationships are encoded into a symmetric binary matrix where $C_{ij} = 1$ indicates contact between components i and j. Lastly, the matrix is visualized as an undirected graph via the NetworkX library.

We tested the cutter knife from [31], comprising six parts, in our tool for validation and the results are shown in Figure 1.

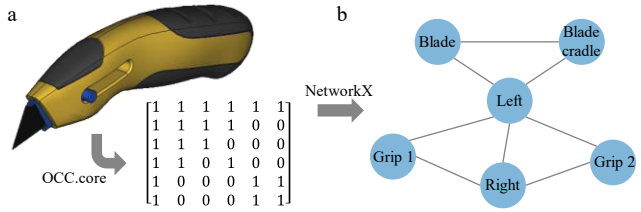


Fig. 1. (a) Cutter knife STEP file and contact matrix; (b) contact graph.

The contact graph matches the one from [31], except in our work the part *Blade* has also contact with the part *Left*, whereas in [31] it only touches the *Blade Cradle.* This might be due to different tolerance values but will be investigated in future work. Our code is publicly available (see *Data availability).*

5. Discrete event simulation of flexible, hybrid (dis)assembly systems

Once the liaison between parts is known, a basic AOG can be created. We employ Breadth-First Search (BFS) to convert the liaison graph into an AOG and created an AOG for the hind leg assembly of our toy dog use case. Figure 2 shows the area of interest in the toy dog assembly and the associated AOG.

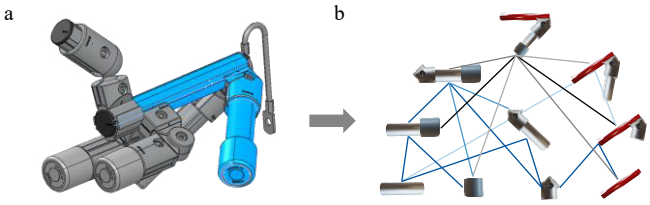


Fig. 2. (a) toy dog STEP file, (b) AND-OR graph.

One leg comprises three parts that must be connected to the dog's body (part 4). Each possible subassembly is represented as a node, and edges represent assembly or disassembly processes. With only four parts in the assembly, there is already a plethora of sequences possible.

AOGs can also be represented as a precedence matrix and thus be fed into our existing discrete event simulation, as presented in [34]. [35] already established that AOGs provide higher operational flexibility in theory. Our simulation is modelled in Siemens' *Tecnomatix Plant Simulation*, and we employ several python scripts to communicate with Plant Simulation's COM Interface, such as a scheduling server.

For this work, we first modelled a 5-station assembly-only layout and compared two assembly approaches: a fixed precedence and a flexible precedence based off an AOG. The Plant Simulation model is shown in Figure 3.

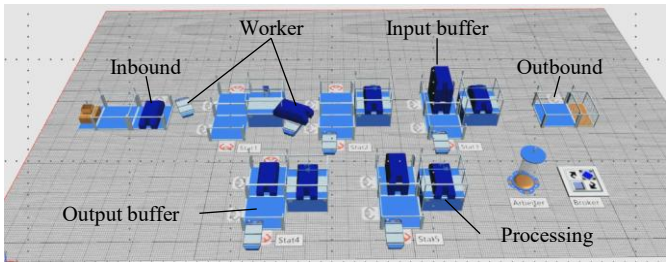


Fig. 3. 5-station assembly layout in Plant Simulation

For each approach 50 jobs were queued, arriving in 45 s intervals. At the inbound source, jobs are released in a fixed interval. Automated guided vehicles (AGVs, called workers) transport jobs between inbound, stations, and outbound. We configured enough workers (11); simulating a logistics bottleneck was no objective of this study but might be in future work. A scheduling server decides per greedy algorithm where a job moves for its next processing step. In our experiment each processing step requires a different technology that only one station provides. However, in the flexible precedence, the order of technologies (and thus stations) is not fixed. Table 2 shows station and technologies for the fixed precedence scenario, and additional capabilities in italics for the flexible precedence scenario.

Table 2. Station and technology mapping for both simulation scenarios, including processing times.

Station	1	2	3	4	5
Technology	A (30 s)	B (60 s)	C (10 s) *B (60 s)* *D (120 s)*	D (120 s)	E (10 s) *B (60 s)* *D (120 s)*

The simulation was conducted on a regular office computer with GPU; a workstation is not required at this complexity level. We compared the two scenarios regarding three KPI:

- Station utilization [%]
- Average job makespan [s]
- Average blocked time [s]

Where blocked time represents time, a station cannot process because a finished job cannot move to its next scheduled station. The results of the two simulation runs are shown in Figure 4.

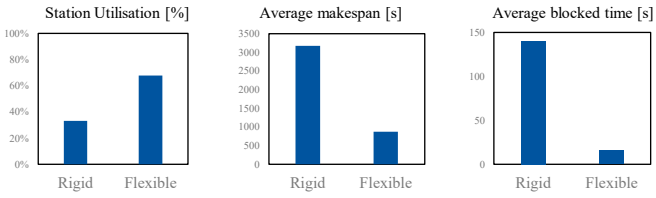


Fig. 4. Results of rigid and flexible precedence simulation runs considering station utilization, average makespan, and average blocked time.

In our case simulation, the flexible precedence outperformed the fixed precedence in every observed metric. Naturally, the results will vary largely with different assembly setups, especially regarding the stations' capabilities and processing times; yet proof of concept for simulating flexible assembly could be demonstrated. In future work, we plan to expand the simulation capabilities and add disassembly.

6. Discussion and outlook

We conducted a systematic literature review to identify works that developed an approach for automated assembly and disassembly sequence planning from CAD files and made their work publicly available. We tested all identified approaches against criteria such as reproducibility, ease-of-use, and capability to create AOGs and concluded that only one approach delivers our desired capability and is fully open source but with a difficult setup. Consequently, we developed a system that exploits open-source capabilities to create contact-based liaison matrices and graphs from STEP assemblies. We successfully compared our tools capability against another openly available tool that relies on a commercial CAD suite.

A contact-based liaison matrix is a basic starting point to generate AOGs. In the future, we plan to extend the AOG generation by moving wedge collision checks, which we deem feasible with BReps and pythonocc. Ideally, a sophisticated AOG generation tool further checks gravitational stability of subassemblies in the process of assembly/disassembly and allows manual intervention of product designers. Furthermore, our current graph generation method was only tested for small assemblies. The toy dog consists of 26 parts or 63 solids and thus it is likely that our algorithm will require mechanism to reduce complexity to compute results within acceptable time, for example by subassembly detection. Expanding our approach by process simulation or semantic capabilities might be necessary for a seamless disassembly assessment pipeline.

Lastly, the simulation modeled only a laboratory assembly case thus far, not considering simultaneous disassembly. Future research must be concerned with simulating real remanufacturing environments and wage economic advantages of line (dis-)assembly versus the added complexity of a flexible (dis-)assembly. To do that, we plan to move the simulation to SimPy to make the entire tool chain open-source and reproducible.

Acknowledgements

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germanys Excellence Strategy – EXC-2023 Internet of Production – 390621612 and Project Human Decision – 543081196 of Priority Programme SPP 2443: Hybrid decision support in product creation.

Author contributions

Julian Haller: conceptualization, investigation, performing the literature search and data analysis, writing– original draft, visualization, and project administration. **Junjie Liang** and **Amon Göppert**: writing– review & editing and supervision. **Robert H. Schmitt**: funding acquisition.

Data availability

The results of this publication are reproducible by the provided information in the methodology section. The PRISMA checklist and database results in *research information system* (RIS) format can be made available upon request. The publication's code is publicly available at https://gitlab.com/TIBHannover/human-decision/source-code/STEP-contact-matrix-extraction

Conflict of interest

The authors have no competing interests to declare that are relevant to the content of this article.

References

[1] Sakao T, Bocken N, Nasr N, Umeda Y. Implementing circular economy activities in manufacturing for environmental sustainability. CIRP Annals 2024;73:457–81. https://doi.org/10.1016/j.cirp.2024.06.002.

[2] Potting J, Hekkert M, Worrell E, Hanemaaijer A. Circular Economy: Measuring innovation in product chains. The Hague: PBL Netherlands Environmental Assessment Agency; 2016.

[3] Parker D, Riley K, Robinson S, Symington H, Tweson J, Jannson K, et al. Remanufacturing Market Study. 2015.

[4] Teunter RH, Flapper SDP. Optimal core acquisition and remanufacturing policies under uncertain core quality fractions. European Journal of Operational Research 2011;210:241–8. https://doi.org/10.1016/j.ejor.2010.06.015.

[5] Tolio T, Bernard A, Colledani M, Kara S, Seliger G, Duflou J, et al. Design, management and control of demanufacturing and remanufacturing systems. CIRP Annals 2017;66:585–609. https://doi.org/10.1016/j.cirp.2017.05.001.

[6] Rizova MI, Wong TC, Ijomah W. A systematic review of decision-making in remanufacturing. Computers & Industrial Engineering 2020;147:106681. https://doi.org/10.1016/j.cie.2020.106681.

[7] Paul M, Eickholt D, Reinhart G. Framework for adaptable remanufacturing. Production & Manufacturing Research 2024;12. https://doi.org/10.1080/21693277.2024.2362692.

[8] Tian Y, Xu J, Li Y, Luo J, Sueda S, Li H, et al. Assemble Them All. ACM Trans Graph 2022;41:1–11. https://doi.org/10.1145/3550454.3555525.

[9] Rodríguez NB, Gabriel C, Gaha R, Favi C. Analysis of disassembly parameters in repairability scores: limitations for engineering design and suggestions for improvement. Procedia CIRP 2023;116:738–43. https://doi.org/10.1016/j.procir.2023.02.124.

[10] Münker S. Automated assembly sequence planning of large-scale assemblies. 2023. https://doi.org/10.18154/RWTH-2023-12022.

[11] Haller J, Baum J, Göppert A, Schmitt RH. Remanufacturing in flexiblen (De-)Montagesystemen: Entscheidungsunterstützung für kreislauffähiges Produktdesign. Zeitschrift Für Wirtschaftlichen Fabrikbetrieb 2025;120:151–4. https://doi.org/10.1515/zwf-2025-1027.

[12] Rashid MFF, Hutabarat W, Tiwari A. A review on assembly sequence planning and assembly line balancing optimisation using soft computing approaches. Int J Adv Manuf Technol 2012;59:335–49. https://doi.org/10.1007/s00170-011-3499-8.

[13] Neb A. Review on Approaches to Generate Assembly Sequences by Extraction of Assembly Features from 3D Models. Procedia CIRP 2019;81:856–61. https://doi.org/10.1016/j.procir.2019.03.213.

[14] Cramer M, Kellens K, Demeester E. APLAN: open assembly planning framework in FreeCAD. Procedia CIRP 2024;127:56–61. https://doi.org/10.1016/j.procir.2024.07.011.
[15] Barbau R, Krima S, Rachuri S, Narayanan A, Fiorentini X, Foufou S, et al. OntoSTEP: Enriching product model data using ontologies. Computer-Aided Design 2012;44:575–90. https://doi.org/10.1016/j.cad.2012.01.008.
[16] Open CASCADE Technology | Collaborative development portal n.d. https://dev.opencascade.org/ (accessed September 3, 2025).
[17] Paviot T. pythonocc 2022. https://doi.org/10.5281/ZENODO.3605364.
[18] Tariki K, Kiyokawa T, Nagatani T, Takamatsu J, Ogasawara T. 3D Model-Based Non-interference Assembly Sequence Generation for Products with a Large Number of Parts. 2019 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM), Bangkok, Thailand: IEEE; 2019, p. 142–7. https://doi.org/10.1109/CIS-RAM47153.2019.9095782.
[19] Tariki K, Kiyokawa T, Ricardez GAG, Takamatsu J, Ogasawara T. 3D Model-Based Assembly Sequence Optimization using Insertionable Properties of Parts. 2020 IEEE/SICE International Symposium on System Integration (SII), Honolulu, HI, USA: IEEE; 2020, p. 1400–5. https://doi.org/10.1109/SII46433.2020.9026210.
[20] Tariki K, Kiyokawa T, Nagatani T, Takamatsu J, Ogasawara T. Generating complex assembly sequences from 3D CAD models considering insertion relations. Advanced Robotics 2021;35:337–48. https://doi.org/10.1080/01691864.2020.1863258.
[21] Page MJ, McKenzie JE, Bossuyt PM, Boutron I, Hoffmann TC, Mulrow CD, et al. The PRISMA 2020 statement: an updated guideline for reporting systematic reviews. BMJ 2021:n71. https://doi.org/10.1136/bmj.n71.
[22] Page MJ, Moher D, Bossuyt PM, Boutron I, Hoffmann TC, Mulrow CD, et al. PRISMA 2020 explanation and elaboration: updated guidance and exemplars for reporting systematic reviews. BMJ 2021:n160. https://doi.org/10.1136/bmj.n160.
[23] Gusenbauer M, Haddaway NR. Which academic search systems are suitable for systematic reviews or meta-analyses? Evaluating retrieval qualities of Google Scholar, PubMed, and 26 other resources. Research Synthesis Methods 2020;11:181–217. https://doi.org/10.1002/jrsm.1378.
[24] Münker S, Schmitt RH. CAD-based AND/OR Graph Generation Algorithms in (Dis)assembly Sequence Planning of Complex Products. Procedia CIRP 2022;106:144–9. https://doi.org/10.1016/j.procir.2022.02.169.
[25] Gonnermann C, Gebauer D, Daub R. CAD-Based Feature Recognition for Process Monitoring Planning in Assembly. Applied Sciences 2023;13:990. https://doi.org/10.3390/app13020990.
[26] Schirmer F, Srikanth VK, Kranz P, Rose CG, Schmitt J, Kaupp T. Towards Automatic Extraction of Product and Process Data for Human-Robot Collaborative Assembly. 2023 21st International Conference on Advanced Robotics (ICAR), Abu Dhabi, United Arab Emirates: IEEE; 2023, p. 520–5. https://doi.org/10.1109/ICAR58858.2023.10406471.
[27] Gaarsdal J, Haurum JB, Wolff S, Madsen CB. AssemblyNet: A Point Cloud Dataset and Benchmark for Predicting Part Directions in an Exploded Layout. 2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA: IEEE; 2024, p. 5824–33. https://doi.org/10.1109/WACV57701.2024.00573.
[28] Vergez L, Polette A, Pernot J-P. Interface-Based Search and Automatic Reassembly of CAD Models for Database Expansion and Model Reuse. Computer-Aided Design 2024;167:103630. https://doi.org/10.1016/j.cad.2023.103630.
[29] Vergez L, Polette A, Pernot J-P. Multi-part kinematic constraint prediction for automatic generation of CAD model assemblies using graph convolutional networks. Computer-Aided Design 2025;178:103805. https://doi.org/10.1016/j.cad.2024.103805.
[30] Münker S, Polikarpov M, Schmitt RH. Virtuelle Demontage für XL-Produkte/Assembly Sequence Generation for XL-Products – Virtual Disassembly for High Numbers of Components. Wt 2020;110:833–7. https://doi.org/10.37544/1436-4980-2020-11-12-97.
[31] Willis KDD, Jayaraman PK, Chu H, Tian Y, Li Y, Grandi D, et al. JoinABLe: Learning Bottom-up Assembly of Parametric CAD Joints. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA: IEEE; 2022, p. 15828–39. https://doi.org/10.1109/CVPR52688.2022.01539.
[32] Tian Y, Willis KDD, Al Omari B, Luo J, Ma P, Li Y, et al. ASAP: Automated Sequence Planning for Complex Robotic Assembly with Physical Feasibility. 2024 IEEE International Conference on Robotics and Automation (ICRA), Yokohama, Japan: IEEE; 2024, p. 4380–6. https://doi.org/10.1109/ICRA57147.2024.10611595.
[33] AutodeskAILab/Fusion360GalleryDataset: Data, tools, and documentation of the Fusion 360 Gallery Dataset n.d. https://github.com/AutodeskAILab/Fusion360GalleryDataset (accessed September 10, 2025).
[34] Rachner J, Kaven L, Voet F, Göppert A, Schmitt RH. Simulation-Based Potential Analysis of Line-Less Assembly Systems in the Automotive Industry. In: Schüppstuhl T, Tracht K, Fleischer J, editors. Annals of Scientific Society for Assembly, Handling and Industrial Robotics 2022, Cham: Springer International Publishing; 2023, p. 41–51. https://doi.org/10.1007/978-3-031-10071-0_4.
[35] Münker S, Göppert A, Schmitt RH. Suitability of CAD-based Disassembly Sequence Plans for Flexible Remanufacturing. Procedia CIRP 2023;120:738–43. https://doi.org/10.1016/j.procir.2023.09.068.