JÖNKÖPING UNIVERSITY
School of Engineering

Scenario-Based Simulation of Automation Configurations in HMLV Assembly.

Main area: Production Engineering and Management
Authors: Abinay Kaniyath Poothakandy, Abhishek Sureshbabu Edamana

This thesis is conducted at the School of Engineering at Jönköping University within Production Engineering and Management. The authors are responsible for the opinions, conclusions and results presented here.

Examiner:	Carin Rösiö
Supervisor:	Paraskeva Wlazlak
Credits:	30 hp
Date:	2026-09-10

Abstract

Purpose: This thesis builds a discrete-event simulation (DES) model of a representative high-mix, low-volume (HMLV) assembly system and uses it to evaluate alternative automation configurations against measurable production performance indicators.

Method: An explanatory and predictive simulation study was carried out. A DES model of a five-station, mixed-model assembly line producing four product variants was built in AnyLogic 8 PLE, incorporating stochastic processing times, sequence-dependent changeovers, a contended operator pool, and constant work-in-process (CONWIP) release control. Three configurations were compared under identical demand: manual, semi-automated (human-robot collaboration, HRC), and automation-assisted. The baseline comparison of the three configurations was replicated five times with fixed random seeds to ensure exact exactly; the sensitivity sweeps used ten replications per point with independently drawn seeds and are reported with 95 per cent confidence intervals. The manual baseline was validated against published HMLV throughput benchmarks. The sensitivity analyses varied arrival rates from 13 to 48 jobs per hour and CONWIP limits from 3 to 12 units and included a reduced-changeover variant.

Findings: Automation did not change throughput under the baseline load, where the differences between configurations remained within the replication confidence intervals, but it reduced lead time and work-in-process substantially. The automation-assisted configuration cut lead time by 56.3 per cent, and by 65.9 per cent once changeover time was halved, mainly through a 97.4 per cent fall in queueing time. Raising demand lifted the capacity ceiling to 41.23 jobs per hour, an increase of 148 per cent. Partial automation nevertheless eased the line without removing the constraint: automating three of the five workstations reduced loading at the remaining operator-paced stations by only 0.2 percentage points.

Implications: In HMLV assembly the capacity ceiling is set by the allocation of tasks rather than by the processing speed of any individual station. Automating non-bottleneck operations reduces queueing and lead time but leaves the system constraint where it is, so investment aimed purely at speed will not raise output. As processing is automated, sequence-dependent changeover becomes the largest remaining component of lead time.

Limitations: The study models a conceptual, representative system rather than a specific industrial case. Economic assessment, ergonomic factors, and learning effects lie outside its scope, and equipment failure at the automated stations is excluded from the baseline configurations and examined only as a sensitivity case.

Keywords – Automation configurations, CONWIP, Discrete-event simulation, High-mix low-volume, Human-robot collaboration, Performance evaluation

Acknowledgment

We would like to express our sincere gratitude to our supervisor, Paraskeva Wlazlak, for her guidance and expertise throughout this thesis. Her insight shaped the direction and structure of our work, and we are grateful for the time and care she devoted to helping us sharpen our ideas.

We also wish to thank our examiner, Carin Rösiö, for her valuable feedback and constructive input, which have significantly strengthened this work.

Finally, we thank our families, friends, and classmates for their continued encouragement and support throughout this journey.

Abinay Kaniyath Poothakandy

Abhishek Sureshbabu Edamana

Table of Contents

List of tables: VII

1. Introduction 1
1.1. Background 1

1.2. Problem Statement 2

1.3. Purpose and Research Questions 2

1.4. Scope and Delimitations 3

1.5. Application of Artificial Intelligence 4

1.6. Outline 4

2. Theoretical Framework 5
2.1. High-Mix, Low-Volume Production Characteristics 5

2.2. Automation in Assembly Systems 6

2.2.1. The Levels of Automation Framework 6
2.2.2. Structural Trade-offs 7
2.3. Discrete-Event Simulation for System Evaluation 8

2.4. Workload Control and CONWIP 9

2.5. Verification and Validation 10

3. Research Method 12
3.1. Research Design and Approach 12

3.1.1. Research Process Overview 12
3.1.2. Research Strategy 12
3.2. Systematic Literature Review 13

3.2.1. Screening Procedure 13
3.2.2. Search 1: Core Parameterisation 14
3.2.3. Search 2: HMLV Automation Literature 14
3.2.4. Search 3: Verification and Validation Methodology 14
3.2.5. Supplementary Sources Outside the Database Searches 15
3.3. Conceptual Model 15

3.3.1. System Structure and Flow Logic 15
3.3.2. Workload Control 16
3.3.3. Resources 16
3.4. Experimental Design 16

3.4.1. Configurations 17
3.4.2. Experimental Variables and Boundary Conditions 18
3.4.3. Run Length, Initialisation and Replication 18
3.4.4. Model Parameters 19

3.5. Verification and Validation 20

3.5.1. Verification of the Implemented Model 21
3.5.2. Validation of the Conceptual Model 22
3.5.3. Sensitivity Analysis 22
3.5.4. Limitations of the Validation Approach 23
3.6. Research Quality and Ethical Considerations 23

3.6.1. Reliability 23
3.6.2. Experimental Control Across Configurations 23
3.6.3. External Validity 24
3.6.4. Objectivity and Ethical Considerations 24

4. Results
4.1. Verification and Validation Results 25

4.1.1. Run-length Sensitivity 25
4.1.2. Internal Consistency Check: Little's Law Estimate Against Recorded Token Holdings 25
4.1.3. Operational Plausibility 26
4.1.4. Comparison with an Analytical Model 26
4.2. Baseline Configuration Comparison 26

4.3. Decomposition of the Lead Time Reduction 28

4.4. Station-Level Utilisation 29

4.5. Sensitivity to the CONWIP Limit 31

4.6. Capacity Ceilings 34

4.7. Reduction of Changeover 38

4.8. Sensitivity to the Assumed Automated Processing Time 39

5. Analysis 41
5.1. Analysis of the Modelling Requirements 41

5.1.1. Mapping of Theoretical Requirements to Model Components 41
5.1.2. Answer to the First Research Question 43
5.2. Analysis of Model Credibility 43

5.2.1. Theoretical Grounding 43
5.2.2. Selection of Techniques and its Justification 43
5.2.3. Evaluation of the Model Against the Identified Constraints 44
5.2.4. Capability of the Combination in Establishing Credibility 45
5.2.5. Limitations of the Validation Approach 45
5.2.6. Answer to the Second Research Question 45
5.3. Analysis of Configuration Performance 46

5.3.1. Throughput is Set by Demand Below the Capacity Ceiling 46
5.3.2. Lead Time Falls Because Work Ceases to Wait 46

5.3.3. Partial Automation Relieves the Line Without Relieving the Constraint 47
5.3.4. Capacity Ceilings Track Task Allocation More Closely Than Processing Speed Does 47
5.3.5. Changeover as the Residual Component of Lead Time 47
5.3.6. Answer to the Third Research Question 48
6. Discussion 49
6.1. Results in Relation to Existing Literature 49
6.2. Practical Implications 50
6.3. Method Discussion 50
6.4. Limitations 51
7. Conclusions 53
7.1. Answers to the Research Questions 53
7.2. Contributions 53
7.3. Future Research 55
References 56
Appendix A: Systematic Literature Review: Search Protocol and Screening IX
Appendix B: Model Parameters and Input Data XIV
Appendix C: Detailed simulation results XVI
Appendix D: Analytical derivation of utilisation and capacity ceilings XXII

LIST OF TABLES:

Table 1: LoA reference scales for mechanical and cognitive tasks 7

Table 2: Model parameters 19

Table 3: Manual configuration at three run lengths (seed 1) 25

Table 4: Internal Consistency Check; Little's Laws 25

Table 5: Performance by configuration 26

Table 6: Changes compared to the manual baseline 27

Table 7: Lead Time Breakdown (Minutes) 28

Table 8: Utilisation by workstation 30

Table 9: Effect of the CONWIP Limit 31

Table 10: Lead time separated into time inside the line and time waiting for release, by CONWIP limit 31

Table 11: Throughput and operator utilisation against offered load 34

Table 12: Capacity ceilings by configuration 35

Table 13: Effect of changeover reduction 38

Table 14: Effect of the automated processing time on system performance 39

Table 15: Theoretical requirements, model components and evidence of necessity ... 41

Table 16: Verification and validation techniques applied, and the credibility each establishes 44

List of figures:

Figure 1: AnyLogic model structure 17

Figure 2: Lead time decomposed into its processing, changeover and queueing components (values in minutes) 29

Figure 3: Station utilisation by configuration. The dashed line marks the manual baseline 30

Figure 4: Effect of the CONWIP limit 32

Figure 5: Throughput achieved versus offered load 35

Figure 6: Capacity ceiling versus number of automated workstations 37

List of Abbreviations:

AL	Assembly line
CNC	Computer numerical control
CONWIP	Constant work-in-process
CPS	Cyber-physical system
CV	Coefficient of variation
DES	Discrete-event simulation
HMLV	High-mix, low-volume
HRC	Human-robot collaboration
KPI	Key performance indicator
LoA	Levels of automation
LT	Lead time
LUMS COR	Lancaster University Management School corrected order release
PLE	Personal Learning Edition (AnyLogic)
POLCA	Paired-cell overlapping loops of cards with authorisation
SD	Standard deviation
Th	Throughput
V&V	Verification and validation
WIP	Work-in-process
S	Workstation

1. Introduction

This chapter introduces the research area. It sets out what is distinctive about high-mix, low-volume assembly, what is meant here by an automation configuration, and why discrete-event simulation is the method used to compare such configurations. The problem statement, purpose, and research questions follow, together with the scope of the study and the structure of the report.

1.1. Background

Manufacturing has grown more complex under pressures that reinforce one another. Competition is global, customers expect products configured to their own requirements, and the products themselves are becoming more advanced and more varied (Johansen et al., 2021). Firms cannot expect a production strategy to sit still under these conditions. Where buyers ask for a product built to their own specification instead of one taken off the shelf, the pressure bites hardest, and it drives manufacturers towards methods for which flexibility, not volume, is the quality worth optimising (Gan et al., 2023). The environment that results combines broader product variety with shorter life cycles and demand that refuses to settle.

Normand and Bradley (2024) observe that such settings call for an uncommon degree of operational flexibility and responsiveness to customers; methods of improvement built for high-volume contexts therefore fail to transfer directly. The difficulty differs in kind from what high-volume manufacturing presents. Within high-mix, low-volume (HMLV) production, the assembly system has to take frequent product changes in stride while holding operational performance steady, yet processing times, routing, and resource use shift from job to job, so planning suffers and control suffers with it (Johansen et al., 2021). Because one variant may need a task sequence, a processing profile, and a route that another variant does not, workstation loads come out uneven, and the bottleneck moves instead of staying put (Monazzam et al., 2022).

How far, then, should automation go? Frohm et al. (2008) answer with a spectrum instead of a switch: stretching from the fully manual to the fully automated, with physical and mental tasks divided between operator and machine in differing proportions along the way. Flexible automation and robotics now reach into production that is complex and variable (Johansen et al., 2021), collaborative robots and vision-guided systems above all. Caution is still owed. Automation brought into an HMLV environment can work against the very objectives that motivated it. The line may grow rigid, and with loads thrown unevenly across the stations a bottleneck can end up relocated rather than removed (Johansen et al., 2021).

Each configuration also changes the system's structure; resource interaction, queue formation, and the system's overall behaviour all change with it. The choice therefore goes beyond whether to automate at all; it is equally a question of degree and of arrangement. Answering that means comparing configurations under identical operating conditions, because the trade-offs among lead time, throughput, work-in-process, and resource utilisation surface only when those conditions are held constant, and the four indicators stack into a hierarchy where performance at system level follows from behaviour at station level (Kang et al., 2016; Monazzam et al., 2022).

Discrete-event simulation (DES) offers a way to evaluate such systems while production keeps running. Stochastic processing times, finite resources, queues, and

operational variability are all represented explicitly in the model (Banks et al., 2004). Combined with scenario-based testing, it runs alternative system designs under one and the same demand and operating conditions, so measurable performance figures exist ahead of any change on the shop floor (Negahban & Smith, 2014).

The present study puts that capability to a specific use: it compares several automation configurations for a single HMLV assembly system, under controlled and identical conditions, without tying the comparison to any particular company. Automation and DES each carry long research histories behind them. Comparisons of exactly this kind, though, are scarce.

1.2. Problem Statement

Considerable flexibility is required of an HMLV assembly system if product variety is to be accommodated without sacrificing operational performance. Automation technology has grown more accessible; deploying it here is still complex. High-volume production usually justifies automation through economies of scale. HMLV systems behave otherwise, marked as they are by variability of product type, of processing time, and of routing sequence (Johansen et al., 2021), and such variability leaves the performance impact of automation harder to predict. Conventional analytical methods presuppose a stability this environment does not offer.

That DES works well for analysing production systems and for weighing automation options is well supported in the literature (Herps et al., 2022; Negahban & Smith, 2014). What the literature mostly contains, however, is optimisation-driven improvement work, single-company industrial case studies, or studies of high-volume automated systems. Studies that set different automation configurations—fully manual, semi-automated, and automation-assisted setups—against one another inside a controlled, representative HMLV assembly environment beholden to no particular company remain few. A result obtained inside one company need not survive transfer to another HMLV setting, where product characteristics and operational constraints differ; generalisability suffers, and with it the findings' worth as grounds for decisions elsewhere (Herps et al., 2022).

There is consequently no structured account of how different automation configurations affect key production performance indicators under the same stochastic and demand conditions. The concept of automation levels has been studied at length from theoretical and human-factors perspectives (Frohm et al., 2008), but its practical consequences for system-level performance in HMLV assembly have not been examined in any depth through simulation. In the absence of a systematic, simulation-based framework for assessing alternative configurations, automation decisions in these settings rest on assumption rather than on measurable performance trade-offs.

1.3. Purpose and Research Questions

The purpose of this thesis is to develop and verify a DES model of a representative HMLV assembly system, and to use that model to evaluate and compare automation configurations against measurable production performance indicators. Three arrangements are tested, namely fully manual, semi-automated, and automation-assisted, and the trade-offs between them are identified across throughput, lead time, work-in-process, and resource utilisation. Because the comparison is made in simulation rather than on a real line, it can be run under controlled conditions and

repeated at no cost, so the consequences of a given configuration are known before any physical investment is committed.

Three research questions follow from that purpose and are addressed in turn.

RQ1: What modelling principles and structural assumptions are required to validly represent key characteristics of HMLV assembly systems in a DES model intended for the evaluation of alternative automation configurations?

This first question concerns the theoretical and architectural design of the model. It asks which features of HMLV systems, among them sequence-dependent setups, stochastic processing variability, and flow constraints, have to be represented explicitly if the simulation is to stand as a proxy that genuinely represents environments of this kind. Whether an assumption is necessary is judged against the specific purpose at hand—comparing automation configurations, if leaving the assumption out would invalidate that comparison, the assumption is required.

RQ2: Which verification and validation techniques can establish the credibility of a DES model of an HMLV assembly system when empirical system data are unavailable?

Credibility is the concern of the second question. The study rests on no industrial case, so every technique that presupposes empirical system data falls away, historical data validation most obviously. Which of the remaining verification and validation methods can stand in their place is one open matter. What each method establishes on its own is another. And once all of them have been applied, some uncertainty will remain; how much is still to be settled.

RQ3: How do different automation configurations (manual, semi-automated, and automation-assisted) affect key production performance indicators, namely throughput, lead time, work-in-process, and resource utilisation, under identical operating conditions?

The third question is comparative, and it is answered through controlled simulation experiments in which every configuration faces the same demand and the same variability. Reading the results means separating differences that stem from the configuration itself off from those that demand alone would produce, while attending to trade-offs, capacity limits, and automation's structural effects on how the system behaves.

Out of these questions the study produces three things. Practitioners gain a procedure for weighing automation configurations ahead of any investment commitment. On the modelling side, the work offers a strategy for validating simulation models that were built without industrial data. It also sharpens the account of how far the level of automation drives flow performance.

1.4. Scope and Delimitations

The model developed here is a DES model of an HMLV assembly system: conceptual, not industrial; representative of its class, not specific. It is used to compare automation setups under controlled operating conditions. The set under evaluation comprises fully manual, semi-automated, and automation-assisted configurations. Stochastic

processing times, together with controlled demand variability, are part of the design, so the features that count in such systems stay represented while overall complexity stays manageable.

Four product variants run through five workstations in series. A sensitivity case joins the three principal configurations: an automation-assisted variant whose changeover time has been reduced. The system's behaviour is examined across arrival rates between 13 and 48 jobs per hour. No real installation sits behind the model, so validation is carried out against benchmark figures published for HMLV lines.

Only measurable production performance indicators are evaluated—throughput, lead time, work-in-process, and resource utilisation. A full factory, or even a multi-line environment, is not represented, and the study involves no real industrial data collection and no real-time synchronisation. Economic assessments are excluded: no investment cost analysis, no return-on-investment calculations. So are human-related factors, among them learning curves, fatigue, and ergonomic assessment. The baseline configurations assume that the automated equipment never fails mechanically; failures at the automated stations get separate treatment, as a sensitivity case.

1.5. Application of Artificial Intelligence

The authors drafted each section independently. Only after that were tools allowed in, and each was given a narrow job. Grammarly and Perplexity handled formatting, sentence structure, and linguistic accuracy. Rayyan—the screening function described in Section 3.2.1 logged and organised the title-and-abstract screening decisions, while Primo AI and Scopus AI helped locate relevant articles and books on the subjects under study. A general-purpose language model served to re-check the arithmetic and the internal consistency of the reported figures. The line was drawn there. No tool generated ideas or analytical insight, modelling decisions, simulation runs, data analyses, or conclusions; every one of those is the authors' own work. The tools acted as writing and organisational aids only, improving clarity, readability, and traceability. The whole arrangement is disclosed here for transparency's sake.

1.6. Outline

The thesis runs to seven chapters. The present chapter has introduced the background, the problem statement, the purpose and research questions, and the scope. Chapter 2 builds the theoretical framework: HMLV production systems, automation in assembly, discrete-event simulation, and workload control. Chapter 3 describes the research method: the research process and the systematic literature review; then the modelling logic, and the verification and validation procedures. Chapter 4 reports the outcomes of the simulation experiments, sensitivity analyses included. Chapter 5 works out what those outcomes mean for the research questions; Chapter 6 then sets the findings against the published literature, which brings in the practical implications plus a critical look at the method and the study's limitations. The closing chapter summarises the contributions and points to directions for further research.

2. Theoretical Framework

This chapter builds the theoretical groundwork for the study, built out of a structured review of the peer-reviewed literature. One question organises that review: what must the simulation model be able to do? Five areas supply the answer, and each section closes by setting out the demands its concept places on the model. Establishing simulation credibility comes last; the classification of automation configurations and workload control as a mechanism of release occupy the middle of the chapter; the operational features of HMLV production open it, with DES as a tool for structural assessment alongside them.

2.1. High-Mix, Low-Volume Production Characteristics

The study evaluates at the level of the assembly cell, leaving the individual workstation aside. That choice rests on a widely held account of a production system as a socio-technical arrangement of material, information, and human resources working together to transform inputs into outputs—with performance depending on the arrangement taken as a whole and not on any single component, as Bellgran and Säfsten (2010) describe it.

HMLV production usually means making a broad range of products in small batches, built to order instead of kept in stock. Sifting 152 publications on HMLV manufacturing in a structured review, Gan et al. (2023) report that production planning has drawn most of the research attention; the remaining aspects of the production process, and of the wider value chain, attract considerably less. The concentration of effort on the planning function suggests that the structural performance of the assembly operations being planned remains comparatively underexplored: the planning of such systems is well covered, whereas the operations being planned are not.

Variability is the primary operational challenge, since each variant demands its own task sequence, processing time, and routing path. Workstation loads are therefore uneven, and the constraint shifts instead of holding still. Yang et al. (2025) examine exactly this behaviour in a discrete-event study of an aerospace assembly workshop, tracing how changes in configuration alter production capacity, workstation utilisation, and the location of the bottleneck. Normand and Bradley (2024) draw out what this means for improvement practice: HMLV environments call for a degree of operational flexibility and customer responsiveness that methods designed for low-mix, high-volume settings were never built to deliver, and those methods therefore cannot be transferred across unchanged.

Changeover adds further variability, and it does so out of proportion to its duration, because small batches spread the time lost to setup over fewer units than high-volume production does. The industrial data reported by Miqueo et al. (2022) for two HMLV assembly lines make this concrete: average setup times of 6.85 and 8.35 minutes against average cycle times of 5.42 and 4.65 minutes. On both lines the average changeover therefore exceeds the time needed to produce a single unit at a workstation. Setup cannot be treated as a second-order correction in this setting, since it is comparable in magnitude to the productive work itself.

A further consequence of variability is that the performance indicators do not move together. Gödri (2022, p.1) frames daily production management in HMLV as the problem of operating “in face of conflicting key performance indicators such as high delivery precision, short lead time, and maximal resource utilization.” An intervention

that improves one of them may well worsen another, so the effect of a structural change cannot be judged from a single indicator.

The three indicators most often used to characterise this behaviour, namely throughput, work-in-process, and cycle time, are interdependent rather than independent, and shop-floor interventions are usually assessed across all three at once (Shafiq et al., 2020). Kang et al. (2016) formalise the relationship by arranging operational indicators into a hierarchy in which system-level measures are derived from station-level ones, making explicit how overall performance depends on the behaviour of individual stations.

Reported performance for HMLV assembly lines varies widely, which limits what any single external figure can be asked to establish. Miqueo et al. (2022) report average cycle times of 5.42 and 4.65 minutes for two industrial lines, implying gross rates of roughly 11.1 and 12.9 units per hour, whereas Miqueo et al. (2023), simulating a manual fixed-worker line for a different product, report 25.5 units per hour. The spread is usually put down to differences in work content rather than to differences in system quality. A model of this kind can therefore be checked for whether it operates within a plausible range, but it cannot be calibrated against any one published figure.

2.2. Automation in Assembly Systems

Automation has traditionally belonged to high-volume, repetitive manufacturing, where standardised processes make mechanisation straightforward. Advances in flexible robotics, reconfigurable fixtures, and vision-guided systems have since opened it to more variable environments. Reviewing the role of automation in HMLV, Johansen et al. (2021) describe the resulting situation as a tension: rising automation and increasingly customised products together press production systems to assemble a high mix in low volume more efficiently, and the industry is still looking for ways to do so.

Individual studies show that the level of automation produces quantifiable differences in the performance of assembly cells. Ames Zegarra and Indukaladharan (2021) simulate a manual assembly line alongside a semi-automated line in which robots perform specific tasks, and report gains in overall efficiency and throughput rate together with a marked difference in work-in-process. Studies of this kind confirm that the effects exist, but not the conditions under which they arise, since each is carried out on a single system under a single demand assumption.

2.2.1. The Levels of Automation Framework

The principal framework for classifying automation setups is the Levels of Automation (LoA) taxonomy proposed by Frohm et al. (2008). Its central claim is that a single measure of automation will not serve in manufacturing, and that automation must instead be split into two dimensions: mechanical LoA, which concerns the replacement of human physical effort, and cognitive LoA, which concerns the replacement of human information processing and control. Each is rated on a seven-level scale, shown in Table 1. Because the two dimensions vary independently, a task can be physically automated while remaining cognitively manual, or the reverse.

Table 1
LoA reference scales for mechanical and cognitive tasks (adapted from Frohm et al., 2008)

Level	Mechanical LoA (physical support)	Cognitive LoA (information support)
1	Totally manual: no tools; muscle power only	Totally manual: user relies on own experience and knowledge
2	Static hand tool: support of static tools	Decision giving: user receives information on what to do
3	Flexible hand tool: support of flexible tools	Teaching: user receives instructions
4	Automated hand tool: support of automated tools	Questioning: technology verifies execution before action
5	Static machine: automatic work by a task-designed machine	Supervision: technology alerts the user
6	Flexible machine: reconfigurable machine	Intervene: technology takes over and corrects if needed
7	Totally automatic: machine handles all deviations	Totally automatic: all information and control handled by technology

Table 1 shows the two scales defined in parallel while remaining mutually independent, and it is that independence which makes the framework usable as an experimental variable. A configuration need not be described by a single ordinal quantity of automation; it can be described instead by which tasks have been reallocated, and to what. That distinction is what determines how the resources of the system come to be contended for.

Empirical parameters for the mechanical dimension come from research on collaborative assembly. Evaluating manual and human-robot collaborative workplaces, Ojstersek et al. (2024) report that manual assembly processing time follows a normal distribution with a mean of 85 s and a standard deviation of 12 s, while the collaborative version has a standard deviation of 6 s. The fall in dispersion, alongside the fall in the mean, can be read as a characteristic signature of mechanical automation, machines being both faster and more consistent than operators.

2.2.2. Structural Trade-offs

Where automation is placed generally weighs as heavily as the amount installed. That priority of placement over quantity surfaces in Miqueo et al. (2023), whose industrial case concerns walking-worker assembly lines run in parallel under HMLV demand: “none of the automated stations are the AL bottleneck. Therefore, in this particular case, the number of automated stations would not significantly impact the AL operational KPIs.” That remark does a narrow job in their paper—it is the reason automation layout drops out of their experimental design—and they offer it in passing, not as a conclusion. Read on its own, though, it names the mechanism: automation applied away from the constraint leaves system-level performance unchanged, at any level of installation.

Trade-offs, rather than uniform gains, are what a higher level of automation tends to purchase in a variable environment. Herps et al. (2022) demonstrate this across three concepts of automation for one HMLV manufacturing system. One design pairs parallel buffers with a dedicated robot, in contrast to shared buffers served by automated guided vehicles (a hybrid of the two completing the set). The spread in profit ran from 10 to 30 per cent. What, then, separates the three is not how much automation each contains but how that automation is arranged, and the arrangement is what decides the outcome. Their layout also incorporates a manufacturing preparation centre: fixture build-ups are prepared there, away from the machine, so setup work stays decoupled from the productive cycle. For reconfigurable matrix production, Perwitz et al. (2022) arrive at a comparable conclusion, quantifying benefits of flexibility that conventional line configurations cannot deliver.

2.3. Discrete-Event Simulation for System Evaluation

Processing times are stochastic, queues form, resources are allocated dynamically; this is what makes manufacturing systems defy static analysis (Banks et al., 2004). DES instead advances the system state only at discrete event points—an arrival, a completion, the release of a resource. Queues, routing logic, and capacity constraints are therefore represented explicitly, not aggregated away. The method is by now well established here. In their survey, Negahban and Smith (2014) reviewed 290 simulation publications published between 2002 and 2013; they fall into manufacturing system design, plus manufacturing system operation, and simulation language development.Reviewing assembly optimisation applications specifically, Prajapat and Tiwari (2017) identify what-if scenario analysis as the method most commonly employed, and time and throughput as the dominant objective functions. That scenario analysis dominates the application literature suggests the approach adopted here is the field's standard treatment of this class of question.

The advantage of simulation over analytical modelling is specific rather than general. Biazen and Gebeyehu (2019) build both an analytical and a discrete-event model of the same assembly line and find close agreement between them on work-in-process, queue cycle time, cycle time, and resource utilisation; the two approaches describe the existing system equally well. The difference appears only once the system is altered. Reassigning operators from idle stations to the bottleneck station reduced waiting time by 12 per cent and work-in-process by 13 per cent, and that is a structural intervention the analytical formulation had no means of expressing. Simulation is therefore not needed in order to describe a system of this kind, but in order to change one and observe what follows.

Two alternative paradigms were considered and set aside. Agent-based simulation is most effective where system behaviour arises from decentralised autonomous decision-making among agents, and system dynamics deals in aggregate flows rather than individual entities. Neither captures the discrete progression of individual products through contested workstations, which is the mechanism under investigation here.

Simulation output is increasingly used as an input to operational decisions rather than solely for offline analysis. Mahmoodi et al. (2024) develop a data-driven, simulation-based decision support system for resource allocation in Industry 4.0 environments, in which simulation results feed allocation decisions directly. The distinction matters when a model is being evaluated: a model supporting a single structural decision has to

be credible about mechanisms, whereas a model embedded in an operational loop has to be accurate about magnitudes as well. The present study belongs to the first category.

Applied prospectively, the method serves as decision support. Goyal et al. (2026) present a scenario-based simulation framework for early-stage production system development, extending simulation from process validation into conceptual design and into the justification of investment under uncertainty. Applying scenario-based simulation to an automated machine cell, Alkanafani and Arabaci (2025) report that changes to the layout improved performance more than increases in equipment speed did, which illustrates both the value of testing configurations virtually and the tendency of structural factors to outweigh factors of rate.

2.4. Workload Control and CONWIP

Releasing work into a production system without control generally produces congestion. Kasper et al. (2023) state the underlying relationship plainly: reducing work-in-process correlates with benefits such as predictable throughput times and better manageability. Their review separates CONWIP and Kanban, designed for repetitive manufacturing, from LUMS COR and POLCA, developed for high-variety manufacturing, and notes that all the conventional methods are hierarchical, in that they settle release and authorisation before dispatching.

The weight of the release decision has been measured. Analysing 576 experimental scenarios on a bottleneck-oriented flow shop with capacity adjustment, Prabhu et al. (2025) attribute 80.2 per cent of throughput performance and 42.84 per cent of flow time performance to the order release method. Release cannot, on that evidence, be treated as a secondary control parameter.

Constant work-in-process is the mechanism used here, on the rationale that controlling how much work is released is a more direct lever on flow than managing the order in which work is processed. Spearman (2014) situates this principle within a broader argument that manufacturing systems follow predictable regularities rather than being governed solely by managerial practice. Frazee and Standridge (2016) describe the mechanism as arguably the simplest form of pull system: a single loop spanning the line, in which the departure of a completed job authorises the release of a new one, with one parameter capping total work-in-process. Its advantage in high-variety settings is that it does not require the products in the loop to share a process flow, which removes the need for the variant-specific card sets that Kanban demands. Their comparison with POLCA shows that POLCA can achieve lower average lead times at a given WIP level, because it constrains work in specific areas rather than only in aggregate, with the two systems converging at higher WIP levels.

Setting the limit is a modelling problem in its own right. Khan and Standridge (2019) use aggregate simulation modelling to determine the CONWIP parameter for an HMLV cell, combining over one hundred product types into a single arrival process and a single operation time distribution per operation. Their findings bear directly on this study: a CONWIP parameter of 50 produced throughput comparable to unconstrained release, while raising it to 58 reduced lead time by roughly 20 per cent, so the limit governed lead time without restricting output.

The evidence on how the mechanism interacts with variety is mixed, and is best reported as such. Harrod and Kanet (2013) compare four workflow control mechanisms

against three queue disciplines in a five-work-centre make-to-order job shop and conclude that the choice of queue discipline matters considerably more than the choice of control mechanism. They also observe that workflow control reduces shop-floor inventory while increasing total order inventory, and they document a behaviour they term lockup, a form of terminal blocking. Mueller and Schmitt (2020) show that order release sequencing in autonomous matrix-structured assembly is sensitive to the similarity between the precedence graphs of product variants, lower similarity improving throughput times and capacity utilisation. Thürer et al. (2017) find that backlog sequencing based on capacity slack improves CONWIP performance considerably where variety is high. The mechanism also reaches beyond conventional discrete assembly: Thürer et al. (2021) apply workload control to additive manufacturing shops in which post-processing rather than printing is the primary constraint, and show that simulation-based assessment is an effective way of evaluating release policies before implementation, particularly where, as here, the constraint does not sit where intuition would place it. Gosavi and Gosavi (2024) argue that conventional threshold-setting for CONWIP fails in digitised environments because it depends on offline optimisation, and propose closed-form, data-driven thresholds in its place.

Taken together, this body of work shows CONWIP to be a mechanism whose effect depends on the setting in which it is used. It limits inventory consistently, but its influence on throughput, lead time, and the placement of queues varies with product variety, with sequencing, and with the relationship between the limit and the system's natural work-in-process level. Whether the particular value of the limit matters in a demand-constrained HMLV environment is examined empirically in Section 4.5.

2.5. Verification and Validation

A simulation model supports conclusions only once its credibility has been established. Sargent (2013) provides the standard approach, distinguishing conceptual model validity, computerised model verification, operational validity, and data validity, and setting out techniques for establishing each. Those techniques include animation, comparison with other models, degenerate tests, event validity, extreme condition tests, face validity, validation against historical data, internal validity, operational graphics, parameter variability and sensitivity analysis, predictive validation, traces, and Turing tests.

The techniques are neither interchangeable nor equally accessible. Several of them, among them historical data validation, event validity, predictive validation, and Turing tests, require access to the operating system being modelled. A study built on a conceptual system rather than an industrial case cannot apply these, so whatever credibility they would have established has to be sought elsewhere. This is the methodological condition of the present study, and it is what RQ2 asks about.

The condition is not peculiar to conceptual studies. Olsen and Raunak (2019) identify it as an instance of the oracle problem: a simulation has to be validated if its results are to be relied on, yet frequently no oracle exists against which to verify it, precisely because the model was built to uncover what the real system cannot disclose. Their answer is metamorphic testing, which constructs pseudo-oracles from relationships that must hold between a model's parameters and its behaviour, so that validity is assessed through internal consistency rather than through anything observed externally. Mielczarek (2026) confronts the same difficulty via model-to-model comparison: a model validated against a second model of the same system, independently built, the

system itself staying unobservable. Both methods stand on a relation that has to hold. They stand there because the alternative is data that cannot be obtained.

Writing about complex systems of systems, Zeigler and Nutaro (2016) argue that any single verification approach used on its own is severely limited in the range of conditions it can test, and that formal analytic and simulation-based methods should be integrated rather than relied on separately. If no single approach suffices, then a model with limited scope for external validation must demonstrate its internal consistency by several independent means rather than by one.

On operational validity, Leal et al. (2011) compile the statistical techniques available for comparing simulated output against reference behaviour. They note that published descriptions of operational validation differ widely in how far they emphasise the process, and in whether the techniques used are objective or subjective. Stating clearly which technique has been chosen, and what it is meant to establish, is therefore more than a procedural matter; it forms part of the contribution.

A further risk to operational validity arises from the conditions under which a simulation starts. Models are typically initialised empty and idle, which does not correspond to the steady-state behaviour of a production system that runs continuously, so statistics collected during the transient that follows are biased towards that artificial starting point (Banks et al., 2004). Two remedies are available. The transient can be identified and removed, often by applying a moving-average method to the output data, or the run can be extended until the effect of the transient becomes negligible against the data gathered later (Banks et al., 2004). Either is defensible—truncating the transient removes the bias, extending the run dilutes it—so long as the choice is stated openly and its sufficiency is demonstrated, not merely assumed. Analytically, one such relation is available. Little's Law states that when a queueing process has finite means and its arrivals are strictly stationary and metrically transitive, the average number of units present equals the arrival rate multiplied by the average time a unit spends in the system (Little, 1961). A simulation measures all three quantities independently, so their agreement checks internal consistency without drawing on external data. It is a proven relationship, not a benchmark; any model violating it is simply wrong, no matter how plausible its outputs look.

There are also limits, and they belong next to what verification can establish. Banks et al. (2004) caution that no amount of logic verification lifts the bound that the quality of a model's inputs places on its predictive value.

3. Research Method

How the study was carried out is the subject of this chapter. Six elements follow in sequence. The research design comes first; then the protocol for the systematic literature review through which data were collected; the conceptual model and how it was derived; the experimental design next; and the verification and validation procedures after that. The chapter closes with the measures taken to ensure research quality.

3.1. Research Design and Approach

3.1.1. Research Process Overview

The study adopts a simulation-based design, both explanatory and predictive at once, which follows the methodological framework set out by Säfsten and Gustavsson (2024). Its explanatory side lies in trying to establish why performance differs betweenautomation configurations rather than merely that it does, and predictive in that the model is used to anticipate behaviour under conditions not yet observed.

The research proceeded in four stages. The opening stage fixed the objectives and gathered the data through three systematic literature searches. Only then did design work begin: the second stage converted the theoretical requirements set out in Chapter 2 into a specification that made system structure, resources, and control logic explicit (Section 3.3)—the conceptual model. Build came third. There the model was implemented in AnyLogic 8PLE and then verified and validated. The fourth stage compared the three automation configurations under identical conditions and examined the behaviour of the model across a range of demand levels and control parameters, to establish whether the observed differences held beyond a single operating point.

The study is quantitative but non-empirical, in that no primary data were collected from an operational production system; the input parameters were taken instead from published studies reporting explicit stochastic parameters. The modelled system is representative rather than a replica of any particular facility. What this design choice implies for validity is discussed below, and the question of how credibility can be established under such a constraint is treated as one of the study's research questions rather than as an incidental limitation.

3.1.2. Research Strategy

Simulation was chosen as the research strategy because the research questions concern the effect of structural change on system-level behaviour. Demonstrating such effects experimentally on an operational line would mean reconfiguring that line three times under controlled demand, which is not practicable. Analytical modelling can describe a system in steady state, but, as Biazen and Gebeyehu (2019) illustrate, it cannot represent the reallocation of resources between stations, which is precisely the intervention under investigation here.

DES was chosen over agent-based simulation and system dynamics for the reasons given in Section 2.3. The information that matters here is where an individual job waits and which resource it waits for, and that information exists only at the level of individual entities and specific resource allocations.

3.2. Systematic Literature Review

Literature was gathered through three separately scoped searches, each following the phased approach described by Säfsten and Gustavsson (2024), which runs from problem definition through structured screening to synthesis. The searches were carried out in sequence rather than in parallel because the scope of each was set by the outcome of the one before: the theoretical framework search followed the definition of the research questions, and the verification and validation searches followed the reformulation of RQ2.

3.2.1. Screening Procedure

Two screening procedures were used across the three searches, reflecting the stage the project had reached when each was carried out.

Search 1 was run during the parameterisation stage, before the review protocol had been formalised. Records were therefore screened manually by title, abstract, and conclusion, following the phased approach of Bairagi and Munot (2019), and duplicates were identified by inspection. Because the search hunted specific numeric inputs—distribution families and their parameters—a single explicit criterion was enough to settle eligibility, and interpretive judgement had little room to enter. Searches 2 and 3 came later. By then the scope had widened from parameter extraction to take in theoretical framing and methodological review. Eligibility now turned on judgements of relevance rather than on the presence of a specified parameter, so a more structured procedure was adopted. Records were exported and imported into Rayyan, a web application built to support the screening phase of systematic reviews (Ouzzani et al., 2016). It served three functions: automated detection of duplicates across databases, structured screening of titles and abstracts with an explicit include, exclude, or maybe decision recorded against every record, and the maintenance of an auditable trail of those decisions.

Screening in Searches 2 and 3 was carried out record by record, independently by both authors, using Rayyan's blind mode, in which each reviewer's decisions stay hidden from the other until screening is complete. Records marked "maybe" at the title and abstract stage were carried forward to full-text assessment.

Dual independent screening was used for these two searches rather than single screening because one reviewer assessing thousands of records against relevance criteria will tend to judge marginal cases inconsistently, and those inconsistencies stay hidden when there is no second assessment to compare against. Blinding prevents the second reviewer from being anchored by the first. The procedure was not applied retrospectively to Search 1, whose eligibility criteria were narrower and left less scope for divergent judgement.

Disagreements were settled by discussion once blind screening had finished, with reference to the full text where the abstract alone was not sufficient. Five conflicts arose in the HMLV automation search and three in the verification and validation search. A conflict rate this low suggests the two reviewers understood the inclusion criteria consistently, though consistency alone does not establish that the criteria were correct, since two reviewers sharing a misunderstanding would agree just as readily.

Using a dedicated screening application rather than a spreadsheet carries an important methodological benefit. Every decision is logged against a specific reference together with its reason, and the two reviewers' decisions are kept separate. The screening process is therefore reproducible: a third party could revisit the same set of records, decisions, and points of disagreement. This addresses a common weakness in literature-based data collection, where the reduction from a large set of retrieved items to a small set of included ones is often reported as a single, unexplained step.

3.2.2. Search 1: Core Parameterisation

The first search aimed to identify the stochastic parameters required to build and calibrate the model. It was run in Scopus and in Primo (Jönköping University Library) using the string:

("High-Mix Low-Volume" OR "HMLV" OR "Job Shop" OR "Make-to-Order") AND ("Discrete-Event Simulation" OR "DES" OR "AnyLogic") AND ("Human-Robot Collaboration" OR "HRC" OR "Cobot" OR "Automation" OR "Matrix Production")

The search returned 1,534 records, which fell to 893 once 641 duplicates had been removed. Full-text assessment was then carried out on the 150 records that met the inclusion criteria, namely explicit stochastic parameters and relevance to HMLV assembly; of these, 11 studies were retained and used to parameterise the model. Appendix A presents the screening process as a flow diagram.

3.2.3. Search 2: HMLV Automation Literature

A second search addressed the theoretical framework. It was run in Web of Science and Scopus, restricted to publications between 2016 and 2026 and to English-language material. Document types were limited in Web of Science to articles; Scopus admitted articles, books, and conference papers.

The first topic string read (“High Mix Low Volume” OR “High Variety Low Volume” OR “Mixed model production”) AND (“Automation Configuration” OR “Automation architecture” OR “Cyber physical production system”) AND (“Discrete Event Simulation” OR “Agent based simulation” OR “Monte Carlo simulation”). An all-fields run in Web of Science brought back far more records than anyone could screen, so I narrowed the string and re-ran it against the topic field alone:

(“High Mix Low Volume” OR “High Variety Low Volume”) AND (“Automation Configuration” OR “Automation architecture” OR “Cyber physical production system”) AND “Discrete Event Simulation”

That narrowed version pulled 2,562 records from Web of Science, plus 133 from Scopus. Rayyan took in all 2,695, flagged and removed 16 duplicates, and left 2,679 for title and abstract screening. Thirty-three records survived to full-text review. Five of those were retained for the theoretical framework; the rest, on full reading, either failed the inclusion criteria or repeated material already covered.

3.2.4. Search 3: Verification and Validation Methodology

The third search took up the verification and validation methodology set out in Section 3.5. Web of Science again, same filters, with a refined query:

“verification and validation” AND (“discrete event simulation” OR “simulation model”) AND (“High Mix Low Volume” OR “job shop assembly”)

It yielded 2,744 records. Scopus had already been tried with a broader query and gave back only five records, and since that database covers these terms so thinly, it was not searched again. Screening of the retrieved set followed the same Rayyan procedure, with 10 duplicates removed. Title and abstract assessment carried 22 records through to full-text review, of which eight were retained.

This search is the source of the verification and validation literature discussed in Section 2.5 and applied in Section 3.5. It followed directly from the wording of RQ2: establishing which validation techniques remain available when empirical system data are unavailable called for a dedicated review rather than reliance on a single methodological source.

3.2.5. Supplementary Sources Outside the Database Searches

Not all of the literature used in this thesis came from the three database searches. The difference is deliberate rather than accidental: three categories of source were obtained by other means.

The reference lists of the studies retained from Searches 2 and 3 were examined for works cited repeatedly as the origin of a concept that the searches had picked up only in application. This is the backward step of the snowballing procedure described by Wohlin (2014), in which a starting set obtained by database search is expanded by following the references it cites. The step was applied selectively rather than exhaustively: a reference was followed only where more than one of the retained papers attributed a concept central to this study to it.

Snowballing also surfaced studies published within the search window that the database queries had not returned, either because they appeared in book series and conference proceedings outside the indexed journal coverage, or because they used terminology that the search strings did not capture. Wohlin (2014) cites this under-retrieval as a reason to supplement database searching rather than rely on it alone.

This route was essential because the window for Searches 2 and 3 opened only in 2016, while several of the foundational works those searches turned up are older: the proof of the queueing relation used for internal validation goes back to Little (1961) and the relation between utilisation and queueing to Kingman (1961), and there is Frohm et al.'s (2008) Levels of Automation taxonomy None of these could have been retrieved by a search confined to the last decade, and excluding them would have meant citing the applications of these frameworks without acknowledging the frameworks themselves.

3.3. Conceptual Model

The conceptual model is an abstraction of the HMLV assembly system, specified before implementation (Banks et al., 2004).

3.3.1. System Structure and Flow Logic

The system is modelled as a serial mixed-model assembly line with five workstations, S1 to S5. Every product variant passes through all five workstations in the same order.

Variety is expressed through variant-specific work content and sequence-dependent changeover rather than through different routings. Four variants are produced, in proportions of 40, 30, 20, and 10 per cent, and they carry work-content factors of 0.8, 1.0, 1.3, and 1.6 respectively, which multiply the base processing time at each station. The mix is deliberately unbalanced, reflecting the loading asymmetry typical of HMLV production: the most common variant is not the most demanding one. Work at each workstation follows a seize-delay-release protocol. A job seizes a resource unit, holds the station while the operation runs, and lets the unit go when it ends. Concurrency at a station is therefore limited by the availability of that resource, not by an assumed one-job-per-station rule. Whenever an incoming job belongs to a different variant from the one the station handled immediately before, a changeover fires. Each station remembers the last variant it saw, and because that memory is local, changeover time accumulates independently along the line, tracking the sequence that actually occurs rather than the mix as a whole. In the baseline configurations, one changeover costs 60 seconds.

3.3.2. Workload Control

A CONWIP loop spanning the whole line governs order release. Entry to S1 opens only when a completed job leaves S5, so total work-in-process stays steady. Mechanically, the loop is a seize-release pair drawn on a token pool: the seize sits before the first station, the release after the last.

Where did the limit come from? Little's Law (Little, 1961), applied to published benchmarks, pointed to roughly 18 units. But the model never gets near that. At the operating demand level of 13 jobs per hour, the manual configuration settles at 4.33 units of work-in-process; a ceiling of 18 would never bind, and the control mechanism would simply stay inactive. So 10 units it was; high enough not to starve a five-station line, low enough to bite once congestion sets in.

That value grew out of the model's own behaviour, not an independent source, so its influence is treated as an experimental variable rather than a fixed assumption; Section 4.5 tests it directly across 3 to 12 units. Every configuration, though, carries the same limit. That way performance differences trace back to automation rather than to varying congestion.

3.3.3. Resources

Two resource pools are defined in the model. Three operators make up the operator pool, shared across every operator-paced workstation, so labour is contended at cell level instead of being an attribute of any one station. The second pool, the automated-station pool, serves stations whose physical work has been automated and which draw no operator capacity.

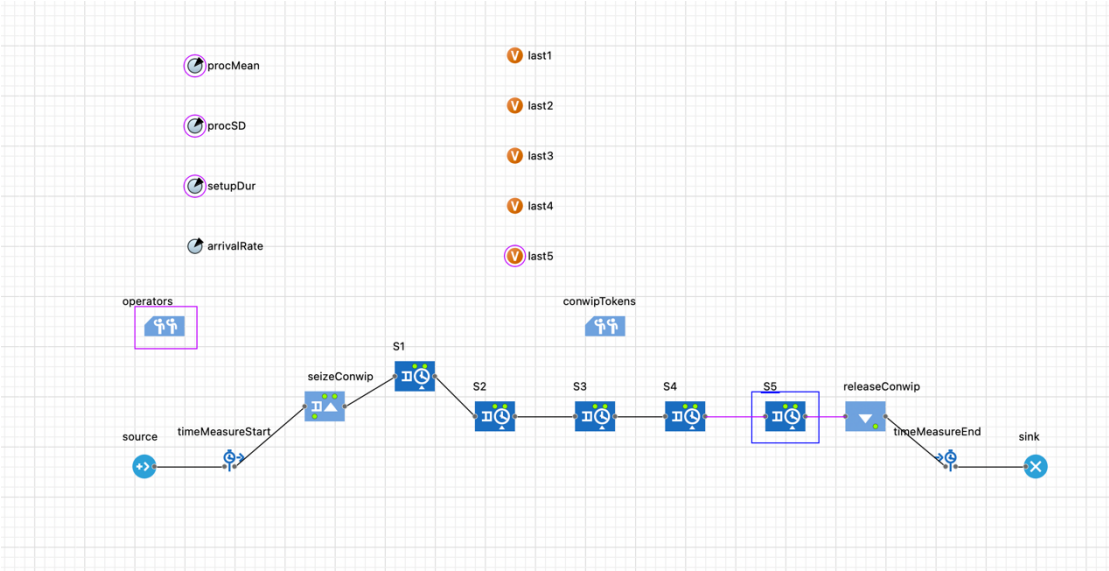
The operator count never moves from three, in any configuration. That is the central design choice of the experiment: with labour held constant while automation increases, a performance difference can only reflect how tasks have been redistributed, never how many resources are available. A design that reduced headcount alongside automation would confound the two effects.

3.4. Experimental Design

The model was implemented in AnyLogic 8 Personal Learning Edition (PLE) using the Process Modelling Library, and its block structure is shown in Figure 1.

Figure 1

AnyLogic model structure: source, CONWIP seize, five workstations, CONWIP release and sink



AnyLogic 8 PLE was selected for three reasons. First, its Process Modelling Library provides the discrete-event constructs the conceptual model requires: source, seize, service, release, and sink, together with resource pools that support contention between stations for a shared operator. Operator contention is central to this study, since the distinction between the semi-automated and automation-assisted configurations rests on whether the operator is released from the processing loop. A tool without first-class contended-resource semantics would have required that behaviour to be coded by hand. Second, AnyLogic exposes the underlying Java, which allowed the sequence-dependent changeover matrix and the CONWIP release loop to be implemented directly rather than approximated. Third, AnyLogic appears throughout the reviewed literature on the simulation of HMLV and human-robot collaborative assembly and was included as a term in Search 1, so the results obtained here are directly comparable to published work.

The Personal Learning Edition carries restrictions: models run in a limited execution mode, and the built-in optimisation experiments are unavailable. Neither affected this study, which required repeated fixed-parameter runs rather than a search over a parameter space. The edition is freely available, so the model and every result reported here can be reproduced by any reader without a commercial licence.

3.4.1. Configurations

Scenario 1: Manual. All five workstations are operator-paced. Each operation requires both the workstation and a free operator from the shared pool of three. Processing time is normally distributed, with a mean of 85.0 seconds and a standard deviation of 12.0 seconds, scaled by the variant work-content factor.

Scenario 2: Semi-automated (human-robot collaboration). Collaborative equipment assists the operator at each workstation, reducing the mean and the standard deviation of processing time to 55.0 seconds and 6.0 seconds respectively. The operator stays engaged throughout every operation at every station, so the operator pool continues to serve all five workstations.

Scenario 3: Automation-assisted. Workstations 1, 2, and 4 are automated. They run at a mean of 40.0 seconds with a standard deviation of 3.0 seconds and draw on the automated-station pool, so they require no operator capacity. Workstations 3 and 5 remain operator-paced, retaining the manual processing time of 85.0 seconds with a standard deviation of 12.0 seconds. The pool of three operators therefore serves two workstations rather than five.

Scenario 3b: Automation-assisted with reduced changeover time. Everything matches Scenario 3, with one exception: changeover drops from 60 seconds to 30, which mirrors the offline preparation of fixtures and tooling described by Herps et al. (2022). A sensitivity case, nothing more.

Scenarios 2 and 3 differ, and the difference is deliberate — it sits at the core of the experiment. Processing time falls in both. Yet only Scenario 3 changes which resource a station competes for. Comparing the two therefore separates the effect of speed from the effect of task reallocation, which the Levels of Automation framework treats as distinct dimensions (Frohm et al., 2008).

Workstations are modelled as work locations rather than as capacity-constrained machines. A workstation places no inherent limit on the number of jobs it can hold, so concurrent processing is constrained only by the availability of the resource pool serving that station. This follows from the study's focus on contended labour rather than on equipment occupancy, and it has two consequences. Reported station utilisation measures the load offered to a workstation rather than the fraction of time it is occupied, and may therefore exceed 100 per cent when more than one job is present. The capacity ceilings reported in Chapter 4 are correspondingly those of the labour pool rather than of the physical line.

3.4.2. Experimental Variables and Boundary Conditions

The independent variable is the automation configuration. The dependent variables, evaluated simultaneously, are throughput, lead time, work-in-process, and resource utilisation.

Demand, product mix, work-content factors, changeover duration, operator headcount, CONWIP limit, and run length are held identical across all configurations. In the baseline experiments, jobs arrive as a Poisson process at a rate of 13 per hour.

Two further aspects are examined. The arrival rate is swept from 13 to 48 jobs per hour to locate the capacity ceiling of each configuration, and the CONWIP limit is swept from 3 to 12 units to test how sensitive the results are to the workload control parameter (Section 4.5).

3.4.3. Run Length, Initialisation and Replication

Each run covers 200 hours of simulated time, which yields roughly 2,600 completed jobs.

Of the two standard responses to initialisation bias, the second was chosen. Rather than truncating the transient with a moving-average procedure, the effect of the empty-and-idle initial state was measured directly by running the manual configuration for three different durations. Throughput was 12.68 jobs per hour at 50 hours, 12.92 at 200 hours, and 13.01 at 500 hours. The 50-hour run is up to 2.5 per cent lower than the longer runs, while the 200-hour and 500-hour runs differ by less than 0.7 per cent. Initialisation bias is therefore negligible at the run length selected, and no warm-up period is required.

Each configuration was replicated five times in the baseline comparison, using fixed random seeds 1 to 5, so that every baseline result can be reproduced exactly. The sensitivity experiments, namely the CONWIP sweep, the load sweep, the automation-depth sweep, and the automated-processing-time sweep, use ten replications per point with independently drawn seeds. The capacity ceilings pool all replications at the loads where throughput no longer tracks the load offered, giving the counts recorded in Table 12. Coefficients of variation across replications ranged from 2.5 to 5.6 per cent, and 95 per cent confidence intervals are reported alongside every mean in Chapter 4.

3.4.4. Model Parameters

Table 2 summarises the model parameters. The full parameter set, including how each value was derived and the distribution fitting, is set out in Appendix B.

Table 2
Model parameters

Parameter	Value	Source
Workstations	5, in series	Model design
Product variants	4	Model design
Product mix	40 / 30 / 20 / 10 %	Model design
Work-content factors	0.8 / 1.0 / 1.3 / 1.6	Model design
Manual processing time	Normal (μ = 85.0 s, σ = 12.0 s)	Ojstersek et al. (2024), Table 2
HRC processing time	Normal (μ = 55.0 s, σ = 6.0 s)	Ojstersek et al. (2024), p. 531
Automated processing time	Normal (μ = 40.0 s, σ = 3.0 s)	Scaled; see note below
Changeover time	60 s when variant differs from predecessor	Model design
Changeover time, Scenario 3b	30 s	Herps et al. (2022), decoupled preparation
Operators	3, shared across operator-paced stations	Model design

CONWIP limit	10 units	Calibrated; see Section 3.3.2
Arrival process	Poisson, 13 jobs/h baseline	Calibrated; see Section 3.5.3
Run length	200 h (≈ 2,600 jobs)	Section 3.4.3
Replications	5, fixed seeds 1–5 (baseline); 10, independent seeds (sensitivity sweeps)	Section 3.4.3

On the HRC processing time, Ojstersek et al. (2024) state in the body of their paper that "processing time is significantly reduced to 55 seconds with a standard deviation of 6 seconds, compared to 85 seconds in the manual assembly operation" (p. 531), whereas Table 3 of the same paper records the corresponding value as 42 seconds. The published text is therefore internally inconsistent. The 55-second figure given in the prose was adopted, being the more conservative of the two, so that the benefit attributed to collaborative equipment is understated rather than overstated.

On the automated processing time, Herps et al. (2022) report that machine processing times for CNC operations in an automated HMLV cell are normally distributed with a mean of 40 minutes and a standard deviation of 30 minutes. That figure describes a much larger and slower class of operation than the assembly tasks modelled here, and is not directly transferable. The value of 40 seconds used in this study was instead set at the workstation level of the modelled cell, preserving the monotonic reduction in both mean and dispersion across the three configurations (85.0/12.0 → 55.0/6.0 → 40.0/3.0 seconds) implied by the Levels of Automation framework. It is a modelling assumption rather than a parameter derived from the literature, and is reported as such.

On equipment reliability, Perwitz et al. (2022) parameterise machine failure risk at 3 per cent per station, with failures occurring roughly every 200 minutes. Equipment failure is excluded from the baseline configurations, in which all automated equipment is assumed to be continuously available; it is examined instead as a sensitivity case in Appendix C, using the mean time between failures of 200 minutes reported in that source together with a mean repair time of 6.2 minutes derived from the 3 per cent unavailability.

3.5. Verification and Validation

Verification and validation answer different questions. Verification asks whether the model was built correctly, that is, whether the implemented model behaves as the conceptual model specifies. Validation asks whether the correct model was built, that is, whether the conceptual model represents the system adequately for the intended purpose (Sargent, 2013). The distinction carries particular weight in this study because the two are not equally accessible. Verification needs only the model and its specification, and can therefore be carried out in full. Validation needs a referent against which correspondence can be judged, and because the study models a conceptual rather than an industrial system, no such referent exists. This asymmetry is a property of the research design rather than a shortcoming of the techniques selected, and it is stated here rather than deferred to the limitations chapter. Sargent (2013) describes seventeen

techniques commonly used in model verification and validation. Four of them, namely historical data validation, event validity, predictive validation, and Turing tests, require access to the operating system being modelled and were therefore unavailable. The techniques that were used, and the reasoning behind their selection, are set out below.

3.5.1. Verification of the Implemented Model

Verification established that the implemented model behaves as the conceptual model specifies, through the checks set out below.

Traces: individual jobs were followed through the model under deterministic inputs. These traces confirmed that entities pass through the five workstations in sequence, that changeover fires only where the previous variant at that station differs, and that resources are seized and released without leakage.

Degenerate and extreme condition tests: the model was run both at arrival rates approaching zero and at rates well beyond the capacity of the system. Under very light load, queues emptied, utilisation fell towards zero, and lead time converged on the sum of the processing times, as expected. Past capacity, the accumulating work stayed at the CONWIP release point and never entered the line itself—confirmation that the release constraint binds where it was intended to.

Constraint verification: work-in-process was monitored at all times, and it never went past the CONWIP limit of 10 units; no new job entered the line until a completed one had left. Resource interaction: deadlock and unintended idle states were watched for in the operator and automated-station pools. One early build deadlocked completely, because a queue limit of one at a workstation choked the flow upstream of every station. Removingthe finite queue restriction resolved this, and the revised model was re-verified against all of the checks above.

Internal consistency of independently recorded quantities: Little's Law states that, in a queueing process with finite means and stationary, metrically transitive arrivals, the average number of units in the system equals the product of the average arrival rate and the average time in the system (Little, 1961). Comparing work-in-process with the product of throughput and lead time is informative only if the two quantities are recorded by different mechanisms, since the model derives work-in-process from that product. The measurement window in this model opens at the time-measurement block preceding the release gate and closes once the token has been released, so two quantities are recorded separately: the mean time in the system, taken from the time-measurement blocks, and the mean number of jobs holding a release token, taken from the utilisation of the token pool. Combining the first with throughput via Little's Law gives the mean number of jobs within the measurement window, while the second gives the mean number within the line itself. The difference between them is the number of jobs waiting for release, which must be non-negative and small under light load. Olsen and Raunak (2019) describe validation in the absence of an external reference as an instance of the oracle problem, and propose assessment against relations that must necessarily hold. The comparison used here is such a relation. It is a verification rather than a validation technique, since it establishes that the implementation is internally coherent rather than that it corresponds to any operating system. Results are reported in Section 4.1.2.

Comparison with an analytical model: operator utilisation and the capacity ceiling of each configuration can be derived in closed form from the input parameters, independently of any simulation output. Operator utilisation is throughput multiplied by the operator time each job consumes and divided by the size of the operator pool, while the capacity ceiling is the operator supply divided by that same quantity. Mielczarek (2026) applies a model-to-model comparison in which the system itself remains unobservable, and the approach taken here is of that kind. Results are reported in Section 4.1.4.

3.5.2. Validation of the Conceptual Model

Operational validity was assessed by comparing the output of the model with published performance figures for industrial HMLV assembly lines.

Reported throughput for such lines varies considerably, and the variation reflects differences in product work content rather than differences in system quality. Miqueo et al. (2022) report mean cycle times of 5.42 and 4.65 minutes for two industrial lines at a white goods manufacturer, which imply gross throughput of approximately 11.1 and 12.9 units per hour. Miqueo et al. (2023), simulating a manual fixed-worker line that assembles a different product, report 25.5 units per hour.

The simulated manual baseline achieves 12.95 ± 0.25 jobs per hour, which falls within that range. The arrival rate was calibrated to 13 jobs per hour so as to sit inside the band reported by industry; at the 10 jobs per hour first trialled, the system was demand-starved and throughput was correspondingly lower.

The comparison shows that the model operates at a feasible industrial rate. It does not demonstrate a match with any specific line, and no such claim is made: no line documented in the literature shares this model's work content, station count, and product mix, so an exact match would be coincidence rather than evidence.

A second criterion, resource utilisation between 44 and 45 per cent, was considered and then dismissed. Herps et al. (2022) report CNC machine utilisation of 44 per cent in a real system and 45 per cent in the corresponding model, but these are validation data points for one specific machining cell rather than utilisation targets for HMLV assembly in general. Applying them alongside a throughput criterion would over-determine the calibration, since with the arrival rate as the only free parameter, throughput and utilisation cannot be set independently. Utilisation is therefore reported as a result in Chapter 4 rather than used as a validation criterion.

3.5.3. Sensitivity Analysis

Parameter variability and sensitivity analyses were carried out on the three parameters not taken directly from the literature: the CONWIP limit, the arrival rate, and the mean processing time at the automated stations. Treating these as experimental dimensions rather than as fixed assumptions allows the conclusions to be framed as conditional on demand and on the assumed automation speed, rather than resting on a single operating point. The CONWIP limit was varied from 3 to 12 units (Section 4.5), the arrival rate from 13 to 48 jobs per hour (Section 4.6), and the automated processing time from 30 to 50 seconds, that is ±25 per cent about the assumed 40 seconds (Section 4.8).

3.5.4. Limitations of the Validation Approach

Taken together, the techniques applied establish that the implemented model corresponds to its specification, that it behaves plausibly beyond the range in which it was operated, that its independently recorded quantities are mutually consistent, that its outputs follow analytically from its inputs, and that its conclusions do not depend on the three parameters that were assumed rather than derived. Zeigler and Nutaro (2016) argue that any single verification approach is severely limited in the range of conditions it can test, and that formal analytic and simulation-based methods should be integrated rather than relied on separately. The combination applied here follows that reasoning.

Four of these five techniques are, however, techniques of verification. Only the comparison against published performance ranges bears on validation, and it establishes plausibility rather than correspondence. Leal et al. (2011) observe that published accounts of operational validation differ considerably in how far they emphasise the process, and in whether the techniques applied are objective or subjective; they argue that stating the choice and its purpose is itself part of the contribution. Stated plainly, the validation evidence in this study rests on a single technique of limited strength, and the credibility of the model therefore rests predominantly on verification.

This follows from the study design rather than from the techniques chosen. A conceptual system offers no referent against which correspondence could be demonstrated, so the ceiling on what validation can establish was fixed at the point where the decision was taken to model a representative cell rather than an actual one. Banks et al. (2004) note that the predictive value of a model remains bounded by the quality of its inputs however thoroughly its logic has been verified, and that bound applies here in full.

3.6. Research Quality and Ethical Considerations

3.6.1. Reliability

Reliability rests on three mechanisms. Data collection followed a documented three-stage literature review protocol, reported in Section 3.2 in enough detail for others to replicate the searches. Screening was carried out independently by two reviewers, with disagreements recorded and resolved rather than set aside, so that study selection does not depend on the judgement of one person. In the simulation environment, the baseline results were produced with fixed random seeds and can be reproduced exactly by re-running the model with the stated seed and parameter values, while the sensitivity experiments use ten replications per parameter combination with independently drawn seeds and are reported as means with 95 per cent confidence intervals rather than as individually reproducible points.

3.6.2. Experimental Control Across Configurations

Internal validity rests on holding every boundary condition constant across the configurations. Demand pattern, product mix, work-content factors, changeover duration, operator headcount, CONWIP limit, run length, and random seeds are identical in all scenarios. The only quantities that change are the processing-time parameters and the resource pool serving each station, so observed differences in

performance can be attributed to the automation configuration. The most important control is the constant operator pool: holding headcount at three while stations are automated means the study measures the effect of reallocating work rather than of removing labour.

3.6.3. External Validity

The study examines a conceptual, representative system rather than a specific facility, so the findings support analytical rather than statistical generalisation (Säfsten & Gustavsson, 2024). They apply to HMLV assembly environments that share the structural features modelled here: serial flow, variant-specific work content, sequence-dependent changeover, a contended labour pool, and global release control. Without further testing, they do not extend to systems with variant-specific routing, parallel stations, or labour dedicated to individual stations.

3.6.4. Objectivity and Ethical Considerations

No primary data were collected from human participants and no confidential industrial data were used, so the study raises no data protection or informed-consent issues. The main ethical requirement in a simulation study is that the model should not be designed to produce a predetermined result, and two aspects of the design address this. The operator pool is held constant, which removes the most obvious means of making automation appear beneficial. Results are also reported regardless of their direction: the findings that throughput is unchanged by automation at the baseline load, and that automating most stations does not move the system constraint, are both presented despite contradicting the study's initial expectations. All secondary data, modelling logic, and parameter values are credited to their sources in accordance with APA 7, and where a parameter was set by the authors rather than drawn from the literature this is stated explicitly.

4. Results

This chapter presents the results of the simulation experiments. Section 4.1 reports the verification and validation outcomes, and Sections 4.2 to 4.4 compare the baseline performance of the three configurations. Sections 4.5 and 4.6 examine sensitivity to the workload control parameter and to demand, Section 4.7 addresses the changeover reduction case, and Section 4.8 examines sensitivity to the automated processing time.

All results were generated in AnyLogic 8 PLE. Unless stated otherwise, each simulation ran for 200 hours at an arrival rate of 13 jobs per hour with a CONWIP limit of 10, and each baseline configuration was replicated five times using fixed seeds 1 to 5. The replication data are given in Appendix C.

4.1. Verification and Validation Results

4.1.1. Run-length Sensitivity

Table 3
Manual configuration at three run lengths (seed 1)

Run length	Completed jobs	Throughput (jobs/h)	Lead time (min)	Deviation from 500 h
50 h	634	12.68	19.87	-2.5 %
200 h	2,584	12.92	19.33	-0.7 %
500 h	6,505	13.01	19.45	

Table 3 reports the manual configuration at 50, 200 and 500 hours under a single seed, to establish the run length used throughout. The 200-hour run length adopted across every experiment gives throughput to within 0.7 per cent, and lead time to within 0.6 per cent, of what a run five times longer returns. At 50 hours the figures sit as much as 2.5 per cent below both. The empty-and-idle starting state still matters there; by 200 hours it no longer does. No warm-up truncation was therefore applied.

4.1.2. Internal Consistency Check: Little's Law Estimate Against Recorded Token Holdings

Table 4
Internal Consistency Check: Little’s Law Estimate Against Recorded Token Holdings

Configuration	λ (jobs/h)	W (h)	L = λW (jobs)	Tokens held (jobs)	Difference (jobs)	Difference (min)
Manual	12.948	0.3342	4.328	4.020	0.308	1.43
Semi-automated	12.906	0.1724	2.225	2.200	0.025	0.12

Automation-assisted	13.128	0.1461	1.918	1.860	0.058	0.27
Automation-assisted with changeover reduction	12.977	0.1140	1.481	1.480	0.001	0.00

Table 4 compares the work-in-process implied by Little's Law with the CONWIP tokens actually held, for each configuration. The difference is positive in each of the configurations, as it must be, and it peaks at the manual baseline, the point where congestion is highest. Because token pool utilisation is recorded only to the nearest whole per cent, a pool capacity of ten units leaves a resolution floor of approximately 0.05 jobs. Differences above that floor—the manual and the automation-assisted cases—count as resolved; the semi-automated and the Scenario 3b differences fall within it, so they are reported as consistent with zero rather than as measured values. A negative difference, which no configuration returns, would mean jobs holding tokens outnumbering those the measurement window actually contains, and none is large enough to suggest jobs accumulating undetected. The measure counts jobs waiting for release, so the ordering makes sense: 0.31 jobs in the manual baseline, tracking congestion, falling to effectively zero in Scenario 3b. What the agreement establishes: throughput, lead time, and work-in-process hold mutually consistent inside the implementation; correspondence with any external system it does not establish.

4.1.3. Operational Plausibility

For the manual baseline, throughput comes out at 12.95 ± 0.25 jobs per hour. For comparison, Miqueo et al. (2022) report, for two industrial HMLV assembly lines, cycle-time means of 5.42 and 4.65 minutes, implying, in gross terms, approximately 11.1 and 12.9 jobs per hour. Miqueo et al. (2023), for a manual fixed-worker line that assembles a different product, give 25.5 jobs per hour. The simulated baseline sits within that range.

4.1.4. Comparison with an Analytical Model

Operator utilisation and the capacity ceilings can be calculated in closed form from the input parameters, as set out in Appendix D, and then compared with the values the simulation returns. Predicted operator utilisation agrees with the simulated value to within 1.3 percentage points in all four configurations. The three capacity ceilings agree to within 0.8 per cent: 16.6 predicted against 16.63 ± 0.04 measured, 21.8 against 21.96 ± 0.06, and 41.4 against 41.23 ± 0.05. The comparison is independent of the consistency check reported in Section 4.1.2, since it works from the input parameters rather than from the recorded outputs.

4.2. Baseline Configuration Comparison

Table 5
Performance by configuration (mean ± 95% confidence interval, five replications)

Configuration	Throughput (jobs/h)	Lead time (min)	WIP (units)	Operator utilisation (%)
Manual	12.95 ± 0.25	20.05 ± 1.88	4.33 ± 0.45	77.4 ± 0.8
Semi-automated	12.91 ± 0.33	10.35 ± 0.44	2.23 ± 0.12	58.2 ± 1.2
Automation-assisted	13.13 ± 0.52	8.77 ± 0.13	1.92 ± 0.10	30.8 ± 1.6
Automation-assisted +changeover reduction	12.98 ± 0.41	6.85 ± 0.02	1.48 ± 0.05	25.6 ± 0.8

Table 6
Changes compared to the manual baseline

Configuration	Throughput	Lead time	WIP	Operator utilisation
Semi-automated	-0.3 %	-48.4 %	-48.6 %	-19.2 pp
Automation-assisted	+1.4 %	-56.3 %	-55.7 %	-46.6 pp
Automation-assisted+changeover reduction	+0.2 %	-65.9 %	-65.8 %	-51.8 pp

Table 5 gives absolute performance by configuration with 95 per cent confidence intervals, and Table 6 restates the differences against the manual baseline—in per cent for throughput, lead time and work-in-process, and in percentage points for operator utilisation. Confidence intervals are given in Table 5 only, so significance should be judged there. Throughput is effectively unchanged across the configurations. The four means lie between 12.91 and 13.13 jobs per hour and their confidence intervals overlap. At an offered load of 13 jobs per hour, all four configurations complete very nearly all of the work presented to them. With five replications the differences cannot be distinguished, and they are therefore reported as no difference.

Lead time, by contrast, falls substantially: by 48.4 per cent in the semi-automated configuration and by 56.3 per cent in the automation-assisted configuration, reaching 65.9 per cent once changeover is reduced as well.

Work-in-process falls almost proportionally with lead time (-48.6, -55.7, and -65.8 per cent against -48.4, -56.3, and -65.9 per cent respectively). This follows from Little's Law: with throughput held constant by demand, λ is fixed, so L can fall only as W falls.

Operator utilisation falls sharply, from 77.4 per cent to 30.8 per cent. The reason is that headcount is held at three while three of the five stations are taken out of the operators' workload.

Dispersion across replications also differs markedly between configurations. The lead-time coefficient of variation is 6.6 per cent in the manual baseline, against 1.0 per cent in the automation-assisted configuration and 0.2 per cent in Scenario 3b. Greater automation therefore shortens lead time and makes it more predictable, the reduction in outcome variability following directly from the lower processing-time variability built into the parameters.

4.3. Decomposition of the Lead Time Reduction

This section separates lead time into three components: processing, changeover, and queueing. Only the total is measured directly in the model. Processing and changeover times are computed analytically from the parameters listed in Table B.1, and queueing time is taken as the residual once those two are subtracted from the total. The decomposition is therefore exact by construction rather than observed independently, which means the residual absorbs any element of lead time that is neither processing nor changeover. The calculations can be checked in full: every parameter and distribution needed is given in Appendix B.

The expected work-content factor (E[w] = 1.040) weights the station processing times, whose sum gives the expected processing time per job. For changeover, the expected time takes the number of stations, multiplies by the changeover duration, and again by the probability that two consecutive jobs differ in variant; for the mix used here that probability is $1 – (0.4^2 + 0.3^2 + 0.2^2 + 0.1^2) = 0.70$.

Table 7
Lead Time Breakdown (Minutes)

Configuration	Processing	Changeover	Queueing	Total	Queueing share
Manual	7.37	3.50	9.18	20.05	45.8 %
Semi-automated	4.77	3.50	2.08	10.35	20.1 %
Automation-assisted	5.03	3.50	0.24	8.77	2.7 %
Automation-assisted +changeover reduction	5.03	1.75	0.07	6.85	1.0 %

Figure 2
Lead time decomposed into its processing, changeover and queueing components (values in minutes)

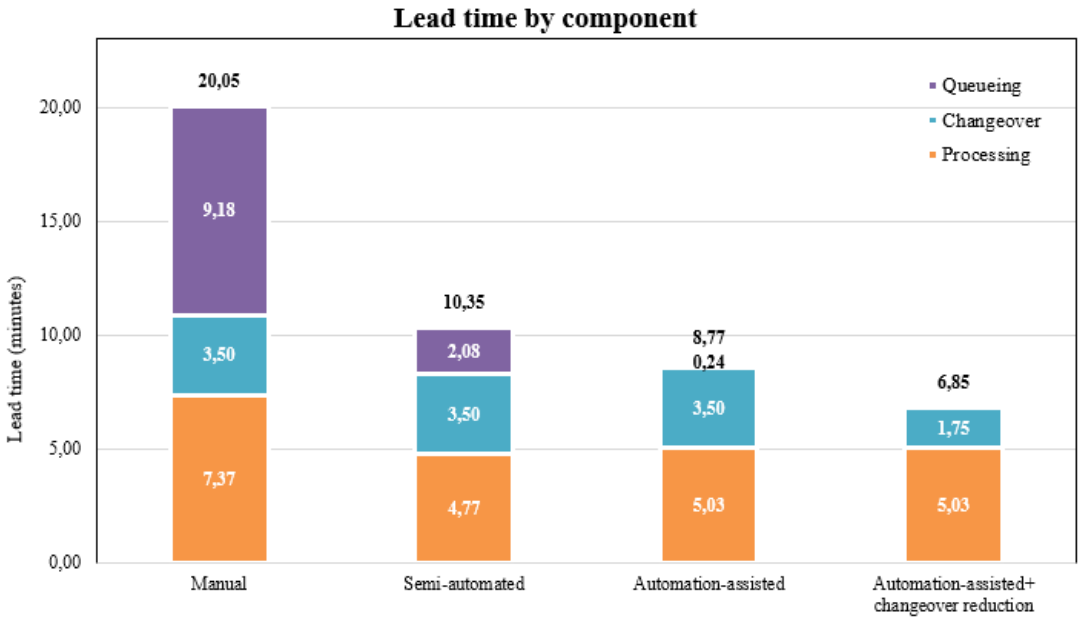


Table 7 gives the decomposition in minutes, and Figure 2 shows the same components graphically. Between the manual and automation-assisted configurations, processing time drops by only 32 per cent — from 7.37 to 5.03 minutes — far short of what the per-station cut from 85 seconds to 40 seconds would suggest. Two of the five stations remain manual: S3 and S5 continue to run at 85 seconds, regardless of what has changed elsewhere on the line.

Queueing time falls by 97.4 per cent — 9.18 minutes down to 0.24 minutes. Of the 11.28 minutes by which total lead time drops between the manual and automation-assisted configurations, queueing alone contributes 8.94 minutes, 79 per cent. In the semi-automated configuration the figure is 73 per cent.

Changeover, meanwhile, stays at 3.50 minutes across Scenarios 1 to 3: how long a setup takes depends on the product sequence and on the setup requirement, never on who or what performs the operation. With the other components shrinking, its slice of total lead time grows, from 17.5 per cent in the manual configuration to 39.9 per cent under automation assistance, at which point it becomes the largest single component.

4.4. Station-Level Utilisation

The Service blocks in AnyLogic do not report station utilisation directly. It is derived from the resource pool statistics by multiplying pool utilisation by pool capacity and dividing by the number of stations that pool serves. The derivation is exact in this model

because a station processes only while holding a resource unit from its pool, which makes resource-busy time and station-busy time identical.

Table 8
Utilisation by workstation

Configuration	S1	S2	S3	S4	S5	Resource serving S1/2/4
Manual	46.4 %	46.4 %	46.4 %	46.4 %	46.4 %	Operators
Semi-automated	34.9 %	34.9 %	34.9 %	34.9 %	34.9 %	Operators
Automation-assisted	29.8 %	29.8 %	**46.2 %**	29.8 %	**46.2 %**	Automated stations
Automation-assisted+ changeover reduction	22.0 %	22.0 %	38.4 %	22.0 %	38.4 %	Automated stations

Figure 3
Station utilisation by configuration. The dashed line marks the manual baseline

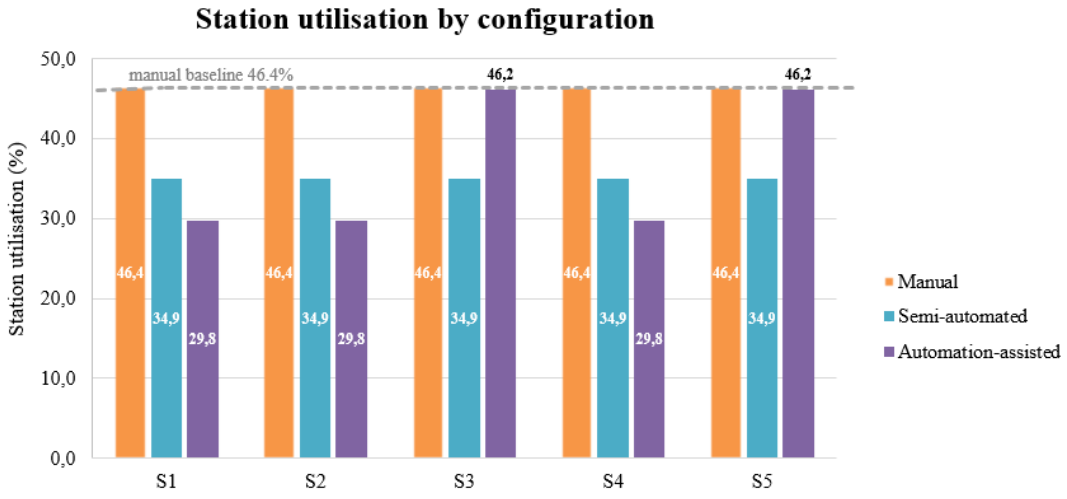


Table 8 reports station utilisation by configuration and Figure 3 plots the same values against the manual baseline. In the automation-assisted configuration, the three automated stations run at 29.8 per cent utilisation. The two stations that were not automated run at 46.2 per cent, within 0.2 percentage points of the 46.4 per cent they recorded in the fully manual configuration.

Automating 60 per cent of the workstations reduces the load on the constraint stations by 0.2 percentage points. The line as a whole is far less burdened, lead time has more than halved, and work-in-process has fallen by 56 per cent, yet the stations that set the capacity of the system are working almost as hard as before.

Reducing changeover in Scenario 3b lowers that figure to 38.4 per cent, a fall of 7.8 percentage points. Changeover reduction reaches the constraint stations because those stations themselves undergo changeover, whereas automating equipment elsewhere on the line does not.

4.5. Sensitivity to the CONWIP Limit

The CONWIP limit was varied from 3 to 12 units in all three configurations, at an arrival rate of 13 jobs per hour and with ten replications at each limit.

Table 9
Effect of the CONWIP Limit on Throughput and Lead Time, Manual Configuration

Limit	Throughput (jobs/h)	Lead time (min)	Token pool utilisation
3	12.91 ± 0.15	16.31 ± 0.80	77.5 %
4	12.91 ± 0.20	17.31 ± 0.64	69.5 %
5	12.95 ± 0.25	18.22 ± 0.62	63.3 %
6	12.89 ± 0.15	18.00 ± 1.13	56.1 %
7	13.09 ± 0.17	19.67 ± 1.35	53.7 %
8	13.01 ± 0.17	19.13 ± 0.93	48.0 %
9	12.97 ± 0.13	19.27 ± 0.80	43.5 %
10	13.16 ± 0.21	19.84 ± 0.70	41.5 %
11	13.03 ± 0.20	19.91 ± 0.53	38.1 %
12	12.91 ± 0.14	20.04 ± 0.71	35.0 %

Table 10
Lead time separated into time inside the line and time waiting for release, by CONWIP limit

Configuration	Limit	Pool util. (%)	Jobs in line	L = λW (jobs)	Jobs at release point	Time inside the line (min)	Time waiting for release (min)
1 Manual	3	78	2.340	3.633	1.293	10.62	5.87
	5	61	3.050	3.551	0.501	14.12	2.32
	8	48	3.840	4.055	0.215	17.52	0.98
	10	39	3.900	4.162	0.262	18.11	1.22
	12	31	3.720	3.788	0.068	17.20	0.31
2 Semi-automated	3	57	1.710	1.978	0.268	8.08	1.27
	5	39	1.950	2.033	0.083	9.33	0.40
	8	27	2.160	2.231	0.071	9.85	0.32
	10	21	2.100	2.163	0.063	9.74	0.29

	12	18	2.160	2.163	0.003	10.02	0.01
3 Automation-assisted	3	51	1.530	1.689	0.159	6.85	0.71
	5	34	1.700	1.738	0.038	7.70	0.17
	8	21	1.680	1.745	0.065	7.74	0.30
	10	18	1.800	1.862	0.062	8.39	0.29
	12	14	1.680	1.745	0.065	7.75	0.30

The same redistribution occurs in the semi-automated and automation-assisted configurations, though on a scale close to the resolution of the recorded utilisation, since both run with far less congestion than the manual baseline. Table 10 shows that only the manual configuration produces a redistribution large enough to be clearly resolved, moving 5.56 minutes of waiting from inside the line to the release point as the limit tightens from 12 units to 3. The corresponding movement in the two automated configurations is under one minute in each case.

Figure 4

Effect of the workload control limit, manual configuration at an offered load of 13 jobs per hour

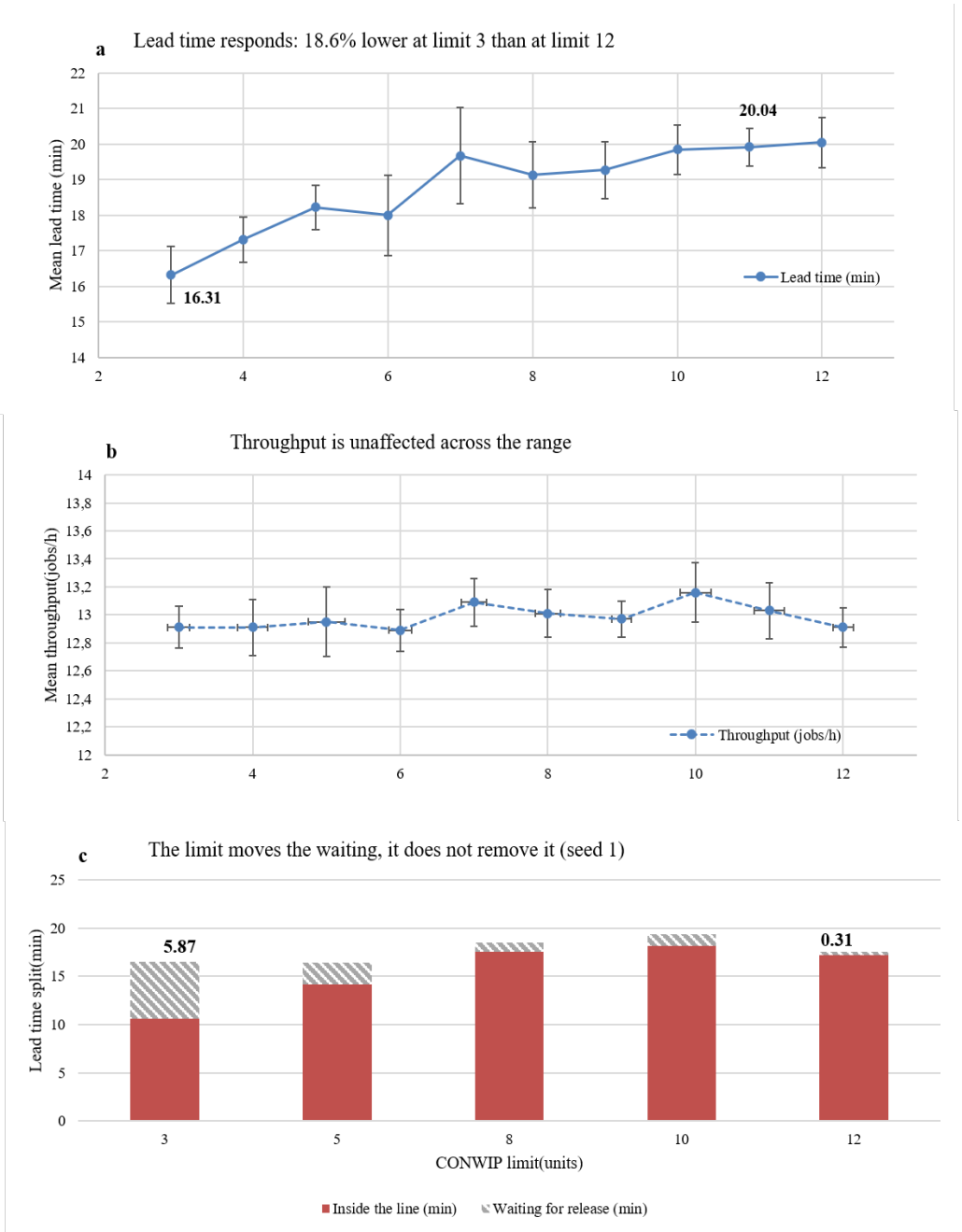


Table 9 reports throughput and lead time against the CONWIP limit for the manual configuration, and Figure 4 plots both responses. Throughput is unaffected by the CONWIP limit anywhere in the range tested. The ten means lie between 12.89 and 13.16 jobs per hour, a spread of 0.27 against replication intervals of approximately 0.17, so no effect on output is detectable at this level of precision.

Lead time is affected. Tightening the limit from 12 units to 3 reduces mean lead time from 20.04 ± 0.71 to 16.31 ± 0.80 minutes, a reduction of 18.6 per cent, and the confidence intervals at the two extremes do not overlap. The response is close to monotone across the ten values tested. This is the behaviour Little's Law implies, since with throughput held by demand a cap on work-in-process is also a cap on time

in the system. Khan and Standridge (2019) report a lead-time change of approximately 20 per cent when the CONWIP limit is varied in an industrial HMLV cell, which indicates an effect of comparable magnitude outside the present model. The direction differs, however, because their cell operates in a region where release still constrains throughput, whereas the configuration modelled here is demand-constrained across the range tested.

The limit of 10 units used in the baseline experiments is therefore conservative. A tighter limit would shorten lead time in every configuration without disturbing the comparison between them, since the same limit applies throughout.

4.6. Capacity Ceilings

The arrival rate was raised from 13 jobs per hour until each configuration saturated, with ten replications at each load except at loads 17, 18, 23, and 24, where five were used.

Table 11
Throughput and operator utilisation against offered load

Offered load	Manual throughput	Manual operator	Semi throughput	Semi operator	Auto throughput	Auto operator
16	15.96 ± 0.21	95.6 %				
17	16.47 ± 0.09	99.2 %				
18	16.65 ± 0.08	99.8 %				
19	16.62 ± 0.05	99.9 %				
22			21.54 ± 0.15	98.6 %		
23			21.96 ± 0.18	99.9 %		
24			22.00 ± 0.18	99.9 %		
25			21.94 ± 0.09	99.9 %	25.09 ± 0.16	60.2 %
28					28.00 ± 0.31	67.0 %
31					31.06 ± 0.20	74.5 %
34					34.18 ± 0.32	81.9 %
42					41.14 ± 0.11	98.6 %

44					41.21 ± 0.12	98.7 %
46					41.27 ± 0.12	98.8 %
48					41.29 ± 0.12	98.8 %

Note. Ten replications per point except at loads 17, 18, 23 and 24, which use five. Values are means with 95 per cent confidence intervals.

Table 12

Capacity ceilings by configuration

Configuration	Ceiling (jobs/h)	Increase over manual	Binding constraint at saturation	Replications
Manual	16.63 ± 0.04		Operator availability, 99.9 %	15
Semi-automated	21.96 ± 0.06	32 %	Operator availability, 99.9 %	20
Automation-assisted	41.23 ± 0.05	148 %	Operator availability, 98.7 %	40

Note. Each ceiling is the mean throughput across all replications at those offered loads where throughput no longer tracks demand.

Figure 5

Throughput achieved versus offered load. Where all offered demand is met, the dashed diagonal marks it, and each configuration departs from that line at saturation

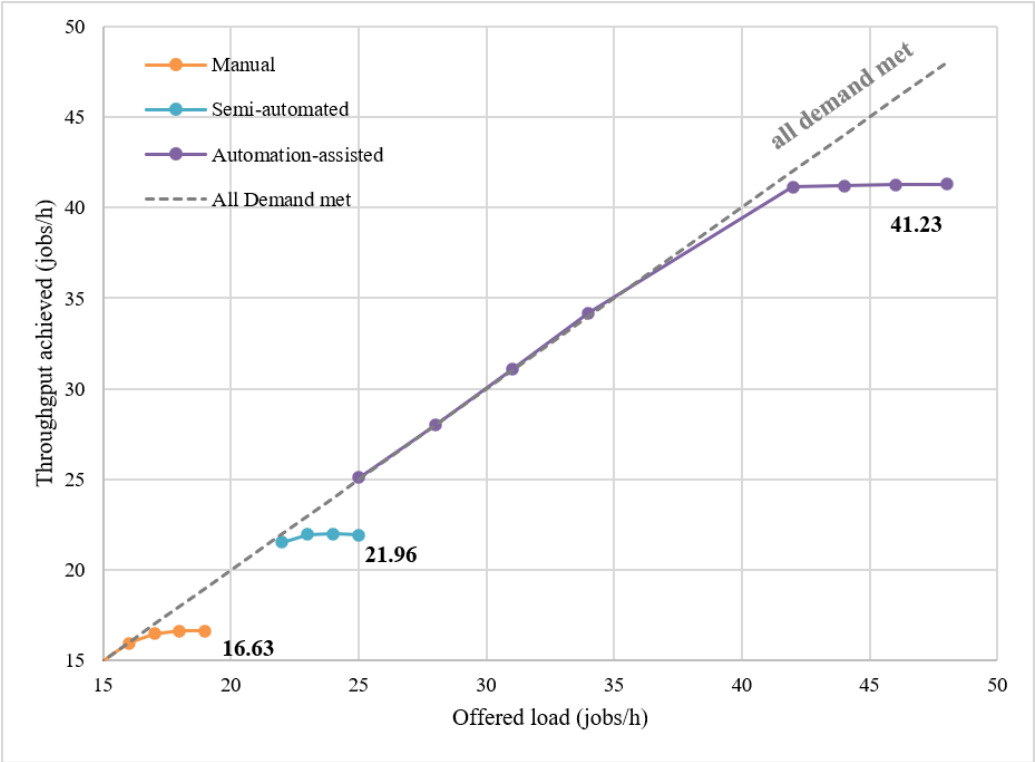


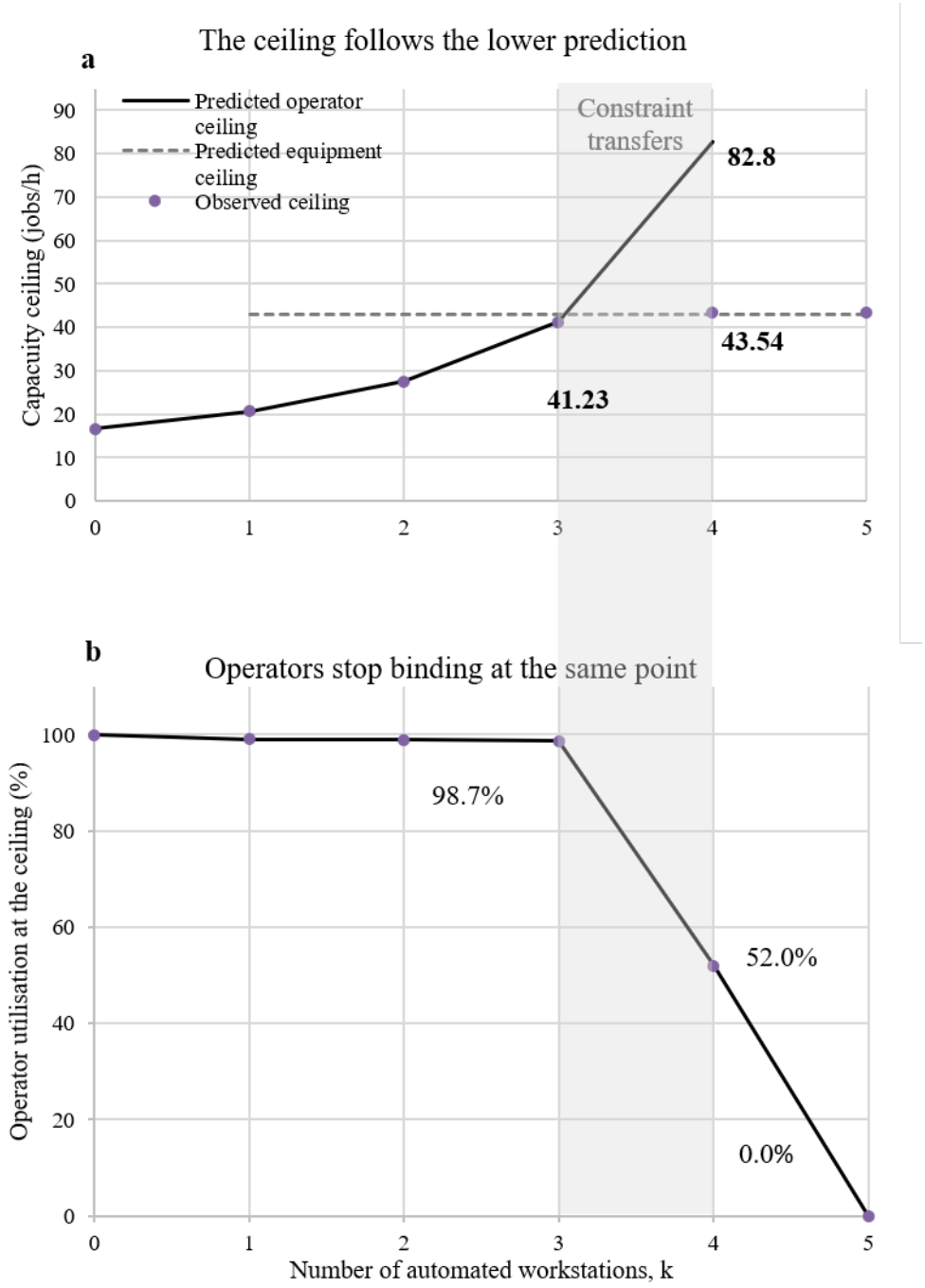
Table 11 reports throughput and operator utilisation across the load sweep, Table 12 summarises the resulting ceilings, and Figure 5 plots throughput achieved against offered load. Saturation arrives as the operator pool nears full utilisation: the manual configuration tops out at 16.63 ± 0.04 jobs per hour, with operator utilisation at 99.9 per cent; the semi-automated reaches 21.96 ± 0.06, also at 99.9 per cent; the automation-assisted, 41.23 ± 0.05 at 98.7 per cent. Under these loads throughput still follows the offered load closely; past them it plateaus, and lead time keeps growing without bound. At saturation, then, operator availability is what binds, in every configuration. Each ceiling works out to the operators' three units divided by the operator time a job consumes, and Appendix D sets out that calculation.

Station utilisation is not the same as pool utilisation and needs careful interpretation. As set out in Section 3.4, a workstation is represented as a work location rather than as a capacity-constrained machine, so the figure reported for a workstation reflects the load offered to it rather than the fraction of time it is occupied. Where more than one job is present at a station, that figure exceeds 100 per cent. In the automation-assisted configuration at saturation, the two operator-paced workstations report 148 and 147 per cent, while the three automated workstations report between 94 and 96 per cent; both sets of figures equal the product of throughput and the respective cycle times. The capacity ceilings reported here are accordingly those of the labour pool. A representation imposing finite station capacity would return lower values, most markedly in the automation-assisted configuration, where the operator pool is least loaded.

The nature of the constraint is the same in all three configurations; only its severity differs. In the manual and semi-automated configurations every workstation requires an operator, so each job consumes operator time at five stations. In the automation-assisted configuration only two workstations require an operator, so each job consumes

operator time at two stations, and the same pool of three operators supports proportionally more jobs. The relationship does not, however, continue indefinitely. Appendix C extends the comparison to one, two, four, and five automated workstations, with ten replications per load point. Ceilings track the operator prediction at zero, one, two, and three automated stations, where the operator pool is at or near full utilisation when throughput plateaus; beyond three, they do not. As Figure 6 shows, at four and five automated stations, the plateau settles at 43.54 ± 0.07 and 43.47 ± 0.08 jobs per hour, while operator utilisation falls to 52.0 and 0.0 per cent, respectively. The binding constraint therefore transfers from the operator pool to the automated equipment, whose ceiling is 3,600 seconds divided by the automated cycle of 83.6 seconds, that is 43.1 jobs per hour, and which does not depend on how many stations are automated. Scenario 3 is thus the deepest level of automation at which releasing operators still raises capacity.

Figure 6
Capacity ceiling versus number of automated workstations



Note. Panel a shows the two analytically predicted ceilings together with the measured plateau. The measured ceiling follows the lower of the two predictions, and the predictions cross between three and four automated stations. Panel b shows operator utilisation at each measured ceiling. Ten replications per load point for k = 1, 2, 4, and 5; the ceilings for k = 0 and k = 3 are those given in Table 12. Error bars are 95 per cent confidence intervals.

4.7. Reduction of Changeover

Scenario 3b halves changeover, from 60 seconds to 30 seconds, in the automation-assisted configuration.

Table 13

Effect of changeover reduction

Metric	Scenario 3	Scenario 3b	Change
Throughput (jobs/h)	13.13 ± 0.52	12.98 ± 0.41	-1.1 %
Lead time (min)	8.77 ± 0.13	6.85 ± 0.02	-21.9 %
WIP (units)	1.92 ± 0.10	1.48 ± 0.05	-22.8 %
Operator utilisation	30.8 %	25.6 %	-5.2 pp
Changeover component (min)	3.50	1.75	-50.0 %
Constraint station utilisation	46.2 %	38.4 %	-7.8 pp

Table 13 compares Scenario 3b with the automation-assisted configuration from which it is derived. Halving changeover produces a further 21.9 per cent reduction in lead time, which exceeds the 15.3 per cent gain obtained by moving from the semi-automated to the automation-assisted configuration. Once processing has been automated, reducing changeover therefore offers more benefit per intervention than automating processing further. It is also the only intervention tested that reaches the constraint stations, lowering their utilisation by 7.8 percentage points, whereas automating three other stations lowered it by 0.2.

4.8. Sensitivity to the Assumed Automated Processing Time

Table 14
Effect of the automated processing time on system performance

Automated Processing Time (S)	Throughput (jobs/h)	Lead time (min)	WIP (units)	Operator utilisation
30	12.95 ± 0.10	8.14 ± 0.05	1.756 ± 0.017	30.7 %
35	13.11 ± 0.15	8.50 ± 0.06	1.856 ± 0.028	31.6 %
40 (baseline)	13.04 ± 0.17	8.72 ± 0.03	1.894 ± 0.030	31.5 %
45	12.96 ± 0.15	9.02 ± 0.04	1.949 ± 0.025	31.2 %
50	13.12 ± 0.16	9.30 ± 0.04	2.033 ± 0.027	31.5 %

Note. Automation-assisted configuration: offered load 13 jobs per hour, CONWIP limit 10, and standard deviation held at 3 seconds. Ten replications per point, with independently drawn seeds. Values are means with 95 per cent confidence intervals.

Table 14 reports the effect of varying the automated processing time by ±25 per cent, from 30 to 50 seconds. The automated processing time of 40 seconds is the only parameter in the model set by the authors rather than taken from a published source, as noted in Section 3.4.4. It was varied by ±25 per cent, from 30 to 50 seconds, with all other parameters held constant.

Throughput is unaffected. The five means lie between 12.95 and 13.12 jobs per hour and all confidence intervals overlap, so no effect on output is observed anywhere in the range. Operator utilisation is likewise flat, holding between 30.7 and 31.6 per cent. None of this surprises. The parameter governs the three automated stations, and since jobs there consume no operator time, the quantity that determines the capacity ceiling is independent of it by construction. The measurement bears this out, confirming that the implementation behaves accordingly.

Lead time increases monotonically, rising from 8.14 to 9.30 minutes across the range, an increase of 14.3 per cent. That response is far smaller than the 66.7 per cent increase in the parameter itself, because the automated stations account for three of the five station cycles and for none of the changeover, and because queueing at the baseline load is already close to zero in this configuration.

The conclusions drawn in Sections 4.2, 4.4, and 4.6 therefore do not depend on the assumed value. Throughput below the ceiling is set by demand at every value tested, the constraint stations remain the two that were not automated, and the operator-limited capacity ceiling is unchanged throughout.

5. Analysis

This chapter examines the results reported in Chapter 4 against the three research questions. Each section corresponds to one research question and, within each, the specific outcomes obtained for the modelled system are generalised to the structural conditions under which they are expected to hold, which reflects the way the research questions were framed for this study.

5.1. Analysis of the Modelling Requirements

5.1.1. Mapping of Theoretical Requirements to Model Components

Table 15 links each requirement to the model component that implements it, to the result that demonstrates its effect, and to a verdict on its necessity. The structure of the table follows the overall logic of the study: a theoretical characteristic gives rise to a modelling requirement, the requirement is realised as a specific model component, and that component then produces an observable effect in the results.

Table 15
Theoretical requirements, model components and evidence of necessity

Requirement	Model component	Evidence in the results	Verdict
Variant-specific stochastic processing times	Normal distribution scaled by work-content factors 0.8 to 1.6	Station loading asymmetry throughout Chapter 4	Necessary
Sequence-dependent changeover	Station-local variable, 60 s incurred when the variant differs	3.50 min of lead time, 39.9 % of the total in Scenario 3	Necessary
Simultaneous multi-indicator evaluation	Four indicators recorded for every run	Throughput flat while lead time falls 56.3 %	Necessary
Automation reduces mean and dispersion	85.0/12.0, 55.0/6.0 and 40.0/3.0 s	Lead-time CV falls from 6.6 % to 0.2 %	Necessary
Contended shared resources	Operator pool of three serving all operator-paced stations	Capacity ceiling rises from 16.63 to 41.23 jobs/h	Necessary, and decisive
Partial automation as a distinct case	Scenario 3, automating S1, S2 and S4 only	Constraint loading unchanged at 46.2 % against 46.4 %	Necessary
Global release constraint	CONWIP seize and release pair spanning the line	Throughput invariant across limits 3 to 12	Necessary for experimental control

Credibility by multiple techniques	Trace, extreme condition, analytical and sensitivity checks	Section 5.2	Necessary

Variant-specific stochastic processing times: the unbalanced product mix and the work-content factors together create the station loading asymmetry observed throughout Chapter 4. Had processing times been deterministic and independent of variant, every station would have carried an identical load in every configuration, and the central finding of Section 4.4, that loading concentrates on the stations which were not automated, could not have emerged.

Sequence-dependent changeover: changeover accounts for 3.50 minutes of lead time in Scenarios 1 to 3 and for 39.9 per cent of the total in the automation-assisted configuration. Had it been modelled instead as a fixed overhead per job, it would have remained constant across configurations and Scenario 3b could not have been constructed. The conclusion that changeover becomes the residual constraint therefore depends entirely on this assumption.

Simultaneous multi-indicator evaluation: Section 4.2 shows throughput unchanged while lead time falls by 56.3 per cent and operator utilisation falls by 46.6 percentage points. A study looking only at throughput would have concluded that automation makes no difference; a study looking only at lead time would have concluded that it transforms the system. Neither conclusion would be correct. This is a concrete instance of what Gödri (2022) describes as the conflicting-indicator condition in high-mix, low-volume management.

Contended shared resources: this is the single most consequential assumption in the model. Had operators been modelled as an attribute of each station rather than as a shared pool, automating S1, S2, and S4 would have left S3 and S5 untouched, and the capacity increase reported in Section 4.6 would not have appeared. The difference between Scenarios 2 and 3, which carries the central comparison of the study, exists only because the resource is contended.

Global release constraint: CONWIP was included on the expectation that the limit would exert real control over output. Section 4.5 shows otherwise. Across limits from 3 to 12 units, and in all three configurations, throughput varies less than it does between replications of a single configuration, even at a limit of 3, where the token pool is occupied 78 per cent of the time and is plainly binding.

The limit governs where work waits, not the amount of waiting done. Lead time contains both pieces — the wait for release and the time spent inside the line — because the measurement window opens before a job seizes a token. Separate them, and they run in opposite directions: reducing the limit from 12 units to 3 shifted 5.56 minutes of waiting from inside the line to the release point. The mechanism therefore functions chiefly as an experimental control rather than as a determinant of the outcome: applying the same limit to every configuration ensures that the configurations are not separated by differences in congestion.

Precise calibration of the release parameter: Khan and Standridge (2019) devote a full study to setting the CONWIP limit for a high-mix, low-volume cell, reporting that a limit of 50 matched unconstrained throughput while raising it to 58 changed lead time by approximately 20 per cent. Their finding that the two indicators respond differently to the same parameter suggests that the release limit acts on congestion rather than on output, and the present results reproduce that structure, showing a range across which throughput remains stable. The lead-time response here is of comparable magnitude but differs in direction, which follows from the difference in operating regime: their cell is capacity-constrained, so a limit set too low starves it, whereas the system modelled here is demand-constrained, and no limit within the range tested causes starvation. Precise calibration of the limit is therefore less critical in a demand-constrained system than in a capacity-constrained one, and which case applies is determined by the relationship between demand and capacity rather than by the control mechanism itself.

5.1.2. Answer to the First Research Question

Five structural assumptions are essential. They are variant-specific stochastic processing times, sequence-dependent changeover, contended shared resources, the simultaneous assessment of multiple performance indicators, and uniform boundary conditions across configurations. Of these, contended shared resources is the most consequential, since it is the mechanism through which automation at one station affects performance at another, and its omission would distort the conclusions severely. It is also the assumption most often neglected in models built at station level. Global release control is required for experimental validity rather than for representational accuracy, and its parameterisation proved less critical than the literature on limit-setting would suggest.

5.2. Analysis of Model Credibility

5.2.1. Theoretical Grounding

Sargent (2013) describes seventeen techniques for model verification and validation, and organises the task into conceptual model validity, computerised model verification, operational validity, and data validity. Olsen and Raunak (2019) treat the absence of an external reference as an instance of the oracle problem, and suggest that validity can then be assessed against relations that must necessarily hold rather than against observed behaviour. Zeigler and Nutaro (2016) argue that relying on any one verification approach alone severely limits the range of conditions that can be tested. Together, these three sources support the position taken here: that credibility in the absence of empirical data must be assembled from several partial checks rather than from a single definitive one.

5.2.2. Selection of Techniques and its Justification

Four of Sargent's seventeen techniques require access to the operating system being modelled: historical data validation, event validity, predictive validation, and Turing tests. None of these could be applied in the present study. The remaining techniques were assessed for applicability, and four were selected against the criteria set out in Table 16.

Table 16
Verification and validation techniques applied, and the credibility each establishes

Technique	Applied as	What it establishes	What it does not establish
Traces	Individual jobs followed with deterministic inputs	The implementation matches the specification	That the specification matches any real system
Degenerate and extreme condition tests	Runs at near-zero and beyond-capacity load	Behaviour remains plausible outside the operating range	Validity within that range
Internal validity	Little's Law and the token-based work-in-process comparison	Mutual consistency of independently recorded quantities	Correspondence with an external referent
Parameter variability and sensitivity	CONWIP limit, arrival rate and automated processing times varied systematically	Conclusions are not artefacts of the three assumed parameters	That the assumed values are themselves correct
Comparison with an analytical model	Operator utilisation and capacity ceilings derived in closed form from the input parameters and compared with simulation output	That the model's resource accounting and capacity behaviour follow from its stated parameters	That the parameters themselves describe any real system

5.2.3. Evaluation of the Model Against the Identified Constraints

The internal validity check warrants particular attention, since it is the strongest substitute available for the external validation the study could not carry out. Because the measurement window opens before a job seizes a CONWIP token and closes once that token has been released, two quantities are recorded separately: the mean time in the system, taken from the time-measurement blocks, and the mean number of jobs holding a token, taken from the utilisation of the token pool. Combining the first with throughput via Little's Law gives the mean number of jobs within the measurement window, while the second gives the mean number within the line itself. Subtract one figure from the other and you have the jobs waiting for release — a quantity that must stay non-negative, and small under light load. It does, everywhere: 0.31 jobs in the manual configuration, under 0.06 jobs in the automated ones. Nor is the agreement merely arithmetical. Different mechanisms in the model record the two figures, so any discrepancy would mean jobs were being lost, or that the measurement window did not span the interval intended.

A second internal check compares the simulation with a closed-form calculation. Operator utilisation is throughput multiplied by the operator time per job and divided

by the size of the operator pool, while the capacity ceiling is the operator supply divided by that same quantity. Predicted operator utilisation agrees with the simulated value to within 1.3 percentage points in all four configurations, and the three capacity ceilings agree to within 0.8 per cent. Mielczarek (2026) applies a model-to-model comparison where the system itself remains unobservable, and this check is of that kind. It is independent of the consistency check described above, since it works from the input parameters rather than from the recorded outputs.

5.2.4. Capability of the Combination in Establishing Credibility

Taken together, the four techniques applied establish that the implemented model corresponds to its specification, that it behaves plausibly beyond the range in which it was operated, that its independently recorded quantities are mutually consistent, and that its conclusions do not depend on the three parameters that were assumed rather than derived. This is a defensible basis for reasoning about the mechanisms the model exhibits, and a stronger one than any of the four techniques would provide alone, as Zeigler and Nutaro (2016) argue.

5.2.5. Limitations of the Validation Approach

The combination does not establish that the model predicts the behaviour of any specific assembly system. Internal consistency, structural plausibility, and parameter robustness jointly establish one thing: the model coherently represents the system as defined. They do not establish that the specification corresponds to a particular factory. That gap matters for how the results should be used — they support reasoning about mechanisms, but not the prediction of outcomes for a specific installation. That narrower claim is nevertheless sufficient for the present study, whose conclusions concern structural relationships and are established through mechanisms rather than through numerical correspondence.

The distinction between verification and validation determines how the results may be used. Verification supports reasoning about the mechanisms the model exhibits, because those mechanisms follow from the specification and the specification has been shown to be faithfully implemented. Validation would be needed to support the prediction of outcomes for a particular installation, and the available evidence does not extend that far. The conclusions of this study are accordingly framed as structural relationships within the analysed system rather than as quantitative predictions, which is what Säfsten and Gustavsson (2024) describe as analytical rather than statistical generalisation. The claim is narrower than a validated model would permit; it is adequate for the purpose, but the boundary should not be left implicit.

5.2.6. Answer to the Second Research Question

When empirical system data are unavailable, credibility can be established through verification by trace and constraint monitoring, through extreme-condition and degenerate testing, through internal validity assessed against analytically derived relations, and through systematic sensitivity analysis of the parameters not established by the literature. Comparison with published performance ranges adds a further check, though a weaker one: what it establishes is plausibility rather than direct correspondence. Thiscombination supports conclusions about structural mechanisms,

but does not justify quantitative predictions for a specific installation. A further element of the answer is that these techniques are not equivalent in what they establish: most of them verify the implementation, and stating which of them bear on validation, together with how little that leaves, is itself part of what credibility under these conditions requires.

5.3. Analysis of Configuration Performance

5.3.1. Throughput is Set by Demand Below the Capacity Ceiling

At an offered load of 13 jobs per hour, all four configurations produce between 12.91 and 13.13 jobs per hour. The point is not that automation cannot raise throughput, but that below the capacity ceiling throughput is set by the arrival rate; Section 4.6 shows the ceiling itself rising from 16.63 to 41.23 jobs per hour.

Ames Zegarra and Indukaladharan (2021), simulating a comparable assembly cell, report a throughput increase when automation is introduced. That a difference appears in their case and not here points to the condition under which it becomes visible: the less automated configuration must already be operating at or near its capacity ceiling. The present results identify that condition explicitly. Below the ceiling the additional capacity exists but stays latent, and the practical consequence is direct: a firm that assesses automation by measuring throughput on a line which is not capacity-constrained will see no benefit at all, and may conclude wrongly that the investment is unjustified.

5.3.2. Lead Time Falls Because Work Ceases to Wait

Section 4.3 decomposes that reduction. Queueing falls by 97.4 per cent and processing by 32 per cent, which places between 73 and 79 per cent of the improvement in queueing rather than in processing.

The mechanism is congestion relief. In the manual configuration, three operators serve five stations, so a job frequently waits for an operator rather than for a workstation. Automating three stations removes those stations from the operators' workload, leaving two operator-paced stations served by the same three operators, and waiting for an operator becomes infrequent as a result. This is the relationship Kingman (1961) describes, in which queueing grows non-linearly with resource utilisation, so that lowering operator utilisation from 77.4 to 30.8 per cent produces a far larger reduction in queueing than a proportional change would.

Frohm et al. (2008) distinguish the mechanical dimension of automation from the cognitive one. For flow performance, the present results point to a narrower decisive distinction: whether the contended resource stays inside the processing loop. Scenario 2 keeps the operator engaged throughout and cuts processing time at every station; lead time falls by 48 per cent. Scenario 3 reduces processing time at only three stations but takes the operator out of them, and lead time falls by 56 per cent. The configuration with the smaller reduction in processing time therefore performs better, because it changes what the work contends for.

5.3.3. Partial Automation Relieves the Line Without Relieving the Constraint

Section 4.4 reports the main finding of the study. Automating S1, S2, and S4 left S3 and S5 at 46.2 per cent utilisation, against 46.4 per cent in the fully manual configuration, a difference of 0.2 percentage points. The system as a whole improved substantially while the stations that set its capacity barely changed at all. Capacity rose because operators were freed from three stations, not because the constraint stations became faster: those stations still run at 85 seconds in both configurations.

This is the mechanism that (Miqueo et al., 2023, p. 10) identify when they exclude automation layout from their own experimental design, noting that "none of the automated stations are the AL bottleneck. Therefore, in this particular case, the number of automated stations would not significantly impact the AL operational KPIs." Their decision to set the condition aside as unsuitable for study indicates that the effect has been recognised but not quantified, and the present results supply that quantification.

Yang et al. (2025), analysing bottleneck shifts under different configurations in aerospace assembly, show that configuration changes relocate constraints. The present results add a boundary condition to that finding: constraints relocate only where the configuration change touches the constraint itself.

5.3.4. Capacity Ceilings Track Task Allocation More Closely Than Processing Speed Does

The three configurations return capacity ceilings of 16.63, 21.96, and 41.23 jobs per hour. Scenario 2 reduces processing time at all five stations and gains 32 per cent, whereas Scenario 3 reduces processing time at three stations and gains 148 per cent. Scenario 3 is therefore slower in aggregate, since two of its five stations still run at manual speed, and yet it has substantially more capacity.

Capacity depends on how much of the contended resource each job uses. In Scenario 2, every job still needs an operator at all five stations, but the operator is occupied for a shorter time, so capacity rises proportionally. In Scenario 3, each job requires an operator at only two stations, which reduces the operator's total engagement per job considerably and increases capacity accordingly. Figure 6 shows this relationship across the range of automation depths tested. The measured ceilings follow the operator time each job consumes rather than the processing speed of the individual stations, but only for as long as the operator pool remains the scarcest resource. Beyond three automated workstations, the operator prediction and the measured ceiling diverge: at four automated stations the predicted operator ceiling rises to 82.8 jobs per hour, whereas the measured plateau remains at 43.54 ± 0.07, with operators only 52.0 per cent loaded. The mechanism therefore has a boundary. Releasing the contended resource raises capacity until that resource is no longer the constraint, at which point the ceiling is set by whatever has replaced it, here the automated equipment, at 43.1 jobs per hour.

5.3.5. Changeover as the Residual Component of Lead Time

Changeover holds at 3.50 minutes in Scenarios 1 to 3, since it depends on the product sequence rather than on who or what performs the operation. As the other components

shrink, its share of total lead time climbs — 17.5 per cent in the manual configuration, 39.9 per cent in the automation-assisted one. Scenario 3b halves it, buying a further 21.9 per cent reduction in lead time; the move from Scenario 2 to Scenario 3 gained 15.3 per cent. In an automated high-mix, low-volume cell, setup reduction is thus the most effective next step, ahead of further automation of processing.

5.3.6. Answer to the Third Research Question

The automation configuration affects the four indicators, but not in the same structural way. Below the capacity ceiling throughput does not move; at the ceiling itself it rises substantially. Lead time and work-in-process drop sharply, and the drop comes chiefly from reduced queueing; shorter processing times contribute less. Resource utilisation falls in proportion to the work removed from the contended resource. What the capacity ceiling tracks within the analysed system is task allocation — how much of the contended resource each job consumes — more closely than processing speed.

6. Discussion

6.1. Results in Relation to Existing Literature

Automation is generally treated in the literature as something that raises throughput, and individual studies back that view: Ames Zegarra and Indukaladharan (2021), introducing automation into an assembly cell, report higher throughput alongside lower work-in-process. The present results qualify that view more than they contradict it. Only above the capacity ceiling of the less automated configuration do differences in throughput become visible; below it, the arrival rate sets throughput, and automation yields latent capacity rather than additional output. The operating pointrelative to the capacity ceiling is therefore a variable worth reporting whenever the throughput effects of automation are compared across studies. Establishing that it accounts for the divergence in reported results would require examining the operating points of those studies directly, which lay outside the scope of this work.

The finding that automating non-constraint stations leaves constraint loading unchanged occupies an unusual position in the literature: it has been observed and then set aside rather than studied. (Miqueo et al., 2023, p. 10) state that "none of the automated stations are the AL bottleneck. Therefore, in this particular case, the number of automated stations would not significantly impact the AL operational KPIs," and give this as their reason for excluding automation layout from their experimental design. The condition that makes their case unsuitable for studying the effect is precisely the condition in which the effect matters most in practice, since it is the situation most firms meet when they automate incrementally.

The present study runs the experiment they set aside, and quantifies the outcome: with three of five stations automated, constraint loading changes by 0.2 percentage points while system lead time more than halves. No unknown mechanism is discovered here; the contribution lies in measuring an acknowledged one, and doing so in the configuration where it carries practical consequences.

Frohm et al. (2008), in the Levels of Automation (LoA) framework, distinguish mechanical from cognitive automation. The present findings suggest a further distinction that matters for flow performance: whether automation removes the contended resource from the processing loop. Scenario 2 raises mechanical LoA at every station and yields a 32 per cent increase in capacity; Scenario 3 raises it at three stations and yields a 148 per cent increase. The ordinal levels of the framework do not separate these two cases, since both are increases in mechanical LoA. For capacity, what matters is not the level itself but whether the contended resource remains inside the loop.

The CONWIP limit does not affect throughput, which is consistent with the mechanism described by Frazee and Standridge (2016): CONWIP caps inventory, and in a demand-constrained system inventory does not cap output. This does not contradict Molenda et al. (2020), who find that capacity-slack-based backlog sequencing improves release performance, or Prabhu et al. (2025), who attribute 80.2 per cent of throughput performance to the order release method, since both studies address capacity-constrained settings in which release does limit output. The present finding marks the

boundary: release control determines where work waits, and affects how much work is completed only where the system is capacity-constrained.

Goyal et al. (2026) present scenario-based simulation as a way of assessing disruptions and configurations before implementation, which helps to justify investment under uncertainty. The present study shows the specific benefit of that approach: the finding that automating most stations would not shift the constraint takes hours to obtain from a simulation, but can be obtained from a physical trial only once the investment has been made. Alkanafani and Arabaci (2025) reach a conclusion of similar shape for an automated machine cell, finding that changes of layout are more effective than increases in speed.

Taken together, these comparisons place the study as one that confirms the direction of previously reported automation effects, qualifies the conditions under which those effects appear, and measures an effect the literature has recognised but set aside.

6.2. Practical Implications

Location of the constraint as a criterion for selecting stations: automating stations away from the constraint improves lead time and work-in-process substantially, but does not raise capacity in proportion to the investment. In this study three of five stations were automated while the constraint stations went on operating as before. A firm seeking additional capacity should automate the constraint; a firm seeking responsiveness and lower inventory may reasonably automate elsewhere. These are different objectives, and they justify different investments.

Do not assess automation by measuring throughput alone on an unsaturated line. Where demand sits below capacity, throughput does not move, and the investment can look ineffective. The benefits sit elsewhere: added capacity headroom, reduced lead time, lower work-in-process. Firms that evaluate on output alone will tend toundervalue automation in exactly the environments where its flow advantages are greatest.

Expect the constraint to move towards changeover. As processing is automated, changeover becomes the largest remaining element of lead time, accounting for 40 per cent of it in the automation-assisted configuration here. Halving it yields a greater improvement than the whole transition from semi-automated to automation-assisted. Reducing setup times is usually cheaper than upgrading equipment, so once initial automation is in place it is likely to be the more cost-effective investment.

Consider whether automation frees labour or merely speeds it up. The two automated configurations differ in kind rather than in degree. Collaborative equipment that accelerates an operator keeps that operator inside the loop, whereas equipment that runs unsupervised takes the operator out of it. The second raises system capacity far more than the first, even though it reaches fewer stations.

6.3. Method Discussion

The study supports analytical rather than statistical generalisation (Säfsten & Gustavsson, 2024). The findings apply to HMLV assembly systems that share the structural features modelled here, and without further testing they do not extend to

systems with variant-specific routing, parallel stations, or labour dedicated to individual stations. The mechanisms identified, such as queueing dominating the reduction in lead time and capacity following contended-resource consumption, are structural, and the results suggest that they are more likely to transfer than the specific magnitudes are, although this has not been tested outside the modelled configuration.

The predictive value of the model depends on the quality of its inputs (Banks et al., 2004). Two parameters are not drawn from the literature: the automated processing time of 40 seconds, which was scaled to the modelled cell rather than sourced, and the CONWIP limit of 10, which was calibrated against the behaviour of the model. Both are identified as assumptions in Section 3.4.4 and were subjected to sensitivity analysis. A further parameter, the HRC processing time of 55 seconds, is drawn from the literature, but from a source that reports two conflicting values; the more conservative of the two was used. These two assumptions form a narrower set than the three parameters subjected to sensitivity analysis in Section 3.5.3. The arrival rate belongs to that set but not to this one, because it was set against published throughput figures for industrial HMLV lines rather than against the model's own behaviour, and is therefore externally anchored rather than assumed.

On the choice of DES: discrete-event simulation was appropriate because the quantity of interest arises at the level of individual resource seizures. Its advantage over analytical modelling is specific rather than general: Biazen and Gebeyehu (2019) show close agreement between the two methods on descriptive measures, with differences appearing only where resources are reallocated between stations, which is precisely the intervention examined here.

6.4. Limitations

Simplification of the system: the model is a serial line with common routing. Real HMLV systems often use variant-specific routing, which creates loading differences that this model does not reproduce. The simplification was necessary in order to isolate the automation effect, but it reduces the applicability of the findings to job-shop settings.

Equipment reliability is modelled as a sensitivity case rather than in the baseline configurations. A mean time between failures of 200 minutes is taken from Perwitz et al. (2022), and a mean repair time of 6.2 minutes is derived from the 3 per cent unavailability reported in the same source, since no repair duration is stated there. Across five replications at the baseline load, no measure differs from the perfect-availability case by more than replication noise: throughput 12.81 ± 0.42 against 13.13 ± 0.52 jobs per hour, lead time 8.74 ± 0.12 against 8.77 ± 0.13 minutes, and constraint-station utilisation unchanged at 46.2 per cent. Within the analysed system, the automated workstations retain sufficient capacity slack that failures at the rate reported do not propagate to system performance. Two boundaries apply. Failures are modelled only at the automated workstations, so the finding does not extend to failures at the stations that set the ceiling, where no comparable slack exists. The repair duration is a derived rather than a reported value, and five replications limit how small a difference the comparison can detect.

Human factors are omitted: fatigue, learning curves, and ergonomic constraints go unrepresented, and operators work at stationary stochastic rates indefinitely. In some respects this favours the manual configuration; in others it penalises it — the net effect is uncertain. Economic evaluation is likewise absent. The study reports differences in performance, not what they cost. A configuration that improves lead time by 56 per cent may or may not justify the capital it requires, and that question is not addressed here.

The sensitivity experiments differ in the statistical support behind them, and the distinction matters for how each result should be read. The load sweep and the CONWIP control sweep use ten replications per point with independently drawn seeds, while the fine load steps used to locate each capacity ceiling use five. Results from these experiments are reported with 95 per cent confidence intervals, and no difference smaller than the interval at a given point is treated as resolved. On that basis, the effect of the release limit on lead time is resolved at 18.6 per cent across the range tested, whereas its effect on throughput is not.

One experiment offers weaker support. The reliability sensitivity uses five replications rather than ten, which limits what it can detect, and no conclusion in Chapter 7 rests on it alone. The automation-depth sweep reported in Appendix C uses ten replications per load point, with independently drawn seeds, for configurations with one, two, four, and five automated workstations. Its ceilings are reported with 95 per cent confidence intervals on the same basis as the load sweep.

One further consideration applies to the CONWIP control sweep. Seeds were drawn independently at each limit rather than from a common set applied across limits. Matched seeds would have reduced the variance of the comparison between limits, so the bound reported on the effect of the limit is conservative rather than optimistic.

Parameter provenance: as set out in Section 6.3, two of the parameters are the authors' own assumptions. Their influence was tested, but they remain assumptions.

Uneven screening procedure: the three literature searches were not screened by the same method. Searches 2 and 3 were screened independently by both authors in Rayyan, with conflict resolution recorded; Search 1, carried out earlier and with narrower eligibility criteria, was screened manually. The parameterisation search is therefore the least protected against single-reviewer selection bias, which matters because it supplied the input values of the model. The risk is limited, since its eligibility criterion was the presence of an explicitly stated stochastic parameter rather than a judgement of relevance, but it is not eliminated.

Workstations are not capacity-constrained, as set out in Section 3.4. A physical workstation processes one job at a time, whereas the representation used here constrains concurrent processing only by resource availability. The capacity ceilings reported in Chapter 4 therefore describe the labour pool rather than the assembly line as a physical installation, and should be read as upper bounds on what the modelled task allocation permits.

7. Conclusions

7.1. Answers to the Research Questions

RQ1: five structural assumptions are necessary if a DES model is to represent an HMLV assembly system adequately for the purpose of comparing automation configurations: variant-specific stochastic processing times; sequence-dependent changeover; contended shared resources; simultaneous evaluation of multiple performance indicators; and consistent boundary conditions across configurations. Omitting contended shared resources would distort the conclusions most severely, since it is the mechanism through which automation at one station affects performance at another. Global release control is required for experimental validity rather than for representational accuracy, and its parameterisation proved less critical for a demand-constrained system than the CONWIP literature would suggest.

RQ2: where empirical system data are unavailable, model credibility can be established through verification by trace and constraint monitoring; testing under extreme and degenerate conditions; internal validity checks against analytically derived relations, of which Little's Law is the most robust for flow systems; and systematic sensitivity analysis over the parameters not taken from the literature. Comparison against published performance ranges establishes plausibility but not exact correspondence. This approach supports conclusions about structural mechanisms but does not permit quantitative predictions for a specific installation, and that distinction must be stated explicitly rather than left implicit.

RQ3: the automation configuration affects the four indicators in structurally different ways. Below the capacity ceiling, throughput is limited by the arrival rate and is unaffected by configuration; at the ceiling, capacity rises from 16.63 to 41.23 jobs per hour across the three principal configurations, and to an equipment-limited 43.5 where four or more workstations are automated. Lead time falls by up to 65.9 per cent and work-in-process by up to 65.8 per cent, chiefly through reduced queueing rather than reduced processing. Utilisation falls in proportion to the work removed from the contended resource. The capacity ceiling follows task allocation rather than processing speed, so that in the configuration studied, automating three of five workstations improved flow substantially while leaving constraint-station loading unchanged to within 0.2 percentage points.

7.2. Contributions

Three findings of this study are not reported in the existing literature. First, the effect of automating workstations that are not the system constraint has been recognised but not measured. (Miqueo et al., 2023, p. 10) state that "none of the automated stations are the AL bottleneck. Therefore, in this particular case, the number of automated stations would not significantly impact the AL operational KPIs," and give this as their reason for setting the automation layout aside in their experimental design. The present study runs that experiment and quantifies the outcome: automating three of five workstations leaves constraint-station loading unchanged to within 0.2 percentage points while reducing lead time and work-in-process by more than half. Second, the resulting separation between improvement in flow and improvement in capacity has not

previously been quantified. Within the analysed system, a configuration that reduces processing time at fewer workstations yields a capacity gain more than four times that of one which reduces it at all five, because it removes the operator rather than merely accelerating them. Third, the study reports what verification and validation can and cannot establish for a simulation model built without an industrial referent, and states that most of the evidence available under that constraint is verification rather than validation. Accounts of this kind are uncommon, since studies conducted without a company partner more often report the techniques applied than the limits of what those techniques support.

Theoretical contribution: The results refine the Levels of Automation framework (Frohm et al., 2008) for flow-performance purposes by identifying a distinction that its ordinal scales do not capture, namely whether automation removes the contended resource from the processing loop. The semi-automated configuration reduces processing time at all five workstations while keeping an operator engaged at each, and raises the capacity ceiling by 32 per cent. The automation-assisted configuration reduces processing time at three workstations, removes the operator from those three, and raises the ceiling by 148 per cent. Both configurations represent increases in mechanical automation, yet the framework does not distinguish them, and within the analysed system their capacity gains differ by more than a factor of four. The distinction holds only within the boundary identified by the same experiment: capacity rises with the release of the operator for as long as operators remain the scarcest resource. Automating a fourth and a fifth workstation pushes the predicted operator ceiling to 82.8 jobs per hour and beyond. The measured ceiling, though, stops at approximately 43.5 in both cases — operators loaded at 52 and 0 per cent, respectively — because the automated equipment has become the constraint. The refinement offered here is therefore not that automation should remove the operator, but that capacity follows whichever resource is scarcest, and that the gain from removing one constraint is bounded by the next.

Methodological contribution: The study also sets out a verification and validation strategy for a simulation model built without access to an industrial system, and states explicitly what that strategy establishes and what it does not. Four of the seventeen techniques catalogued by Sargent (2013) are unavailable by construction, namely historical data validation, event validity, predictive validation, and Turing tests. The combination applied in their place establishes that the implementation matches its specification, that behaviour remains plausible outside the operating range, that independently recorded quantities are mutually consistent, and that model outputs follow analytically from model inputs. It further establishes that the conclusions do not depend on the parameters that were assumed rather than derived. What it does not establish is correspondence with any particular installation, and Section 3.5.4 states that most of the evidence gathered is verification rather than validation. Because a substantial proportion of student and exploratory simulation studies are conducted without a company partner, an explicit account of what can and cannot be claimed under that constraint has value beyond the present case.

Practical contribution: the study offers a procedure for comparing automation configurations before investment, together with a specific warning. Within the analysed system, automating workstations away from the constraint reduced lead time and work-in-process by more than half while leaving constraint-station loading unchanged to

within 0.2 percentage points. Evaluating the same configuration by throughput alone, on a line that was not saturated, showed no benefit whatsoever, even though substantial capacity headroom had been created. Goyal et al. (2026) frame scenario-based simulation as a means of assessing configurations before implementation in order to support investment justification, and the present results give a worked example of what such an assessment can reveal in advance, where a physical trial would reveal it only after the investment had been committed.

7.3. Future Research

Extend equipment reliability beyond the automated stations: the sensitivity case reported in Appendix C. Models failures at the automated workstations only and finds no effect at the baseline load. The stations that determine the capacity ceiling carry no comparable slack, so failures occurring there would not be absorbed in the same way. Testing this is the natural extension.

Automate the constraint: the obvious alternative experiment is to automate S3 and S5 rather than S1, S2, and S4, and then to compare the capacity gain per station automated. The present results suggest a considerably larger gain from the smaller number of stations.

Introduce variant-specific routing: extending the model to accommodate routings that differ by variant would test whether the constraint-migration findings still hold where the location of the constraint itself varies from one product to another.

Add economic evaluation: linking the performance model to capital and operating costs would convert the comparison of configurations into a comparison of investments.

Validate against an industrial case: access to an operating HMLV line would allow the use of historical data validation, event validity, and predictive validation, the three techniques that are unavailable here, and would establish whether the structural findings are quantitatively accurate as well as mechanistically valid.

References

Alkanafani, A., & Arabaci, Ç. (2025). *Scenario-based simulation and optimization of an automated machine cell* [Master's thesis, Jönköping University]. DiVA. https://urn.kb.se/resolve?urn=urn:nbn:se:hj:diva-68417

Ames Zegarra, C., & Indukaladharan, A. (2021). *Simulation of assembly cell: Simulation based evaluation of automation solutions in an assembly cell* [Master's thesis, Jönköping University]. DiVA. https://urn.kb.se/resolve?urn=urn:nbn:se:hj:diva-53862

Bairagi, V., & Munot, M. V. (2019). *Research methodology: A practical and scientific approach*. CRC Press. https://doi.org/10.1201/9781351013277

Banks, J., Carson, J. S., Nelson, B. L., & Nicol, D. M. (2004). *Discrete-event system simulation* (4th ed.). Prentice Hall.

Bellgran, M., & Säfsten, K. (2010). *Production development: Design and operation of production systems*. Springer.

Biazen, M. A., & Gebeyehu, S. G. (2019). Comparative analysis of analytical and discrete-event simulation models of assembly line systems. *Journal of Engineering, Project, and Production Management, 9*(2), 132–141. https://doi.org/10.2478/jeppm-2019-0015

Frazee, T., & Standridge, C. (2016). CONWIP versus POLCA: A comparative analysis in a high-mix, low-volume (HMLV) manufacturing environment with batch processing. *Journal of Industrial Engineering and Management, 9*(2), 432–449. https://doi.org/10.3926/jiem.1248

Frohm, J., Lindström, V., Winroth, M., & Stahre, J. (2008). Levels of automation in manufacturing. *Ergonomia – An International Journal of Ergonomics and Human Factors, 30*(3). https://urn.kb.se/resolve?urn=urn:nbn:se:hj:diva-7253

Gan, Z. L., Musa, S. N., & Yap, H. J. (2023). A review of the high-mix, low-volume manufacturing industry. *Applied Sciences, 13*(3), 1687. https://doi.org/10.3390/app13031687

Gödri, I. (2022). Improving delivery performance in high-mix low-volume manufacturing by model-based and data-driven methods. *Applied Sciences, 12*(11), 5618. https://doi.org/10.3390/app12115618

Gosavi, A., & Gosavi, A. A. (2024). CONWIP control in the digitized world: The case of the cyber-physical jobshop. *International Journal of Production Economics, 270*, 109169. https://doi.org/10.1016/j.ijpe.2024.109169

Goyal, A., Acar, Ö. F., Yamamoto, Y., & Aslanidou, I. (2026). Scenario-based simulation for developing new production systems: A methodological framework. *IFIP Advances in Information and Communication Technology, 769*, 526–540. https://doi.org/10.1007/978-3-032-03550-9_35

Harrod, S., & Kanet, J. J. (2013). Applying work flow control in make-to-order job shops. *International Journal of Production Economics, 143*(2), 620–626. https://doi.org/10.1016/j.ijpe.2012.02.017

Herps, K., Dang, Q. V., Martagan, T., & Adan, I. (2022). A simulation-based approach to design an automated high-mix low-volume manufacturing system. *Journal of Manufacturing Systems, 64*, 1–18. https://doi.org/10.1016/j.jmsy.2022.05.013

Johansen, K., Rao, S., & Ashourpour, M. (2021). The role of automation in complexities of high-mix in low-volume production – A literature review. *Procedia CIRP, 104*, 1452–1457. https://doi.org/10.1016/j.procir.2021.11.245

Kang, N., Zhao, C., Li, J., & Horst, J. A. (2016). A hierarchical structure of key performance indicators for operation management and continuous improvement in production systems. *International Journal of Production Research, 54*(21), 6333–6350. https://doi.org/10.1080/00207543.2015.1136082

Kasper, T. A. A., Land, M. J., & Teunter, R. H. (2023). Non-hierarchical work-in-progress control in manufacturing. *International Journal of Production Economics, 257*, 108768. https://doi.org/10.1016/j.ijpe.2022.108768

Khan, S., & Standridge, C. (2019). Aggregate simulation modeling with application to setting the CONWIP limit in a HMLV manufacturing cell. *International Journal of Industrial Engineering Computations, 10*(2), 149–160. https://doi.org/10.5267/j.ijiec.2018.10.002

Kingman, J. F. C. (1961). The single server queue in heavy traffic. *Mathematical Proceedings of the Cambridge Philosophical Society, 57*(4), 902–904. https://doi.org/10.1017/S0305004100036094.

Leal, F., Costa, R. F. da S., Montevechi, J. A. B., Almeida, D. A. de, & Marins, F. A. S. (2011). A practical guide for operational validation of discrete simulation models. *Pesquisa Operacional*, *31*(1), 57–77.https://doi.org/10.1590/s0101-74382011000100005.

Little, J. D. C. (1961). A proof for the queuing formula: L = λW. *Operations Research, 9*(3), 383–387. https://doi.org/10.1287/opre.9.3.383

Mahmoodi, E., Fathi, M., Tavana, M., Ghobakhloo, M., & Ng, A. H. C. (2024). Data-driven simulation-based decision support system for resource allocation in industry 4.0 and smart manufacturing. *Journal of Manufacturing Systems, 72*, 287–307. https://doi.org/10.1016/j.jmsy.2023.11.019

Miqueo, A., Torralba, M., & Yagüe-Fabra, J. A. (2022). Models to evaluate the performance of high-mix, low-volume manual or semi-automatic assembly lines. *Procedia CIRP, 107*, 1461–1466. https://doi.org/10.1016/j.procir.2022.05.175

Miqueo, A., Yagüe-Fabra, J. A., Torralba, M., Oliveros, M.-J., & Tosello, G. (2023). Parallel walking-worker flexible assembly lines for high-mix low-volume demand. *Processes, 11*(1), 172. https://doi.org/10.3390/pr11010172

Mielczarek, B. (2026). Operational validation of social simulation models: Model-to-model-comparison approach to validate pension models. *Journal of Simulation*. Advance online publication. https://doi.org/10.1080/17477778.2026.2653194.

Molenda, P., Mezger, T., Oechsle, O., Koller, J., & Döpper, F. (2020). Backlog-sequencing: A comparison between workload control and CONWIP using a simulation approach. *Procedia CIRP, 93*, 664–669. https://doi.org/10.1016/j.procir.2020.04.097

Monazzam, N., Alinezhad, A., & Adibi, M. A. (2022). Simulation-based optimization using DEA and DOE in production systems. *Scientia Iranica, 29*(6E), 3470–3488. https://doi.org/10.24200/sci.2021.55499.4253

Mueller, D., & Schmitt, T. V. (2020). Production planning in autonomous and matrix-structured assembly systems: Effects of similarity of precedence graphs on order release sequencing. *Procedia CIRP, 93*, 1358–1363. https://doi.org/10.1016/j.procir.2020.06.002

Negahban, A., & Smith, J. S. (2014). Simulation for manufacturing system design and operation: Literature review and analysis. *Journal of Manufacturing Systems, 33*(2), 241–261. https://doi.org/10.1016/j.jmsy.2013.12.007

Normand, A., & Bradley, T. H. (2024). An experimental investigation of Lean Six Sigma philosophies in a high-mix low-volume manufacturing environment. *PLOS ONE, 19*(5), e0299498. https://doi.org/10.1371/journal.pone.0299498

Ojstersek, R., Javernik, A., & Buchmeister, B. (2024). Integrating simulation modelling for sustainable, human-centred Industry 5.0: ESG-based evaluation in collaborative workplaces. *Advances in Production Engineering & Management, 19*(4), 527–538. https://doi.org/10.14743/apem2024.4.522

Olsen, M., & Raunak, M. (2019). Increasing validity of simulation models through metamorphic testing. *IEEE Transactions on Reliability, 68*(1), 91–108. https://doi.org/10.1109/tr.2018.2850315

Ouzzani, M., Hammady, H., Fedorowicz, Z., & Elmagarmid, A. (2016). Rayyan—a web and mobile app for systematic reviews. *Systematic Reviews, 5*, 210. https://doi.org/10.1186/s13643-016-0384-4

Perwitz, J., Sobottka, T., Beicher, J. N., & Gaal, A. (2022). Simulation-based evaluation of performance benefits from flexibility in assembly systems and matrix production. *Procedia CIRP, 107*, 693–698. https://doi.org/10.1016/j.procir.2022.05.047

Prabhu, A., Raghunandana, K., Sadanand, R. V., & Barboza, A. B. V. (2025). Workload control in a bottleneck-oriented flow shop with capacity adjustments. *Production Engineering Archives, 31*(4), 455–465. https://doi.org/10.30657/pea.2025.31.42

Prajapat, N., & Tiwari, A. (2017). A review of assembly optimisation applications using discrete event simulation. *International Journal of Computer Integrated Manufacturing, 30*(2–3), 215–228. https://doi.org/10.1080/0951192X.2016.1145812

Säfsten, K., & Gustavsson, M. (2024). *Research methodology 2.0 for engineers and other problem solvers* (2nd ed.; R. Ehnsiö, Trans.). Studentlitteratur.

Sargent, R. G. (2013). Verification and validation of simulation models. *Journal of Simulation, 7*(1), 12–24. https://doi.org/10.1057/jos.2012.20

Shafiq, M., Waqas, M., Shahzad, K., Rashid, Z., Ali, A., Awais Islam, M., Bilal, M., & Hameed, U. (2020). Optimizing throughput, WIP, and cycle time: A case study of utensils manufacturing shop floor. *Journal of Engineering Research, 8*(1). https://doi.org/10.1016/S2307-1877(25)00458-4

Spearman, M. L. (2014). Of physics and factory physics. *Production and Operations Management, 23*(11), 1875–1885. https://doi.org/10.1111/poms.12188

Thürer, M., Fernandes, N. O., Stevenson, M., & Qu, T. (2017). On the backlog-sequencing decision for extending the applicability of ConWIP to high-variety contexts: An assessment by simulation. *International Journal of Production Research, 55*(16), 4695–4711. https://doi.org/10.1080/00207543.2017.1281462

Thürer, M., Huang, Y., & Stevenson, M. (2021). Workload control in additive manufacturing shops where post-processing is a constraint: An assessment by simulation. *International Journal of Production Research, 59*(14), 4268–4286. https://doi.org/10.1080/00207543.2020.1761038

Wohlin, C. (2014). Guidelines for snowballing in systematic literature studies and a replication in software engineering. In *Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering* (pp. 1–10). Association for Computing Machinery. https://doi.org/10.1145/2601248.2601268

Yang, S., Yin, S., Wang, J., Li, W., & Xu, Z. (2025). Production simulation and configuration optimization based on discrete event simulation: A case study of an aerospace assembly workshop. *Journal of Mechanical Science and Technology, 39*(4), 2063–2073. https://doi.org/10.1007/s12206-025-0331-1

Zeigler, B. P., & Nutaro, J. J. (2016). Towards a framework for more robust validation and verification of simulation models for systems of systems. The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology, 13(1), 3–16. https://doi.org/10.1177/1548512914568657.

Appendix A: Systematic Literature Review: Search Protocol and Screening

I. Overview

Three searches were conducted, each with a different scope. Table A.1 summarises the protocol and results of each search. The screening flow for Search 1 is shown in Section A.2; for Searches 2 and 3, the corresponding counts are given in Table A.1. Section A.3 lists the studies cited from each search, together with those screened but not cited, and records the route by which each was retrieved. Several studies were retrieved by more than one search because the three searches address overlapping literatures. Therefore, the totals in Table A.1 are per search rather than counts of unique studies. Works obtained outside database searches are identified as such in Section A.3 and are described in Section 3.2.5.

Table A.1
Summary of the three searches

	Search1 Parameterisation	Search 2 HMLV automation	Search 3 Verification and validation
Purpose	Stochastic parameters for model construction	Theoretical framework	V&V methodology
Databases	Scopus; Primo (JU Library)	Web of Science; Scopus	Web of Science
Date range	Not restricted	2016–2026	2016–2026
Language	English	English	English
Document types	All	Articles (WoS); articles, books, conference papers (Scopus)	Articles
Records identified	1,534	2,695	2,744
Duplicates removed	641	16	10
Duplicate detection	Manual	Rayyan	Rayyan
Screening procedure	Manual, title/abstract/conclusion	Dual independent, Rayyan blind mode	Dual independent, Rayyan blind mode
Reviewer conflicts	n/a	5	3
Records screened	893	2,679	2,734
Full-text assessed	150	33	22
Studies retained	**11**	5	8

II Search 1: Core Parameterisation

Screening flow:

RECORDS IDENTIFIED THROUGH DATABASE SEARCHING N =1,534
↓
DUPLICATES REMOVED N = 641
↓
RECORDS SCREENED BY TITLE, ABSTRACT AND CONCLUSION N=893
↓
RECORDS EXCLUDED AT SCREENING N = 743
↓
FULL-TEXT ARTICLES ASSESSED FOR ELIGIBILITY N = 150
↓
Full-text articles excluded N = 139, no explicit stochastic parameters, outside HMLV / assembly scope, performance indicators not comparable
↓
STUDIES INCLUDED IN PARAMETERISATION N = 11

III Studies Retained from the Literature Searches

Table A.2
Search 1: Studies used for parameterisation (n = 11)

#	Study	Contribution to the model
1	Ojstersek et al. (2024)	Manual and HRC processing time distributions
2	Herps et al. (2022)	Automated cycle time reference; decoupled setup preparation
3	Miqueo et al. (2022)	Cycle and setup times for industrial HMLV lines; validation reference
4	Miqueo et al. (2023)	Throughput reference for manual and semi-automated HMLV lines
5	Khan & Standridge (2019)	CONWIP limit derivation in an HMLV cell
6	Frazee & Standridge (2016)	CONWIP mechanism and parameterisation
7	Shafiq et al. (2020)	Throughput, WIP and cycle time relationships
8	Perwitz et al. (2022)	Equipment failure parameters; implemented as the sensitivity case in Appendix C
9	Yang et al. (2025)	Configuration effects on capacity and bottleneck location
10	Biazen & Gebeyehu (2019)	Analytical versus DES model comparison

11	Gödri (2022)	Conflicting KPI structure in HMLV

Table A.3
Studies screened for the theoretical framework, by route of retrieval (n = 24)

#	**Study**	**Retrieved via**	**Cited in**
1	Gan et al. (2023): A review of the high-mix, low-volume manufacturing industry	Snowballing	2.1
2	Harrod & Kanet (2013): Applying work flow control in make-to-order job shops.	Snowballing	2.4
3	Gödri (2022): Improving delivery performance in HMLV manufacturing	Searches 1 and 2	2.1, 5.1
4	Frazee & Standridge (2016): CONWIP versus POLCA	Searches 1, 2 and 3	2.4
5	Gosavi & Gosavi (2024): CONWIP control in the digitized world	Snowballing	2.4
6	Kasper et al. (2023): Non-hierarchical work-in-progress control	Snowballing	2.4
7	Molenda et al. (2020): Backlog-sequencing: WLC versus CONWIP	Snowballing	2.4, 6.1
8	Mueller & Schmitt (2020): Production planning in autonomous and matrix-structured assembly	Snowballing	2.4
9	Prabhu et al. (2025): Workload control in a bottleneck-oriented flow shop	Snowballing	2.4, 6.1
10	Thürer et al. (2017): Backlog-sequencing for extending CONWIP to high-variety contexts	Snowballing	2.4, 6.1
11	Prajapat & Tiwari (2017): A review of assembly optimisation applications using DES	Snowballing	2.3
12	Goyal et al. (2026): Scenario-based simulation for developing new production systems	Snowballing	2.3, 6.1
13	Biazen & Gebeyehu (2019): Analytical and DES models of assembly line systems	Search 1	2.3, 6.3
14	Herps et al. (2022): Simulation-based design of an automated HMLV system	Searches 1 and 2	2.2.2, 3.4
15	Miqueo et al. (2022): Models to evaluate HMLV assembly line performance	Search 1	2.1, 3.5.2

#	Study	Retrieved via	Cited in
16	Yang et al. (2025): Production simulation and configuration optimisation, aerospace	Search 1	2.1, 5.3
17	Perwitz et al. (2022): Performance benefits from flexibility in matrix production	Search 1	2.2.2, 6.4
18	Shafiq et al. (2020): Optimizing throughput, WIP and cycle time	Search 1	2.1
19	Assembly flow design and process data digitalisation for process industries	Search 2	Screened, not cited
20	Intelligent order release in matrix-structured assembly systems (2024)	Search 2	Screened, not cited
21	Solution space exploration in HMLV manufacturing (2025)	Search 2	Screened, not cited
22	Novel availability and performance ratio for job shop processes	Search 2	Screened, not cited
23	Key performance indicators for facility performance assessment	Search 2	Screened, not cited
24	Berretini & Gobbo Junior (2025): Autonomicity in cyber-physical systems	Snowballing	Screened, not cited

Table A.4
Studies screened for verification and validation, by route of retrieval (n = 24)

#	Study	Retrieved via	Cited in
1	Sargent (2013/2014): Verifying and validating simulation models	Snowballing	2.5, 3.5, 5.2
2	Zeigler & Nutaro (2016): Robust V&V for systems of systems	Search 3	2.5, 5.2
3	Olsen & Raunak (2019): Increasing validity through metamorphic testing	Search 3	2.5, 3.5.1, 5.2
4	Leal et al. (2011): A practical guide for operational validation of DES models	Snowballing	2.5, 3.5.4
5	Mielczarek (2026): Operational validation via model-to-model comparison	Search 3	2.5
6	Thürer et al. (2021): Workload control in additive manufacturing shops	Searches 2 and 3	2.4
7	Mahmoodi et al. (2024): Data-driven simulation-based decision support	Searches 2 and 3	2.3

8	Miqueo et al. (2023): Parallel walking-worker flexible assembly lines	Searches 1, 2 and 3	2.1, 2.2.2, 3.5.2, 6.1
9	Khan & Standridge (2019): Aggregate simulation modeling, CONWIP limit	Searches 1 and 3	2.4, 3.3.2
10	Frazee & Standridge (2016): CONWIP versus POLCA	Searches 1, 2 and 3	2.4
11	A scenario-based visual modeling method for complex products lifecycle	Search 3	Screened, not cited
12	Simulation-validated genetic algorithm scheduling	Search 3	Screened, not cited
13	Simulation model for job shop production process improvement	Search 3	Screened, not cited
14	Modelling and simulation of multi-resource flexible job-shop scheduling	Search 3	Screened, not cited
15	Setting optimal production lot sizes and planned lead times in a job shop	Search 3	Screened, not cited
16	Graduation-inspired manufacturing system for hybrid assembly cell lines	Search 3	Screened, not cited
17	Multi-agent reinforcement learning for distributed FJSP	Search 3	Screened, not cited
18	Evolution strategies-guided deep RL for hybrid flow-shop scheduling	Search 3	Screened, not cited
19	AdaBoost-inspired differential evolution for reconfigurable FJSP	Search 3	Screened, not cited
20	A dynamic adaptive selection method for multi-objective FJSP	Search 3	Screened, not cited
21	A discrete improved grey wolf optimisation algorithm for dynamic scheduling	Search 3	Screened, not cited
22	An improved genetic algorithm for flexible job shop scheduling	Search 3	Screened, not cited
23	A hybrid multi-objective evolutionary algorithm for adaptive scheduling	Search 3	Screened, not cited
24	A genetic algorithm enhanced with neighbourhood structure for general FJSP	Search 3	Screened, not cited

Appendix B: Model Parameters and Input Data

I. Processing time distributions

Processing time at station *j* for a job of variant *v* is:

$$t(j,v) = \max(1, N(\mu_c , \sigma_c)) \times w_v$$

where μ_c and σ_c are the configuration-specific mean and standard deviation, and w_v is the work-content factor of variant *v*. The lower bound of one second prevents the normal distribution from producing non-positive samples in the tail.

Table B.1
Processing time parameters by configuration

Configuration	Stations	μ (s)	σ (s)	CV
1 Manual	S1-S5	85.0	12.0	0.141
2Semi-automated (HRC)	S1-S5	55.0	6.0	0.109
3 Automation-assisted	S1, S2, S4	40.0	3.0	0.075
	S3, S5	85.0	12.0	0.141

The coefficient of variation falls monotonically across the configurations (0.141 → 0.109 → 0.075), which is the dispersion-reduction signature of mechanical automation identified in Section 2.2.1.

II. Changeover

Changeover is incurred at station *j* when the variant of the arriving job differs from the variant of the job immediately preceding it at that station:

$$s(j) = S \text{ if } v_{current} \neq v_{previous(j)}, \text{ otherwise } 0$$

Each station maintains an independent record of the last variant processed, so changeover accrues from the realised sequence rather than from the aggregate mix.

Table B.2
Changeover parameters

Parameter	Value
S, Scenarios 1–3	60 s
S, Scenario 3b	30 s
Expected changeover frequency	0.70 per station per job

The expected frequency follows from the mix: the probability that two consecutive jobs share a variant is $0.4^2 + 0.3^2 + 0.2^2 + 0.1^2 = 0.30$, so a changeover occurs on 70 % of transitions in the long run.

III. Resource pools

Table B.3
Resource pools

Pool	Capacity	Serves	Configurations
operators	3	All operator-paced stations	All
autoStations	3	S1, S2, S4	Scenario 3, 3b only

Operator headcount is held at three in every configuration. This is the study's central experimental control. Labour is never withdrawn as automation increases, so any observed performance difference reflects the reallocation of work rather than a change in available resources.

IV. Model structure in AnyLogic

Table B.4
AnyLogic block structure

Block	Function
SOURCE	Poisson arrivals; assigns variant and work-content factor on exit
SEIZECONWIP	Seizes one token from the CONWIP pool; blocks release when the cap is reached
S1–**S5**	Service blocks; seize resource, delay by processing plus changeover, release
TIMEMEASURESTART / **TIMEMEASUREEND**	Lead time measurement across the full line
RELEASECONWIP	Releases the CONWIP token on job completion
SINK	Job exit and throughput counting

Variant assignment is performed on exit from the source:

```
double r = uniform();
if      (r < 0.4) { agent.variant = 0; agent.wf = 0.8; }
else if (r < 0.7) { agent.variant = 1; agent.wf = 1.0; }
else if (r < 0.9) { agent.variant = 2; agent.wf = 1.3; }
else              { agent.variant = 3; agent.wf = 1.6; }

agent.setupTime = (lastN == agent.variant) ? 0 : setupDur;
lastN = agent.variant;
```

Appendix C: Detailed simulation results

All runs: AnyLogic 8 PLE, 200 hours simulated, arrival rate 13 jobs per hour, CONWIP limit 10 unless otherwise specified. Section I uses fixed seeds 1 to 5 and can be reproduced exactly. Sections II to VII report sweeps; where a seed is stated the value is a single replication, and where a confidence interval is given the point is a mean over independently seeded replications.

I. Replication-level results, baseline configurations

Table C.1
All replications, three configurations plus the changeover-reduction case

Configuration	Seed	Completed jobs	Lead time (s)	Op. util (%)	CONWIP util (%)	Throughput (jobs/h)	Lead time (min)	WIP (units)
Manual	1	2,584	1,160	77	39	12.92	19.33	4.163
	2	2,548	1,190	77	39	12.74	19.84	4.213
	3	2,568	1,142	77	39	12.84	19.03	4.073
	4	2,615	1,341	78	42	13.07	22.35	4.870
	5	2,634	1,183	78	42	13.17	19.72	4.329
	Mean			**77.4**	**40.2**	**12.95**	**20.05**	**4.330**
	95 % CI			± 0.8		**± 0.25**	**± 1.88**	**± 0.45**
	CV			0.7 %		1.3 %	6.6 %	7.3 %
Semi-automated	1	2,587	602.0	58	21	12.94	10.03	2.163
	2	2,504	611.5	57	21	12.52	10.19	2.127
	3	2,592	641.2	58	23	12.96	10.69	2.308
	4	2,629	609.7	59	22	13.14	10.16	2.226
	5	2,595	639.5	59	23	12.97	10.66	2.305

	Mean			**58.2**	**22.0**	**12.91**	**10.35**	**2.226**
	95 % CI			± 1.2		**± 0.33**	**± 0.44**	**± 0.12**
	CV			1.4 %		1.8 %	3.0 %	3.7 %
Automation-assisted	1	2,576	520.3	30	18	12.88	8.672	1.862
	2	2,561	528.3	30	18	12.80	8.806	1.879
	3	2,709	531.7	32	20	13.54	8.861	2.000
	4	2,582	520.1	30	18	12.91	8.669	1.865
	5	2,701	529.5	32	19	13.51	8.825	1.986
	Mean			**30.8**	**18.6**	**13.13**	**8.767**	**1.918**
	95 % CI			± 1.6		**± 0.52**	**± 0.13**	**± 0.10**
	CV			3.6 %		2.8 %	1.0 %	3.6 %
Automation-assisted+ changeover reduction	1	2,658	410.3	26	15	13.29	6.838	1.515
	2	2,522	411.3	25	14	12.61	6.855	1.441
	3	2,650	411.5	26	15	13.25	6.859	1.515
	4	2,571	411.3	26	15	12.86	6.855	1.469
	5	2,576	409.4	25	15	12.88	6.823	1.465
	Mean			**25.6**	**14.8**	**12.98**	**6.846**	**1.481**
	95 % CI			± 0.8		**± 0.41**	**± 0.02**	**± 0.05**
	CV			2.1 %		2.2 %	0.2 %	2.2 %

Notes. Throughput = completed jobs ÷ run length. Lead time (min) = mean lead time (s) ÷ 60. WIP = throughput × lead time ÷ 60, by Little's Law. Confidence intervals are 95 %, computed as $t_{(0.025,4)} \times s \div \sqrt{5}$ with t = 2.776. Automated-station pool utilisation in Scenario 3 was 29, 29, 31, 29 and 31 per cent across the five seeds (mean 29.8 %), and 23, 21, 22, 22 and 22 per cent in Scenario 3b (mean 22.0 %).

II. CONWIP Sweep

Table C.2
Effect of the CONWIP limit, all three configurations, seed 1

Configuration	Limit	Completed jobs	Lead time (s)	CONWIP util (%)	Throughput (jobs/h)	Lead time (min)
1 Manual	3	2,645	989.5	78	13.22	16.49
	5	2,593	986.6	61	12.96	16.44
	8	2,630	1,110	48	13.15	18.50
	10	2,584	1,160	39	12.92	19.33
	12	2,596	1,051	31	12.98	17.51
2 Semi-automated	3	2,539	560.8	57	12.70	9.346
	5	2,507	583.6	39	12.54	9.727
	8	2,632	610.2	27	13.16	10.17
	10	2,587	602.0	21	12.94	10.03
	12	2,587	602.0	18	12.94	10.03
3 Automation-assisted	3	2,680	453.8	51	13.40	7.564
	5	2,648	472.5	34	13.24	7.875
	8	2,606	482.0	21	13.03	8.034
	10	2,576	520.3	18	12.88	8.672
	12	2,600	483.2	14	13.00	8.053

Throughput range across the entire sweep: manual 12.92–13.22 (2.3%), semi-automated 12.54–13.16 (4.9%), automation-assisted 12.88–13.40 (4.0%). In each case, the range is comparable to or smaller than the between-replication range at a fixed limit, so no throughput effect is detectable.

III. Load sweep

Table C.3
Throughput, lead time and utilisation against offered load, seed 1

Configuration	Offered load	Completed jobs	Lead time (s)	Op. util (%)	Autostationutil (%)	Throughput (jobs/h)	Lead time (min)	Demand met	Saturated
Manual	13	2,584	1,160	77		12.92	19.33	99.4%	No
	16	3,094	1,827	91		15.47	30.45	96.7%	Yes
	19	3,348	41,620	99		16.74	693.6	88.1%	Yes
	22	3,341	80,920	99		16.70	1,349	75.9%	Yes
	25	3,353	121,300	99		16.77	2,021	67.1%	Yes
Semi-automated	13	2,587	602.0	58		12.94	10.03	99.5%	No
	16	3,168	744.6	71		15.84	12.41	99.0%	No
	19	3,719	1,005	83		18.59	16.75	97.9%	No
	22	4,260	4,184	96		21.30	69.73	96.8%	Yes
	25	4,402	39,830	99		22.01	663.8	88.0%	Yes
Automation-assisted	13	2,576	520.3	30	29	12.88	8.672	99.1%	No
	16	3,255	528.5	38	37	16.27	8.808	100%	No
	19	3,789	552.1	45	43	18.95	9.202	99.7%	No
	22	4,382	570.4	52	50	21.91	9.507	99.6%	No
	25	4,916	599.1	58	56	24.58	9.985	98.3%	No
	28	5,565	637.4	66	64	27.82	10.62	99.4%	No

IV. Run-length robustness

Table C.4
Manual configuration, seed 1

Run length (s)	Run length (h)	Completed jobs	Lead time (s)	Throughput (jobs/h)	Lead time (min)
180,000	50	634	1,192	12.68	19.87
720,000	200	2,584	1,160	12.92	19.33
1,800,000	500	6,505	1,167	13.01	19.45

V. Station-level utilisation and its derivation

AnyLogic Service blocks do not report utilisation. Station utilisation was derived from resource pool statistics:

> **station utilisation = (pool utilisation × pool capacity) ÷ stations served by that pool**

The derivation is exact because a station processes only while holding a resource unit from its pool, so resource-busy time equals station-busy time.

Table C.5
Derivation of the figures reported in Table 8

Configuration	Pool	Pool util	Capacity	Stations served	Derived station util
1 Manual	operators	77.4 %	3	5	46.4 %
2 Semi-automated	operators	58.2 %	3	5	34.9 %
3 Automation-assisted	operators	30.8 %	3	2 (S3, S5)	46.2 %
	autoStations	29.8 %	3	3 (S1, S2, S4)	29.8 %
3b + changeover reduction	operators	25.6 %	3	2 (S3, S5)	38.4 %
	autoStations	22.0 %	3	3 (S1, S2, S4)	22.0 %

VI. Automation depth

Table C.6
Capacity ceiling against number of automated workstation

k	Operator paced stations	Predicted operator ceiling	Predicted equipment ceiling	Observed ceiling	Operator utilisation at the ceiling
0	5	16.6		16.63 ± 0.04	99.9 %
1	4	20.7	43.1	20.63 ± 0.09	99.0 %
2	3	27.6	43.1	27.58 ± 0.09	98.9 %
3	2	41.4	43.1	41.23 ± 0.05	98.7 %
4	1	82.8	43.1	43.54 ± 0.07	52.0 %
5	0	No operator paced station	43.1	43.47 ± 0.08	0.0 %

Note: Workstations were automated in the order S1, S2, S4, S3, S5, so that k = 3 reproduces Scenario 3. The automated-station pool is scaled with k at one unit per automated station, preserving the one-to-one ratio used in Scenario 3. The predicted operator ceiling is the operator supply of 3 × 3600 s divided by the operator time per job, which equals the number of operator-paced stations multiplied by the manual cycle of 130.4 s. The predicted equipment ceiling is 3600 s divided by the automated cycle of 83.6 s; it does not depend on k, because pool capacity and station count rise together. Ceilings for k = 1, 2, 4 and 5 are the mean throughput across the offered loads at which throughput ceased to track the offered load, with ten replications per load point and independently drawn seeds. Ceilings for k = 0 and k = 3 are those reported in Table 12.

VII. Equipment reliability sensitivity

Table C.7
Automation-assisted configuration with and without equipment failures, offered load 13 jobs per hour

Measure	Perfect availability	With failures
Throughput (jobs/h)	13.13 ± 0.52	12.81 ± 0.42
Lead time (min)	8.77 ± 0.13	8.74 ± 0.12
Operator utilisation	30.8 ± 1.6 %	30.7 ± 1.1 %
Constraint station utilisation	46.2 %	46.2 %

Note. Five replications in each condition. Failures are modelled at the automated workstations only, with a mean time between failures of 200 minutes and a mean repair time of 6.2 minutes, derived from Perwitz et al. (2022).

Appendix D: Analytical derivation of utilisation and capacity ceilings

I. Derived parameters

The mean work factor across the product mix is the sum of each variant's probability multiplied by its work-content factor:

$$0.4(0.8) + 0.3(1.0) + 0.2(1.3) + 0.1(1.6) = \mathbf{1.040}$$

The probability that two consecutive jobs are of different variants, and therefore that a changeover is incurred, is one minus the sum of the squared mix probabilities:

$$1 - (0.4^2 + 0.3^2 + 0.2^2 + 0.1^2) = 1 - 0.30 = \mathbf{0.70}$$

Mean changeover per workstation is therefore 0.70×60 s = **42.0 s** in Scenarios 1 to 3, and 0.70×30 s = **21.0 s** in Scenario 3b.

II. Workstation cycle times

Table D.1
Mean cycle time per workstation by station type

Station type	Nominal	Processing	Changeover	Cycle
Manual	85 s	88.4 s	42.0 s	130.4 s
Collaborative	55 s	57.2 s	42.0 s	99.2 s
Automated	40 s	41.6 s	42.0 s	83.6 s
Manual, reduced changeover	85 s	88.4 s	21.0 s	109.4 s

Note. Processing is the nominal time multiplied by the mean work factor of 1.040.

III. Operator time consumed per job

Table D.2
Operator time per job by configuration

Configuration	Operator-paced stations	Cycle each	Operator time per job
Manual	5	130.4 s	652.0 s
Semi-automated	5	99.2 s	496.0 s
Automation-assisted	2	130.4 s	260.8 s
Automation-assisted with changeover reduction	2	109.4 s	218.8 s

IV. Predicted operator utilisation

Utilisation is throughput multiplied by operator time per job, divided by the operator pool size of three and by 3600 seconds.

Table D.3
Predicted and simulated operator utilisation at an offered load of 13 jobs per hour

Configuration	Throughput (jobs/h)	Predicted	Simulated	Difference
Manual	12.948	78.2 %	77.4 %	0.8 pp
Semi-automated	12.906	59.3 %	58.2 %	1.1 pp
Automation-assisted	13.128	31.7 %	30.8 %	0.9 pp
Automation-assisted with changeover reduction	13.290	26.9 %	25.6 %	1.3 pp

The predicted figure exceeds the simulated figure in every case. This is expected, since throughput counts completed jobs while some work remains within the system at the end of the run.

V. Predicted capacity ceilings

The ceiling is the operator supply divided by the operator time each job consumes, that is 3 × 3600 s divided by the figures in Table D.2.

Table D.4
Predicted and measured capacity ceilings

Configuration	Calculation	Predicted	Measured	Difference
Manual	10800 ÷ 652.0	16.6	16.63 ± 0.04	0.18 %
Semi-automated	10800 ÷ 496.0	21.8	21.96 ± 0.06	0.73 %
Automation-assisted	10800 ÷ 260.8	41.4	41.23 ± 0.05	0.41 %

VI. Zero-variation verification

Both processing-time standard deviations were set to zero; changeover duration was set to zero; every job was assigned to variant A; and the arrival rate was reduced to 5 jobs per hour, so the model operated well below its capacity ceiling. Under these conditions, the time in the system of a job that never waits is determined entirely by the input parameters, namely five workstations at 85 s each and the variant A work factor of 0.8, giving exactly 340 s. Across 985 completed jobs, the observed minimum was 340.000 s, matching the analytical value to the resolution reported by the model. The mean was 341.703 s, with a maximum of 454.568 s. The excess above the minimum arose from occasional waiting for an operator, since three operators serve five operator-paced workstations and arrivals are stochastic. The operator pool stood at 15 per cent utilisation, and the token pool at 4 per cent, so the excess comes from contention, not saturation. The minimum is the quantity that verifies the calculation, because it is the only observation guaranteed to contain no queueing.