Multi-objective analysis of manufacturing systems

TNO Public › TNO 2024 P11832
13 December 2024

ICT, Strategy & Policy
www.tno.nl
+31 88 866 50 00
info@tno.nl

TNO 2024 P11832 – 13 December 2024

Multi-objective analysis of manufacturing systems

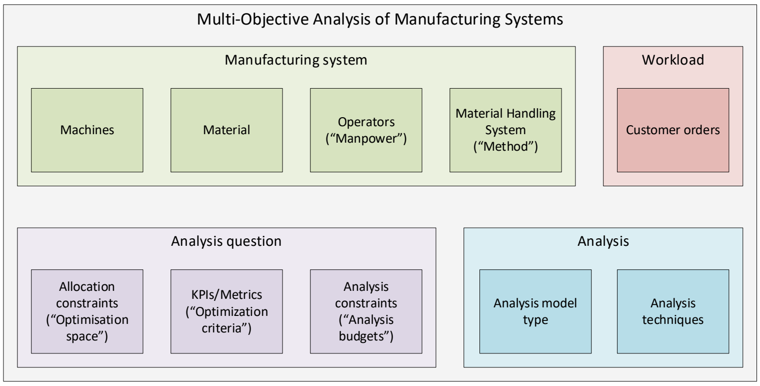
Author(s)	Jacques Verriet
Classification report	TNO Public
Title	TNO Public
Report text	TNO Public
Number of pages	53 (excl. front and back cover)
Number of appendices	0
Project name	Effective Performance Engineering 4 CPPS
Project number	060.61073/01.01

Summary

The systems being developed by the high-tech industry do not operate in isolation. They are combined with other systems to provide value to their owners. This report is concerned with an important type of systems of systems of which high-tech systems are a part, namely manufacturing systems (of systems). The report addresses the analysis of the performance of such manufacturing systems. The performance of a manufacturing system typically involves a system-specific combination multiple objectives, which we will refer as the system's effectiveness or the system's effective performance. As the effectiveness of a manufacturing system (of systems) cannot be expressed by a single value, this report considers the multi-objective analysis of manufacturing systems.

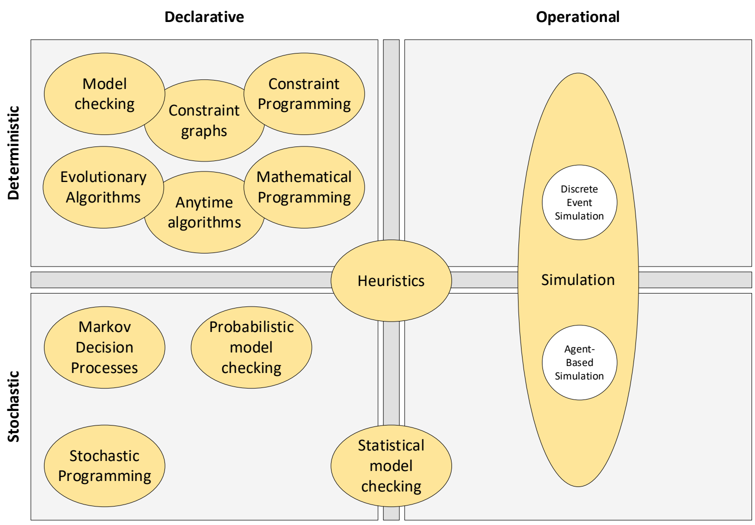
Landscape

The report introduces the following landscape to capture the relevant aspects of a manufacturing system (of systems) and to reason about these aspects using multi-objective analysis.



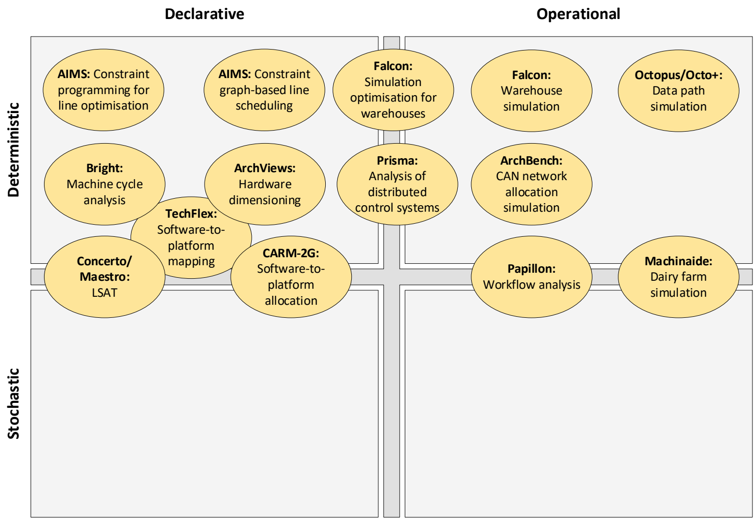
State of the Art

The first part of this report involves an analysis of the state of the art on multi-objective analysis of manufacturing systems. This analysis of the state of the art shows that each of the landscape's aspects individually involves many academic publications. As it is impossible to be complete, this report includes a limited literature study for a selection of the landscape's analysis-related aspects. The state of the art on analysis methods is summarised using the classification along the axes deterministic vs. stochastic and declarative vs. operational. This classification is shown in the picture below.



TNO-ESI knowledge

The second part of this report considers the expertise related to multi-objective analysis of manufacturing systems that TNO-ESI has gained in its 20+ years of projects. The picture below summarises this knowledge using the analysis method classification introduced earlier.



Recommendation

The figure above shows that TNO-ESI's projects over the past 20 years primarily focused on deterministic analysis approaches. Deterministic approaches work well for low-mix high-volume (LMHV) manufacturing systems, involving production of large batches of few different products.

High-mix low-volume (HMLV) systems, involving production of small batches of many different products, have a greater variety of behaviour. For such systems, stochastic analysis techniques may be valuable. To address this, we recommend performing a study project to assess the added value of stochastic analysis techniques for analysing HMLV manufacturing systems, e.g. the ones of TNO-ESI's industrial partners' customers. Special attention should be given to the variability of resources and the workload: the study project should (1) assess the influence of stochastic behaviour in the different manufacturing systems and (2) evaluate the added value of existing stochastic analysis techniques compared to deterministic analysis techniques.

Contents

Abbreviations

Abbreviation	Meaning
ABC	Activity-Based Costing
ABS	Agent-Based Simulation
ACP	Algebra of Communicating Processes
AGV	Automated Guided Vehicle
AI	Artificial Intelligence
AIMMS	Advanced Interactive Multidimensional Modeling System
AMPL	A Mathematical Programming Language
APT	Actual Processing Time
AUPT	Actual Unit Processing Time
BDD	Binary Decision Diagram
CAPEX	Capital Expenditure
CP	Constraint Programming
CPN	Coloured Petri Nets
CPPS	Cyber-Physical Production System
CPS	Cyber-Physical System
CSP	Communicating Sequential Processes
DECN	Discrete Event Control Network
DES	Discrete Event Simulation
DPN	Dataflow Process Network
EPT	Effective Process Time
ESI	Embedded Systems Institute
FIFO	First In First Out
GLPK	GNU Linear Programming Kit
GPE	Global Production Effectiveness
HMLV	High-Mix Low-Volume
ILP	Integer Linear Programming
KPI	Key Performance Indicator
LMHV	Low-Mix High-Volume
mCRL2	Micro Common Representation Language 2
MDP	Markov Decision Process
MEE	Machining Equipment Effectiveness
MILP	Mixed Integer Linear Programming
ML	Machine Learning
MOM	Manufacturing Operations Managements
MPSE	Multiproduct Production System Effectiveness
NEE	Net Equipment Effectiveness
NSGA	Non-dominating Sorting Genetic Algorithm
nuSMV	New Symbolic Model Verifier
OAE	Overall Asset Effectiveness
OEE	Overall Equipment Effectiveness

OEE_{Flex}	Flexibility-induced equipment effectiveness
OOEML	Overall Equipment Effectiveness of a Manufacturing Line
OFE	Overall Fab Effectiveness Overall Factory Effectiveness
OLE	Overall Line Effectiveness
OPE	Overall Plant Effectiveness
OTE	Overall Throughput Effectiveness Overall Transport Effectiveness
PBT	Planned Busy Time
PEE	Production Equipment Effectiveness
Promela	Process Meta Language
SDF	Synchronous Dataflow
SDF3	Synchronous Dataflow for Free
SDS	System Dynamics Simulation
SOAPS	Simulation Optimisation Applied to Production Systems
SPIN	Simple Promela Interpreter
TEEP	Total Equipment Effectiveness Production
TOEE	Total Overall Equipment Effectiveness
WIP	Work in Process

1 Introduction

The systems being developed by the high-tech industry are not standalone systems: they need to be combined into a larger whole, a system of systems, to provide value to their owners. In this report, we will investigate the performance of one important type of these larger wholes: i.e. manufacturing systems (of systems) or cyber-physical production systems (CPPSs). Manufacturing systems are production facilities that consist of multiple (high-tech) machines that cooperatively manufacture products according to a production process.

The focus of the report will be on the effective usage of the resources of a production system, i.e. a usage that optimises the production system's performance objectives. To optimise the effective resource usage of a (manufacturing) system of systems, it is not sufficient to optimise the contained (high-tech) systems individually: one needs an integral look at the entire manufacturing system of system [1]. In this report, we will consider the aspects that play a role in the effective usage of a manufacturing system's resources to fulfil its customers' orders and ways to evaluate these aspects.

1.1 Scope

Manufacturing is a process using resources to perform operations on materials to produce products [2]. Manufacturing systems involve humans, machinery, and equipment connected via flows of material and information [3]. For the optimal functioning of a manufacturing system, many decisions must be made. The 4 Ms of manufacturing [4] can be used to classify decision aspects (see Table 1.1):
1. Manpower: The labour of people
2. Method: Production processes
3. Machine: Tools and equipment
4. Material: Raw material, components and consumables

Table 1.1: 4 Ms of manufacturing [5]

M characteristic	Description	Performance aspects
1. Manpower	Labour of people involved in delivering products and services for production	Efficiency of the operators
2. Machine	Equipment, facilities, tools employed for production	Usage and maintenance scheduling, avoiding unnecessary downtime
3. Method	Process, shipping, schedule, procedure	Products flow efficiently through the production line, finished products quickly exit the line
4. Material	Raw materials, consumables, components used to satisfy production	Parts and materials close to workstation, while not overcrowding the operator

Later, two additional Ms were introduced [6]:
5. *Milieu*: Environmental events
6. Measurement: Inspection and other measurements

According to Maier [7], manufacturing systems are directed systems of systems, i.e. they are centrally managed to fulfil a certain purpose. This central management must make many decisions with respect to the optimal usage of a manufacturing system's resources. The types of decisions to be made about the 6 Ms have different timelines and different scopes. For the timelines, Rouwenhorst et al. [8] distinguish strategic, tactical and operational decisions:

- *Strategic* decisions have a long-term impact and concern high investments. The two main groups are the decisions concerning the design of the process flow and the decisions concerning the selection of the machine types.
- *Tactical* decisions include the dimensions of resources (storage system sizes but also number of employees) and the determination of a layout.
- At the *operational* level, processes must be carried out within the constraints set by the strategic and tactical decisions made at the higher levels. The main decisions at this level concern assignment and control problems of people and equipment.

This report sketches a landscape with respect to the analysis of a manufacturing system's effectiveness. The report's focus is on the effective usage of a manufacturing system's resources, i.e. the original 4 Ms of manufacturing: manpower, method, machine and material. Musselman [1] identifies two main steps in manufacturing system resource scheduling: allocation and sequencing.

- *Allocation*[1] is the selection which resources to use to fulfil a piece of manufacturing work.
- *Sequencing* determines the order in which work is being fulfilled by the corresponding resources.

Musselman [1] states that allocation is dominant in make-to-order/make-to-engineer manufacturing processes, in which customised products are manufactured upon receiving a customer order. Sequencing is dominant in make-to-stock manufacturing processes with a high product mix. In this report, we will generally use the term allocation for the combination of allocation and sequencing.

Effectiveness is a measurement which is different per manufacturing system. It typically involves a combination of multiple aspects. Examples of such aspects include costs, timing, machine and personnel utilisation, product quality, sustainability and robustness against uncertainty/exceptions.

As the report's focus is on the effective usage of a manufacturing system's resources, the report hence mainly considers operational decision making, but it also touches strategic and tactical decision making.

1.2 Outline

This report sketches a landscape for multi-objective analysis of manufacturing systems and identifies which parts of the landscape have (not) been addressed by TNO-ESI's projects. The multi-objective manufacturing system analysis landscape is presented in Chapter 2. Chapters 3 and 4 investigate the state of the art of two of the landscape's main elements, KPIs and analysis methods. Chapter 5 looks at the coverage of TNO-ESI's (past and current) projects of the landscape to identify opportunities to explore new fields of knowledge, which help the high-tech ecosystem optimise their systems in a system-of-systems context.

[1] Musselman [1] uses the term synchronisation instead of allocation.

2 Manufacturing system analysis landscape

In this chapter, we will sketch a landscape for the multi-objective analysis of manufacturing systems. In the literature, one can find several classifications of manufacturing systems. We use the classifications of Dietrich [2] and Ghasemi et al. [9] to identify relevant elements for our landscape. Ghasemi et al. [9] consider four categories with respect to simulation optimisation applied to production systems (SOAPS):

- Problem and modelling: Production environment (e.g. flow shop, job shop), optimisation objective(s), sources of uncertainty, modelling strategies.
- Solving methodology: Optimiser scope (i.e. local or global), optimiser type, simulator type, integration method.
- Real applications: Case studies.
- Experimenting: Validation strategy.

Dietrich [2] proposes a detailed taxonomy for discrete manufacturing systems. The three main categories of her taxonomy are the following.

- Production process involves the operations needed to produce a product. This includes operations, material flow, information flow, and contention for resources.
- System management involves the way a manufacturing system is managed. Dietrich identifies requirements generation, WIP control policy, and distribution of information and control as the main aspects of this category.
- System behaviour involves the behaviour exhibited by an operational manufacturing system. Important elements are operational data and material handling data.

Both Ghasemi et al. [9] and Dietrich [2] observe that the operations of a manufacturing system can be both deterministic and stochastic and that they depend on each other, e.g. sequence-dependent setup times. What they both do not explicitly distinguish is controllable and uncontrollable behaviour. Some behaviour cannot be controlled. Examples are the arrival of customer orders, the products being ordered, machine breakdowns and variation of operation durations.

In Chapter 1, we discussed the 4 Ms of manufacturing: Manpower, Method, Machine, and Material. The 4 Ms capture the resources of a manufacturing systems and some restrictions on how they may be used. They correspond to the problem and modelling category of Ghasemi et al. [9] and the production process and system management categories of Dietrich [2]. The 4 Ms are, however, not sufficient for the analysis of a manufacturing system, especially not if the analysis involves the influence of different allocations of work to a manufacturing system. One must also consider the workload of a manufacturing system. In addition, the landscape should include aspects concerning the analysis itself. These corresponds to the solving methodology of Ghasemi et al.'s classification [9].

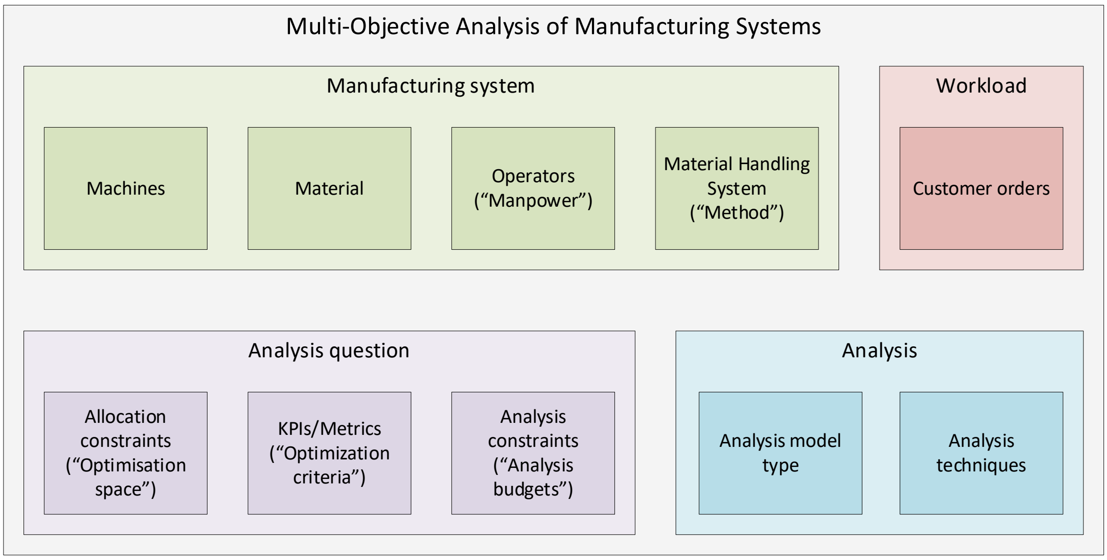


Figure 2.1 Landscape for multi-objective analysis of manufacturing systems

Figure 2.1 shows a landscape for multi-objective analysis of manufacturing systems. The landscape consists of four interdependent elements:

1. The *Manufacturing system* element corresponds to the 4 Ms of manufacturing. We distinguish four aspects.
   a. The *Machines* aspect represents the equipment that transforms materials to create products.
   b. The *Material* aspect represents all materials being handled in the systems. These include raw materials, partial products and final products as well as consumables.
   c. The *Operators* aspect represent the people in a manufacturing system and the services they provide. These services include operating a machine, servicing a machine and transporting materials.
   d. The *Material Handling System* aspect represents the transportation in a manufacturing system. Transportation may be fully automated transportation using e.g. conveyor belts and AGVs, or it involves operators moving materials between different machines.
2. The *Workload* element represents that work that needs to be performed by a manufacturing system. The work involves the fulfilment of customers ordering products. These customer orders need to be translated into manufacturing recipes that can be executed by the manufacturing system’s equipment. Depending on the scope of the analysis, the workload may involve the orders of a short or long period of time.
3. The *Analysis question* element represents the question that a decision maker wants to answer for a given manufacturing system and a corresponding workload. There are many different questions. Examples include which manufacturing recipe to use, which resources to use to execute a recipe, and the order in which a recipe’s operations are executed execution. We have identified three aspects:
   a. The *Variation points* aspect represents the alternatives that the decision maker needs to select from. These variation points may involve variations of the 4 Ms, e.g. variations of the layout of the production system or different workloads. The variation points may also involve the way the 4 Ms are used in the production process, e.g. the allowed allocations of work to the resources. The number of alternatives is typically very large, too large to be analysed individually. E.g. the number of different operation sequences grows exponentially with the number of operations.

   b. The *KPIs/Metrics* aspect represents the criteria that a decision maker uses to select one of the available alternatives. In case a decision maker has multiple decision criteria, we have a multi-objective analysis problem at hand.
   c. The *Constraints* aspect involves the constraints a decision maker defines to limit the number of allowed alternatives. These limits may also involve the resources used for finding an alternative, e.g. the time available for finding a feasible solution.
4. The *Analysis* element represents the techniques and tools used to answer an analysis question for a manufacturing system and a corresponding workload. We distinguish two aspects.
   a. The *Analysis model type* aspect represents the model that is used to address an analysis question. This includes the formalism in which the analysis question is translated.
   b. The *Analysis techniques* aspect represents the techniques that are applied to an analysis model to answer an analysis question.

In Sections 2.1, 2.2, 2.3, and 2.4, the four main elements are discussed in more detail. Section 2.1 describes the manufacturing system element, Section 2.2 the workload element, Section 2.3 the analysis question element, and Section 2.4 the analysis element.

2.1 Manufacturing system element

The landscape's manufacturing system element of Figure 2.1 covers the resources of a manufacturing system. A manufacturing system can be seen as a collection of production devices, transport devices, human operators and materials. These (interdependent) aspects correspond to the original 4 Ms of manufacturing: Manpower, Method, Machine, and Material. The aspects are described in Sections 2.1.1, 2.1.2, 2.1.3, and 2.1.4.

2.1.1 Machines aspect

The production devices, corresponding to the Machine M, are represented by the Machines aspect. These devices perform operations on materials. Dietrich [2] distinguishes seven types of operations according to the type of material being handled: bulk, kitting, fabrication, assembly, by-product, distribution and consumption. One production device may be able to perform multiple operations; Van De Ginste et al. [10] refer to this variability as process flexibility.

The focus of this report is multi-objective analysis of manufacturing systems. To facilitate this, it is not sufficient to know what operations machines can perform. One also needs to know the "costs" of these operations with respect to the optimisation criteria (of the KPIs/Metrics aspect of the Analysis question element). These costs can be of various types, as will be explained in Section 2.3.2.

There are various types of manufacturing systems depending on the amount/volume and the variation/mix of the work they need to fulfil. Cost-effective variants with totally different characteristics are Low-Mix High-Volume (LMHV) and High-Mix Low-Volume (HMLV) manufacturing systems (also see Section 2.2). LMHV systems have a highly repetitive workload: long batches of the same product. HMLV systems, on the other hand, have a highly varying workload, which involves changing between different operations or different materials being processed. Especially for such manufacturing system, the Machines aspect should also include the costs of setting up the machines for the next operation to be performed. For timing-related optimisation criteria, these costs are referred to as sequence-dependent setup times [11].

Manufacturing systems come with a lot of variability and uncertainty. Part of this variability and uncertainty originates from the machines and should be part of the description of a manufacturing system's machines. There is typically machine variability with respect to the costs of a performed operation. The most studied stochastic manufacturing system behaviour is variation of machine processing times [9]. Moreover, there may be variability in the quality of the performed operations and the availability of a machine (for a certain operation) [12].

2.1.2 Material aspect

The material aspect represents all the materials handled in a manufacturing system. Manufacturing systems are facilities that transform raw materials into final products. These are, together the partial products that are created during the manufacturing process, the most important materials.

However, there are other materials that play a role in a manufacturing system. Consumables are materials that are not normally included in bills of material or are not individually accounted for in specific production requests [13]. One can think of the glue in a gluing process and the ink in the printing process, but also the lubricants of a machine's motor.

A third type of material involves waste materials. Operations like cutting and drilling come with a loss of material, which may play an important role in the manufacturing process. Waste materials certainly play a role with respect to cost and sustainability of operations.

Not all materials in a manufacturing system can be transported easily; some materials have dimensions, shapes or weights to make them difficult to transport [14]. These so-called non-conveyable materials can be made conveyable by storing them in or on top of a special carrier. One can think of bins in which small objects can be placed or pallet on top of which large objects can be placed. These reusable carriers are another type of materials, and these may involve a complex logistic flow.

2.1.3 Operators aspect

A third type of resource in a manufacturing system involves its labour force. To obtain a smoothly operating manufacturing system, human operators play multiple roles. This is not expected to change with the upcoming development in AI [15] [16]. In normal operation, operators may be involved in performing operation at/by a machine and they may transport materials from one location to another. Operators may also play an important role in service, e.g. the replenishment of consumables, and maintenance, e.g. the cleaning and repair of machines.

Like machines, different operators may have different capabilities. Advanced operations may require more experienced operators whereas simple operations can be performed by all operators. This is especially relevant when a manufacturing system has a flexible workforce. There are manufacturing systems whose workload varies periodically. E.g. manufacturing systems producing for the end customer often have a peak workload around the Christmas holidays. To scale up production, temporary staff is employed, and these labourers are typically less skilled than the permanent staff.

The behaviour of human operators varies more than machine behaviour, which can be highly predictable [17]. This means that there may be a higher need for stochastic analysis for manufacturing systems with a lower degree of automation.

2.1.4 Material Handling System aspect

Manufacturing a product involves multiple operations that need to be executed in a certain sequence (see Section 2.2). As most machines can perform only one type of operation, transport of materials (see Section 2.1.2) between production machines (see Section 2.1.1) is required. This is handled by the material handling system. In their glossary of terms [18], SRSI define material handling as “the movement, storage, control, and protection of materials and products throughout the process of their manufacturing, distributing, consumption and disposal.” The transport between machines can be performed by mobile devices like AGVs and by stationary devices like conveyor belts. Human operators (see Section 2.1.3) may also perform transportation tasks, typically using reach trucks, pallet trucks or forklifts.

The material handling system defines the (allowed and/or possible) transport paths between the machines. Note that the allowed paths are partially determined by the stationary transport devices. The rest is determined by the type of process that is to be supported, i.e. the Method M of the 4 Ms of manufacturing [5]. Flexible processes allow more paths than fixed processes; Van De Ginste et al. [10] refer to allowing multiple routes as routing flexibility.

Flow shops and job shops are common types of manufacturing systems. Both involve jobs that involve strict sequences of operations. In flow shops, these sequences are identical for all jobs and the jobs visit the same machines in the same order [19]. Job shops have more flexibility than flow shops: the sequences of operations may differ per job as well as the allocation of these operations to machines [20]. Note that job shops require a lot of routing flexibility, whereas flow shops require little routing flexibility.

Like for the machines (see Section 2.1.1), there is uncertainty and variability in the material handling system, and this may have a significant influence on the effectiveness of a manufacturing system. There is variability with respect to the costs of transport, e.g. the duration. In addition, transport equipment may be unavailable because of failures and operators may be unavailable because of (unplanned) breaks.

2.2 Workload element

The landscape’s workload element refers to the customer orders that a manufacturing system must fulfil. These orders can be characterised along two dimensions: mix and volume. Mix refers to the number of different products that need to be produced, volume to the number of identical products. There are two extremes with restrict to this characterisation. Low-Mix High-Volume (LMHV) refers to large quantities of identical products and High-Mix Low-Volume (HMLV) refers to small quantities of a high variety of products [21]. The former is referred to as mass production, whereas the latter as mass customisation.

There is a clear relation between the process supported by a manufacturing system and its workload. The Hayes-Wheelwright matrix (see Table 2.1) indicates that a higher mix of products require a more flexible manufacturing process and that a very high volume requires little flexibility. Hayes and Wheelwright [22] argue that combinations far away from the matrix' diagonal are rare.

Table 2.1: Hayes-Wheelwright matrix [22]

Product structure Process structure	Low Volume Unique products	Low Volume Multiple products	High Volume Standardised products	Very High Volume Commodity products
Jumbled Flow	Job shop		*More process flexibility than required, so higher cost*	
Disconnected Line Flow		Batch		
Connected Line Flow	*Less process flexibility than required, so higher cost*		Assembly line	
Continuous Flow				Continuous

Besides mix and volume, customer orders may have due dates, i.e. the time at which they need to be fulfilled. Customer orders with early due dates, i.e. rush orders, can be very disruptive to a manufacturing process. Allowing rush orders requires a flexible manufacturing system and may requires continuous (re-)allocation of work.

There is a lot of uncertainty with respect to customer orders. It is unknown when they arrive, and this is especially important for rush orders. When making more tactical decisions, the exact customer order profile is typically unknown, and one needs to predict the workload. Such predictions can be based on historical customer order data.

To allocate customer orders to manufacturing system resources (see Section 2.1), one needs to translate the customer orders into (sequences of) operations that can be performed by manufacturing machines. As handling many small customer orders may involve a large overhead, customer orders can be combined. Similar customer orders can be combined into a larger order using *batching*. Another approach is *nesting* or *ganging*, which involves the fulfilment of multiple orders using the same piece of (raw) material, i.e. two A4 images being printed on one A3 paper sheet paper [23] or multiple parts being cut from the same sheet of metal.

2.3 Analysis question element

The manufacturing system and workload elements (see Sections 2.1 and 2.2) describe a manufacturing system and its workload. These are the most important inputs for an analysis. However, there can be very many ways in which a workload can be allocated to the resources of a manufacturing system. The analysis question element addresses this variety. We consider three aspects: allocation constraints, KPIs/metrics, and analysis constraints. These are described in Sections 2.3.1, 2.3.2, and 2.3.3, respectively.

2.3.1 Allocation constraints aspect

The resources of a manufacturing system have certain capabilities, which restricts the way in which they can be used. For instance, an operation should only be allocated to machine that can perform this operation. As a manufacturing system analyst, you may want to restrict the allowed allocations beyond the capabilities of the manufacturing system's resources. Such constraints are specified in the allocation constraints aspect.

The number of constraints depends on the type of analysis. In case one wants to analyse one concrete scenario, the allocation must be given completely including the order of the allocated work of every resource. By leaving allocation freedom, the analysis involves an optimisation/exploration. In that case, the allocation constraints aspect describes an optimisation/configuration space. The available allocation freedom may involve allowing multiple resources to be used for an operation, either manufacturing or transport, or the resources are specified, but the order in which the operations are to be allocated is open.

2.3.2 KPIs/Metrics aspect

An analysis is meant to assess the quality of a concrete allocation or to finds an allocation with a certain quality. The quality of an allocation can be defined in terms of KPIs and metrics. These quantify performance aspects of an allocation. Common manufacturing performance criteria include latency, throughput, flow times, costs, and OEE. However, an analyst may also define proprietary KPIs that need to be optimised.

Per KPI, one can either specify that it should be optimised, i.e. minimised or maximised, or one can specify a budget, i.e. a minimum and/or a maximum value.

Chapter 3 addresses key performance indicators in more detail.

2.3.3 Analysis constraints aspect

The analysis constraints aspect involves constraints that do not involve the allocation. It involves the constraints of the analysis itself. Here one should think of the resource usage of the analysis itself, i.e. the budgets available for performing the analysis. These resources include computational resource and time. In an online setting, there is a limited time to find an (optimal) allocation. For instance, a production printer's scheduler must decide when to release the next sheet of paper within hundreds of milliseconds. Finding a high-quality allocation in so little time involves a large challenge.

Analysis constraints also include the accuracy and reliability of the analysis results. In case of stochastic manufacturing system behaviour, the actual values of KPIs may deviate from their predicted values. By considering the behaviour's stochastics, one can calculate the expected deviation from the actual KPIs.

A final analysis constraint involves the number of solutions to be found. In case of multi-objective analysis or stochastic system behaviour, there is value in providing multiple analysis results. In case of multiple KPIs, multiple solutions allow decision makers with the possibility to trade off different objective functions; in case of stochastic behaviour, expected system performance can be traded off against performance stability/uncertainty.

2.4 Analysis element

The analysis element represents addressing an analysis question (see Section 2.3) for a given manufacturing system (see Section 2.1) and a given workload (see Section 2.2). Within the analysis element, we distinguish model types (see Section 2.4.1) and analysis techniques (see Section 2.4.2).

2.4.1 Analysis model type aspect

The analysis model type aspect involves how an analysis question regarding the allocation of a workload to a manufacturing system is captured in a model. Essential in such an allocation are the progress of the production plan and the availability of resources over time; these aspects need to be captured by the analysis model. There are many modelling formalisms that can capture this aspect.

We distinguish two main classes of model: declarative models and operational models.
- *Declarative models* describe a system in terms of mathematical formulas/constraints. These describe the boundaries of the behaviour of a manufacturing system. Some types of declarative models can be used to explore a configuration space to find a high-quality system configuration.
- *Operational models* describe a system in terms of its executable behaviour. For manufacturing systems, operational models are often specified as a kind of state transition model, e.g. state machines, Petri nets, activity diagrams. Operational models can only be used to evaluate individual scenarios, e.g. the execution of a specific allocation of work to a manufacturing system configuration.

Another distinction of models is between deterministic and stochastic models. In deterministic models, all system behaviour is known upfront. A deterministic model does not have any uncertainties: the model's behaviour is determined purely by its inputs. In stochastic models, some behaviour is uncertain. This uncertain behaviour is subject to a probability distribution, which may be known or unknown.

More details about analysis models can be found in Chapter 4, which addresses analysis methods for manufacturing system analysis.

2.4.2 Analysis method aspect

The analysis method aspect represents analysis techniques and how they are applied to an analysis model (see Section 2.4.1) to answer an analysis question. In this report, we distinguish two main categories of analysis methods.
- *Prediction methods* predict the performance of a manufacturing system for one specific scenario.
- *Optimisation methods* consider multiple scenarios: they explore a configuration space to search for a configuration with an optimum performance.

Although prediction methods evaluate only one scenario, they can be used for optimisation by combining them with an optimisation method, which explores the configuration space. The combination of simulation and optimisation is quite common; it is called simulation optimisation [9]. In this setting, an optimisation method selects promising candidate solutions, which are evaluated using simulation. The results of the simulation are used by the optimisation method to continue the exploration.

Chapter 4 addresses analysis methods in more detail.

2.5 Summary

In this chapter, we have introduced a landscape that captures the most important aspects of multi-objective analysis of manufacturing systems. Some of the landscape's elements are discussed in more detail in subsequent chapters. Chapter 3 addresses KPIs commonly used in production systems. Chapter 4 discusses techniques for multi-objective analysis and optimisation of manufacturing systems including the types of models used. In Chapter 5, we discuss which parts of the landscape has been covered by TNO-ESI's (past and current) projects.

3 Key Performance Indicators

The focus of this report is on the effective usage of a manufacturing system's resources. This effectiveness typically involves a combination of multiple aspects, which may be different for each manufacturing system. In this chapter, we give an overview of KPIs (Key Performance Indicators) for manufacturing systems. We will distinguish two types of KPIs: simple KPIs and composite KPIs. Simple KPIs involve performance measures of a single aspect, where composite KPIs are performance measurements for combinations of multiple aspects. Simple KPIs are discussed in Section 3.1, composite KPIs in Section 3.2. Section 3.3 discusses standards which define KPIs in the manufacturing domain; these involve both simple and composite KPIs. Section 3.4 summarises the findings regarding KPIs.

3.1 Simple KPIs

Simple KPIs are measurements of individual performance aspects. There are many such aspects. In their well-known book Factory Physics, Hopp and Spearman [24] consider several KPIs to evaluate the performance of manufacturing systems. For activity-based costing (ABC), they distinguish several costs: labour costs, material costs, overhead costs.

For factory dynamics, Hopp and Spearman [24] consider several KPIs. Throughput (rate) is the average output (of sufficient quality) of a production process. Capacity is the upper bound of a system's throughput. Work in process (WIP) is the inventory of a system between the start and end of a production process. The cycle time of a product is the time it spends between the start and end of a production process, i.e. the time is part of the WIP. Lead time equals the amount of time reserved for production. Fill rate is defined as the fraction of orders that is fulfilled from stock. The utilisation of a piece of equipment is the fraction of the time that it is not idle.

Musselman [1] discusses a production system which wants to optimise the number of late orders, average order lateness, machine utilisation, total setup time, and work in process (WIP). Pitombeira Neto and Vila Gonçalves Filho [25] present a cellular manufacturing system design to optimise three KPIs: work-in-progress (WIP), the number of intercell moves and the capital investment (CAPEX).

3.2 Composite KPIs

Section 3.1 gave an overview of common simple KPIs. To assess the performance of a manufacturing system, a single KPIs is typically not sufficient. To address multiple aspects , one can consider multiple simple KPIs simultaneously. Alternatively, one can consider KPIs that combine multiple aspects in one formula. The KPIs discussed in this section combine multiple performance aspects to provide an aggregated performance evaluation.

Hopp and Spearman [24] consider multiple composite KPIs. For factory dynamics, they define cycle time as the ratio between a system's throughput and its average inventory.

They define service level as the probability that the cycle time does not exceed the lead time. Herps et al. [26] define the revenue of a manufacturing system as the difference of the revenue of finished products and the cost of buffered products.

Sections 3.2.1 and 3.2.2 discuss variants of common composite KPIs: Overall Equipment Effectiveness (OEE) and Overall Factory Effectiveness (OFE).

3.2.1 Overall Equipment Effectiveness

A well-known composite KPI is *Overall Equipment Effectiveness* (OEE) [12]. OEE is defined as a relative measure of the performance of a manufacturing system. It relates the actual performance of a system to its (theoretical) maximum performance. OEE distinguishes three types of performance loss:

- *Speed loss*: Performance loss due to idling and minor stops and running at a reduced speed.
- *Availability loss*: Performance loss due to (unplanned) downtime and setup/adjustment time.
- *Quality loss*: Performance loss due to production of production of defective products and rework of products.

The OEE of a system is computed as the product of the availability rate AR, the performance efficiency PE, and the quality rate QR: $OEE = OR \times PE \times QR$ [12]. These ratios can be computed as follows:

- $AR = \frac{OT-DT-ST}{OT}$, where OT represents the system's planned operation time, DT the system's delay time, and ST the system's setup time.
- $PE = \frac{PA \times CT}{OT}$, where OT represents the system's planned operation time, PA the produced number of produced products, and CT the ideal cycle time per product.
- $QR = \frac{PA-DA}{PA}$, where PA represents the produced number of products and DA the number of defectives products produced.

Muchiri and Pintelon [26] discuss two adaptations of OEE: *Total Equipment Effectiveness Production* (TEEP) includes planned downtime in the planned time horizon and *Production Equipment Effectiveness* (PEE) weights for the three loss categories. They also discuss *Overall Asset Effectiveness* (OAE) and *Overall Plant Effectiveness* (OPE), which include business-related losses.

As OEE and most of its derivatives (including those discussed in Section 3.2.2) are a relative measure, one needs to define a reference to which the performance of a system can be compared. This is an additional challenge is measuring a system's performance. Roser [27] warns that OEE values can be fudged: by changing the reference performance, one can obtain a high OEE. He mentions that an OEE of 40-60 percent is normal and lower values are not uncommon either. Williamson also warns about the abuse of OEE [28]; he argues that OEE should not be used as a benchmark for equipment performance, but only to compare a piece of equipment's performance over time.

3.2.2 Overall Factory Effectiveness

OEE is originally intended for the performance of a single machine running long batches of the same product [29]. Several authors observe the limitations of OEE to measure the effectiveness of a production system and discuss *Overall Factory Effectiveness* (OFE) [30] [31]. Many factory-effectiveness KPIs have been defined; Muchiri and Pintelon [26] give an overview of effectiveness metrics for production facilities. In this section, we will discuss some of these KPIs. This section is by no means exhaustive; many OEE-like KPIs have been defined by researchers from academia. It is highly likely that most of these KPIs are not used in industrial practice.

Scott and Pisa [32] introduce *Overall Factory Effectiveness* (OFE) to address a holistic approach to semiconductor fab optimisation. Oechsner et al. [33] introduce the term *Overall Fab Effectiveness* (OFE). Both Overall Factory Effectiveness and Overall Fab Effectiveness are broader than OEE in two ways. On the one hand, both OFEs consider a full fab instead of an individual machine. On the other hand, the OFEs include additional performance aspects, e.g. a semiconductor fab's total cost of ownership and readiness for technological developments. Neither Scott and Pisa [32] nor Oechsner et al. [33] provide formulas for computation of Overall Factory/Fab Effectiveness.

Huang et al. [30] discuss *Overall Throughput Effectiveness* (OTE) to address activities performed by multiple machines. Like OEE, OTE equals the ratio of a factory's actual performance and its attainable performance. Huang et al. [30] and Aleš et al. [34] explain how a plant's OTE can be computed from its machines' OEE; they consider series and parallel composition. In a series subsystem, OTE is dominated by the least effective machine; in a parallel subsystem, the effectiveness of the parallel machine is summed. Huang et al. [30] use simulation to run experiments to optimise an assembly line's performance. Muthiah et al. [31] extend the result of Huang et al. with two additional patterns: assembly and expansion (i.e. disassembly). The results of Muthiah et al. do not suffice to compute any factory's OTE: not all (possible) topologies can be expressed with these four considered patterns.

Nachiappan and Anantharaman [35] introduce *Overall Line Effectiveness* (OLE) and Braglia et al. [36] introduce *Overall Equipment Effectiveness of a Manufacturing Line* (OEEML). Both consider the effectiveness of manufacturing systems that are organised as production lines. Both recognise that not all machines in a production line are equally important: a bottleneck machine determines a line's effectiveness. TOEE and OEEML assesses the performance of a production line by relating the ideal performance of a bottleneck machine and the number of good products produced by the last machine in the line. Braglia et al. [36] indicate that OEEML can be misleading if the machines in a production line are decoupled by large buffers making the machine independent of each other.

Lanza et al. [37] introduce Global Production Effectiveness (GPE) of globally network manufacturing systems. GPE is a combination of manufacturing effectiveness, sourcing effectiveness, transportation effectiveness, stock effectiveness and personnel effectiveness. The manufacturing effectiveness is computed from the OEE of its constituting machine using series, parallel, joining and expansion patterns. Depending on the context, the manufacturing effectiveness uses OEE, TEEP or TOEE as a basis. The sourcing effectiveness measures the quality and timeliness of supplier deliveries. The stock effectiveness measures the damages, service level of logistics and availability of storage locations. Personnel effectiveness measures personnel's availability and a productivity index. Lanza et al. [37] propose formulas to compete each of the constituting aspects of GPE.

Jauregui Becker et al. [38] consider high-mix low-volume manufacturing environments. The introduce the *Machining Equipment Effectiveness* (MEE). Like OEE, MEE is the product of an availability rate, a performance rate and a quality rate. To deal with multiple products being produced, MEE's performance rate divides a machine's work into periods in which only one product in produced. The performance loss is computed per period and aggregated into an overall performance rate. MEE's quality rate relates the cost involved in repairing defective products to the economic value. As MEE's quality rate can be negative, Jauregui Becker et al. propose to use it as a signalling function, not as an absolute measurement.

Li et al. [39] also address multi-product production systems. They propose *Multiproduct Production System Effectiveness* (MPSE). MPSE is computed by looking at the different products individually. MPSE is a combination of the effectiveness of the production of individual products. Both MEE and MPSE do not address situations in which systems are working on different products simultaneously.

Foit et al. [40] consider two related KPIs. *Overall Factory Effectiveness* (OFE) assesses the performance of a factory. It is defined as the ratio between the number of products produced and the (theoretically) maximum number of products produced. *Overall Transport Effectiveness* (OTE) KPI assesses the effectiveness of the (AGV-based) transport within a manufacturing system. OTE is calculated relative to the production capacity of a manufacturing system: it is the ratio between the number of transports performed and the number of transports needed to achieve the (theoretically) maximum number of products produced.

Van De Ginste et al. [41] propose OEE_{Flex}, flexibility-induced equipment effectiveness, which addresses equipment flexibility. This is especially relevant for HMLV manufacturing systems, which have a quick switch between different products. OEE_{Flex} is a weighted combination of three aspects: mobility, uniformity and range. Mobility and uniformity provide a measure for machine effectiveness and range for the capability of a system to adapt.

3.3 Standards

3.3.1 NEN-ISO 22400-2:2014

NEN-ISO 22400-2:2014 [42] defines KPIs for manufacturing operations management (MOM). MOM addresses workflow/recipe control to produce desired products as well as measuring and optimising the process. MOM distinguishes four types of operations:
- Production operations,
- Maintenance operations,
- Quality Operations, and
- Inventory operation.

NEN-ISO 22400-2:2014 defines 34 KPIs for MOM, which are built up hierarchically [43]. An overview of all NEN-ISO 22400-2:2014's KPIs can be found in Table 3.1. The overview contains both simple KPIs and composite KPIs. Many of the KPIs in Table 3.1 are relative KPIs: they relate an actual value to a planned or ideal value, or they relate effort spent effectively to total effort spent. Many of these KPIs in Table 3.1 refer to work units. These can be individual machines, production cells, production lines, areas or sites.

Table 3.1 NEN-ISO 22400-2:2014 KPIs [42]

KPI	Description
Worker efficiency	Ratio of the actual work time and the actual attendance time of an employee.
Allocation ratio	Ratio of the complete actual busy time of all work units for an order and the actual order execution time of this order.

KPI	Description
Throughput rate	Ratio of the produced quantity of an order and the actual execution time of this order.
Allocation efficiency	Ratio of the actual busy time of a work unit and the unit's planned busy time.
Utilisation efficiency	Ratio of a unit's actual production time and its busy time.
Overall equipment effectiveness index	Combination of a work unit's availability, effectiveness, and quality ratio.
Net equipment effectiveness index	Combination of a work unit's ratio between processing time and planned busy time, effectiveness, and quality ratio.
Availability	Ratio of a work unit's actual production time and its planned busy time.
Effectiveness	Ratio of the ideal production time of a work unit and the actual production time.
Quality ratio	Ratio of the number of good-quality products produced by a work unit and the total number of products produced by it.
Setup ratio	Ratio of the actual setup time of a machine and its processing time.
Technical efficiency	Ratio of the actual production time of a work unit and the sum of the work unit's actual production time and the actual delay time.
Production process ratio	Ratio between actual production time of all work units and the throughput time of a production order.
Actual to planned scrap ratio	Ratio of the actual scrap quantity and the planned scrap quantity.
First pass yield	Percentage of produced products that do not require any rework.
Scrap ratio	Ratio of the scrap quantity and the total production quantity.
Rework ratio	Ratio of the rework quantity and the total production quantity.
Fall off ratio	Percentage of products that start but do not finish a sequence of production steps.
Machine capability index	Ratio of the dispersion of the specification limits of a machine characteristic and 6 times the standard deviation of a series of measurements of this characteristic.
Critical machine capability index	Ratio of the dispersion of the specification limits of a machine characteristic with respect to its average and 3 times the standard deviation of a series of measurements of this characteristic.
Process capability index	Ratio between the dispersion of the specification limits of a process characteristic and 6 times the standard deviation of a series of measurements of this characteristic.
Critical process capability index	Ratio of the dispersion of the specification limits of a process characteristic with respect to its average of averages and 3 times the standard deviation of a series of measurements of this characteristic.
Comprehensive energy consumption	Ratio of all energy consumed in a production cycle and the production quantity.
Inventory turns	Ratio between throughput and the average inventory level.
Finished good ratio	Ratio of good quality produced and consumed material quantity.
Integrated goods ratio	Ratio of integrated good quantity and consumed material quantity.

KPI	Description
Production loss ratio	Ratio of material quantity lost during production and material quantity consumed.
Storage and transportation loss ratio	Ratio of material quantity lost during transport and storage and material quantity consumed.
Other loss ratio	Ratio of material quantity lost not during production, transport and storage and material quantity consumed.
Equipment load ratio	Ratio of produced quantity and equipment production capacity.
Mean operating time between failure	Average of time between failure measurements of a work unit.
Mean time to failure	Average of time until failure measurements of a work unit.
Mean time to repair	Average of time between failure repair measurements of a work unit.
Corrective maintenance ratio	Ratio between the corrective maintenance time and the total maintenance time.

The overview in Table 3.1 includes two overall manufacturing performance criteria: the Overall Equipment Effectiveness (OEE) index and the Net Equipment Effectiveness (NEE) index. Both overall performance criteria are defined as the product of three ratios: an availability ratio, an effectiveness ratio and a quality ratio.

- The *effectiveness ratio* equals $\frac{PRI \times PQ}{APT}$, where PRI equals the planned run per item, PQ the number of items produced, and APT equals the actual production time.
- The *quality ratio* equals $\frac{GQ}{PQ}$, where GQ equals the number of good-quality items produced and PQ the total number of items produced.
- The difference between OEE and NEE concerns the *availability ratio*. The availability ratios equal $\frac{APT}{PBT}$ for OEE and $\frac{AUPT}{PBT}$ for NEE, where PBT is the planned busy time, APT the actual processing time and $AUPT$ the actual unit processing time. The difference between APT and $AUPT$ as follows: $AUPT$ includes setup times whereas APT does not.

3.4 Summary

In this chapter, we have given an overview of KPIs that are commonly used to evaluate the performance of a production system. Section 3.1 gave an overview of common simple KPIs. To assess the performance of a manufacturing system, a single KPIs is typically not sufficient. To address multiple aspects, one can either use multiple simple KPIs or a single composite KPI. Such KPIs are addressed in Sections 3.2. From a performance analysis point of view, using a single composite KPI is simpler than using multiple basic KPIs. Unfortunately, the common, OEE-based, composite KPIs are not usable under all circumstances. On the one hand, the OEE-based KPIs are using average information, which works well for LMHV systems, but poorly for HMLV systems. On the other hand, some of the OEE-based KPIs are not meant for performance analysis, but only for identifying performance issues.

4 Analysis methods

In this chapter, we present methods to analyse the performance of manufacturing systems. These correspond to the analysis element of the landscape introduced in Chapter 2 and both its aspects. In Chapter 2, we distinguished two categories of analysis methods: declarative models and operation models. In Chapter 2, we also distinguished deterministic and stochastic models. This classification is shown in Figure 4.1, which shows a grid with four quadrants for all combinations of the model types. The figure also places the formalisms discussed in this chapter in the grid. Note that analysis techniques may be in multiple quadrants. The classification in Figure 4.1 shows which techniques have been applied to which types of techniques. It is not meant to select an analysis technique, but it may help in making such a decision.

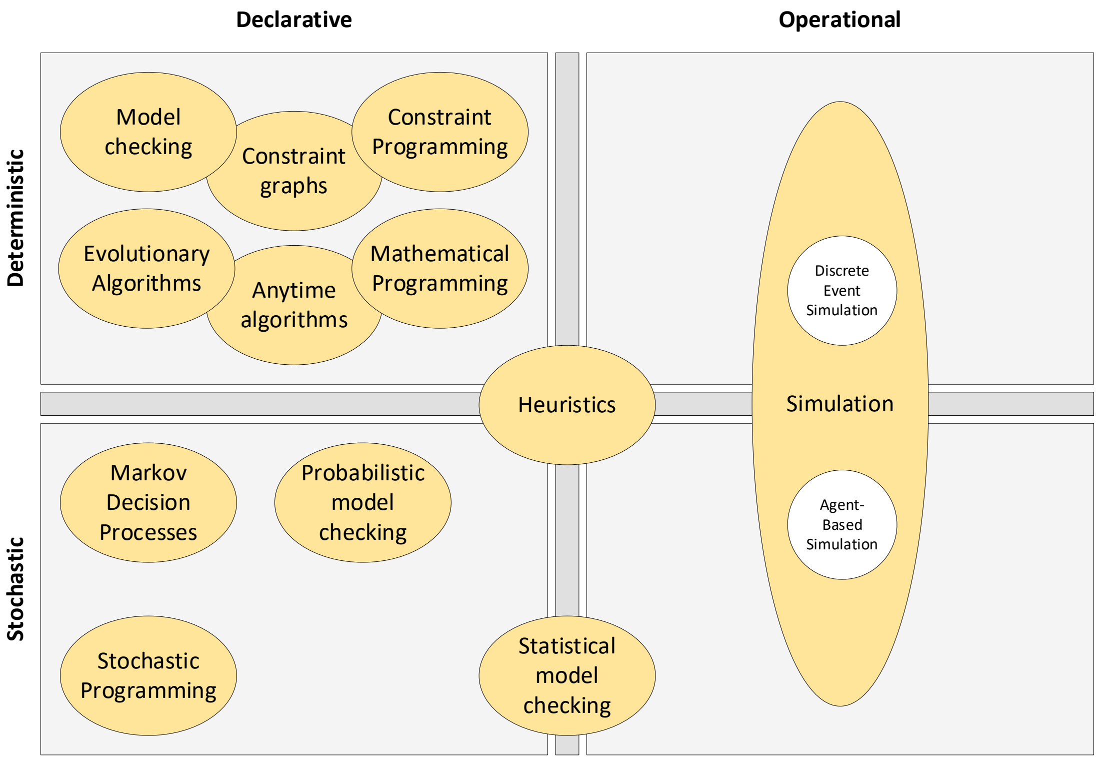


Figure 4.1 Classification of analysis techniques

This chapter is organised along one of the axes in Figure 4.1. Section 4.1 describes declarative analysis methods that can be/have been applied in the manufacturing domain. Section 4.2 gives an overview of operational analysis methods, which as one can see in Figure 4.1 only involves simulation. Section 4.3 is concerned with multi-objective analysis; it describes methods of trading off multiple performance criteria. Section 4.4 summarises the chapters and reflects on the findings.

4.1 Declarative analysis methods

This section considers declarative analysis methods that have been or could be applied in the manufacturing domain. As explained in Chapter 2, declarative methods use systems that are specified in terms of mathematical formulas/constraints.

4.1.1 Mathematical programming

Mathematical programming, or mathematical optimisation, involves a collection of techniques in which an optimisation space is described in terms of mathematical formulas. The goal of mathematical programming is finding the optimum, i.e. the point in this optimisation spaces which minimises or maximises a specific optimisation criterion, the objective function [44].

In its generic form, the constraints and objective functions of mathematical programming instance involve arbitrary mathematical functions. However, there are many variants that are more restricted. Common is the restriction to linear functions; this variant of mathematical programming is called *Integer Linear Programming* (ILP) if all variables are integer and *Mixed Integer Linear Programming* (MILP) if some variables are integer. MILP is the most used mathematical programming technique for scheduling problems [45].

The formulas used in mathematical program need not be deterministic; mathematical programming with stochastic formulas is called *Stochastic Programming* [46]. Stochastic programming aims at optimizing the expected value of a stochastic objective function. Sahinidis [47] compares several stochastic programming variants: stochastic linear programming, stochastic integer programming, stochastic non-linear programming and robust stochastic programming. The latter addresses a decision maker's risk tolerance.

Nearly all mathematical programming variants are NP-hard. For instance, ILP with binary variables is already NP-hard [48]. Hence heuristics are used to find an optimum. These heuristics typically involve evolutionary algorithms (see Section 4.3).

There are several tools that solve ILP problems. Well-known commercial ILP solvers are IBM ILOG CPLEX Optimization Studio [49] and Gurobi Optimizer [50] and AIMMS [51]. Free ILP solvers include lpsolve [52] and GLPK (GNU Linear Programming Kit) [53]. AMPL (A Mathematical Programming Language) [54] provides a unified interface to several commercial and open-source mathematical programming tools.

Fattahi et al. [55] present multiple MILP models for flexible job shop scheduling. They can solve small instances. To allow solving instances of a realistic size, they combine MILP with metaheuristics (see Section 4.1.10). Their results show that hierarchical approaches which consider allocation, i.e. the assignment of jobs to machines, and sequencing, i.e. the ordering of allocated jobs, separately provides better solutions that integrated approaches that combine allocation and sequencing.

Birgin et al. [56] propose an MILP solution for a generalisation of the flexible job shop, which involves arbitrary precedence constraints. They perform computational experiments using IBM ILOG CPLEX Optimization Studio for jobs involving assembly only, disassembly only and disassembly followed by assembly. These patterns were inspired by common job in the printing industry.

4.1.2 Constraint programming

Constraint Programming (CP) is another technique commonly used for solving scheduling and other combinatorial optimisation problems. Like mathematical programming, CP involves specifying the optimisation space in terms of constraints. However, the constraints used by CP are more general than those for mathematical programming and they are not limited to numerical constraints. This makes specification of constraint programs simpler than specifying MILP programs.

A well-known commercial constraint optimiser is IBM ILOG CP Optimizer [57]. Hexaly [58] is a commercial constraint optimiser specialised in scheduling, routing and packing. A well-known free constraint optimiser is OR-Tools CP-SAT Solver [59]. MiniZinc [60] provides a unified interface to multiple constraint optimisers, including several free ones. Laborie et al. [61] give an overview of scheduling problems that have been addressed using IBM ILOG CP Optimizer. This including several shop scheduling applications. They also compare multiple constraint solvers, and this comparison shows that (their) IBM ILOG CP Optimizer outperforms other solvers.

Lunardi et al. [62] and Naderi et al. [45] compare ILP (see Section 4.1.1) and CP for scheduling of production systems. Both have used IBM ILOG CPLEX [49] for ILP and IBM ILOG CP Optimizer [57] for CP. Both conclude that constraint programming is more scalable than ILP: using CP, larger problem instances can be optimised. Hexaly has been compared to other constraint solvers; it performs very well on selected job shop scheduling problems [63] [64].

4.1.3 Constraint graphs

Constraint graphs are a special kind of constraint program (see Section 4.1.2), which can be analysed very efficiently. Constraint graphs are directed graphs of which the nodes represent events, and the arcs represent minimum delays between the events [65]. These events can be the start and end of an operation [66]. The delays between events can be both positive and negative. Positive delays represent release dates, i.e. an event must occur some time after another event, where negative delays represent due date, i.e. an event must occur at most some time before another event.

A constraint graph with positive cycles corresponds to infeasible timing constraints. If a constraint graph does not have positive cycles is feasible, an earliest schedule can be computed using Bellman-Ford’s shortest path algorithm [67].

4.1.4 Model checking

Prediction methods (see Section 4.1) provide an estimate for the performance of a system. Such predictions are not fully accurate: the actual performance is probably close to the actual performance, but not identical to it. Model checking is a method to guarantee that certain properties are satisfied. Model checking involves exhaustively exploring a system’s state space for state that violate a desired property or satisfy an undesired property. As model checking involves an exhaustive search, it suffers from the same scalability problems as mathematical programming (see Section 4.1.1) and constraint programming (see Section 4.1.2).

Castillo and Smith [68] give an overview of formal modelling methodologies for manufacturing systems using cells. They observe that many formalisms used for model checking are based on some notion of states and state transitions. For manufacturing systems, they distinguish four types of desirable properties:

- Progress involves processes eventually performing an action if it can.
- Liveness involves the ability of a process to make a transition.
- Deadlock occurs when a system is not able to proceed.
- Partial deadlock involves situations in which some processes never progress.

The overview of Castillo and Smith [68] distinguish two types of formalisms: machine-based formalisms and, language-based based formalisms. The machine-based formalisms are based on notions of states and events. Castillo and Smith [68] consider automata theory [69], statecharts [70], Petri nets [71], and Discrete Event Control Networks (DECN) [72]. The language-based formalisms involve on synchronous languages, e.g. Lustre [73] and Esterel [74], process algebras, e.g. Communicating Sequential Processes (CSP) [75], and temporal logic. Castillo and Smith [68] compare the different formalisms with respect to level of abstraction, expressiveness, verification power and their applicability in a system's lifecycle.

In the manufacturing domain, model checking has often been done using Petri nets. Moore and Gupta [76] provide an overview of applications of Petri nets in the manufacturing domain. They consider both qualitative and quantitative analysis. Qualitative analysis involves non-numeric properties like reachability, liveness and freedom of deadlocks. An example is the work of Viswanadham et al. [77], who use Petri net to design a deadlock avoiding controller. Using quantitative analysis, Petri nets can also be used to optimise numeric properties like latency and throughput. An example is the work of Lei et al. [78] who use Petri nets to combine a deadlock-free controller and a makespan-minimising heuristic for flexible manufacturing systems.

There are many model checking tools; most are academic tools, but some have commercial support. Uppaal [79] [80] is a model checker with both academic and commercial licensing based on timed automata [81]. mCRL2 [82] is based on the process algebra ACP, NuSMV [83] on binary decision diagrams (BDDs) and Spin on Promela [84]. Statemate [85] is a model checker based on statecharts [70], which has become part of IBM Engineering Systems Design Rhapsody [86].

Probabilistic model checking calculates the likelihood of events during system execution [87]. PRISM is a stochastic model checker [88], which can analyse several types of probabilistic models including Markov decision diagrams (see Section 4.1.5) and probabilistic timed automata. Uppaal's statistical model checking extension uses techniques from the statistical domain for model checking of timed automata of stochastic systems [89].

4.1.5 Markov Decision Processes

Markov Decision Processes (MDPs) are models for decision making when outcomes are uncertain [90]. An MDP involves states, actions, probabilistic transitions and rewards: selecting an action in a state give a reward and determines the next state based on a probabilistic transition function. The goal of using MDPs is finding an optimal policy, i.e. a policy that selects the (expected) best possible action for a state. In case of finite MDPs, an optimum policy can be found in polynomial time, e.g. via dynamic programming or linear programming [91]. This is possible because MDPs are memoryless, which means that the transition probabilities of a state are independent of how that state was reached. This is also called the Markov property.

In the context of manufacturing, MDPs can be used for maintenance and repair planning, production control and control of queues. Kallenberg [90] illustrates how MDPs can be used to decide when to replace components, how many products to produce and how to optimally control queues. He also provides pointers on how MDPs can be used for pre-emptive scheduling of (independent) jobs with stochastic processing times.

4.1.6 Dataflow Process Networks

Dataflow Process Networks (DPNs) involve a general model of computation for distributed (computing) systems [92]. DPNs consist of parallel processes that communicate via FIFO queues. A process can fire if there are sufficient tokens in its input queues; after firing, it produces token in its output queues. There are several variants of dataflow. In synchronous dataflow (SDF) all firings of a process consume and produce the same number of tokens. This property allows the static computation of a finite schedule that is executed repeatedly for a network that processes an infinite stream of tokens [93]. More expressive variants like cyclo-static dataflow (CSDF) [94] and scenario-aware dataflow (SADF) also allow design-time analysis [95].

Design-time analysis of dataflow networks has been applied to schedule digital signal processing workloads on a processing platform. It is unknown whether it has been/can be applied to allocation and scheduling of manufacturing operations onto a manufacturing system's resources.

4.1.7 Petri nets

Petri nets are a formalism to describe distributed systems [71]. Petri nets are directed bipartite graphs with two types of nodes: places and transitions. Places contain tokens; transitions consume tokens from their input places and produce tokens in their output places. There are many types of Petri nets. Workflow nets are used to model the workflow of processes; these have a unique source place, i.e. a place without incoming transitions, and a unique sink place, i.e. a place without outgoing transitions [96].

Timed Petri nets have been used for scheduling [97]. In timed Petri nets, tokens have timestamps at which they become available, and transitions have fixed time durations, which are called firing delays. A transition's firing is instantaneous, but the tokens they produce only become available after the transition's firing delay. It is assumed that transitions are eager, i.e. they will fire as soon as possible. When several transitions are enabled at a moment in time, one of them will fire (possibly disabling the other transitions). SNAKES is a Python-based simulator for timed Petri nets [98].

Van der Aalst [97] explains how scheduling problems can be described by Petri nets. In particular, he describes how to capture the notions of resource allocation and precedence constraints. Reachability graphs can be used compute lower and upper bounds for the optimum makespan, i.e. the schedule length. As reachability graphs contain all possible transition sequence, they can become very large. Van der Aalst [97] proposes ways to reduce their size.

4.1.8 Heuristics

Heuristics are pragmatic algorithms to find, typically suboptimal, solutions to optimisation problems. Heuristics are commonly used for scheduling, i.e. the assignment of resources to tasks. This section discusses two well-known scheduling heuristics, list scheduling and the shifting bottleneck heuristic. Both can also be used for the allocation of work to the machines of a manufacturing systems. These and other scheduling heuristics are described by Ruiz [99], who give an overview of scheduling heuristics.

A well-known greedy heuristic is list scheduling [100]. List scheduling involving creating a list of the operations to be scheduled. The first element for all its predecessors has been allocated is allocated at the earliest time possible considering the constraints regarding precedence relations, release and due dates, resource availability, etc. This step is repeated until all elements in the list have been allocated. Default list scheduling allocates operations at their earliest possible start time. Alternatively, one can also allocate an operation such that its finishes as early as possible [101]. The main challenge lies in the creation of the priority list, which greatly determines the quality of a schedule. Panwalkar and Iskander [102] discuss and classify more than 100 dispatching rules which can be used to prioritise operations.

The shifting bottleneck heuristic is a heuristic used for the minimisation of the makespan of (shop) scheduling problems in which operations must compete for resources, typically machines [103]. The heuristic involves a disjunctive graph [104], which has directed arcs representing precedence relationships and disjunctive edges for operations that require the same resources. The shifting bottleneck heuristic performs multiple steps to determine the order of the operations connected by disjunctive edges. It first determines which resource is the bottleneck and fixes the sequence of the operations of that machine. The latter is done using a one-machine scheduling algorithm which minimise the operations' maximum lateness, i.e. the maximum exceedance of operations' due date. The operations' due date is computed from the starting times of their successor operations; these should be delayed as little as possible.

4.1.9 Anytime algorithms

Some of the analysis methods described in this chapter require a lot of time and/or computational resources to find a feasible system configuration. These resources are not always available. Sometimes, a limited amount of time is available to find a solution: Musselman [1] presents a manufacturing system in which shift schedules need to be produced within minutes. In addition, when a rush order comes in, one quickly needs to reschedule the work.

Anytime algorithms are algorithms that take into the available time; they can be interrupted at any point to return a result whose quality is a function of computation time [105]. Baruwa et al. [106] present an anytime algorithm for scheduling of flexible manufacturing systems. Their algorithm quickly finds a feasible schedule and improves this schedule over time. Efstathiou [107] presents anytime algorithm to repair manufacturing schedules after the occurrence of a dynamic event. This algorithm aims at repairing the earliest faults first and minimising the repair disruption on the existing schedule.

4.1.10 Metaheuristics

The optimisation methods considered in Sections 4.1.1 and 4.1.2 use specification of a configuration space and aim the find the optimum configuration in this space. This is called global optimisation [9]. Global optimisation has a drawback that it does not scale well. Local optimisation tries to find configurations in a neighbourhood of a candidate configuration.

There are several local optimisation approaches. Genetic algorithms are a well-known local optimisation approach, which is inspired by natural selection. From an existing population of so-called chromosomes, representing a configuration in a configuration space, a new population is generated based using genetics-inspired operations crossover, mutation, and inversion. Crossover exchanges subparts of two chromosomes; mutation randomly changes the values of some locations in the chromosome; and inversion reverses the order of a contiguous section of the chromosome [108]. A selection mechanism selects promising chromosomes for further reproduction. This is continued until a specified number of steps have been made, a certain minimum criterion is met, or no improvements are made. NSGA-II (Non-dominating Sorting Genetic Algorithm) [109] is an efficient genetic algorithm implementation.

Simulated annealing [110] also uses mutation to search for an optimal configuration; instead of a population of configurations, it considers one configuration, which gets mutated. If mutation leads to an improved configuration, then simulated annealing will accept the new configuration. However, to avoid getting stuck in local optima, simulated annealing may also allow deteriorations of configurations. If a mutation leads to worse configuration, it is accepted with a certain probability. This probability decreases with the number of steps: worse configurations are more likely to be accepted early in the process.

Tabu search [111] is like simulated annealing. A difference is its mechanism to avoid running in cycles: tabu search maintains a so-called tabu list, a list of configurations, which it has recently found. The algorithm does not allow configurations on the tabu list to be considered shortly after they have been found.

4.2 Operational analysis methods

This section gives an overview of operational analysis methods that have been used in the manufacturing domain.

4.2.1 Simulation

Simulation is a very flexible analysis method, which can analyse a single system scenario. There are different types of simulation: discrete-event simulation (DES), agent-based simulation (ABS) and system dynamics simulation (SDS). For the planning and scheduling perspective, DES and ABS are the most interesting types of simulation; SDS is more interesting for simulation of physical processes. DES can be used to capture a CPPS's production flow, whereas ABS can be used to capture distributed intelligence [112].

The models underlying a discrete event simulation are typically based on a directed graphs in which the nodes represent work elements that needs to be performed and the arcs represent the precedence constraints between the work elements. There are many formalisms that has such a structure: statecharts [70], activity diagrams, Petri nets (see also Section 4.1.7) and event graphs.

As observed in Chapter 2, a manufacturing system include both controllable and uncontrollable behaviour. In case of modelling stochastic behaviour, such as a machine's unplanned downtime, one needs to perform multiple experiments to get a good indication for the values of the KPIs of interest. The classical central limit theorem provides an indicator for the accuracy of the series of independent (simulation) samples: for a series $(y_1, \dots, y_n)$ of n samples, the series' standard deviation can be approximated by $\frac{s}{\sqrt{n}}$, where $s^2 = \sum_{i=1}^{n} \frac{(y_i - \bar{y})^2}{n-1}$

is the series' sample variation. The central limit theorem also provides an indication for the number of samples needed. Given a desired error ϵ and a confidence level of $1-\alpha$, the number of simulations needed equals $\left(\frac{z_{\alpha/2}\times\sigma}{\epsilon}\right)^2$, where $z_{\alpha/2}$ is the value of the standard normal variable with a cumulative probability level $(1 - \frac{\alpha}{2})$ and σ is the standard deviation [113].

Negahban et al. [114] present a literature overview of manufacturing system simulation. Their overview addresses two main types of simulations: manufacturing system design and manufacturing system operation. Manufacturing system design addresses the placement of machinery and a material handling system inside a facility. They observe that material handling system design is especially difficult for system with a high degree of flexibility. High flexibility is seen in cellular manufacturing systems used for semiconductor manufacturing. Cellular manufacturing system involve a material handling system with many re-entrant loops. Pitombeira Neto and Vila Gonçalves Filho [25] present a simulation to evaluate a cellular manufacturing system design. This simulation considers three KPIs: level of WIP, the number of intercell moves and the capital investment. An evolutionary algorithm is used to explore the configuration space and the corresponding trade-offs of these KPIs.

System design involves a long-term decision which involves uncertainty. When a system is being designed, its actual workload is not known yet. Jithavech and Kumar Krishnan [115] address this uncertainty by simulating a manufacturing system for many workloads and taking the average performance. Koo and Jang [116] use simulation to evaluate AGV dispatching rules for AGV-based material handling systems with stochastic travel times.

Glatt et al. [117] present a simulation concept that combines material flow simulation and physical phenomena. This combination allows them to predict disturbances in the material flow. This could be very valuable in describing a manufacturing system's behaviour in more detail. Unfortunately, their concept has not been implemented and assessed.

Roda and Macchi [29] observe that the aggregate performance number provided by OEE, and its variants can only be measured from an actual plant. They introduce a stochastic simulation approach to predict Overall Factory Effectiveness (OFE). Their simulation includes the availability behaviour of a factory's machines as well as their suboptimal modes.

Baines et al. [17] observe that the accuracy of simulations of manufacturing systems with a large degree of manual labour is quite poor. They argue that this is because human operators are modelled as machine resources, which are typically very predictable. In reality, human behaviour fluctuates more than machine behaviour. Baines et al. [17] propose simple mechanisms to account for an operation's age and his/her circadian rhythm. Especially operator age has a large influence on operational performance: in their model and that of Zülch and Becker [118], the performance of a 65-year-old operator is 35 percent lower than a 20-year-old operator. The circadian rhythm accounts for a performance difference of at most a few percent. Katiraee et al. [119] consider the influence of four human factors on the timing and cost of manufacturing systems: skill level, age, gender and anthropometric measures. They argue that more research effort should be spent on making accurate human aspect models. Baines et al. [17] note that validating a simulation model becomes harder when the human factor is dominant, on top of that comes the ethical aspect of modelling humans.

Paape et al. [112] compare the functionality of multiple simulation tools for CPPSs with distributed intelligence. For such manufacturing systems, Anylogic [120] is the most promising simulation tool and they present a case study inspired by a poultry fillet processing line. Herps et al. [26] use Anylogic for the simulation of a high-mix low-volume manufacturing system. Mourtzis et al. [121] present a survey of the usage of simulation using during the design and operation of manufacturing systems. The former includes manufacturing system layout design and material flow design. The operation includes material flow simulation. They evaluate five well-known commercial simulators with respect their capabilities and usability: Anylogic [120], Arena [122], FlexSim [123], Siemens Tecnomatix Plant Simulation [124] and Witness [125]. In this evaluation, Anylogic, FlexSim and Siemens Tecnomatix Plant Simulation score well. Other commercial simulation packages include Simio [126] and Simul8 [127]. Jaamsim [128] is an open-source simulation package. SimPy [129] and salabim [130] are Python-based discrete event simulators. NetLogo [131] is an agent-based modelling environment which can be used to create simulations.

Some simulators include simulation optimisation functionality [9]. For instance, FlexSim [123] has integrated OptTek Systems' OptQuest simulation optimisation software [132] and Anylogic [120] provides simulation optimisation via its experiments.

4.3 Trade-off methods

The optimisation methods reviewed in Section 4.1 aim at optimising a single objective function. However, Chapter 3 shows that there are many relevant criteria when optimising a manufacturing system. This section describes methods to trade off different optimisation criteria.

A straightforward way of trading off is by combining all criteria in a single criterion, e.g. using weights, which is called scalarisation [133]. This is, for instance, done in OEE and many of its variants discussed in Chapter 3. This combination changes a multi-objective optimisation problem into a single-objective optimisation problem. A drawback of this approach is that it is not simple to define a combination that captures an optimiser's preferences, which may be different per optimiser or even per configuration space.

Another straightforward of trading off multiple objective functions is to set bounds on all (but one) objective. These bounds can then be used as constraints of a mathematical program, a constraint program or an evolutionary algorithm. These could then solve a constrained single-objective optimisation problem.

If one wants to get more insight into the trade-off between multiple objectives in a configuration space, another method is needed. Pareto analysis allows identification of a Pareto front, which contains all configurations which are not dominated by other configuration. A configurator dominates another if the former is better than the latter with respect to one objective and not worse than the latter with respect to the other objectives [134]. Geilen et al. [135] extend the dominance relation to sets of configurations and present an algebra to compose such sets.

Visualisation of a configuration space with more than three objectives is challenging. For such configuration spaces, one can use radar/spider chart [136] or parallel coordinate plots [137]. These visualisations can be used to visualise and compare different configurations.

Efatmaneshnik et al. [138] observe that when the number of optimisation criteria grows it becomes difficult to keep an overview of the dominant configurations (e.g. in the Pareto front). They propose a metric to limit the number of non-dominated solutions to alleviate the task of (early) decision making, when decision makers do not have a view on the most important objectives.

4.4 Summary

In this chapter, we have given an incomplete overview of methods that can be used for multi-objective analysis of manufacturing systems. This overview shows that this field is very extensive. There are many (possibly) relevant techniques and the literature regarding each of the techniques individually is very large. An upcoming technique, which was left out of the overview is Artificial Intelligence (AI): Machine Learning (ML) techniques are expected to become very interesting for allocating and sequencing the work of a manufacturing system.

When looking at the overview, there seems to be only one operational analysis method, i.e. simulation. On the other hand, simulation itself is not one method: there are several distinct types of simulation, several of which could play a role in addressing allocation and sequencing questions in the manufacturing domain.

What is striking is the apparent limited number of techniques that can deal with multiple optimisation criteria. Pareto fronts are a means to really trade off multiple optimisation criteria, but this is challenging when the number of criteria is too high. Hence multi-objective analysis questions are transformed into a single-objective analysis question by combining criteria in one high-level criterion or introducing budget constraints.

Which method(s) to select for a certain analysis question is typically be addressed by the corresponding analyst’s experience and preferences. It is unknown whether there is literature supporting this method selection question.

It should be noted that a single method is unlikely to be sufficient to answer a (complex) multi-objective analysis problem. Often multiple methods are combined into a methodology. A concrete example is simulation optimisation, which combines simulation to analyse individual scenarios, and search heuristics, like evolutionary algorithms, to find an optimum scenario.

5 TNO-ESI landscape coverage

This chapter presents an overview of TNO-ESI's experience regarding multi-objective analysis of manufacturing systems and related topics. A high-level summary of this experience is presented in Figure 5.1, which shows the methods used in TNO-ESI's past projects onto the classification introduced in Figure 4.1. Figure 5.2 shows the timeline of these projects. A more detailed description of the work done in TNO-ESI's projects can be found in the following sections. Section 5.14 reflects on TNO-ESI's experience.

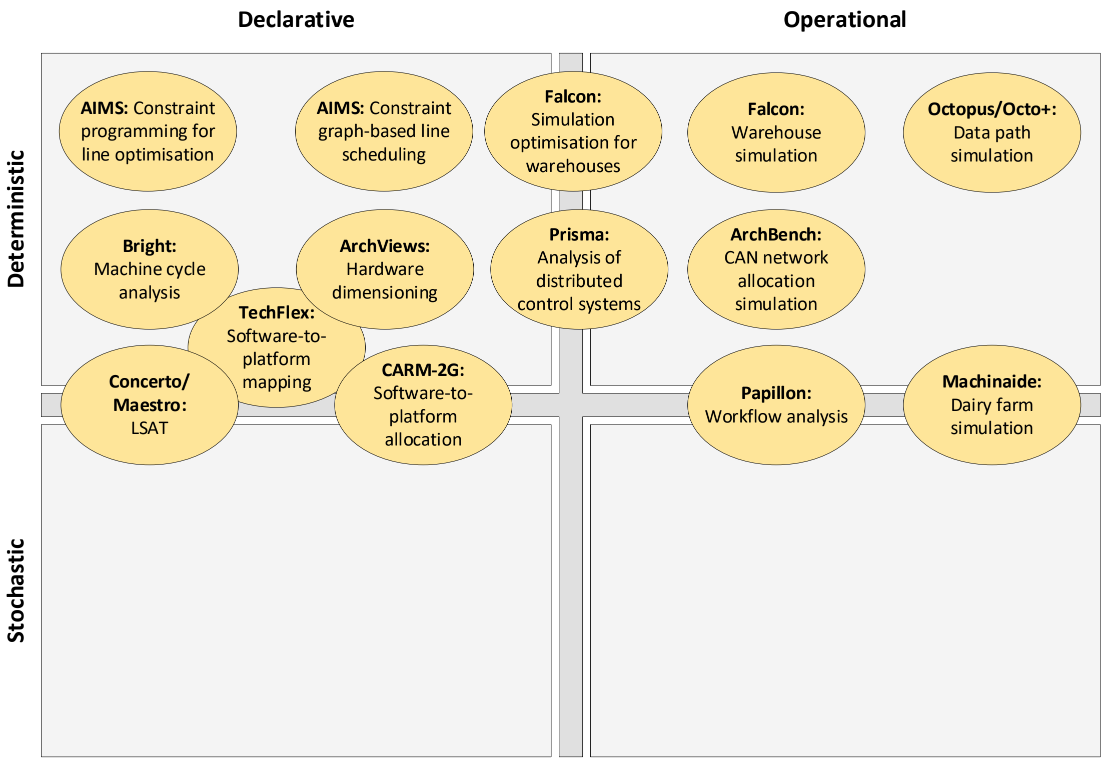


Figure 5.1 TNO-ESI projects related to multi-objective analysis of manufacturing systems

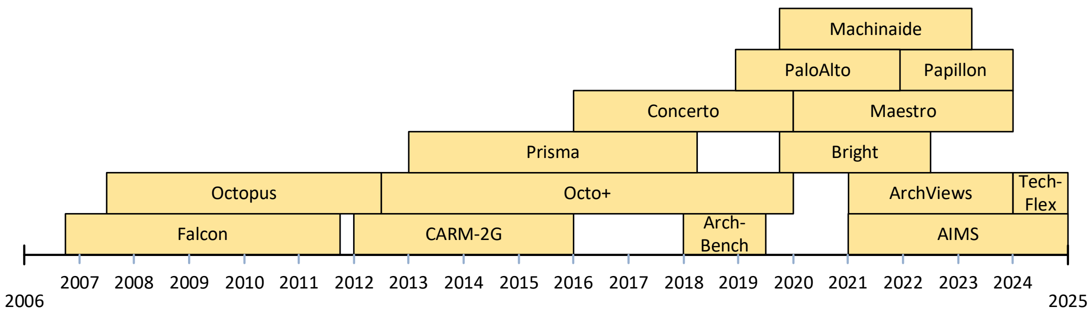


Figure 5.2 Timeline of TNO-ESI projects related to multi-objective analysis of manufacturing systems

5.1 AIMS

The AIMS project is a running collaboration of TNO-ESI and Canon Production Printing. It is concerned with the timing performance of manufacturing systems. To perform such analysis, it applies a Y-chart-based approach [139]: it separately describes a manufacturing system's orders and its equipment. From an allocation of orders to the equipment, a constraint graph is generated, which is used to compute the fastest schedule of tightly coupled production systems [140].

Constraint graphs are more expressive than LSAT [141], which had also been considered as an analysis tool. In LSAT, one can specify release dates, i.e. a minimum time between two events, but one cannot express due dates, i.e. a maximum time between two events. The latter is needed for tightly coupled manufacturing systems, i.e. systems that do not have internal buffers.

From the latest version of the specification of a manufacturing system's orders and equipment, a MiniZinc constraint program [60] is generated which can be used to find an optimum allocation and sequencing. This transformation is still to be documented.

5.2 ArchBench

The ArchBench project was a collaboration of TNO-ESI and DAF Trucks, which ran from 2017 until 2019. In this project, a tool was developed to analyse the allocation of functionality in a CAN bus network. The tool included a deterministic discrete event simulation, which captured the CAN bus network communication. Using the tool, a system architect could reason about different allocations in terms of network load, costs, and weight. Details about the tool can be found in the paper of Bijlsma et al. [142].

5.3 ArchViews

The ArchViews project was a collaboration of TNO-ESI and Thales, which ran from 2020 until 2023. The general goal of the project was to study how to guarantee that a system delivers on its specification, with system performance requirements as a carrying example. The most relevant result in the context of this project was a study on how to make a microservice architecture observable using standardised telemetry tools, such as OpenTelemetry [143], and how to use the collected logs and metrics to verify that the system conforms to its specification. This involved a model-based approach to specify system flows, chains of executing services and their interactions along with timing requirements, as sequence diagrams using PlantUML [144]. Automation was provided to parse these specifications and extract relevant traces and metrics to validate that the interactions between services followed the specification and that the timing requirements were satisfied. Details can be found in Andrade's master thesis [145].

ArchViews also addressed a hardware dimensioning problem where the performance of system flows was predicted using analytical model considering computation and communication costs for alternative mappings between software processes and compute nodes. Infrastructure was developed to automatically profile the communication and computation costs of services and their interactions on the real system, using the observability infrastructure developed in the project. Details can be found in Vollaard's master thesis [146].

5.4 Bright

The Bright project was a collaboration of TNO-ESI and ITEC, which ran from 2019 until 2022. In the Bright project, LSAT [147] [141] was used to model the machine cycle of ITEC's die bonders [148]. Bright did not use LSAT's stochastic critical analysis. The Bright project also include further development of LSAT, e.g. to deal with pools of identical pieces of equipment.

5.5 CARM-2G

The CARM-2G project was a collaboration of TNO-ESI and ASML, which ran from 2012 until 2016. The project developed a methodology based on the Y-chart [139] to allocate control applications to an execution platform. ASML's applications involve thousands of precedence-constrained tasks with strict latency requirements that needed to be mapped onto a general-purpose processing platform. To deal with background tasks, the tasks do not fully use the platform's processors. This makes the allocation tolerant to worst-case task execution times [149].

5.6 Concerto/Maestro

The projects Concerto and Maestro were collaborations of TNO-ESI and ASML. Concerto ran from 2016 until 2022 and Maestro from 2019 until 2023. In these projects, the tool LSAT [147] [141] was developed. LSAT is a tool to analyse the timing performance of logistic systems. Such systems are specified in terms of a machine's equipment and its actions, timing settings of machine actions, activities consisting of machine actions, and the dispatching sequence of activities.

LSAT is mainly used for deterministic timing analysis. However, LSAT's timing settings allow the specification of probability distributions. These are used for stochastic critical path analysis [150]: using Monte Carlo simulation, it is assessed how frequently a machine action lies on a logistic system's critical path.

Within Concerto and Maestro, LSAT has been used to analyse the wafer handler of ASML's lithography systems.

5.7 Falcon

The Falcon project was a collaboration of ESI,[2] Vanderlande Industries, Demcon, Delft University of Technology, Eindhoven University of Technology, Utrecht University, and Twente University. Falcon ran from 2006 until 2011.

The Falcon project addressed the topic of flexible logistic systems with warehousing as a reference case. In the Falcon project, several warehouse simulations have been developed. Two types of agent-based simulation were studied in the project. One is a hierarchical simulation focussing on the allocation of order to workstations, the second focussed on agent organisations. In addition, Falcon included a discrete-event simulation focussing on allocation of order to workstations, their execution and the corresponding transportation. Simulation optimisation using an evolutionary algorithm was used to find the optimum warehouse configuration [151]. Constraints were introduced to deal with multiple optimisation criteria.

[2] The independent research institute Embedded Systems Institute (ESI) became the TNO department Embedded Systems Innovation (TNO-ESI) in 2013.

In addition, an effective process time (EPT) approach was applied to capture probability distributions of warehouse operations. The EPT approach was used to calibrate stochastic simulation models.

Details and further references can be found in the book by Hamberg and Verriet [152].

5.8 Machinaide

The Machinaide project was an ITEA project with TNO, Additive Industries, Cordis Suite, Eindhoven University of Technology, KE-chain and Lely Industries as the Dutch partners in the consortium. The project ran from 2019 until 2023.

In the Machinaide project, a stochastic discrete event simulation based on queueing theory was developed. This simulation, which captured the interaction of cows, robots, and people in a dairy farm, has strong similarities to a manufacturing system. Details can be found in Buermann's master thesis [153].

5.9 Octopus/Octo+

The Octopus project was a collaboration of ESI,[2] Océ Technologies, Delft University of Technology, Eindhoven University of Technology, and Twente University. The Octo+ project was a continuation involving only ESI and Océ Technologies. Octopus ran from 2007 until 2012 and Octo+ from 2012 until 2019.

In the Octopus project, a design space exploration methodology was developed based on the Y-chart [139], which separate the application to be executed and the platform involving the execution resources. In the Octopus project, this pattern has been applied to the data path of a printer, but it can also be applied to a manufacturing system's jobs and its equipment. To analyse allocations of image processing functionality to a computation platform, several analysis tools were used: CPN Tools [154], Uppaal [80] and SDF3 [155]. Details and further references regarding the Octopus project can be found in the book by Basten et al. [156].

In the Octo+ project, a deterministic discrete event simulation for data path analysis was developed [157]. This can be seen as the continuation of the design space exploration methodology of the Octopus project.

5.10 PaloAlto

The PaloAlto project was a collaboration of TNO-ESI and Thermo Fisher Scientific, which started in 2018 and ended in 2021. In the PaloAlto project, workflow models were created in the context of reference architecting. Architectural trade-off analysis was used to reason about the influence of improving the speed of a piece of one component on the overall throughput of a workflow. The creation of the workflow models and the architectural trade-off analysis were both done using the Daarius methodology [158].

5.11 Papillon

The Papillon project was a collaboration of TNO-ESI and Thermo Fisher Scientific. The project, which started in 2022 and ended in 2023, can be seen as a follow-up of the PaloAlto project.

In the Papillon project, a methodology was developed to specify and analyse customer workflows. The workflows are specified in an extension of Capella [158] using its functional chains as a starting point. From a workflow specification, a Petri net simulation is generated based on the SNAKES simulator [98]. The workflow analysis considers three optimisation criteria, makespan and two different costs, which are visualised using a parallel coordinates plot [137]. To deal with stochastic timing, the analyst can specify the number of simulations of the same workflow; the average makespan is computed from these runs. More details can be found in the conference paper of Hooman et al. [159].

5.12 Prisma

The Prisma project was a collaboration of TNO-ESI and Philips Lighting, which ran from 2013 until 2017. In Prisma, a methodology was developed to specify large-scale distributed control systems. From such a specification, models were generated to analyse the control system's behaviour. Two types of analysis were used. Deterministic model checking, using Uppaal [80], was used to detect scenarios leading to unwanted system behaviour. Deterministic discrete event simulation, using Java, was used to analyse system usage scenarios. More details can be found in the project's conference papers [160] [161].

5.13 TechFlex

The TechFlex project, a collaboration of TNO-ESI and Thales started in 2024, is the continuation of the ArchViews project (see Section 5.3). TechFlex addresses the challenge of reducing the time and cost associated with system diversity and evolution at the level of the software platform. A model-based approach to specification and automation with two steps is proposed: 1) technology-agnostic specification of software configurations to create custom software deployments with minimum manual intervention, and 2) deployment optimisation that improves the mapping to software processes to compute nodes to ensure technical performance requirements are satisfied. The optimisation problem is formulated as an MDP (see Section 4.1.5) and is solved using model-based reinforcement learning [162] combined with Monte Carlo Tree Search [163].

5.14 Summary

This chapter gives an overview of knowledge and methodologies related to multi-objective analysis of manufacturing systems in TNO-ESI's (past and running) projects. Not all projects discussed in this chapter considered manufacturing systems. Many of these projects involved the allocation of computational work to a computer platform. This context is different than the manufacturing domain, as computational tasks allow more flexibility, e.g. pre-emption, than physical tasks in a manufacturing system. Yet, the analysis techniques used for the allocation of computational tasks may be relevant for the allocation of manufacturing tasks.

This chapter's overview, especially the visualisation in Figure 5.1, shows that TNO-ESI's projects have focussed on deterministic methods to handle the allocation of operations to a manufacturing system or a computational platform. Deterministic approaches work well for low-mix high-volume (LMHV) systems, involving large batches of the same operations. High-mix low-volume (HMLV) systems, involving production of small batches of many different products, have a greater variety of behaviour. The influence of an event, e.g. an unexpected breakdown of a machine, may be high when one product is being manufactured and low after the manufacturing system has switched to another product. For such scenarios, stochastic analysis techniques may be valuable. An assessment of the added value of stochastic analysis in the context of HMLV manufacturing systems is recommended.

6 Conclusion

TNO-ESI's partners are increasingly looking into the performance of their products in the context in which they are used, typically a manufacturing system. The performance of such systems is not captured by a single criterion but involves several criteria. This report introduced a landscape for the multi-objective analysis of manufacturing systems. The landscape considers questions that can be addressed by allocation of workload to a manufacturing system's resources. It was used to conduct a survey of the existing literature and an inventory of TNO-ESI's knowledge regarding this topic. A summary of the report and the main findings can be found in Section 6.1. A recommendation on how to follow up on these findings are listed in Section 6.2.

6.1 Summary

In Chapter 2 of this report, we introduced a landscape with the main aspects of multi-objective analysis of manufacturing systems. For two of the landscape's aspects, a literature study was performed. The literature study on KPIs in Chapter 3 showed that there are many relevant (simple) KPIs in a manufacturing system. Which ones to select for an analysis question depends on the analysis question at hand. To define a manufacturing system's performance, several composite KPIs, i.e. combinations of KPIs, have been defined. The well-known composite KPIs, i.e. OEE and its variants, look at the average (multi-objective) performance of a manufacturing systems. Because they consider average behaviour, they are suited for low-mix high-volume (LMHV) systems, but they may not be suited for high-mix low-volume (HMLV) systems. Note that the manufacturing systems that include equipment developed by TNO-ESI's industrial partners are mainly HMLV systems.

Chapter 4 contains the results of the literature study into analysis techniques. To structure the study, analysis methods were classified along two axes: deterministic vs. stochastic and declarative vs. operational. On the declarative side, there are many, typical specialised, analysis methods and each of the methods comes with a large body of (academic) knowledge. On the operational side of the overview, simulation is the dominant analysis technique; simulation is a very flexible analysis technique, which can be applied in both a deterministic and a stochastic context.

Chapter 5 studied the expertise that TNO-ESI has built up during its 20+ years existence. The corresponding overview shows that TNO-ESI's knowledge is dominated by deterministic analysis methods. High-mix low-volume (HMLV) systems, involving production of small batches of many different products, have a greater variety of behaviour. For such systems, stochastic analysis techniques may be valuable.

6.2 Recommendation

To assess whether stochastic analysis techniques have an added value for the analysis of industrial HMLV manufacturing systems, we recommend starting a project to address this question. Special attention should be given to the variability of resources and the workload: the project should (1) assess the influence of stochastic behaviour in the different manufacturing systems and (2) evaluate the added value of existing stochastic analysis techniques compared to deterministic analysis techniques in TNO-ESI's knowledge. Manufacturing systems of ESI's industrial partners' customers can be used as use cases for such a project.

7 References

[1] K. J. Musselman, “Complex scheduling of a printing process,” *Computers & Industrial Engineering,* vol. 39, no. 3-4, pp. 273-291, 2001.

[2] B. L. Dietrich, “A Taxonomy of Discrete Manufacturing Systems,” *Operations Research,* vol. 39, no. 6, pp. 886-902, 1991.

[3] S. Chatti, L. Laperrière, G. Reinhart and T. Tolio, Eds., CIRP Encyclopedia of Production Engineering, Second ed., Berlin, Heidelberg: Springer, 2019.

[4] Lean Enterprise Institute, “Four Ms,” 2024. [Online]. Available: https://www.lean.org/lexicon-terms/four-ms/.

[5] Lean Community, “4M Method Meaning,” 2022. [Online]. Available: https://leancommunity.org/4m-method-meaning/.

[6] Kaufman Global, “6Ms of Production,” 2024. [Online]. Available: https://www.kaufmanglobal.com/glossary/6ms-production-man-machine-material-method-mother-nature-measurement/.

[7] M. W. Maier, “Architecting principles for systems-of-systems,” *Systems Engineering,* vol. 1, no. 4, pp. 267-284, 1998.

[8] B. Rouwenhorst, B. Reuter, V. Stockrahm, G. van Houtum, R. Mantel and W. Zijm, “Warehouse design and control: Framework and literature review,” *European Journal of Operational Research,* vol. 122, no. 3, pp. 515-533, 2000.

[9] A. Ghasemi, F. Farajzadeh, C. Heavey, J. Fowler and C. T. Papadopoulos, “Simulation optimization applied to production scheduling in the era of industry 4.0: A review and future roadmap,” *Journal of Industrial Information Integration,* vol. 39, 2024.

[10] L. Van De Ginste, J. Goos, M. Schamp, A. Claeys, S. Hoedt, K. Bauters, A. Biondi, E.-H. Aghezzaf and J. Cottyn, “Defining Flexibility of Assembly Workstations Through the Underlying Dimensions and Impacting Drivers,” *Procedia Manufacturing,* vol. 39, pp. 974-982, 2019.

[11] M. E. Kurz and R. G. Askin, “Scheduling flexible flow lines with sequence-dependent setup times,” *European Journal of Operational Research,* vol. 159, no. 1, pp. 66-82, 2004.

[12] S. Nakajima, Introduction to TPM: Total Productive Maintenance, Cambridge, MA: Productivity Press, 1988.

[13] ISO, “IEC 62264-1:2013 Enterprise-control system integration Part 1: Models and terminology,” ISO, 2013.

[14] FedEx Express, *Non-conveyable Shipments Guide,* 2024.

[15] Assembly Magazine, “The Indispensable Role of Humans in Manufacturing,” 2024. [Online]. Available: https://www.assemblymag.com/articles/98261-the-indispensable-role-of-humans-in-manufacturing.

[16] E. Hozdić and I. Makovec, “Evolution of the Human Role in Manufacturing Systems: On the Route from Digitalization and Cybernation to Cognitization,” *Applied System Innovation,* vol. 6, no. 2, 2023.

[17] T. Baines, S. Mason, P.-O. Siebers and J. Ladbrook, “Humans: the missing link in manufacturing simulation?,” *Simulation Modelling Practice and Theory,* vol. 12, no. 7-8, pp. 515-526, 2004.

[18] SRSI, “Glossary of Terms,” 2024. [Online]. Available: https://www.srsi.com/glossary-of-terms/.

[19] H. Emmons and G. Vairaktarakis, Flow Shop Scheduling: Theoretical Results, Algorithms, and Applications, New York: Springer, 2013.

[20] H. Xiong, S. Shi, D. Ren and J. Hu, “A survey of job shop scheduling problem: The types and models,” *Computers & Operations Research,* vol. 142, 2022.

[21] Matics, “High-Mix Low-Volume (HMLV) Manufacturing,” 2024. [Online]. Available: https://matics.live/glossary/high-mix-low-volume-hmlv-manufacturing/.

[22] R. H. Hayes and S. C. Wheelwright, “Link Manufacturing Process and Product Life Cycles,” *Harvard Business Review,* January 1979.

[23] N. North, “Future-proof your Workflow: Batch Management,” 2024. [Online]. Available: https://global.ricohsoftware.com/futureproof-your-workflow-batch-management/.

[24] W. J. Hopp and M. L. Spearman, Factory Physics, Third ed., Boston, MA: McGraw-Hill/Irwin, 2008.

[25] A. R. Pitombeira Neto and E. Vila Gonçalves Filho, “A simulation-based evolutionary multiobjective approach to manufacturing cell formation,” *Computers & Industrial Engineering,* vol. 59, no. 1, pp. 64-74, 2010.

[26] K. Herps, Q.-V. Dang, T. Martagan and I. Adan, “A simulation-based approach to design an automated high-mix low-volumne manufacturing system,” *Journal of Manufacturing Systems,* vol. 64, pp. 1-18, 2022.

[27] P. Muchiri and L. Pintelon, “Performance measurement using overall equipment effectiveness (OEE): literature review and practical application discussion,” *International Journal of Production Research,* vol. 46, no. 13, pp. 3517-3535, 2008.

[28] C. Roser, “Top Three Methods on how to Fudge Your OEE,” 2013. [Online]. Available: https://www.allaboutlean.com/fudge-oee/.

[29] R. M. Williamson, “Using Overall Equipment Effectiveness: the Metric and the Measures,” Strategic Work Systems, Inc., Columbus, NC, 2006.

[30] I. Roda and M. Macchi, “Factory-level performance evaluation of buffered multi-state production systems,” *Journal of Manufacturing Systems,* vol. 50, pp. 226-235, 2019.

[31] S. H. Huang, J. P. Dismukes, J. Shi, Q. Su, M. A. Razzak, R. Bodhale and D. E. Robinson, “Manufacturing productivity improvement using effectiveness metrics and simulation analysis,” *International Journal of Production Research,* vol. 41, no. 3, pp. 513-527, 2003.

[32] K. M. N. Muthiah, S. H. Huang and S. Mahedevan, “Automating factory performance diagnostics using overall throughput effectiveness (OTE) metric,” *International Journal of Advanced Manufacturing Technology,* vol. 36, pp. 811-824, 2008.

[33] D. Scott and R. Pisa, “Can overall factory effectiveness prolong Moore's Law?,” *Solid State Technology,* vol. 41, pp. 75-82, 1998.

[34] R. Oechsner, M. Pfeffer, L. Pfitzner, H. Binder, E. Müller and T. Vonderstrass, “From overall equipment efficiency (OEE) to overall Fab effectiveness (OFE),” *Materials Science in Semiconductor Processing,* vol. 5, no. 4-5, pp. 333-339, 2002.

[35] Z. Aleš, J. Pavlů, V. Legát, F. Mošna and V. Jurča, “Methodology of overall equipment effectiveness calculation in the context of Industry 4.0 environment,” *Maintenance and Reliability,* vol. 21, no. 3, pp. 411-418, 2019.

[36] R. Nachiappan and N. Anantharaman, “Evaluation of overall line effectiveness (OLE) in a continuous product line manufacturing system,” *Journal of Manufacturing Technology Management,* vol. 17, no. 7, pp. 987-1008, 2006.

[37] M. Braglia, M. Frosolini and F. Zammori, “Overall equipment effectiveness of a manufacturing line (OEEML): An integrated approach to assess systems performance,” *Journal of Manufacturing Technology Management,* vol. 20, no. 1, pp. 8-29, 2008.

[38] G. Lanza, J. Stoll, N. Stricker, S. Peters and C. Lorenz, “Measuring Global Production Effectiveness,” *Procedia CIRP,* vol. 7, pp. 31-36, 2013.

[39] J. M. Jauregui Becker, J. Borst and A. van der Veen, “Improving the overall equipment effectiveness in high-mix-low-volume manufacturing environments,” *CIRP Annals - Manufacturing Technology,* vol. 64, pp. 419-422, 2015.

[40] X. Li, G. Liu and X. Hao, “Research on Improved OEE Measurement Method Based on the Multiproduct Production System,” *Applied Sciences,* vol. 11, no. 2, 2021.

[41] K. Foit, G. Gołda and A. Kampa, “Integration and Evaluation of Intra-Logistics Processes in Flexible Production Systems Based on OEE Metrics, with the Use of Computer Modelling and Simulation of AGVs,” *Processes,* vol. 8, 2020.

[42] L. Van De Ginste, E.-H. Aghezzaf and J. Cottyn, “The role of equipment flexibility in Overall Equipment Effectiveness (OEE)-driven process improvement,” *Procedia CIRP,* vol. 107, pp. 289-294, 2022.

[43] ISO, “NEN-ISO 22400-2 Automation systems and integration - Key performance indicators (KPIs) for manufacturing operations management - Part 2: Definitions and descriptions,” ISO, 2014.

[44] N. Kang, C. Zhao, J. Li and J. A. Horst, “A hierarchical structure of key performance indicators for operation management and continuous improvement in production systems,” *International Journal of Production Research,* vol. 54, no. 21, pp. 6333-6350, 2016.

[45] J. A. Snyman and D. N. Wilke, Practical Mathematical Optimization, Second ed., Berlin, Heidelberg: Springer, 2018.

[46] B. Naderi, R. Ruiz and V. R. , “Mixed-Integer Programming vs. Constraint Programming for Shop Scheduling Problems: New Results and Outlook,” *INFORMS Journal on Computing,* vol. 35, no. 4, pp. 817-843, 2023.

[47] A. Shapiro and A. Philpott, “A tutorial on stochastic programming,” 2007.

[48] N. V. Sahinidis, “Optimization under uncertainty: state-of-the-art and opportunities,” *Computers and Chemical Engineering,* vol. 28, no. 6-7, pp. 971-983, 2004.

[49] R. M. Karp, “Reducibility among Combinatorial Problems,” in *Complexity of Computer Computations*, R. Miller, J. Thatcher and J. Bohlinger, Eds., Boston, Springer, 1972, pp. 85-103.

[50] IBM, “IBM ILOG CPLEX Optimization Studio,” 2024. [Online]. Available: https://www.ibm.com/products/ilog-cplex-optimization-studio.

[51] Gurobi Optimization, “Gurobi Optimizer,” 2024. [Online]. Available: https://www.gurobi.com/solutions/gurobi-optimizer/.

[52] AIMMS, “Optimization Software for Supply Chain & Manufacturing Planning,” 2024. [Online]. Available: https://www.aimms.com/.

[53] Sourceforge, “lpsolve Mixed Integer Linear Programming (MILP) solver,” 2021. [Online]. Available: https://sourceforge.net/projects/lpsolve/.

[54] GNU, “GLPK (GNU Linear Programming Kit),” 2024. [Online]. Available: https://www.gnu.org/software/glpk/.

[55] AMPL, “AMPL Optimization,” 2024. [Online]. Available: https://ampl.com/.

[56] P. Fattahi, M. S. Mehrabad and F. Jolai, “Mathematical modeling and heuristic approaches to flexible job shop scheduling problems,” *Journal of Intelligent Manufacturing,* vol. 18, no. 3, pp. 331-342, 2007.

[57] E. G. Birgin, P. Feofiloff, C. G. Fernandes, E. L. d. Melo, M. T. I. Oshiro and D. P. Ronconi, “A MILP model for an extended version of the Flexible Job Shop Problem,” *Optimization Letters,* vol. 8, no. 4, pp. 1417-1431, 2014.

[58] IBM, “IBM ILOG CP Optimizer,” 2024. [Online]. Available: https://www.ibm.com/products/ilog-cplex-optimization-studio/cplex-cp-optimizer.

[59] Hexaly, “Hexaly,” 2024. [Online]. Available: https://www.hexaly.com/.

[60] Google OR-Tools, “CP-SAT Solver,” 2024. [Online]. Available: https://developers.google.com/optimization/cp/cp_solver.

[61] MiniZinc, “MiniZinc,” 2024. [Online]. Available: https://www.minizinc.org/.

[62] P. Laborie, J. Rogerie, P. Shaw and P. Vilím, “IBM ILOG CP optimizer for scheduling - 20+ years of scheduling with constraints at IBM/ILOG,” *Constraints,* vol. 23, no. 2, p. 210–250, 2018.

[63] W. T. Lunardi, E. G. Birgin, P. Laborie, D. P. Ronconi and H. Voos, “Mixed Integer linear programming and constraint programming models for the online printing shop scheduling problem,” *Computers & Operations Research,* vol. 123, 2020.

[64] Hexaly, “Hexaly vs Gurobi on the Flexible Job Shop Scheduling Problem (FJSP),” 2023. [Online]. Available: https://www.hexaly.com/benchmark/hexaly-vs-gurobi-flexible-job-shop-scheduling-problem-fjsp.

[65] Hexaly, “Hexaly, CP Optimizer, OR-Tools, Gurobi, Cplex on very large-scale instances of the Job Shop Scheduling Problem (JSSP),” 2024. [Online]. Available: https://www.hexaly.com/benchmark/hexaly-vs-cp-optimizer-vs-or-tools-on-the-job-shop-scheduling-problem-jssp.

[66] D. C. Ku and G. De Micheli, “Relative Scheduling under Timing Constraints: Algorithms for High-Level Synthesis of Digital Circuits,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems,* vol. 11, no. 6, pp. 696-718, 1992.

[67] S. E. Elmaghraby and J. Kamburowski, “The Analysis of Activity Networks Under Generalized Precedence Relations (GPRs),” *Management Science,* vol. 38, no. 9, pp. 1245-1263, 1992.

[68] R. E. Bellman, “On a Routing Problem,” *Quarterly of Applied Mathematics,* vol. 16, no. 1, pp. 87-90, 1958.

[69] I. Castillo and J. S. Smith, “Formal Modeling Methodologies for Control of Manufacturing Cells: Survey and Comparison,” *Journal of Manufacturing Systems,* vol. 21, no. 1, pp. 40-57, 2002.

[70] J. E. Hopcroft, R. Motwani and J. D. Ullman, Introduction to automata theory, languages, and computation, Third ed., Boston, MA: Pearson Education, Inc., 2007.

[71] D. Harel, “Statecharts: A visual formalism for complex systems,” *Science of Computer Programming,* vol. 8, no. 3, pp. 231-274, 1987.

[72] T. Murata, “Petri nets: Properties, analysis and applications,” *Proceedings of the IEEE,* vol. 77, no. 4, pp. 541-580, 1989.

[73] S. G. Shanmugham and C. A. Roberts, “Application of graphical specification methodologies to manufacturing control logic development: A classification and comparison,” *International Journal of Computer Integrated Manufacturing ,* vol. 11, no. 2, pp. 142-152, 1998.

[74] P. Caspi, D. Pilaud, N. Halbwachs and J. A. Plaice, “LUSTRE: a declarative language for real-time programming,” in *Proceedings of the 14th ACM SIGACT-SIGPLAN symposium on Principles of programming languages (POPL '87)*, Munich, 1987.

[75] F. Boussinot and R. de Simone, “The ESTEREL language,” *Proceedings of the IEEE,* vol. 79, no. 9, pp. 1293-1304, 1991.

[76] C. A. R. Hoare, “Communicating Sequential Processes,” *Communications of the ACM,* vol. 21, no. 8, pp. 666-677, 1978.

[77] K. E. Moore and S. M. Gupta, “Petri net models of flexible and automated manufacturing systems: a survey,” *International Journal of Production Research,* vol. 34, no. 11, pp. 3001-3035, 1996.

[78] N. Viswanadham, Y. Narahari and T. L. Johnson, “Deadlock prevention and deadlock avoidance in flexible manufacturing systems using Petri net models,” *IEEE Transactions on Robotics and Automation ,* vol. 6, no. 6, pp. 723-733, 1990.

[79] H. Lei, K. Xing, L. Han, F. Xiong and Z. Ge, “Deadlock-free scheduling for flexible manufacturing systems using Petri nets and heuristic search,” *Computers & Industrial Engineering,* vol. 72, pp. 297-305, 2014.

[80] G. Behrmann, A. David and K. G. Larsen, “A Tutorial on Uppaal,” in *Formal Methods for the Design of Real-Time Systems*, M. Bernardo and F. Corradini, Eds., Berlin, Heidelberg, Springer, 2004, pp. 200-236.

[81] VeriAal, “UPPAAL,” [Online]. Available: https://uppaal.veriaal.dk/.

[82] J. Bengtsson and W. Yi, “Timed Automata: Semantics, Algorithms and Tools,” in *Lectures on Concurrency and Petri Nets*, J. Desel, W. Reisig and G. Rozenberg, Eds., Berlin, Heidelberg, Springer, 2003, pp. 87-124.

[83] J. F. Groote and M. R. Mousavi, Modeling and analysis of communicating systems, Cambridge, MA: The MIT Press, 2014.

[84] A. Cimatti, E. Clarke, F. Giunchiglia and M. Roveri, “NuSMV: a new symbolic model checker,” *STTT International Journal on Software Tools for Technology Transfer,* vol. 2, pp. 410-425, 2000.

[85] G. J. Holzmann, “The model checker SPIN,” *IEEE Transactions on Software Engineering,* vol. 23, no. 5, pp. 279-295, 1997.

[86] D. Harel, H. Lachover, A. Naamad, A. Pnueli, M. Politi, R. Sherman, A. Shtull-Trauring and M. Trakhtenbrot, “STATEMATE: a working environment for the development of complex reactive systems,” *IEEE Transactions on Software Engineering ,* vol. 16, no. 4, pp. 403-414, 1990.

[87] IBM, “IBM Engineering Systems Design Rhapsody,” [Online]. Available: https://www.ibm.com/products/systems-design-rhapsody.

[88] M. Kwiatkowska, G. Norman and D. Parker, “Stochastic Model Checking,” in *Formal Methods for Performance Evaluation*, M. Bernardo and J. Hillston, Eds., Berlin, Springer, 2007, pp. 220-270.

[89] University of Oxford, “PRISM model checker,” 2024. [Online]. Available: https://www.prismmodelchecker.org/.

[90] A. David, K. G. Larsen, A. Legay, M. Mikučionis and D. B. Poulsen, “Uppaal SMC tutorial,” *International Journal on Software Tools for Technology Transfer,* vol. 17, no. 4, pp. 397-415, 2015.

[91] L. Kallenberg, “Markov Decision Processes,” Leiden University, Leiden, 2021.

[92] C. H. Papadimitriou and J. N. Tsitsiklis, “The Complexity of Markov Decision Processes,” *Mathematics of Operations Research,* vol. 12, no. 3, pp. 441-450, 1987.

[93] E. A. Lee and T. M. Parks, “Dataflow Process Networks,” *Proceedings of the IEEE,* vol. 83, no. 5, pp. 773-799, 1995.

[94] E. A. Lee and D. G. Messerschmitt, “Static scheduling of Synchronous Data Flow Programs for Digital Signal Processing,” *IEEE Transactions on Computers,* Vols. C-36, no. 1, pp. 24-35, 1987.

[95] T. M. Parks, J. L. Pino and E. A. Lee, “A Comparison of Synchronous and Cyclo-Static Dataflow,” in *Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, 1995.

[96] B. D. Theelen, E. F. Deprettere and S. S. Bhattacharyya, “Dynamic Dataflow Graphs,” in *Handbook of Signal Processing Systems*, Third ed., S. S. Bhattacharyya, E. F. Deprettere, R. Leupers and J. Takala, Eds., Cham, Springer, 2019, pp. 1173-1210.

[97] K. Salimifard and M. Wright, “Petri net-based modelling of workflow systems: An overview,” *European Journal of Operational Research,* vol. 134, no. 3, pp. 664-676, 2001.

[98] W. M. P. van der Aalst, “Petri net based scheduling,” *OR Spectrum,* vol. 18, no. 4, pp. 219-229, 1996.

[99] F. Pommereau, “SNAKES,” 2024. [Online]. Available: https://snakes.ibisc.univ-evry.fr/.

[100] R. Ruiz, “Scheduling Heuristics,” in *Handbook of Heuristics*, R. Martí, P. Panos and M. G. C. Resende, Eds., Cham, Springer, 2020.

[101] R. L. Graham, “Bounds on Multiprocessor Timing Anomalies,” *SIAM Journal on Applied Mathematics,* vol. 17, no. 2, pp. 416-429, 1969.

[102] J. M. J. Schutten, “List scheduling revisited,” *Operations Research Letters,* vol. 18, no. 4, pp. 167-170, 1996.

[103] S. S. Panwalkar and W. Iskander, “A Survey of Scheduling Rules,” *Operations Research,* vol. 25, no. 1, pp. 45-61, 1977.

[104] J. Adams, E. Balas and D. Zawack, “The shifting bottleneck procedure for job shop scheduling,” *Management Science,* vol. 34, no. 3, pp. 391-401, 1988.

[105] J. Błażewicz, E. Pesch and M. Sterna, “The disjunctive graph machine representation of the job shop scheduling problem,” *European Journal of Operational Research,* vol. 127, no. 2, pp. 317-331, 2000.

[106] T. Dean and M. Boddy, “An analysis of time-dependent planning,” in *Proceedings of the Seventh AAAI National Conference on Artificial Intelligence (AAAI'88)*, Saint Paul, MN, 1988.

[107] O. T. Baruwa, M. A. Piera and A. Guasch, “Deadlock-Free Scheduling Method for Flexible Manufacturing Systems Based on Timed Colored Petri Nets and Anytime Heuristic Search,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems,* vol. 45, no. 5, pp. 831-846, 2015.

[108] J. Efstathiou, “Anytime heuristic schedule repair in manufacturing industry,” *IEE Proceedings - Control Theory and Applications,* vol. 143, no. 2, pp. 114-124, 1996.

[109] M. Mitchell, An Introduction to Genetic Algorithms, Cambridge, Massachusetts: The MIT Press, 1999.

[110] K. Deb, A. Pratap, S. Agarwal and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation,* vol. 6, no. 2, pp. 182-197, 2002.

[111] P. J. M. van Laarhoven and E. H. L. Aarts, Simulated annealing: Theory and Applications, Dordrecht: Reidel, 1987.

[112] F. Glover, "Tabu search - Part 1," *ORSA Journal on Computing.,* vol. 1, no. 2, pp. 190-206, 1989.

[113] N. Paape, J. van Eekelen and M. Reniers, "Review of simulation software for cyber-physical production systems with intelligent distributed production control," *International Journal of Computer Integrated Manufacturing,* vol. 37, no. 5, pp. 589-611, 2024.

[114] J. Zhang and S. Cui, "Investigating the Number of Monte Carlo Simulations for Statistically Stationary Model Outputs," *Axioms,* vol. 12, no. 5, 2023.

[115] A. Negahban and J. S. Smith, "Simulation for manufacturing system design and operation: Literature review and analysis," *Journal of Manufacturing Systems,* vol. 33, no. 2, pp. 241-261, 2014.

[116] I. Jithavech and K. Kumar Krishnan, "A simulation-based approach for risk assessment of facility layout designs under stochastic product demands," *The International Journal of Advanced Manufacturing Technology,* vol. 29, pp. 27-40, 2010.

[117] P.-H. Koo and J. Jang, "Vehicle Travel Time Models for AGV Systems under Various Dispatching Rules," *The International Journal of Flexible Manufacturing Systems,* vol. 14, no. 3, pp. 249-261, 2002.

[118] M. Glatt, G. Kasakow and J. C. Aurich, "Combining physical simulation and discrete-event material flow simulation," *Procedia CIRP,* vol. 72, pp. 420-425, 2018.

[119] G. Zülch and M. Becker, "Impact of ageing workforces on long-term efficiency of manufacturing systems," *Journal of Simulation,* vol. 4, no. 4, pp. 260-267, 2010.

[120] N. Katiraee, M. Calzavara, S. Finco, D. Battini and O. Battaïa, "Consideration of workers' differences in production systems modelling and design: State of the art and directions for future research," *Journal of Production Research,* vol. 59, no. 11, pp. 3237-3268, 2021.

[121] The Anylogic Company, "Anylogic," 2024. [Online]. Available: https://www.anylogic.com/.

[122] D. Mourtzis, M. Doukas and D. Bernidaki, "Simulation in Manufacturing: Review and Challenges," *Procedia CIRP,* vol. 25, pp. 213-229, 2014.

[123] Rockwell Automation, "Arena Simulation Software," 2024. [Online]. Available: https://www.rockwellautomation.com/en-us/products/software/arena-simulation.html.

[124] FlexSim Software Products, Inc., "3D Simulation Modeling and Analysis Software," 2024. [Online]. Available: https://www.flexsim.com/.

[125] Siemens Digital Industries Software, "Tecnomatix Plant Simulation," 2024. [Online]. Available: https://plm.sw.siemens.com/en-US/tecnomatix/products/plant-simulation-software/.

[126] Lanner Inc, "WITNESS Simulation Modeling Software," 2024. [Online]. Available: https://www.lanner.com/en-us/technology/witness-simulation-software.html.

[127] Simio LLC, "Manufacturing Simulation and Scheduling Software," 2024. [Online]. Available: https://www.simio.com/applications/manufacturing-simulation-software/.

[128] Simul8 Corporation, “Simul8,” 2024. [Online]. Available: https://www.simul8.com/.

[129] JaamSim Software Inc, “JaamSim,” 2023. [Online]. Available: https://jaamsim.com/.

[130] Team SimPy, “SimPy discrete event simulation for Python,” 2024. [Online]. Available: https://simpy.readthedocs.io/en/latest/.

[131] salabim, “salabim discrete event simulation,” 2024. [Online]. Available: https://www.salabim.org/.

[132] U. Wilensky, “NetLogo,” 2023. [Online]. Available: https://ccl.northwestern.edu/netlogo/.

[133] OptTek Systems, Inc., “OptQuest,” 2024. [Online]. Available: https://www.opttek.com/products/optquest/.

[134] N. Gunantara, “A review of multi-objective optimization: Methods and its applications,” *Cogent Engineering,* vol. 5, no. 1, 2018.

[135] A. P. Pimentel, “Methodologies for Design Space Exploration,” in *Handbook of Computer Architecture*, Singapore, Springer, 2022.

[136] M. Geilen, T. Basten, B. Theelen and R. Otten, “An algebra of Pareto points,” in *Fifth International Conference on Application of Concurrency to System Design (ACSD'05)*, Saint Malo, 2005.

[137] Wikipedia, “Radar chart,” 2024. [Online]. Available: https://en.wikipedia.org/wiki/Radar_chart.

[138] Wikipedia, “Parallel coordinates,” 2024. [Online]. Available: https://en.wikipedia.org/wiki/Parallel_coordinates.

[139] M. Efatmaneshnik, N. Chitsaz and L. Qiao, “Decision making for multi-objective problems: Mean and median metrics,” *Systems Engineering,* vol. 26, no. 6, pp. 814-829, 2023.

[140] B. Kienhuis, E. Deprettere, K. Vissers and P. van der Wolff, “An approach for quantitative analysis of application-specific dataflow architectures,” in *1997 IEEE International Conference on Application-Specific Systems*, Zurich, 1997.

[141] J. Verriet, “Compositional Performance Prediction of Tightly Coupled Manufacturing Systems,” TNO, Eindhoven, 2023.

[142] Eclipse Foundation, “Eclipse LSAT,” 2021. [Online]. Available: https://eclipse.dev/lsat/.

[143] T. Bijlsma, B. v. d. Sanden, Y. Li, R. Janssen and R. Tinsel, “Decision support methodology for evolutionary embedded system design,” in *2019 International Symposium on Systems Engineering*, Edinburgh, 2019.

[144] OpenTelemetry Authors, “OpenTelemetry High-quality, ubiquitous, and portable telemetry to enable effective observability,” 2024. [Online]. Available: https://opentelemetry.io/.

[145] PlantUML, “PlantUML at a Glance,” 2024. [Online]. Available: https://plantuml.com/.

[146] R. Andrade, “Real-Time Conformance Checking for Microservice Applications,” Eindhoven University of Technology, Eindhoven, 2024.

[147] M. Vollaard, “Hardware Dimensioning for Microservice-based Cyber-Physical Systems: A Profiling and Performance Prediction Method,” Vrije Universiteit, Amsterdam, 2023.

[148] B. van der Sanden, Y. Blankenstein, R. Schiffelers and J. Voeten, “LSAT: Specification and Analysis of Product Logistics in Flexible Manufacturing Systems,” in *IEEE 17th International Conference on Automation Science and Engineering*, Lyon, 2021.

[149] J. Verriet, B. van der Sanden, G. van der Veen, A. van Splunter, S. Lousberg, M. Hendriks and T. Basten, “Experiences and Lessons from Introducing Model-Based

Analysis in Brown-Field Product Family Development," in *11th International Conference on Model-Based Software and Systems Engineering*, Lisbon, 2023.

[150] S. Adyanthaya, M. Geilen, T. Basten, R. Schiffelers, B. Theelen and J. Voeten, "Fast Multiprocessor Scheduling with Fixed Task Binding of Large Scale Industrial Cyber Physical Systems," in *2013 16th Euromicro Conference on Digital System Design*, Los Alamitos, CA, 2013.

[151] J. Bastos, B. van der Sanden, O. Donk, J. Voeten, S. Stuijk, R. Schiffelers and H. Corporaal, "Identifying bottlenecks in manufacturing systems using stochastic criticality analysis," in *2017 Forum on Specification and Design Languages*, Verona, 2017.

[152] E. Reehuis and T. Bäck, "Mixed-integer evolution strategy using multiobjective selection applied to warehouse design optimization," in *12th annual conference on genetic and evolutionary computation*, Portland, OR, 2010.

[153] R. Hamberg and J. Verriet, Automation in Warehouse Development, London: Springer, 2012.

[154] M. Buermann, *A flexible simulation framework for modeling a feed-pushing AGV in a dairy farm,* Eindhoven: Eindhoven University of Technology, 2022.

[155] CPN Tools, "CPN Tools," 2024. [Online]. Available: https://cpntools.org/.

[156] Electronic Systems Group, "SDF3," 2024. [Online]. Available: https://www.es.ele.tue.nl/sdf3/.

[157] T. Basten, R. Hamberg, F. Reckers and J. Verriet, Model-Based Design of Adaptive Embedded Systems, New York: Springer, 2012.

[158] M. Hendriks, T. Basten, J. Verriet, M. Brassé and L. Somers, "A blueprint for system-level performance modeling of software-intensive embedded systems," *International Journal on Software Tools for Technology Transfer,* vol. 18, no. 1, pp. 21-40, 2016.

[159] TNO-ESI, "DAARIUS methodology," 2024. [Online]. Available: https://esi.nl/research/output/methods/daarius-methodology.

[160] Eclipse Foundation, "Open Source Solution for Model-Based Systems Engineering," 2024. [Online]. Available: https://mbse-capella.org/.

[161] J. Hooman, K. Kanters, A. Vasenev and J. Verriet, "MBSE-Based Design Space Exploration for Productivity Improvement Using Workflow Models," in *2023 Conference on Systems Engineering Research*, Hoboken, NJ, 2023.

[162] J. Verriet, L. Buit, R. Doornbos, B. Huijbrechts, K. Ševo, J. Sleuters and M. Verberkt, "Virtual Prototyping of Large-Scale IoT Control Systems Using Domain-Specific Languages," in *7th International Conference on Model-Driven Engineering and Software Development*, Prague, 2019.

[163] J. Sleuters, Y. Li, J. Verriet, M. Velikova and R. Doornbos, "A Digital Twin Method for Automated Behavior Analysis of Large-Scale Distributed IoT Systems," in *2019 14th Annual Conference System of Systems Engineering*, Anchorage, AK, 2019.

[164] A. Plaat, W. Kosters and M. Preuss, "Deep model-based reinforcement learning for high-dimensional problems, a survey," *arXiv,* vol. 2008.05598, 2020.

[165] M. Kemmerling, D. Lütticke and R. H. Schmitt, "Beyond games: a systematic review of neural Monte Carlo tree search applications," *Applied Intelligence,* vol. 54, no. 1, pp. 1020-1046, 2024.

ICT, Strategy & Policy

High Tech Campus 25
5656 AE Eindhoven
www.tno.nl

innovation
for life