



Testing in the incremental design and development of complex products

Khadija Tahera¹ · David C. Wynn² · Chris Earl³ · Claudia M. Eckert³

Received: 24 January 2018 / Revised: 18 June 2018 / Accepted: 28 July 2018 / Published online: 6 August 2018
© The Author(s) 2018

Abstract

Testing is an important aspect of design and development which consumes significant time and resource in many companies. However, it has received less research attention than many other activities in product development, and especially, very few publications report empirical studies of engineering testing. Such studies are needed to establish the importance of testing and inform the development of pragmatic support methods. This paper combines insights from literature study with findings from three empirical studies of testing. The case studies concern incrementally developed complex products in the automotive domain. A description of testing practice as observed in these studies is provided, confirming that testing activities are used for multiple purposes depending on the context, and are intertwined with design from start to finish of the development process, not done after it as many models depict. Descriptive process models are developed to indicate some of the key insights, and opportunities for further research are suggested.

Keywords Testing · Design and development · Incremental complex products · Case studies · Descriptive process model

1 Introduction

Numerous models of engineering design and development processes present them as sequences or networks of tasks (Wynn and Clarkson 2018). In industry practice, these tasks include numerous testing activities ranging from material and part testing, to testing of main functional components and subsystems, through to whole-system testing against customer requirements (O'Connor 2001). Testing activities can trigger design change, and also help companies respond to design change by ensuring compliance with requirements is maintained (Lévárdy et al. 2004). However, despite the importance of testing in practice, most design process models do not explicitly emphasise the integration of testing activities throughout product development. From the point of view of testing, these process models are partial not only

in the activities that they depict, but also in their structures which typically do not emphasise the relationships between design and testing. We argue that this is an important gap, because a number of researchers have reported that testing is a very significant cost factor in engineering design and development (Tahera et al. 2017).

In engineering practice, the relationship between design and testing is not straightforward. Testing has a variety of purposes as well as taking many forms, including assessing performance of materials and components, assessing the function and performance of subsystems and ensuring the overall capability of the product for complex user needs. Some testing is typically done by an Original Equipment Manufacturer (OEM), while other testing is done by subsystem suppliers. These tests, both virtual and physical, are important to support verification and validation (Shabi and Reich 2012), and can also guide design decisions if results are available in good time (Kennedy 2008). Computer-aided engineering (CAE) simulations constitute virtual tests that are often used alongside physical tests of components, subsystems, and whole product prototypes. This concurrency accelerates product development (Tahera 2014) while also increasing confidence in test results (Thomke 1998).

As suggested above, testing is not limited to the end of a development process, but is done on an ongoing basis and

✉ Khadija Tahera
k.tahera@hud.ac.uk

¹ Department of Computing and Engineering, University of Huddersfield, Huddersfield HD1 3DH, UK

² Department of Mechanical Engineering, University of Auckland, Auckland 1010, New Zealand

³ School of Engineering and Innovation, The Open University, Milton Keynes MK7 6AA, UK

overlaps with design activities throughout product development (Tahera et al. 2017). This is particularly evident for products which are developed incrementally. Here, this means that a new product is designed using a previous product as the starting point. Modifications and changes are introduced to meet new market and customer requirements, incorporate new technologies, improve manufacturing quality, or reduce costs, for example. Companies typically attempt to limit the proportion of new components in these incremental design processes (Wyatt et al. 2009), which has implications for the testing that is required. The integration of testing with design activities is also especially close for complex products with many interdependent components and functional subsystems. Many such interactions means that design modifications in a component or subsystem are more likely to propagate to other areas of the product (Clarkson et al. 2004), leading to more tests being required to address all modified components. The effects of design complexity on the integration of testing and design are further emphasised when customer requirements cover numerous use cases, perhaps satisfied through different product variants. This further prompts careful integration of testing with design, to adequately account for the full context of use while keeping the testing effort and cost manageable.

1.1 Contribution of this article

Overall, the complexity of incremental product development means that it is both possible and necessary to embed testing closely within the design process. However, there are few detailed empirical studies of engineering testing in the research literature, and the interplay between design and testing is not emphasised in the well-known procedural models of the design and development process. The present paper aims to address these gaps, drawing on case studies to examine how testing occurs in practice and introducing descriptive process models to textually and graphically summarise some of the key issues that are discussed. The paper provides a significant expansion on preliminary findings published in two conference papers (Tahera et al. 2012, 2015), incorporating new insight from an additional case study and an expanded literature review.

1.2 Research method

A search was undertaken to find research literature relevant to testing in engineering design. This considered both descriptive and normative work as well as treatments of testing in design and development process models. Beginning with the reviews recently published in Tahera et al. (2017) and Wynn and Clarkson (2018), the bibliographies of those articles were studied, relevant original sources were revisited and their bibliographies studied in turn. Google Scholar

and Scopus were used to find additional relevant sources and key journals including *Research in Engineering Design*, *Journal of Engineering Design*, and *IEEE Transactions on Engineering Management* were considered. Sections 2 and 3 were incrementally synthesised as the literature was found and digested.

Insights from the literature study were complemented by three industry case studies. The case study companies incrementally design products in the automotive domain: diesel engines, forklift trucks, and turbochargers. The studies focused on testing relating to mechanical design issues, although all the products also have significant electrical and electronic aspects. More information on the case study methodology is provided in Sect. 4.1.

1.3 Outline

The paper begins by discussing research literature on testing in engineering design to summarise current state-of-the-art understanding of this topic. Drawing on this literature, Sect. 2 highlights some of the main factors impacting testing and establishes the mutual dependence between test activities and related activities such as design, verification, and validation. Section 3 reviews design process models from the point of view of testing, concluding that the complex relationships between design and test are not adequately represented in many of these models. This sets the context for the case studies of testing and design outlined in Sect. 4. Section 5 draws together insights from the previous three sections to develop descriptive models of the product development process for incremental and complex products, intended to summarise the main types of testing observed in the studies, and to show their relationship with other design activities. Implications and limitations are explored in Sect. 6. Section 7 offers concluding remarks.

2 Background

Although testing is important in many design and development domains, this paper focuses primarily on testing during *engineering* design and product development. The focus is set on testing itself as well as closely related tasks, thereby including design analysis, modelling and simulation, physical tests, and analysis of test data. The literature study undertaken for this paper supports earlier comments by Engel (2010) that relatively little work has been published on testing in this specific context, as well as the observation of Lévárdy et al. (2004) that testing has received significantly less research attention than the associated design and analysis tasks of product development and systems engineering.

We contend that the topic deserves further research, because it has been suggested that testing is an important

component of product development/systems engineering that can consume substantial proportions of the overall effort and cost (Thomke and Bell 2001; Engel and Barad 2003). Furthermore, several researchers agree that there seems to be opportunity for improvement. For example, Shabi and Reich (2012) write that testing, although required in most product development projects, is seldom done in an optimal manner. O'Connor (2001) suggest that a contributing factor is that there is no consistent set of principles, approaches, and methodology for testing, while more recently, Shabi et al. (2017) similarly argue that there is a “lack of a structured approach” in the engineering domain. In contrast, there is substantial work on structured testing in the context of verification and validation in software engineering (see, e.g., Bertolino 2007, for an overview of key issues). In some respects, software testing is different from hardware testing, for example, because many tests can be automated and run rapidly at low cost in the context of software, which is often not possible when testing hardware (Shabi and Reich 2012). Despite the differences, some of the insights are applicable to both domains (Engel and Barad 2003), since some objectives of testing are broadly similar in both cases, and it can also be noted that most complex engineered products comprise a significant proportion of software. Although software testing is not the main focus of this paper, the literature on this topic is, therefore, mentioned where appropriate to support the discussion.

The next subsections provide an overview of engineering testing as it appears in research literature, and place it in the context of related concepts such as validation, verification, and quality. We then establish that interactions exist between how testing is (or should be) done and characteristics of the design and development context.

2.1 Roles of testing in design and development

Testing is often discussed in the context of verification and validation (V&V), whose purpose is “to ensure the customer receives a quality product and neither the customer nor the supplier is burdened with the cost of rework due to failures of the product after delivery” (Shabi et al. 2017). In this context, testing is typically viewed as one of several methods that can be used in support of V&V (Hoppe et al. 2007) and is often the most data-intensive of those methods (Pineda and Kilicay-Ergin 2010). Complementing testing, other V&V methods include analysis, inspection of properties and simple demonstration of functionality (Engel 2010).

In the V&V context, tests are often intended to demonstrate that a design behaves as expected and as specified in technical requirements, before its release. As such testing is closely related to quality (Engel and Barad 2003). In particular, quality may be viewed as the ability of a product to meet explicit or implicit needs, and testing can help to

identify and eliminate quality failures. Engel and Shachar (2006) write that the total cost associated with quality can be decomposed into three components: (1) the cost of running Verification, Validation and Testing (VVT) activities; (2) the cost of rework to correct quality failures discovered during those VVT activities; and (3) the cost of handling quality failures that are not discovered by the VVT. The total cost of quality can be minimised by appropriate planning of VVT activities, which potentially impacts all three components described above. The planning can be assisted by methods such as Design of Experiments for micro-level planning of individual tests (e.g., Luna et al. 2013), Design Failure Modes and Effects Analysis (DFMEA) to help identify, where testing may be useful (e.g., Shankar et al. 2016), and methods to optimise the overall VVT strategy (e.g., Engel and Shachar 2006; Shabi and Reich 2012). These and other methods are discussed in greater detail in forthcoming sections. Individual tests or the testing plan can themselves contain flaws or be poorly designed such that tests provide incorrect or incomplete information. Overall, therefore, attention to the quality of testing and of the broader V&V plan is important to assure quality of a designed product or system.

To further clarify the relationship between testing, verification, validation, and quality, Table 1 summarises definitions from several sources including the IEEE standard glossary of software terminology (IEEE 1990), the ISO 9000 Quality Management System (Hoyle 2009), the telecommunications community (Engel 2010) and the modelling and simulation community (Balci 1998). This paper adopts the definition of testing from the IEEE standard glossary, which is similar to the definition of Shabi et al. (2017) and those used in the case study companies discussed in Sect. 4. In particular:

Testing: “an activity in which a system or component is executed under specified conditions, the results are observed or recorded, and an evaluation is made of some aspect of the system or component” (IEEE 1990)

As well as V&V, testing also has other roles in a development process:

- Testing may be undertaken for learning with respect to the design, e.g., to understand or refine new technology, or to explore the design space. Similarly, user testing can be integrated into the design process to gain stakeholder feedback on an emerging design, which can be helpful in situations where needs are difficult to express and/or formalise as technical requirements.
- Testing may generate information and knowledge about the test apparatus and testing method, thereby contributing to insights about how future tests can be most usefully conducted.

Table 1 Definitions of verification, validation and testing

Source	Definition(s)
IEEE standard glossary (IEEE 1990)	<p>Testing: An activity in which a system or component is executed under specified conditions, the results are observed or recorded, and an evaluation is made of some aspect of the system or component</p> <p>Verification: The process of evaluating a system or component to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase</p> <p>Validation: The process of evaluating a system or component during or at the end of the development process to determine whether it satisfies specified requirements</p>
ISO 9000 Quality management system (Hoyle 2009)	<p>Testing: (Definition not found)</p> <p>Verification: a process to ensure through the provision of objective evidence that specified requirements has been fulfilled</p> <p>Validation: a process to confirm that resulting product is capable of fulfilling the requirements for the specified application or intended use where known</p>
Telecommunication community (Engel 2010)	<p>Testing: (1) Physical measurements taken to verify conclusions obtained from mathematical modelling and analysis. (2) Physical measurements taken for developing mathematical models</p> <p>Verification: (1) Comparing an activity, a process, or a product with the corresponding requirements or specifications. (2) the process of comparing two levels of an information system specification for proper correspondence (e.g., security policy model with top-level specification, top-level specification with source code or source code with object code)</p> <p>Validation: (1) Test to determine whether an implemented system fulfils its requirements. (2) The checking of data for correctness or for compliance with applicable standards, rules and conventions</p>
Modelling and simulation community Balci (1998)	<p>Testing: ascertaining whether inaccuracies or errors exist in a model. The model is subjected to test data or test cases to determine if it functions properly. “Test failed” implies a problem with the model, not the test. A test is devised and testing is conducted to perform validation, verification or both. Some tests are intended to judge the accuracy of model transformation from one form into another (verification)</p> <p>Verification: Substantiating that the model is transformed from one form into another, as intended, with sufficient accuracy. Model verification deals with building the model right</p> <p>Validation: Model validation is substantiating that the model, within its domain of applicability, behaves with satisfactory accuracy consistent with the modelling and simulation objectives. Model validation deals with building the right model</p>
Systems Engineering/Engineering design (Shabi et al. 2017)	<p>Testing: “Operating or activating a realized product or system under specified conditions and observing or recording the exhibited behaviour”</p> <p>Verification: “Evaluat[ing] a realized product and prov[ing] its compliance with engineering requirements”</p> <p>Validation: “Evaluating a product against specified (or unspecified) customer requirements to determine whether the product satisfies its stakeholders”</p>

- Testing may reveal information about models or assumptions used in the design process, thereby helping to improve them.

Testing can be relatively straightforward for simple designs, or can involve complex processes in its own right. For example, in aircraft programs, large-scale bespoke test rigs must be designed, built and commissioned considering the specifics of each design. Testing is also required in production

processes as an integral part of quality assurance, although discussion of this is beyond the scope of the present article.

Another activity closely related to testing is prototyping (Menold et al. 2018). A prototype, defined as a “pre-production representation of some aspect of a concept or final design” (Camburn et al. 2017), is required for many testing situations although prototyping can also have other purposes, such as to support design communication. For more information on research into design prototyping and its relationship to testing, the reader is referred to the literature

reviews by Zorriassatine et al. (2003) and Camburn et al. (2017).

2.2 Factors that influence how testing is done

Testing is not done in the same way for every development project. This section draws on the literature to discuss some factors that influence how testing occurs.

2.2.1 Design complexity

An increase in design complexity can be expected to lead to an increase in testing complexity. In particular, each additional component or subsystem in a design requires additional testing and validation (Novak and Eppinger 2001). Similarly, more complex interfaces mean that more problems can be expected during the integration process and the more important the role of testing may be to uncover and eliminate them (Pineda and Kilicay-Ergin 2010).

Another issue relating to complexity in testing is the number of combinations of test parameters required to uncover all flaws. Often a system will be subject to multiple factors when in use, and in a design with significant internal complexity the flaws are unlikely to be revealed by varying these conditions one-at-a-time. For example, in a study of a NASA database application, only 67% of flaws were revealed by one-factor-at-a-time testing; 93% by testing all possible pairwise combinations of factors deviating from nominal values; and 98% by all three-way combinations (Kuhn et al. 2008). This example is from the software domain. However, the same issues can be expected in testing of hardware products involving significant software components. Where the number of permutations combined with individual test costs are too large to permit exhaustive testing, as in most real situations, it may be especially important to consider how the testing strategy can be optimised (Thomke and Bell 2001).

Testing can focus on parts of a system (called unit testing in the context of software engineering) on particular combinations of parts (called integration testing) or on the whole-system behaviour. In physical testing of complex hardware products, whole-system tests in particular can be costly and time consuming. In some cases, running tests on a whole system before deployment may not be possible at all. Luna et al. (2013) argue that testing and related activities are especially challenging in a Systems of Systems context and this requires a different approach to testing in traditional systems engineering, because many constituent systems are involved, subsystems evolve over time with significant managerial independence such that emergent behaviours can be expected, and the operating environment is expected to involve significant future unknowns. In such contexts, exhaustive testing is not possible and test complexity needs

to be managed by identifying and focusing attention on critical subsystems and interactions (Luna et al. 2013).

2.2.2 Product architecture

Ulrich (1995) defines product architecture as (1) the arrangement of functional elements comprising the product; (2) the mapping from functional elements to physical components; and (3) the specification of the interfaces among interacting physical components. A number of authors have argued that product architecture affects how a design should be tested (e.g., Baldwin and Clark 2000; Loch et al. 2001; Thomke and Bell 2001; Sosa et al. 2003; Lévárdy et al. 2004). For example, Loch et al. (2001) argue that modular architecture can help to reduce the overall cost of testing. Contributing factors are that interfaces are generally clearer and interactions between subsystems are generally simpler in a modular design. The testing strategy and test plan can also be designed to exploit the modularity of a product architecture (Baldwin and Clark 2000). Care is required when focusing testing on modules as this suggests, because reducing system-level testing arguably presents an increased risk of finding faults later in use (Jones 2010).

Another architectural issue impacting testing is commonality of modules within a product or across a product family. For example, Pineda and Kilicay-Ergin (2010) argue that increased commonality reduces the number of unique parts and interfaces, and thus may reduce the number of required tests. On the other hand, commonality also increases the number of conditions each part must operate under, which may increase the need for testing.

2.2.3 Degree of novelty

The testing to be performed also depends on the degree to which a design is innovative or incremental. Projects including significant innovative elements often require tests to experiment, learn about and refine new technologies (Song and Montoya-Weiss 1998). In principle, this should occur towards the beginning of a development project. Information developed from such tests might substantially impact the evolution of the design process. This is also the case for user-centered approaches, in which user tests of an emerging design are expected to guide future iterations. On the other hand, when a product is developed incrementally from a previous version, less technology development is typically needed within the project, tests are more likely to focus on verification rather than learning, and as such testing may be anticipated to generate less disruption to the planned work. In addition, parts or subsystems that are carried across from a previous design and not modified can be easily verified without testing them again on the unit level. Shabi et al.

(2017) describe this as “verification by comparison” of the reused subsystem to a previously verified identical design.

In the context of incrementally designed complex products, insights and established processes from previous projects typically inform the testing plan and test activities. Thus, incremental development and familiarity with the technology may be helpful to improve the quality of Verification, Validation and Testing (VVT) (Pineda and Kilicay-Ergin 2010).

2.2.4 Timing of testing

Shabi and Reich (2012) argue that VVT should take place “on a continuous basis starting as early as possible in the product lifecycle” and should “involve different methods as the needs of those tasks change along the development lifecycle”. Lu et al. (2000) write that two types of test that occur at different times can be distinguished: analysis tests are done early in the development process to resolve risks, while validation tests, done later, aim to confirm existing predictions that the product will perform as intended.

Some authors have undertaken studies that support the importance of appropriately positioning different tests in the Product Development (PD) project timeline. For example, Thomke (1998) reports an empirical study of integrating testing into the development process, based on a case study and survey of integrated circuit development. He focuses on the point in a project when a company moves from simulated tests to begin physical testing. He argues that when physical testing is very time consuming and costly, designers try to work out as many problems as possible using simulation prior to creating a physical prototype. On the other hand, when physical testing is less costly, designers use it earlier in the design process. (We note that the capabilities of simulation tools have increased significantly in the years since this study). A related issue is that high-fidelity tests may often not be possible early in the design process, because the required design information is not yet available (Thomke and Bell 2001). In system-level testing of complex products, the need for mature design information and high cost of testing means that significant tests are typically planned very late in the process, to verify a prototype behaves as the designers predict (Reinertsen 1997). This can cause significant problems for the project if major design flaws are discovered at the test.

Proponents of set-based concurrent engineering (SBCE) argue that physical or virtual testing should take place as early as possible in the process to learn about the design space and take informed decisions, to avoid the costly rework that occurs if testing reveals flaws late in the design process. This strategy has been called test-then-design (Kennedy 2008) and focuses not on verifying and validating, but on using tests to systematically uncover the

tradeoffs and limitations inherent in the chosen technology before making design decisions. In this context, the cost of testing can be reduced if the tests are carefully designed to allow rapid exploration of the design space, for example, with reconfigurable apparatus (Ward et al. 1995).

2.2.5 Susceptibility to design change

As mentioned above, many engineering projects (and our case studies reported in Sect. 4) constitute incremental modifications of an existing product. In such cases not everything needs to be tested. With the necessary changes known from the outset, appropriate test activities can in principle be identified and built into the testing plans. However, changes also arise unexpectedly during a project, from (a) modified requirements or (b) faults, which themselves may be revealed by testing (Eckert et al. 2004).

In terms of requirements, noting that V&V test plans are designed to verify that a design meets specific technical requirements, frequently changing requirements may have significant impact on the test plan. Pineda and Kilicay-Ergin (2010) accordingly argue that VVT quality is also negatively impacted by unclear, ambiguous or volatile requirements. In terms of faults identified through tests, the flaw must be corrected which requires a design change. Regardless of how a change is triggered, because every part of a design is connected at least to one other part, a change in one part can propagate and require changes in others, so that the parts can continue to work together (Clarkson et al. 2004). Change propagation may need to be considered when creating or revising a testing plan, to ensure that all potentially impacted parts and interactions are appropriately tested (Shankar et al. 2016).

3 Testing in design and development process models

Section 2 reveals a consensus in the literature that testing is intertwined with design activities in complex ways. Testing is used for different purposes, and several factors impact on how testing unfolds during a development process. This section moves on to consider how testing is treated in some well-established design and development process models. Although a number of mathematical and simulation models do consider some aspects of testing discussed in Sect. 2, we find that the complexities of testing revealed in the previous section are not emphasised in the main procedural models of the design and development process. This establishes the gap towards which the present article contributes.

3.1 Mathematical and simulation models of testing

A number of mathematical and simulation models consider the role of testing in design and development. In the following, these models are discussed and key insights are extracted. The discussion is organised according to the overall purpose of the models, following the approach of Wynn and Clarkson (2018). In particular Sect. 3.1.1 discusses models that develop research insight into testing through study of representative situations, but are arguably not intended to support practitioners wishing to model and analyse the detail of their processes. Section 3.1.2 moves on to discuss publications that work towards providing support for decision making about testing, based on modelling and analysis of a specific company situation.

3.1.1 Models of testing focused on research insight

Some early work offers research insight relevant for testing through discussion of related issues such as design review. Ha and Porteus (1995), for example, develop a model to study the optimal timing of design reviews in a concurrent process. They consider that the advantages of frequent reviews are to enable concurrency by allowing design information to be released, while uncovering design flaws so they do not incur downstream wasted effort. At the same time, frequent reviews increase the costs related to setting up and executing them. Ha and Porteus (1995) show that the optimal timing of reviews depends on whether the concurrency or quality issues dominate. Their model is extended by Ahmadi and Wang (1999) to also consider how resource is allocated to different design stages. In this case, the model is used to consider how the reviews should be scheduled with a view to minimise the risk of missing targets. The insights for design reviews are also applicable to tests, considering the level of resolution of these models. Also generating a model that is set in the context of testing, but without extensive discussion of the specific detail of tests, Yassine et al. (2003) model product development as a generation–test cycle, focusing on the situation in which a number of subsystem design teams continuously provide information to a system-level team who are responsible for integration and test of the whole system. They consider that the system-level team provides feedback to the subsystem teams about how well their designs work together, but only intermittently, due to the time required to complete the tests as well as other factors. Through simulation they conclude that increased delays between the design generation and release of test results, or decrease in the quality of the test results, not only causes more rework for subsystem teams but can also drive churn in the overall process (a situation in which problems are generated as fast or faster than they can be

resolved). Although set in the context of testing, the model focuses on the interaction between generate and test and does not extensively discuss the testing activity itself.

Loch et al. (2001) focus more explicitly on testing, considering when testing of design alternatives should be done in parallel (as per SBCE) allowing quick convergence to a solution, or sequentially, which allows for learning from each test to inform the next in a process of iterative improvement. Their model shows that parallel testing is most useful if the cost of tests is low or the time required to complete each test is significant, and if the tests are effective in revealing information about the designs. Erat and Kavadias (2008) build on this work to show that the similarities of design alternatives being tested impacts the optimal test schedule. Further considering the varying effectiveness of tests, Thomke and Bell (2001) develop an algebraic model to optimise the timing and fidelity of tests in product development, considering that the main role of testing is to create information about design flaws that impact technical requirements or customer needs. They focus on the balance between the cost of adding more tests early in the development process, and the benefit of rework that could be avoided. High-fidelity tests are more expensive and time consuming to run, but also may reveal more information. Among other points, these authors conclude that the overall benefits of frequent testing are increased if the scope of tests can be overlapped such that consecutive tests support one another.

Finally, Qian et al. (2010) develop a model for optimising the amount of testing in each of two overlapped stages in product development. They consider that each stage comprises a development period followed by a testing period in which design flaws are identified and corrected. It is assumed that as soon as the development part of the upstream phase is complete, the design information can be released to the downstream stage. Then, the downstream work can begin concurrently with the upstream testing. However, Qian et al. (2010) point out that it may be appropriate to wait until some testing is complete, because this allows some flaws to be found before they are built into the downstream work, thus reducing rework. A mathematical model is formulated to determine the optimal amount of testing in upstream and downstream phases as well as the amount of overlapping, considering the costs of testing, the costs of correcting errors, and the rate at which testing can reveal design flaws. The rate of finding flaws is considered to decrease exponentially as tests continue.

Overall, the mathematical and simulation models discussed above emphasise the importance of (1) timing; (2) fidelity; (3) amount of testing in the process, and how this depends on various situation-specific factors. However, the models are arguably focused on generating research insight and do not provide support or guidance for specific situations.

3.1.2 Models of testing focused on support for practitioners

Models in this section aim to provide decision support about testing by allowing practitioners to model and analyse their specific situation.

Shabi and Reich (2012) consider testing in the context of system verification activities. They develop a model in which a number of VVT activities are required for a design, derived from the requirements. Each of these activities can be performed by one or more VVT methods, selected from the following: verification by comparison to a similar design; verification by analysis/simulation (i.e., testing in a computational environment); verification by physical test (i.e., testing in laboratory setting); verification by demonstration (i.e., testing in a situation that mimics the context of use); and verification by directly examining the properties, such as weight. Shabi and Reich (2012) formulate an optimisation model to assist in choosing the best combination and number of VVT methods that should be used, considering the constraints of maximum acceptable cost and risk. Overall, the article demonstrates the importance of selecting the appropriate test methods for the particular context and shows how this can be formulated as a decision problem, but does not explicitly consider the timing of the VVT activities or methods. Shabi et al. (2017) extend the approach with a more comprehensive knowledge elicitation procedure, and apply it to a case study in the aerospace industry with promising feedback. Engel and Barad (2003) also formulate VVT planning as a decision problem, placing greater emphasis on eliciting the risks that VVT is intended to avoid, and arguing that the ideal VVT strategy is strongly dependent on the project context (for example, the number of units produced).

Shankar et al. (2016) develop a test planning method focusing on the need to systematically generate and prioritize test activities following design change in a product that can be assembled in different configurations. The key issue is that potential failure modes differ according to the configuration at hand. The method starts with identification of requirements against which the changed design is to be verified, and mapping of these requirements against elements of the design. To identify test activities required to verify the design following change, the element at which the change initiates and others that might be involved in change propagation are identified. All possible assembly configurations for these elements are listed, considering the possible variants of each element and the possible suppliers of each variant. From there, requirements impacted by the change are identified and hence the recommended test activities. The tests are then sequenced in order of priority using an FMEA-style prioritisation number called the Verification Complexity Index.

Luna et al. (2013) focus on the Systems of Systems context and argue that to test an SoS, attention should be focused on the most critical interactions between its constituent systems as well as testing the systems individually. They suggest these critical interactions can be identified by mapping the information flows among systems in a DSM and searching for strongly connected clusters. Furthermore, they argue that once critical interactions are known, a test suite should be carefully designed using Design of Experiments methodology to efficiently address them.

Engel (2010) discusses a compendium of VVT activities and methods, extending the discussion beyond testing in system design and development to testing that is implemented throughout the lifecycle of systems. He discusses VVT Methodology Guidelines—a collection of 41 VVT activities and 31 methods, as well as a VVT Process Model. The process model, described in detail by Hoppe et al. (2003), provides a method for selecting the VVT activities to be optimally undertaken at each process stage (referred to as the VVT strategy). In the method, the costs of VVT activities at each stage are balanced against risk impact avoided, by choosing the best option from a Monte Carlo simulation of alternatives.

Lévárdy and Browning (2005) build on this work to develop a discrete-event simulation model called the Adaptive Test Process (ATP), which is further developed in Lévárdy and Browning (2009). These authors focus on the role of V&V activities in reducing technical uncertainty about the performance parameters of a design. Noting that tests may be performed at different points in a project, and use different modes depending on the information available at the time, they develop a model to optimise the placement of test and design activities within a project. The crux of the model is that each activity should be selected dynamically when the state of the project allows it, with consideration to maximising the value added by the next step.

More recently, Tahera et al. (2017) focuses on the overlapping of testing with downstream design activities. Expanding the well-known model of Krishnan et al. (1997), they develop a simulation using the Applied Signposting Model (Wynn et al. 2006) to investigate the optimal way to organise the overlapping. They conclude that virtual testing should be carried out concurrently with physical testing, so that preliminary results from the latter can be used to calibrate the former, thereby allowing more accurate preliminary results to be released to downstream design.

3.2 Testing in procedural models of design and development processes

Other models have been developed to offer guidance on how the design and development process should be carried out, i.e., to describe standards and supposed best practices (Wynn

and Clarkson 2018). Such models are commonly used in industry and education and hence have significant influence on how processes are perceived and managed. In this section, some key models of this type are considered from the point of view of how they account for testing.

3.2.1 Testing in structured sequential models

Some of the most well-known process models present testing primarily as a verification activity. In other words, all tests would be successful in the ideal project and rework following test failure would be avoided. The models suggest that careful planning, control and documentation of a project's progression can help to ensure this.

For example, in textbooks commonly used in undergraduate education, such as Pahl et al. (2007), stage-based models of the engineering design process convey the recommendation that designers should finalise certain decisions in each stage and aim (as far as possible) to freeze the information on which subsequent stages will build. For example, if concept design is properly completed, detail design can commence and the designer can be assured, at least in principle, that the chosen concept will not need further attention. In many such models—including Pahl et al. (2007), Pugh (1991), and French (1999)—testing does not appear explicitly in the main process model diagram and hence is arguably not conveyed as an integral part of the process, although it is typically mentioned in the text accompanying each model.

Second, zooming out from the engineering design process to include all of product introduction, the stage-gate model depicts a new product process as a series of stages separated by formal gate reviews (Cooper 1990). The model, shown in Fig. 1, aims to support control of product development by ensuring that a project does not progress from one stage to the next until it has demonstrated viability. Thus, unjustified investment in immature and risky projects can in principle

be avoided. Stage-gate models conventionally depict testing at the end of the process, just before sign-off. Testing may also be required to satisfy the gate review criteria, although this is not explicitly depicted. The general form and principles of the stage-gate model have been adopted and adapted by many companies (Unger and Eppinger 2011). The model is focused on project control and is not intended to provide detailed insight into how testing should be performed.

Third, the Vee model (Forsberg et al. 2005) is widely known in systems engineering practice. The model, reproduced in Fig. 2, emphasises the structured decomposition of a system, development of the subsystems, and testing in the context of V&V. The different levels of development and testing complement one another across the Vee. Therefore, for example, subsystems are developed individually and tested against their individual requirements, then integrated, and the integrated system is tested to verify it against technical requirements, and finally validated against customer need. This model graphically suggests test planning is done concurrently with development activities, while testing itself takes place after the respective development activity at each level.

Overall, it can be noted that structured sequential models graphically present testing after the respective development activities. A detailed representation of testing, its timing and contingencies is outside the intended scope of these models.

3.2.2 Testing in iterative models

While the structured sequential models aim to control projects and treat iteration as undesirable, other models are explicitly based on exploiting iteration (Wynn and Clarkson 2018). These are sometimes known as iterative incremental development (IID) models. In contrast to the structured sequential models discussed above, they more explicitly integrate testing and more explicitly indicate its role in the process.

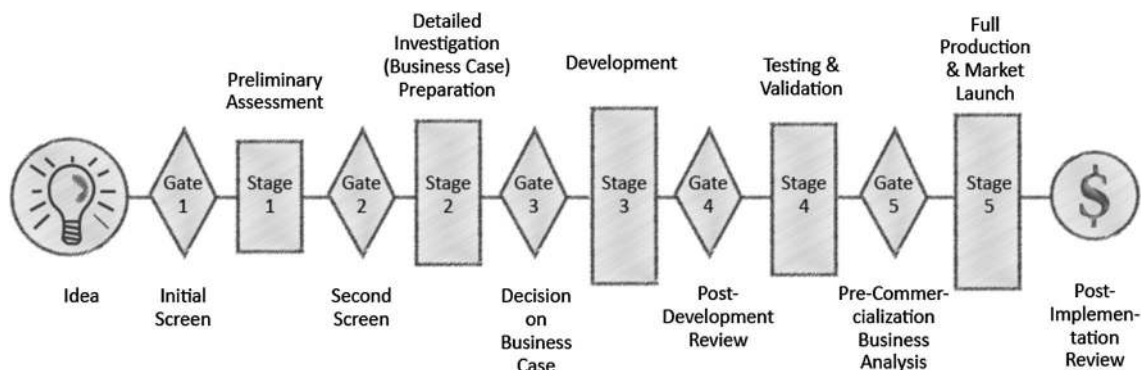


Fig. 1 Stage-gate model of the new product process depicts testing towards the end of the process, after development is complete. Reproduced from Cooper (1990) with permission of Elsevier

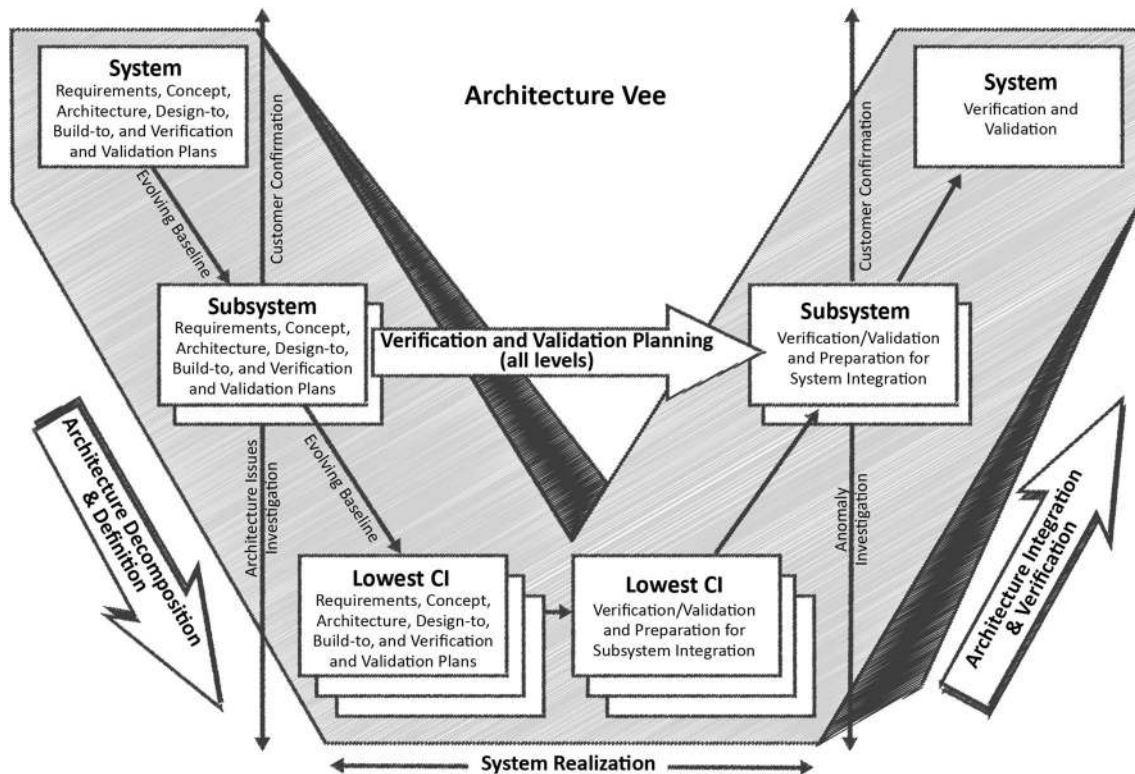


Fig. 2 Vee model of the systems engineering process. Figure reproduced from Forsberg et al. (2005) with permission of Wiley. Copyright ©2005 by John Wiley & Sons, Inc

For example, the Spiral model (Boehm 1988) recommends a repeated sequence of steps, essentially setting objectives for the iteration, development, followed by integration and testing. The number of loops and specific steps can vary from one implementation to another. This model positions integration and testing at the end of each iteration around the spiral. The intention of this spiral model is to manage integration and minimise risks, especially relating to the market, by the repeated creation and test of progressively more mature prototypes (Unger and Eppinger 2011). With each loop, the work in progress is evaluated and suggestions are made for its improvement (Boehm 1988). This model has proven effective in software development but for complex engineering products, several iterations of building prototypes might raise the development cost significantly.

Also incorporating an inherently iterative structure, Agile development process models have been advocated since the 1990s to create more flexibility in product development, with a view to handling changing customer demands (Thomke and Reinertsen 1998). Similar to spiral in its iterative structure, key features of agile models include very rapid iterations and ensuring that customers/users are involved throughout the development process, such that the product can be tried out as it is progressively developed and the developers can make rapid adjustments based on

feedback. Although originating in software, some authors have proposed that essential features of agile models can be useful in product development as well (Suss and Thomson 2012). Cooper (2014) proposes to combine agile principles with stage-gate processes, where each stage comprises an iterative process of building a prototype, testing the product with customers, and revising requirements if necessary. In the model, the emphasis of the test varies between gates. Cooper (2014) argues this model is most appropriate for fast moving highly innovative products, whereas classical stage-gate processes are most well suited to mature and complex products.

3.3 Summary and critique

Researchers have explored various issues relating to testing in engineering design and development, as summarised in Table 2. Section 3.1 discussed a number of mathematical and simulation models that have considered these issues. The main conclusion with respect to this paper is that features of the design context are recognised to be important in determining the ideal testing strategy. Additional research opportunities relating to analytical models of testing will be discussed in Sect. 6, drawing on case study observations.

Table 2 Summary of issues in the literature related to testing, with selected references for further reading

Issue	Selected publications (e.g.)
Relationship to quality and V&V	Shabi et al. (2017); Hoppe et al. (2007); Pineda and Kilicay-Ergin (2010); Engel and Shachar (2006); Shabi and Reich (2012); Engel (2010); Engel and Barad (2003)
Relationship to prototyping	Zorriassatine et al. (2003); Camburn et al. (2017)
Relationship to design complexity	Novak and Eppinger (2001); Thomke and Bell (2001); Kuhn et al. (2008); Pineda and Kilicay-Ergin (2010); Luna et al. (2013)
Relationship to modularity	Loch et al. (2001); Baldwin and Clark (2000); Pineda and Kilicay-Ergin (2010)
Relationship to design novelty	Song and Montoya-Weiss (1998)
Relationship to change propagation	Shankar et al. (2016)
Optimising strategy/timing/integration of testing in the engineering design and development process	Ha and Porteus (1995); Thomke (1998); Ahmadi and Wang (1999); Thomke and Bell (2001); Hoppe et al. (2003); Barad and Engel (2006); Engel and Shachar (2006); Qian et al. (2010); Shabi and Reich (2012); Tahera et al. (2017)

The most well-known macro-level process models of product development, namely the stage-gate, Vee, Spiral and Agile models, do not graphically emphasise the role of testing—and this is not their intention. In particular the following issues discussed in the literature are important to understand testing and its role, but are not all emphasised in any of these models:

- Testing happens throughout a development process, not just after the respective design activities.
- Different types of tests are used at different points for different purposes.
- Test results may lead to redesign or design changes.
- Testing strategy changes according to the design context.

Overall, there is a lack of research literature that comprehensively describes the process of testing during product development and considers how the different aspects appear in an industrial context, and there is no graphical model that depicts the key issues in a simplified manner, that could be used to communicate them in practice and in education. The next sections contribute towards addressing these gaps.

4 Observations on testing in industry practice

The literature review revealed that descriptive empirical study of testing practice in engineering companies has not been a significant focus of prior research, with a few exceptions. One such exception is Thomke (1998)—but this study was published two decades ago when virtual testing was significantly more limited and is set in the domain of microprocessor development, not mechanical engineering design as here. Another exception is Huizinga et al. (2002), who focus fairly narrowly on virtual fatigue testing in one automotive company. A third exception is Engel and Barad (2003) who provide a quantitative study of VVT in an avionics upgrade

project. Their study focuses on eliciting the cost of quality and quantitatively assessing a pilot implementation of the SysTest approach discussed earlier.

Complementing this prior empirical work and seeking additional qualitative insights, case studies were undertaken in three companies. The objectives were to qualitatively investigate testing processes in engineering practice, to identify key issues that manufacturing companies are facing regarding testing and to consider whether these issues are adequately captured in existing models of the design and testing process. Observations from the first two of the three case studies were previously published in two short conference papers (Tahera et al. 2012, 2015); this section summarises those findings and additionally contributes new results from a third case study, prior to drawing on all three cases to synthesise common insights.

4.1 Case study method

Three case studies were undertaken. The first was in a diesel engine design and manufacturing company. Two additional studies in a forklift truck manufacturer and a manufacturer of turbocharging systems for automobile industries were carried out to corroborate the findings. These companies were recruited to the study opportunistically, building on prior contacts of the researchers.

The case studies each involved a series of interviews ranging from 40 to 180 min in duration (Table 4).

Interviewing was led by the first author; the third and fourth authors also attended some interviews in the diesel engine company and one of the interviews in the forklift truck manufacturer. Each interview involved between 1 and 3 interviewees, most of whom participated in several of the sessions. The participants, whose roles are indicated in Table 3, were selected based on suggestions of the study sponsors in each company and considering our objective to investigate the testing processes in those companies. All interviews from the first two companies were recorded and

Table 3 Interview participants in the diesel engine manufacturer (DE) forklift truck manufacturer (FL) and turbocharger manufacturer (TC)

Code	Role
DE1	End-to-End Process Architect
DE2	Verification and Validation Manager
DE3	Validation Team Leader—Tier4 Interim/Final
DE4	Development Engineer
DE5	QFD and FMEA Manager
DE6	Master FMEA Owner and Test Planning
DE7	Business Transformation Manager
DE8	Core Engine Mechanical System Team Leader
FL9	Test and Validation Leader
FL10	Mathematical Modelling and Simulation Engineer
TC11	Design Manager
TC12	Senior Project Engineer
TC13	Lead Engineer—Product Development

Table 4 Summary of individual and group interviews

#	Date	Duration (min)	Participant(s)
1	28 Feb 2011	127	DE1
2	19 May 2011	159	DE2, DE3
3	13 Jul 2011	76	DE2, DE3, DE4
4	13 Jul 2011	65	DE3, DE4
5	23 Aug 2011	135	DE2, DE3, DE5
6	15 Dec 2011	39	DE1, DE2, DE3
7	15 Dec 2011	148	DE1
8	03 Feb 2012	105	DE2, DE3
9	03 Feb 2012	40	DE3
10	27 Mar 2012	82	DE2, DE3
11	27 Mar 2012	75	DE1
12	04 May 2012	142	DE3, DE6
13	17 Sep 2012	119	DE1, DE7
14	16 Jan 2013	52	DE1
15	18 Mar 2013	117	DE1
16	09 Oct 2013	160	DE3, DE8
17	18 Oct 2013	151	DE1
18	21 Feb 2014	86	DE1
19	-	-	FL9
20	02 Jul 2013	75	FL9, FL10
21	09 Sep 2015	40	TC11
22	15 Oct 2015	90	TC11, TC12
23	09 Nov 2015	180	TC11, TC12, TC13
24	18 Nov 2015	120	TC11, TC12

Date and duration of interview #19 cannot be provided due to loss of data

written notes were taken, while interviews in the third company were not recorded due to confidentiality concerns.

- In the diesel engine design and manufacturing company, eighteen semi-structured individual and group interviews were carried out at the company premises from February 2011 to February 2014. Eight engineers including a senior engineer, a development engineer, a business manager, a verification and validation manager and a validation team leader were interviewed. The researcher's understanding of the company context was informed by a previous series of interviews in the same company, focusing on system architecture (Wyatt et al. 2009).
- In the forklift truck design and manufacturing company, two semi-structured interviews were carried out at the company premises with (a) the test and validation leader and (b) a mathematical modelling and simulation engineer. This study was carried out between 2013 and 2014. In addition, several informal discussions took place at the Open University with the mathematical modelling and simulation engineer. These were not recorded and are not shown in Table 4.
- In the company that designs and manufactures turbo-charging systems, four semi-structured interviews involving a senior project engineer, design manager and CAE and validation manager were undertaken in 2015. Some of these interviews took place at the company premises, while others took place at Huddersfield University.

The interview format was semi-structured. Instead of asking a specific set of questions, the areas of interest and enquiry were set down beforehand and used by the interviewer to guide each interview. In particular, interviews in each company began by asking the participants to discuss the relationship between virtual and physical testing and proceeded to investigate areas of interest that were raised. After each interview, the data were analysed and topics for the next interview were identified.

Documents were also provided by the companies. Based on these documents and the information elicited during interviews, the product development process of each company was modelled to pinpoint where testing occurs in the product development processes and to provide a concrete basis for discussions about the role of testing. The process models were refined and developed throughout each interview series, by presenting them to the participants and asking for feedback and clarification.

4.2 Testing and design in the case study companies

This subsection describes the testing and design processes in the three case study companies, prior to synthesising insights common to all three cases.

4.2.1 Testing and design in the diesel engine manufacturing company

The first company offers a range of diesel and gas engines and power packages from 8.2 to 1886 kW and has the capacity to produce up to 800,000 units per year. These engines are used in many off-road applications such as agriculture, construction, material handling, marine, general industrial and electric power. The largest portion of sales is into the European market although sales in the Asian market are rapidly increasing. A key challenge for the company is to comply with new tiers of environmental legislation across all their markets. Over the years, this has led to considerable technological changes accompanied by a significant decrease in product development time. A typical product development project in this company lasts about 18–24 months.

The interviewees described their products as “incrementally improved” designs. They referred to the fact that the functional capability of the diesel engine does not improve dramatically with each new version, but frequent incremental changes in the technology improves the performance and lowers the cost of ownership. Accordingly, two important factors in a new product development process are “newness” of the design and problems identified in existing products. First, in terms of newness, the company measures innovation in terms of the degree of change that happens between products, or equivalently, how much a product varies from previous versions. Newness is increased when new components are introduced, or existing components are used in a different context, e.g., at a higher combustion temperature (Wyatt et al. 2009). For a New Product Introduction (NPI) programme, the company sets newness targets in the range 18–25% and stipulates that the targets must not be exceeded. Second, in terms of problems with existing products, engineers start with an existing analysis of the previous generation of products and a current product issue (CPI) database. The CPI database provides information about failure modes and effects of current products that need special attention for the next generation of products.

Two company processes relevant to this paper are the New Technology Introduction (NTI) process and the New Product Introduction (NPI) process. First, the NTI process takes place as a general research and development exercise in their R&D department or in collaboration with universities before the NPI process starts. Legislation is a major influence on technology development—as one interviewee commented, “*legislation is driving the technology*” (DE1). New technologies, such as new fuel injection techniques or aftertreatment equipment, then need to be integrated with the engine system. Second, when an NPI project is used to develop a specific product, the company uses a stage-gate process. As shown in Fig. 3, this process has seven stages from the identification of market needs through to realising

the product and reviewing its performance in the market (see Tahera (2014) for further discussion). Each stage leads to a formal gate review as indicated in the figure. Ideally, the key testing activities should start only after requirements are finalised, and finish before manufacturing processes are put in place. However, often these activities spread further across the process, as shown in Fig. 3.

In keeping with the focus of the present paper, the following description focuses on the NPI process stages that involve significant testing activities. Complementing Fig. 3, a more detailed flow diagram of these stages showing interactions between testing and the related activities of design and CAE analysis is presented in Fig. 4. As depicted, these activities undergo at least three iterations from stage 2 to stage 4. There are a large number of tests. Some tests are grouped and some are undertaken individually. At each stage, Performance and Emission (P&E) tests start first, followed by mechanical durability tests and reliability testing. Often, durability and reliability tests in particular are time consuming and expensive to run. For example, the company needs to run a gross thermal cycling test for a core component for 2 months continuously, which is a significant cost in fuel alone. The company, therefore, aims to reach a point through analysing information gathered from component level testing and simulations that this reliability test will be successful and no further changes will be required following or during the test. Diesel engines are highly regulated, such that the company cannot sell into particular markets unless the engines meet the applicable regulatory requirements. Therefore, performance testing concerning regulatory requirements also plays a prominent role in this company.

The testing happens at different levels. Component level testing happens primarily at suppliers of components, although the case study company also carries out testing to investigate areas of design concern. Engine level testing involves standalone engines on a test bed. Machine level testing involves engines mounted in a machine or vehicle to reproduce expected conditions of use.

Figure 4 indicates how engine level and machine level testing are mainly conducted in parallel in the three consecutive stages of System/Concept Demonstration (SD), Design Verification (DV) and Product Validation (PV):

- *The System/Concept Demonstration (SD) stage* is primarily concerned with demonstrating that the technology can deliver the required performance. Alternative concepts are analysed and evaluated. Combinations of old and new parts are built into a new product prototype called an MULE. This is tested to verify the performance of new parts. The product specifications evolve during this phase as design decisions are made. As more new parts become available, old parts are replaced in the MULE and testing continues. It is

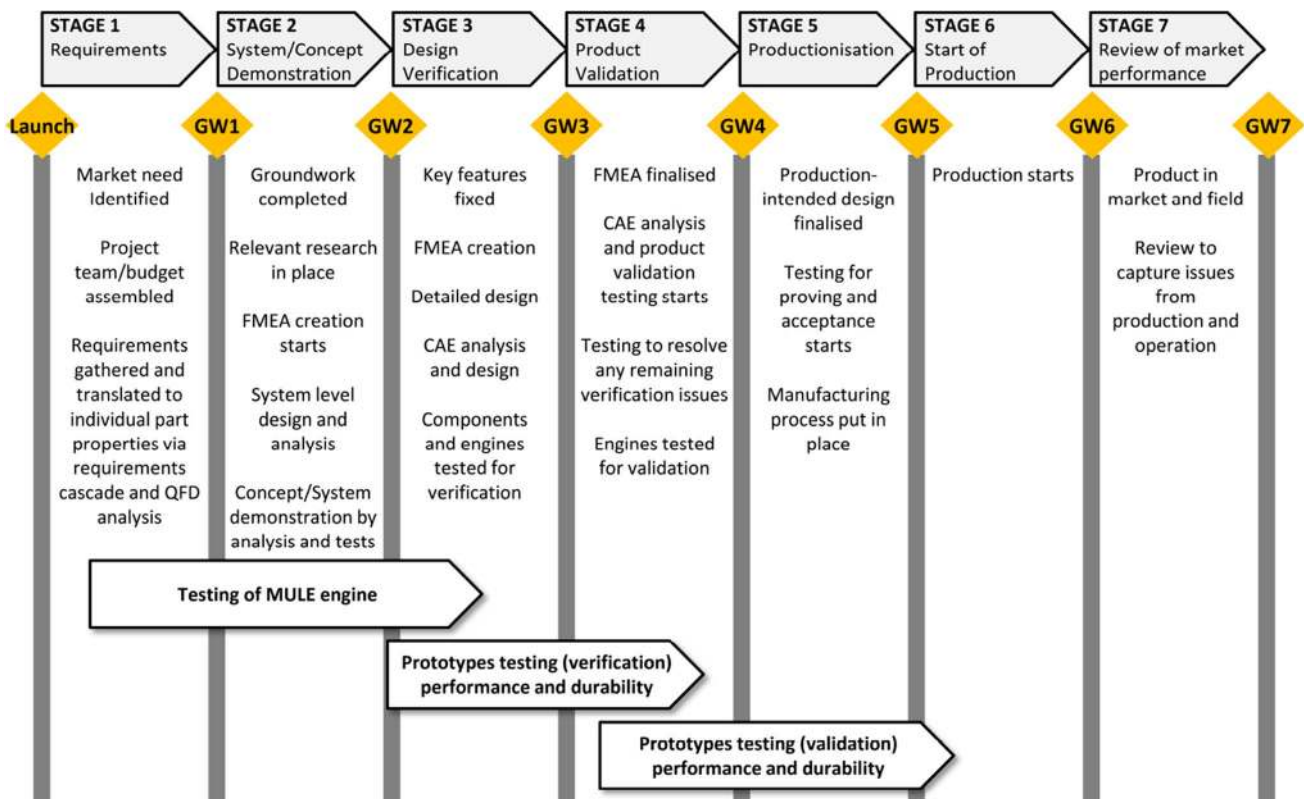


Fig. 3 Depiction of the diesel engine manufacturer's Stage-Gate NPI process. Adapted from Tahera (2014). Key: GW gateway, QFD Quality Function Deployment, FMEA Failure Modes and Effects Analysis, CAE Computer-Aided Engineering

assumed that by GW2, the concept will be selected, the components will be specified and a complete prototype will be built with at least some production parts, and will be ready to be tested for Design Verification (DV).

- *The Design Verification (DV) stage* is primarily intended to help develop optimal performance and verify hardware at the optimised performance. The aim is to ensure that design outputs meet the given requirements under different use conditions. At this stage, testing focuses on the single chosen design, involving analysis and testing of stress, strength, heat transfer and thermodynamics, etc. The design is thereby verified before committing to expensive production tooling.
- *The Product Validation (PV) stage* focuses on the effect of production variability on performance and dealing with any remaining hardware changes. In this stage, hardware testing is limited to late design changes and emissions conformance testing. Comprehensive testing for reliability and durability also occurs and the product is validated. The mandatory tests required for regulatory compliance usually occur during this stage.

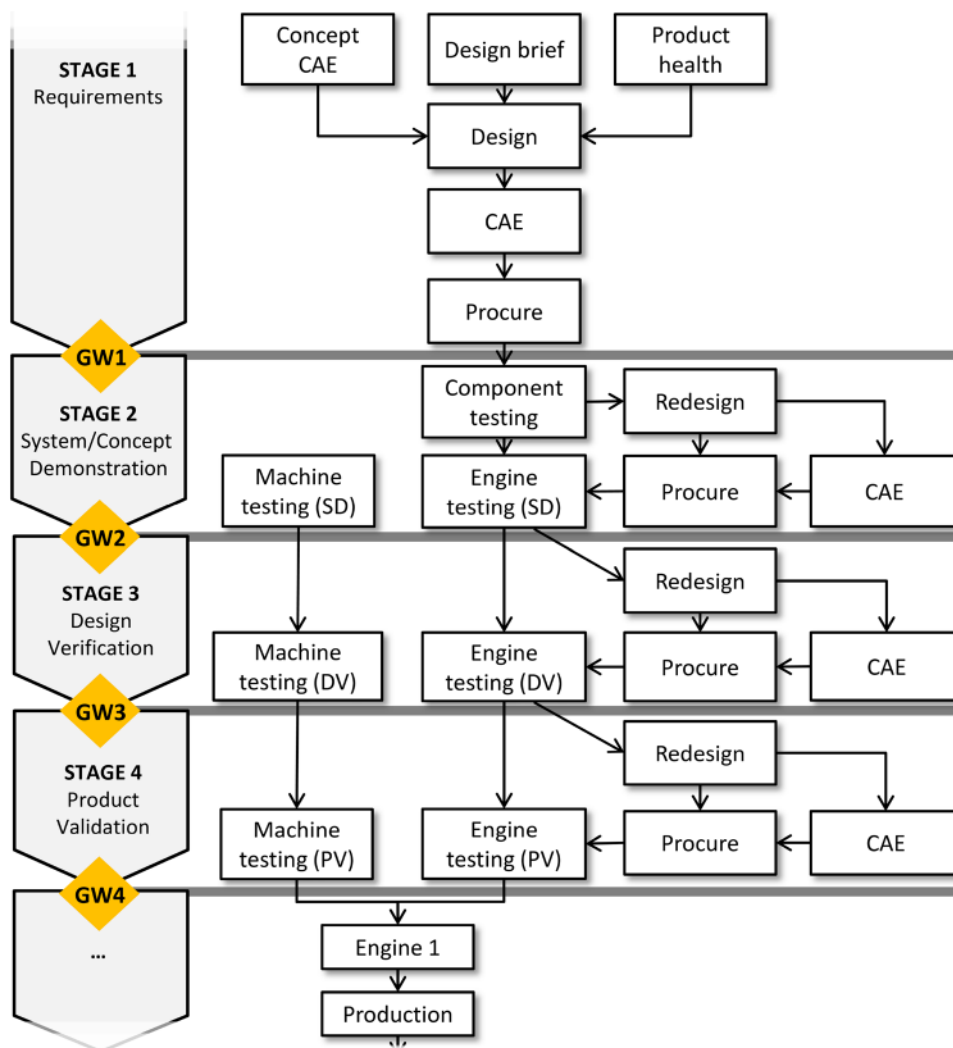
4.2.2 Testing and design in the forklift truck manufacturing company

The forklift trucks designed and manufactured by the second case study company are designed to meet the needs of light to medium duty operating environments for markets in Eastern Europe, Asia-Pacific, Latin America and China. Compared to diesel engines, a forklift truck is a relatively simple product in the sense that there are fewer components and the functionality is more straightforward. A typical product development project in this company lasts about 18 months. Regulations related to safe use and operations of the forklift trucks are important driving factors for new product development.

This company also uses a stage-gate development process, in this case comprised of six stages, as depicted in Fig. 5. It is called a Review Gate Process within the company. Figure 5 shows that like the diesel engine manufacturer, this company starts physical testing early and continues it throughout the development stages.

There are at least three major iterations of prototype testing in a development project in this company. First, the initial concept design is analysed through CAE and simulation and modelling during the Requirements stage (stage

Fig. 4 Flow diagram of testing and related activities in the diesel engine manufacturer, modelled from the company’s actual workflow diagram. Adapted from Tahera et al. (2015). Key: *GW* gateway, *CAE* Computer-Aided Engineering, *SD* System/Concept Demonstration, *DV* Design Verification, *PV* Product Validation



2 in Fig. 5). Second, an MULE truck is produced with a combination of new and old components and physically tested to verify the design. Referred to as Prototype A, this occurs during stage 3 and stage 4, and testing mainly focuses on performance. Third, Prototype B, which is production-tooled, is tested in stage 4 and stage 5 as part of product validation and certification.

This company extensively uses CAE analyses such as structural analysis, hydraulic modelling and simulation during the concept development phases before committing to building prototypes. These analyses are used particularly to explore the design opportunities and for concept selection. Later on, during the design verification and product validation stages, the company largely depends on physical testing.

4.2.3 Testing and design in the turbocharger manufacturing company

Finally, the turbocharger manufacturing company designs and manufactures exhaust gas turbochargers with output

power range of 20–1000 kW. It offers products for passenger cars and commercial vehicles as well as for industrial, locomotive and marine engines. These products are offered worldwide, although the largest proportions of sales are in the US, Canada, Europe, and Asia. Each new product takes approximately 18 months to develop.

The company has developed a product development process called Development and Release of Turbochargers (DaRT). This was launched in 2005 and at the time of the study was being implemented in stages.

DaRT is a stage-gate process similar in many respects to the processes of the other two companies. It has five stages, respectively: Definition, Concept Proof, Functional Proof, Technology Proof and Process Proof (Fig. 6). New concepts are developed and validated through physical and virtual tests in the first two stages of Definition and Concept Proof. The new project is released after the gateway review at the end of Concept Proof. Before the turbocharger is finally released for production, it undergoes a series of tests in stages 3, 4 and 5. These include tests

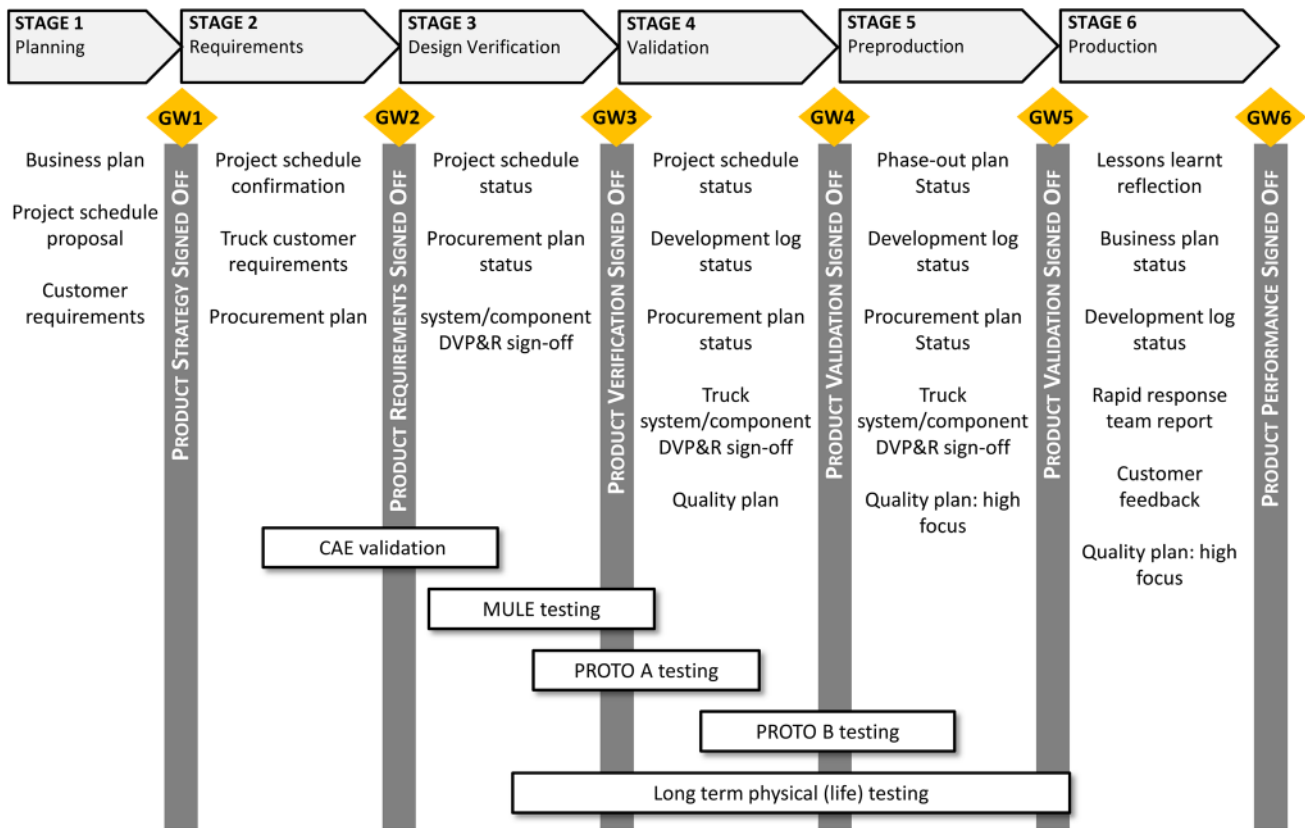


Fig. 5 Testing and design in the forklift truck manufacturer's PD process. Adapted from Tahera et al. (2015). Key: GW Gateway, CAE Computer-Aided Engineering, DVP&R Design Verification Plan and Report

of individual components, tests on the test bench and tests on the vehicle.

In particular, in the Functional Proof stage, first prototypes of turbochargers are designed and built with the combination of existing and newly designed parts and components, and validated through virtual and physical tests. The emphasis of this stage is set on analysing and validating the functional performance of the turbocharger design. A second set of prototypes are designed and built for validating the durability of turbochargers in the Technology Proof stage. At this stage, both virtual and physical testing are required but most validation is done through physical testing. Unlike the other companies' processes, the DaRT process includes a formal mid-stage review gate, which appears in the Technology Proof stage. This intermediary review ensures that any new technology verification and validation is completed, thus allowing mechanical testing for durability (endurance) to begin with sufficient confidence that the technology will work. Finally, in the Process Proof stage, production quality samples of the turbochargers and any new parts and components, if required, are built and validated through further physical tests.

4.3 Insights common to the three case studies

The primary case study in the diesel engine manufacturing company and the two supporting studies in the forklift truck and turbocharger systems manufacturing companies yielded some insights into testing that are common across the cases. These are summarised in the next subsections.

4.3.1 Positioning of testing in the design process

The case study companies all use customised stage-gate processes, although the names and numbers of stages differ. All three companies have at least three key stages, respectively corresponding to concept development, system-level design and detail design, in which iterative design and testing activities are performed. As discussed in Sect. 4.2, testing is used for different purposes in each of these stages such as for learning, demonstration, verification, and validation. Although we did not study the three case study companies as a supply chain, forklift trucks incorporate diesel engines and diesel engines have turbochargers. From the perspective of a new product also involving new subsystems, the integrated

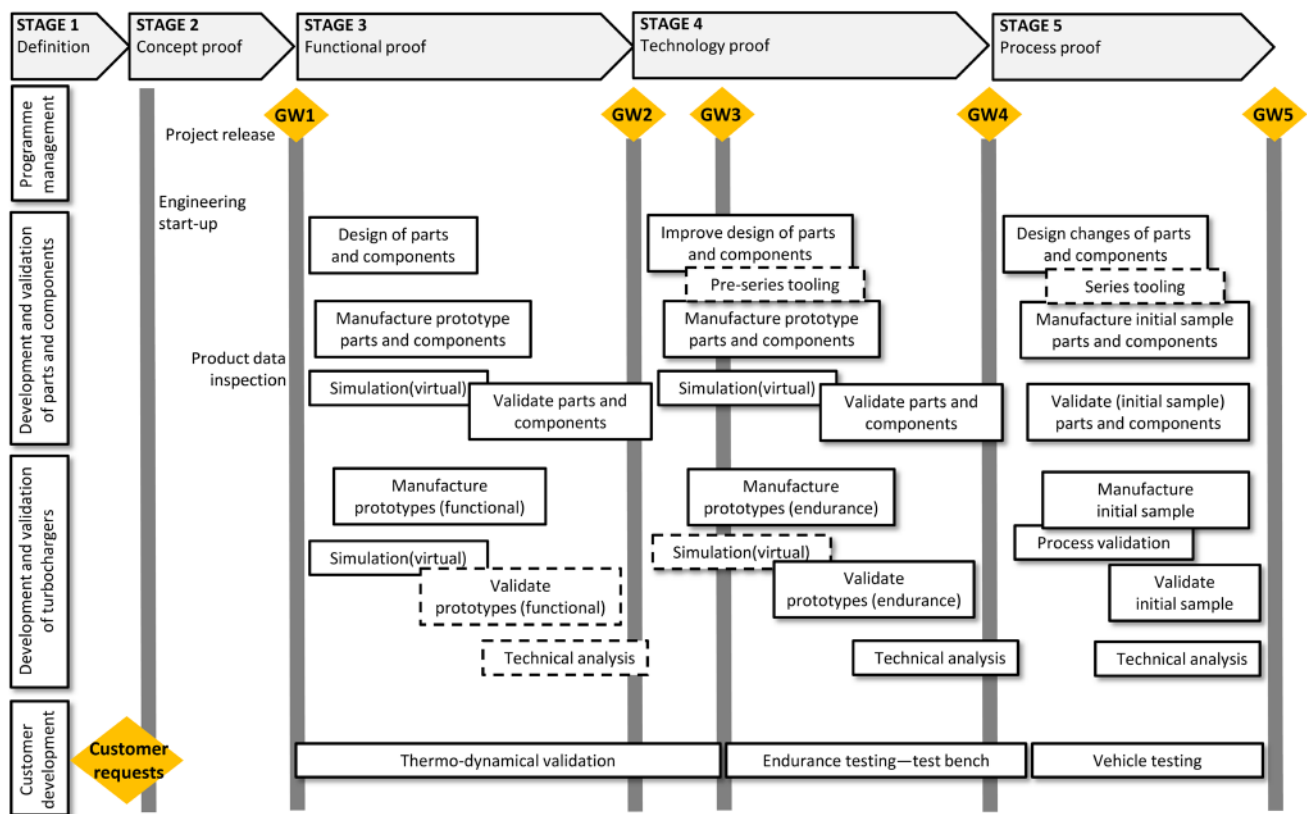


Fig. 6 Testing and design in the turbocharger manufacturing company’s PD process. Key: *GW* Gateway. Solid outlines indicate mandatory activities. Dashed outlines indicate activities that may or may not be required

design and testing processes may, therefore, occur concurrently at different levels of the supply chain.

The case studies all highlighted that testing is a major driver in the engineering product design and development process. Testing does not occur after respective product development activities are complete, as is indicated in the established procedural models reviewed in Sect. 3.2, but is used to assure the quality of design during each stage of the development process. As an interviewee in the turbocharger manufacturing company mentioned:

We don’t simply test quality when everything is finished. We include it in our planning from the very start (TC12).

4.3.2 Managing complexity in testing

Diesel engines are more complex products than forklift trucks and turbochargers, containing a greater number of parts. Despite this, all three companies perform most testing in three key stages. The main difference is that the diesel engine company starts physical testing earlier than the other two companies.

The products produced by these companies are all used in many different applications and need to perform in different environmental conditions. This affects how the products are designed and especially how they are tested. Multiple iterations of physical testing to cover the whole range of applicability and environmental conditions are excessively costly and time consuming. These companies, therefore, carry out physical tests for a small number of the extreme use cases, which bundle multiple adverse extreme conditions into one concrete scenario.

The companies also may choose to carry out early physical testing for a baseline product and use simulations for multiple variations developed for specific use cases. The baseline product is the standard product without adjustments for specific customer specifications. The results from the physical baseline tests are then used to calibrate the simulations. For example, as described by an interviewee in the diesel engine company:

The baseline product definition is physically tested and that information is fairly adequate for the simulation to run for multiple variables for a longer time to find the optimum setup. Then a physical test is required to validate the product as well as simulated results (DE1).

4.3.3 Suppliers and testing

The companies tend to involve suppliers in the process of component testing. Suppliers mostly conduct initial component testing. The case study companies specify certain software and certain processes of verification and validation to the supplier who validates their design against these criteria. For the example of an oil pump to be used in a specific diesel engine, the case study company defines the working and boundary conditions and expects the supplier to perform all the durability analysis and reliability assessment based on those conditions. The supplier's product validation testing significantly reduces the amount of component level testing required in the case study company. Access to the suppliers' testing results and data also brings better understanding of the component or module properties and capabilities.

A new component or part needs to be proven to be compatible not only with the rest of the product but also with the operational and environmental conditions. Any component level testing at the parent company happens when engineers want to investigate a concern relating to one of these issues. For instance, a gross thermal test is performed on a diesel engine to examine the thermal fatigue of a cylinder head and cylinder head gasket; the focus of this test is the cylinder head and cylinder gasket but all the components will be subjected to this system-level test. Such tests focus on the integration and emerging behaviour of components within the system.

4.3.4 Incremental and radical innovation and testing

The case study companies all incrementally improve their products. Nevertheless, radically new subsystems (from the perspectives of the companies) are frequently added to the designs, often due to changes in emissions legislation. For example, development of an off-highway diesel engine added a completely new after-treatment system for Tier4 engines. Major subsystems also go through a periodical redesign when the existing design fails to meet new requirements. In both cases, these companies will follow a standard test plan for product development. However, the emphases of the testing activities are different for radical and incremental subsystems. For a radically new subsystem or component, there can be a significant number of learning, experimentation and demonstration types of tests towards the beginning of the development process. For the incrementally developed subsystems, most of the tests are focused on verification and validation.

The type of product innovation or the degree of product changes influence the types of tests used in the companies, i.e., virtual or physical or a combination of both. As explained by one of the interviewees in the diesel engine company:

Incremental design changes can be managed with standard testing or validated simulations, whereas step changes may require a new test plan. For example, if the company is designing a cylinder block which is a scaled version of a previous product, critically stressed areas would be already known. Thus it might be possible to assess the risk accurately through simulation, whereas a new cylinder block will be physically tested (DE2).

4.3.5 Design changes and testing

The studies confirmed that emerging design changes can lead to retesting and changes in future testing plans. In particular, a change may nullify some of the already-completed tests, may necessitate more testing, and may raise questions regarding whether completed testing was adequate or was performed in the right way. For example, if a component fails to perform according to specification in the concept development, engineers will need to improve the design of that component, while analysing how those changes might affect other components or the performance of the whole product. The validation manager will require testing to be planned both for that particular component and for affected components. Engineers might not necessarily perform the same testing activities as in a previous stage but incorporate new testing parameters. Retesting might happen in a different mode, for instance, CAE analysis might be enough to verify a design change and physical retesting might not always be necessary, as suggested in the example of the previous subsection.

4.3.6 Timing and planning of testing

At each stage, key testing activities are scheduled with a view to the gateway timeline. Some activities might have flexibility; for example, CAE analysis can sometimes finish before schedule as the desired outcome may have already achieved. On the other hand, most physical testing is restricted to planned timeframes. A physical test must run for the time stipulated by the test plan, unless a failure occurs before that. Even if a failure occurs, the failed component may be replaced and the rest of the test will often be continued to learn about other components' behaviours and durabilities. Therefore, in the case of a physical test starting later than planned there is little chance that this test can be shortened. Delay in testing activities in one process stage can thereby affect the gateway schedule and the progress of subsequent stages.

Because companies often share testing facilities across several projects, the planning for the tests and test facilities happens very early in the process. If a testbed is occupied longer than planned, then the next batch of tests is disturbed.

Depending on the priority placed on finishing a particular project, the testing of a new component or a product might occur urgently, displacing other scheduled tests and potentially impacting other projects' schedules.

5 Depicting testing in the incremental development of complex products

The three cases all exemplify engineering products developed incrementally, with medium to high levels of design complexity. In this section, common aspects of their processes are synthesised into a schematic process flow diagram intended to frame how testing is integrated into the development process for incremental and complex products more generally. To complement the process flow diagram and set it in context, the section also introduces a macro-level process depiction that emphasises both iteration and overlap among design and testing activities in product development. The section is a substantial evolution and expansion of preliminary findings earlier published in Tahera et al. (2015), incorporating new insight gained from the expanded literature review and additional empirical work.

5.1 A descriptive model of interactions between design and test

As discussed previously, the three case study companies all use stage-gate processes. Each of the cases presents a unique structure of design and testing activities. Nevertheless, a common pattern was distinguished. A schematic process flow diagram was developed to represent this common

pattern (Fig. 7). The diagram indicates the three common design process stages that were observed, summarised as concept development, system-level design and finally detail design. Although these stages were common, their specific names vary across the cases. For example, the diesel engine company spreads these activities from their stage 2 to stage 4, while the forklift truck and turbocharger manufacturers spread them from stage 3 to stage 5 of their respective processes.

Figure 7 indicates how at each stage, the design is developed further through iterations involving design and analysis using CAE simulation (Flow A in Fig. 7). These CAE analyses enable companies to carry out design optimisation earlier in the product development cycle than would otherwise be the case, and improve the maturity of specifications sent to suppliers (Flow B). Advanced types of CAE, which are referred to as virtual testing, are performed in parallel to the physical tests shown in the bottom part of Fig. 7. These types of CAE complement and assist the physical testing (Flow C). As design and CAE progress, information is released to suppliers allowing procurement of prototypes. This release process occurs progressively (Flow D) with a view to ensuring that suppliers of test prototypes are working with design information which is up-to-date but do not need to wait until the respective design activity is finalised.

Each procured prototype can involve a mix of old and new parts depending on the project progress (Flow E) as explained previously. The prototypes go through two layers of testing at each stage (shown in the bottom part of Fig. 7):

- *Performance testing* assesses how well the emerging product can satisfy requirements including legislative

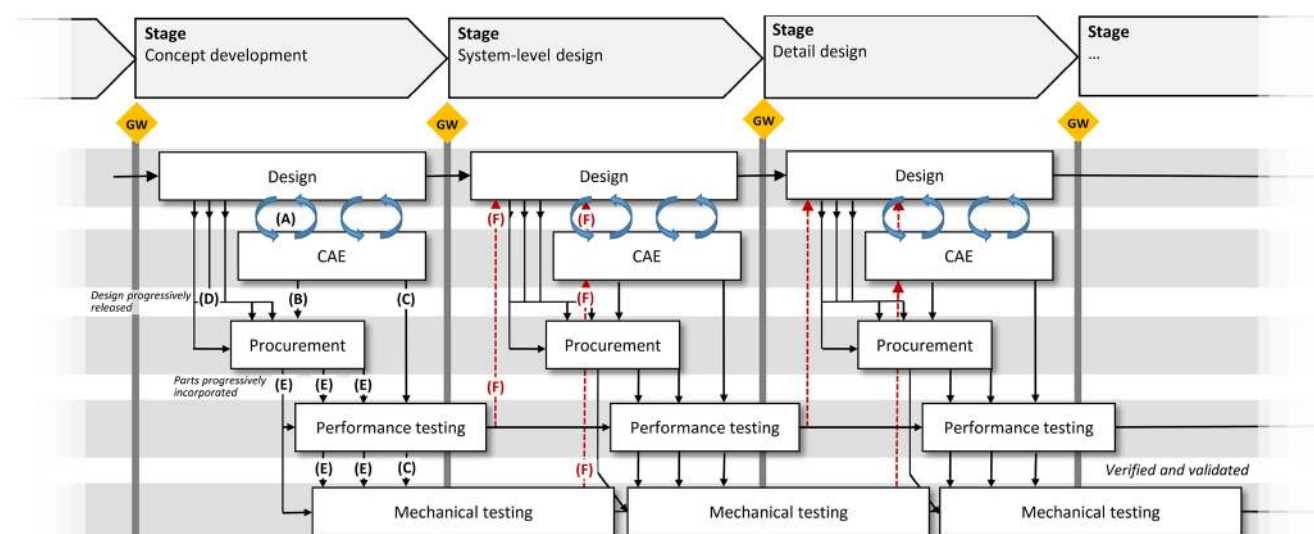


Fig. 7 Process flow diagram of the design and testing activities in the generic stages of concept development, system-level design and detail design. The meanings of flows labelled A–F are discussed in Sect. 5.1

requirements. Performance testing focuses on characteristics such as speed, efficiency, vibration, and noise.

- *Mechanical testing* is mainly undertaken to ensure the reliability and durability of the emerging product. Reliability tests ensure the product's ability to perform the specified function without failing over a period of time. Durability is a particular aspect of reliability—durability tests assure that “the product will work—given proper maintenance, for a given duration”.

As shown in Fig. 7, performance testing starts slightly before mechanical testing and runs almost in parallel. Any design issues identified through these physical tests are dealt with in the next stage of the process (Flow F).

Overall, Fig. 7 indicates that the interactions among design and testing activities are complex and iterative, and this is a significant driving factor of overlapping between development process stages. Other key issues indicated in the model are (1) the importance of both CAE (virtual testing) and physical testing in the design process; (2) the long duration of many physical tests; (3) the overlapping of testing and design. These issues are discussed in further detail below.

5.2 CAE compared to physical testing

It was clear in all three case studies that Computer-Aided Engineering (CAE) and associated computational models and simulations have growing roles in product development. As an interviewee in the diesel engine company commented:

CAE is becoming increasingly important to the companies to minimise the effort and expense involved in product development (DE3).

This is particularly evident in the way that CAE simulations are used as virtual testing to complement and in some instances to replace lengthy and complex physical tests. For example, the modelling and simulation engineer interviewed in the forklift truck manufacturer commented:

There might be 20–30 variations of one product. We can't build and test all of those, we may build three or four variants. If we can validate CAE or FEA against those physical trucks we have built, then we can sign-off the entire range (FL10).

The lead engineer interviewed in the Turbocharger manufacturer stressed that while the virtual tests performed in simulations are taken into account, physical tests are still vital for validation:

Despite today's advanced computer technology and detailed calculation programs, it is testing which finally

decides on the quality of the new aerodynamic components (TC13).

To summarise, CAE analysis and virtual testing have an important role in reducing the number of physical tests as well as increasing the scope of the scenarios that can be covered (see also Becker et al. 2005). Even so, physical tests remain very important in practice. Their long duration, complexity and cost present a significant constraint on how design and test activities can be integrated in product development processes.

5.3 Lengthy physical tests

Two issues were identified that are related to lengthy physical testing. First, the long lead time for procurement and/or manufacture of test equipment and second, the long duration of some physical tests.

In terms of procurement and manufacture, because physical testing needs physical objects, time is required for procuring the parts and building the equipment. For instance, for the gross thermal cycling test mentioned in Sect. 4.2.1, the diesel engine company needs 3 months to procure the components and 2 weeks to build the prototype engine.

In terms of the duration of tests themselves, reliability tests in particular are extremely lengthy, because these tests are designed to predict the lifetime behaviour of the products. For example, the gross thermal cycling test mentioned above is a validation test for determining the thermal fatigue resistance of core components by putting an engine on a test bed for 2 months in a stressed condition. In addition, this test needs to run at least three times for three different extreme use cases. That means three engines are required to run in these three different specifications of this test. These tests can be run in parallel if enough test beds and personnel are available. In this case a final 2 weeks of post-processing is also needed. In total, 6 months are required to complete this test, which is significant relative to a project duration.

Reliability tests such as the gross thermal cycling test are very expensive. Some engineers think they are very valuable and they can learn a lot from them. However, others think that the durations for these tests could be reduced by increasing the extremity of test conditions (a strategy known as accelerated stress testing, see, e.g., Lu et al. 2000 for further discussion) and that this would deliver similar insights. These companies generally perform non-destructive tests and follow the overall procedures and guidance of the business, which involves lengthy physical tests of customer use cases.

5.4 Overlapping between testing and design

Ideally, testing in one stage should be finished before the design of the next stage can start. However, as indicated in Fig. 7, design activities may often start before finishing

the testing of the previous stage. Companies often have no choice but to overlap design tasks with testing, due to the lengthy procurement process. Companies might also overlap lengthy physical testing activities to accelerate the process following delays, or simply to minimise the total duration of testing. At the same time, companies may need to begin procurement and manufacture of test-related items before the corresponding design activities are fully complete. Such overlapping in both directions between testing and design contributes to many uncertainties in the process and can contribute to unplanned delays (Tahera et al. 2017). CAE and virtual testing can play an important role to reduce these uncertainties, especially through the monitoring of emerging test results to assess confidence in using preliminary results for subsequent (re)design. However, in the case study companies, the flow of information from testing to design was not always very clearly defined, so that potential learning from the tests did not always reach the designers when they would most benefit from it.

Overall, the case studies suggest a picture of how concurrent design and test are woven together in product development. Physical tests are often lengthy, complex and expensive and pose severe constraints on this integration. However, virtual tests can help to break down these constraints, thereby supporting overlapping of design and physical tests.

5.5 Observations on macro-level process structure

Some of the most well-established macro-level project structures, namely the stage-gate, spiral and agile models were summarised in Sect. 3.2. In the case studies, all three companies in fact use a hybrid of stage-gate, spiral and agile models that incorporates the advantages of all three. For these incrementally developed engineering products in which quality is critical, the stage-gate system provides structured steps and reviews that may help to avoid design flaws propagating through a project. Within each stage, design–build–test iterations occur to incorporate learning and feedback as in the spiral model. The frequencies of iterations among design and test are higher in the earlier stages of the process than the later stages, permitting experimentation and adaptation to changes before decisions are locked in. Elements of the agile model are incorporated by providing prototypes to the customers to test in their machines at each stage of the process. This helps to perform upfront verification and validation with the customer and reduces the need to have a specific stage for test and validation at the end of the process (as it is often represented in conventional stage-gate process models, e.g., Fig. 1).

Formal gate reviews in the stage-gate processes are used by all the companies for assessment and monitoring. A product must pass prescribed criteria through final gate review

before the project proceeds to the next stage. However, as physical tests often cause delays in completing stages, these gateway reviews tends to be less restrictive by allowing overlapping of activities across gates, and allowing for conditional decisions in addition to go/no-go decisions. The outcomes of these conditional reviews often require additional interim reviews that take place between the formal gates.

5.6 A macro-level depiction of testing in incremental engineering design

As discussed in the previous subsections, Fig. 7 is intended to represent the complex interactions between key activities in the product development process. To complement this perspective and depict in a simplified way some of the macro-level issues discussed above, a second descriptive diagram was developed, as shown in Fig. 8. This diagram is an abstraction from Fig. 7 which zooms out and places it in context. Inspired by the general form of a diagram by Unger and Eppinger (2011), but here refocused on testing, Fig. 8 emphasises concurrency among the processes of designing, analysing, and testing of prototypes. The black thick line represents the progress of the development project that was observed to occur iteratively throughout all the process stages. The tight spirals in the Concept Development stage indicate the more rapid testing iterations that take place at this stage.

As indicated, the foci of testing activities differ in each of the stages. At the Planning stage, initial design and CAE analysis activities are undertaken to assess whether the requirements could be met. Any test at this stage is for assessing technological capability. The focus of Concept Development is on ensuring the practicality of the new concept, and so performance testing concentrates on ensuring that initial concepts are viable. The full mechanical durability and reliability required for production would not typically be achieved at this stage. However, the testing may reveal that the initial design may fail to meet customer requirements, may have technical design faults, or may have potential issues regarding manufacturability and maintainability. Uncovering such issues informs redesign in the next stage. For instance, if testing in Concept Development identifies a failure or mismatch with specification, then in the subsequent System-Level Design stage, engineers undertake redesign to overcome those issues while concurrently addressing the system-level design itself. At the System-Level Design and Detail Design stages, performance and mechanical testing focus on ensuring that the designs will meet specifications under a range of use conditions that are likely to occur in practice, and are reliable. Companies may run these tests in testbeds and/or by putting their products into the customers' machine. The latter may help companies to perform the verification and validation tests with their customer earlier.

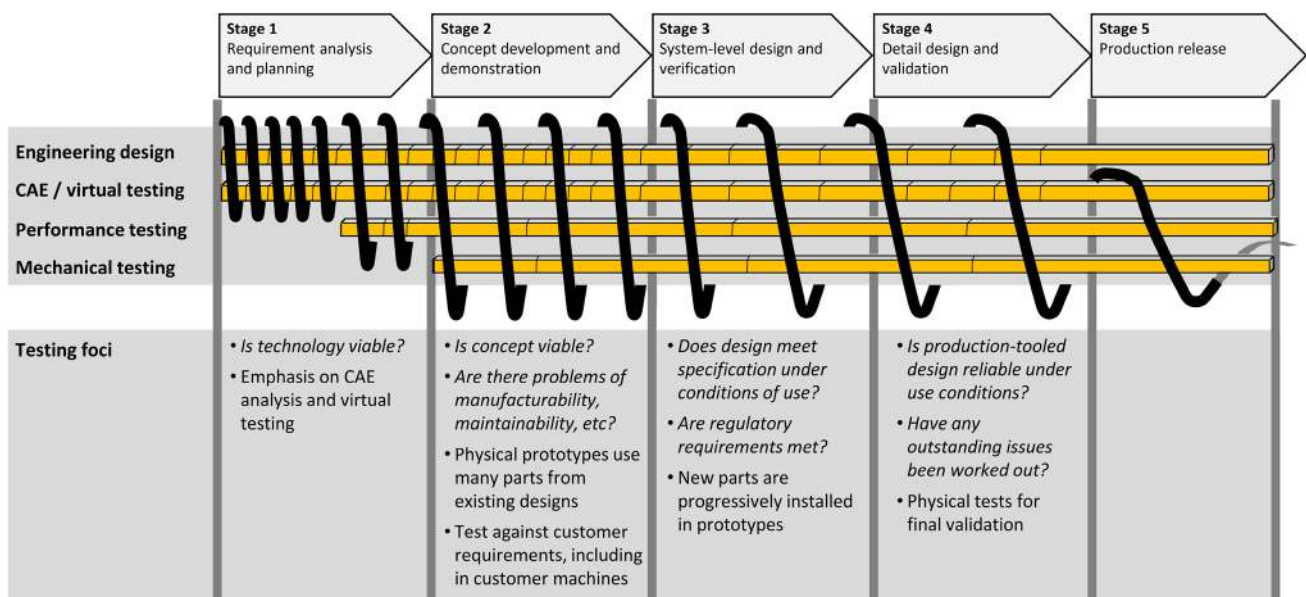


Fig. 8 Macro-level depiction of incremental engineering product development illustrating concurrency and iterations among design, CAE, virtual testing and physical testing within a stage-gate framework

6 Discussion

6.1 Importance of testing as a topic for research

Testing is a significant cost factor in product development. To illustrate, a senior engineer at the diesel engine company indicated the high expenditure that is incurred around testing:

To develop the Tier4 engines can cost R&D alone an excess of over [...] million, I would break it down to design and engineering is probably 15%, the material is probably around 30%, and actually testing around performance is the rest—around 55%. Therefore, most of the money in R&D is goes into testing for performance and durability (DE1).

Physical tests are expensive in their own right, in terms of the resources and time required to carried out the test. The cost of needing to repeat tests is also a significant contributor to the costs of design iteration. Optimising testing and its positioning in the design process is, therefore, potentially a means to releasing resources for other design activities. For all these reasons, we believe testing is an important topic for research.

6.2 Recap of contributions

This paper makes three main contributions. First, we provide a structured overview of the literature on engineering design testing, considering (1) definitions of testing and factors that influence how it is performed; (2) mathematical and

simulation models of testing, and (3) treatments of testing in established macro-level models of the development process. Although this is not an exhaustive review, the key points and publications are covered. This paper thereby collects starting points for further reading on testing in engineering design and development.

Second, noting that few empirical studies of testing practice have been reported in the literature, we provide a description of design and testing practice in three UK-based companies. It was found that testing is closely intertwined with design throughout the development process, and furthermore, that CAE-based virtual testing and physical testing activities both have significant importance. In particular, the cost of testing designs to be used in different applications and varied environmental conditions is reduced by CAE analysis, while physical testing remains the main method of verifying and validating a design. Overall, the observations and literature study provide insight into current practice (summarised in Table 5) and suggest future directions for research into engineering testing, some of which are discussed in the next section.

Third, it was shown that well-known procedural process models of product development, such as those discussed in Section 3.2, do not explicitly emphasise CAE analysis and testing, either because they are below the level of resolution of the models or because they are considered to be supporting activities and not depicted. Because of the importance of testing in product development, we argue there is a need to depict these key activities more explicitly in PD process models. We believe this is needed to emphasise the importance of testing for

Table 5 Summary of key insights gained from literature and case studies

Sect.	Insight(s)
2.2.1	Increase in design complexity increases testing complexity (Novak and Eppinger 2001)
2.2.1	Whole-system testing before deployment may not always be possible (Luna et al. 2013)
2.2.1	Test designs must be optimised when comprehensive testing is not practicable (Thomke and Bell 2001)
2.2.2	Modular architecture may help reduce the costs of testing (Loch et al. 2001)
2.2.2	Commonality reduces number of parts to be tested in a new project (Pineda and Kilicay-Ergin 2010)
2.2.3	Incremental projects allow reuse of test plans, enhancing V&V quality (Pineda and Kilicay-Ergin 2010)
2.2.4	Testing occurs throughout a project using different methods as per context (Shabi and Reich 2012)
2.2.4	High-fidelity tests are not possible until sufficient design information is available (Thomke and Bell 2001)
2.2.4	Significant tests are often planned late in a project for large scale, complex designs (Reinertsen 1997)
2.2.4	Early tests to learn about design space can reduce the likelihood of problems later (Kennedy 2008)
2.2.5	Frequent design changes and requirement changes reduce V&V quality (Pineda and Kilicay-Ergin 2010)
2.2.5	Change propagation must be considered when revising a test plan following a change (Shankar et al. 2016)
3.1.1	Decisions on test frequency should balance benefits of revealing flaws vs. test costs (Ha and Porteus 1995)
3.1.1	Increased delays between design and release of test results drives rework and churn (Yassine et al. 2003)
3.1.1	Parallel testing of alternatives is useful if cost of tests is low or duration of tests is high (Loch et al. 2001)
3.1.1	Frequent testing is most useful if tests support each other in uncovering flaws (Thomke and Bell 2001)
3.1.1	Overlapping testing with downstream design risks wasting effort on flawed design (Qian et al. 2010)
3.1.2	VVT methods should be matched to requirements, considering acceptable risk & cost (Shabi and Reich 2012)
3.1.2	Tests should be focused on critical interactions in a complex system of systems (Luna et al. 2013)
3.1.2	Tests can be prioritised using an FMEA-style prioritisation number (Shankar et al. 2016)
3.3	Macro-level models of design and development processes do not emphasise the role and complexities of testing.
4.3.1	In the case studies, testing occurs throughout the process and is a major driver of the development process
4.3.2	In the case studies, companies bundle multiple extreme adverse scenarios to reduce physical test costs
4.3.2	In the case studies, physical tests are done on baseline cases and used to calibrate virtual tests for other cases
4.3.3	In the case studies, companies use suppliers for most part-level testing, but may investigate specific concerns
4.3.4	In the case studies, “radically new” parts/subsystems involve testing to experiment & demonstrate technology
4.3.4	In the case studies, “incremental” parts/subsystems involve testing focused on V&V
4.3.5	In the case studies, design changes lead to retesting and revising test plans. Retests may be in a different mode
4.3.6	In the case studies, physical testing cannot typically be accelerated in case of delays
4.3.6	In the case studies, physical test facilities are shared across projects so physical tests must be planned early
4.3.6	In the case studies, physical test facilities can be bottlenecks and can cause delays to propagate across projects
5.1	In the case studies, information is progressively released from design for procurement of prototypes for test
5.1	In the case studies, two important types of testing are performance testing and reliability/durability testing
5.2	In the case studies, while virtual testing is important, physical testing is still required for final V&V
5.2	In the case studies, the long durations of physical tests pose significant constraints on design processes
5.3	In the case studies, reliability and durability tests are often the most time consuming
5.6	In the case studies, the frequencies of design–build–test iterations are higher in earlier process stages
5.6	In the case studies, issues identified through physical testing are often addressed by rework in the next stage

practice, research and education. The text and diagrams in Sect. 5 contribute towards this need. These insights are common to the three case studies, which are all representative of incrementally developed complex products, and are thus expected to have a degree of generality in this context. In the future, we hope to undertake further empirical work to refine, assess and further generalise the insights.

6.3 Limitations and future work

It was already mentioned that the insights reported in this paper were developed from interviews and document analysis in only three companies. Therefore, the generality cannot be proven. The case studies were also all undertaken in a specific domain, namely incremental design in the automotive sector with a focus on mechanical engineering issues.

Thus, it is expected that the insights will not all be applicable in other engineering design contexts. For example, in very large-scale projects such as aircraft design or system-of-system development, system-level testing from the beginning of the process (e.g., using the equivalent of the MULE trucks/engines discussed earlier) is typically not possible. The insights presented here would need to be extended to cover this and other differences, and further empirical work is needed to do so.

Additional future research directions may be suggested. First, there is a need to develop prescriptive process models to convey best practices relating to testing and its integration with the design process. The literature study of Sect. 3 revealed that testing practice depends (or should depend) on features of the design context, and this should be reflected in such models. Second, comparison of the empirical insights with the mathematical and simulation models of testing reviewed in Sect. 3.1 also reveals some gaps that could be explored using such models, and accordingly some opportunities for research. One such gap is that the reviewed models all assume that there is sufficient upfront understanding of testing tasks that can be used. They do not explicitly consider the situation in which testing methods must be developed during a project. In addition, the case studies revealed that many tests are preplanned and then largely followed as planned. Tests are even carried out when there is an expectation that the design would be changed, in the hope that learning can be incurred through testing. However, the benefit of this learning was not quantified in the case study companies. Further research could develop pragmatic models with a view to better inform companies on the costs and benefits of testing, considering what tests to use, how to integrate them effectively through the stages of the design process, and how to manage preliminary information passing in both directions between testing and design.

7 Concluding remarks

Testing is an important issue in engineering product development, required for several purposes ranging from assessing technology capability to verifying and validating the design. It is closely connected to other important issues in the design and development process, such as activity overlapping and iteration, but has received significantly less attention in research literature to date.

Common macro-level process models of design and development do not emphasise testing. Meso-level models deal with issues including overlapping testing and (re) design, but these models are relatively abstract, each focusing on selected issues, and do not take account many of the complexities and constraints of practice as revealed in the case studies. Noting these shortcomings in both levels of

model, the study of industry cases and of existing models has been used to synthesise a new descriptive model that emphasises the role of testing in the incremental design context. The generalised model is summarised in text and as two complementary diagrams.

Overall, it was identified that the testing process is closely intertwined with design activities and is an integral part of the product development process. Because testing is often a significant cost factor in product development, we contend that a perspective on the development process which emphasises testing could reveal many opportunities for research and improvement.

Acknowledgements The authors wish to thank all participants in the three case studies. We also thank the reviewers and Editor for their feedback on the manuscript. Some content in this paper was evolved from preliminary findings previously reported in two conference papers, Tahera et al. (2012) and Tahera et al. (2015), as explained in the text. Figure 1 is reprinted from Business Horizons, Vol. 33, RG Cooper, Stage-gate systems: a new tool for managing new products, Pages 44–54, Copyright 1990, with permission from Elsevier.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

- Ahmadi R, Wang R (1999) Managing development risk in product design processes. *Oper Res* 47(2):235–246
- Balci O (1998) Verification, validation and testing. In: Banks J (ed) *Handbook of simulation: principles, methodology, advances, applications and practice*. Wiley, New York, pp 335–396
- Baldwin CY, Clark KB (2000) *Design rules: the power of modularity*, vol 1. MIT Press, Boston
- Barad M, Engel A (2006) Optimizing VVT strategies: a decomposition approach. *J Oper Res Soc* 57(8):965–974
- Becker MC, Salvatore P, Zirpoli F (2005) The impact of virtual simulation tools on problem-solving and new product development organization. *Res Policy* 34(9):1305–1321
- Bertolino A (2007) Software testing research: achievements, challenges, dreams. In: 2007 future of software engineering, IEEE Computer Society, pp 85–103
- Boehm BW (1988) A spiral model of software development and enhancement. *Computer* 21(5):61–72
- Camburn B, Viswanathan V, Linsey J, Anderson D, Jensen D, Crawford R, Otto K, Wood K (2017) Design prototyping methods: state of the art in strategies, techniques, and guidelines. *Des Sci* 3:e13. <https://doi.org/10.1017/dsj.2017.10>
- Clarkson PJ, Simons C, Eckert CM (2004) Predicting change propagation in complex design. *J Mech Des* 126(5):788–797
- Cooper RG (1990) Stage-gate systems: a new tool for managing new products. *Bus Horiz* 33(3):44–54
- Cooper RG (2014) What's next?: after stage-gate. *Res Technol Manag* 57(1):20–31

- Eckert CM, Zanker W, Clarkson PJ (2004) Change and customisation in complex engineering domains. *Res Eng Des* 15(1):1–21
- Engel A (2010) Verification, validation and testing of engineered systems. Wiley, Hoboken
- Engel A, Barad M (2003) A methodology for modeling VVT risks and costs. *Syst Eng* 6(3):135–151
- Engel A, Shachar S (2006) Measuring and optimizing systems' quality costs and project duration. *Syst Eng* 9(3):259–280
- Erat S, Kavadias S (2008) Sequential testing of product designs: implications for learning. *Manag Sci* 54(5):956–968
- Forsberg K, Mooz H, Cotterman H (2005) Visualizing project management: models and frameworks for mastering complex systems, 3rd edn. Wiley, Hoboken
- French MJ (1999) Conceptual design for engineers, 3rd edn. Springer, London
- Ha A, Porteus E (1995) Optimal timing of reviews in concurrent design for manufacturability. *Manag Sci* 41(9):1431–1447
- Hoppe M, Lévárdy V, Vollerthun A, Wenzel S (2003) Interfacing a verification, validation, and testing process model with product development methods. *INCOSE Int Symp Wiley Online Lib* 13:188–205
- Hoppe M, Engel A, Shachar S (2007) SysTest: improving the verification, validation, and testing process—assessing six industrial pilot projects. *Syst Eng* 10(4):323–347
- Hoyle D (2009) ISO9000 quality systems handbook, 6th edn. Routledge, Abingdon
- Huizinga F, Ostaijen RV, Slingeland AVO (2002) A practical approach to virtual testing in automotive engineering. *J Eng Des* 13(1):33–47
- IEEE (1990) IEEE standard glossary of software engineering terminology, IEEE Std 610.12-1990. Tech. rep., The Institute of Electrical and Electronics Engineers, New York, NY
- Jones H (2010) Integrated systems testing of spacecraft, SAE technical paper 2007-01-3144. Tech. rep., SAE. International. <https://doi.org/10.4271/2007-01-3144>
- Kennedy M (2008) Ready, set, dominate: implement Toyota's set-based learning for developing products and nobody can catch you. Oaklea Press, Richmond
- Krishnan V, Eppinger SD, Whitney DE (1997) A model-based framework to overlap product development activities. *Manag Sci* 43(4):437–451
- Kuhn R, Lei Y, Kacker R (2008) Practical combinatorial testing: beyond pairwise. *IT Profess* 10(3):19–23
- Lévárdy V, Browning TR (2009) An adaptive process model to support product development project management. *IEEE Trans Eng Manag* 56(4):600–620
- Lévárdy V, Browning TR (2005) Adaptive test process—designing a project plan that adapts to the state of a project. *INCOSE 2005*
- Lévárdy V, Hoppe M, Browning TR (2004) Adaptive test process: an integrated modeling approach for test and design activities in the product development process. In: ASME 2004 international design engineering technical conferences and computers and information in engineering conference, Salt Lake City, Utah, USA, Sep 28–Oct 2, American Society of Mechanical Engineers, vol 3a, pp 241–250
- Loch CH, Terwiesch C, Thomke S (2001) Parallel and sequential flexing of design alternatives. *Manag Sci* 47(5):663–678
- Lu Y, Loh HT, Brombacher AC, den Ouden E (2000) Accelerated stress testing in a time-driven product development process. *Int J Prod Econ* 67(1):17–26
- Luna S, Lopes A, Tao HYS, Zapata F, Pineda R (2013) Integration, verification, validation, test, and evaluation (IVVT&E) framework for system of systems (SoS). *Procedia Comput Sci* 20(Supplement C):298–305
- Menold J, Simpson TW, Jablolkow K (2018) The prototype for x framework: exploring the effects of a structured prototyping framework on functional prototypes. *Res Eng Des*. <https://doi.org/10.1007/s00163-018-0289-4>
- Novak S, Eppinger SD (2001) Sourcing by design: product complexity and the supply chain. *Manag Sci* 47(1):189–204
- O'Connor P (2001) Test engineering: a concise guide to cost-effective design, development and manufacture. Wiley, Chichester
- Pahl G, Beitz W, Feldhusen J, Grote K (2007) Engineering design: a systematic approach, 3rd edn. Springer, London
- Pineda RL, Kilicay-Ergin N (2010) System verification, validation, and testing. In: Kamrani AK, Azimi M (eds) Systems engineering tools and methods. CRC Press, Boca Raton, pp 81–109
- Pugh S (1991) Total design: integrated methods for successful product engineering. Addison-Wesley, Boston
- Qian Y, Xie M, Goh TN, Lin J (2010) Optimal testing strategies in overlapped design process. *Eur J Oper Res* 206(1):131–143
- Reinertsen D (1997) Managing the design factory: a product developer's toolkit. The Free Press, New York
- Shabi J, Reich Y (2012) Developing an analytical model for planning systems verification, validation and testing processes. *Adv Eng Inform* 26(2):429–438
- Shabi J, Reich Y, Diamant R (2017) Planning the verification, validation, and testing process: a case study demonstrating a decision support model. *J Eng Des* 28(3):171–204
- Shankar P, Summers JD, Phelan K (2016) A verification and validation planning method to address change propagation effects in engineering design and manufacturing. *Concurr Eng* 25(2):151–162
- Song XM, Montoya-Weiss MM (1998) Critical development activities for really new versus incremental products. *J Prod Innov Manag* 15(2):124–135
- Sosa ME, Eppinger SD, Rowles CM (2003) Identifying modular and integrative systems and their impact on design team interactions. *ASME J Mech Des* 125(2):240–252
- Suss S, Thomson V (2012) Optimal design processes under uncertainty and reciprocal dependency. *J Eng Des* 23(10–11):826–848
- Tahera K (2014) The role of testing in engineering product development processes. PhD Thesis, The Open University
- Tahera K, Earl C, Eckert C (2017) A method for improving overlapping of testing and design. *IEEE Trans Eng Manag* 64(2):179–192
- Tahera K, Earl C, Eckert C (2012) The role of testing in the engineering product development process. In: Horváth I, Rusák Z, Albers A, Behrendt M (eds) Proceedings of TMCE 2012, Karlsruhe, Germany, May 7–11, pp 893–904
- Tahera K, Eckert C, Earl C (2015) Highlighting the importance of testing in the product development process. In: Weber C, Husung S, Cantamessa M, Cascini G, Marjanović D, Venkataraman S (eds) Proceedings of the 20th international conference on engineering design (ICED 15), Milan, Italy, Jul 27–30, Design Society, vol 2, pp 441–450
- Thomke S (1998) Managing experimentation in the design of new products. *Manag Sci* 44(6):743–762
- Thomke S, Bell DE (2001) Sequential testing in product development. *Manag Sci* 47(2):308–323
- Thomke S, Reinertsen D (1998) Agile product development: managing development flexibility in uncertain environments. *Calif Manag Rev* 41(1):8–30
- Ulrich K (1995) The role of product architecture in the manufacturing firm. *Res Policy* 24(3):419–440
- Unger D, Eppinger SD (2011) Improving product development process design: a method for managing information flows, risks, and iterations. *J Eng Des* 22(10):689–699
- Ward A, Liker J, Cristiano JDS II (1995) The second Toyota paradox: how delaying decisions can make better cars faster. *Sloan Manag Rev* 36(3):43–61

- Wyatt DF, Eckert CM, Clarkson PJ (2009) Design of product architectures in incrementally developed complex products. In: Norell Bergendahl M, Grimheden M, Leifer L, Skogstad P, Lindemann U (eds) DS 58-4: Proceedings of ICED 09, the 17th international conference on engineering design, Palo Alto, CA, USA, Aug 24–27, vol 4, pp 167–178
- Wynn DC, Eckert CM, Clarkson PJ (2006) Applied Signposting: a modeling framework to support design process improvement. In: ASME 2006 international design engineering technical conferences and computers and information in engineering conference, Philadelphia, Pennsylvania, USA, Sep 10–13, American Society of Mechanical Engineers, vol 4a, pp 553–562
- Wynn DC, Clarkson PJ (2018) Process models in design and development. *Res Eng Design* 29(2):161–202
- Yassine A, Joglekar N, Braha D, Eppinger SD, Whitney D (2003) Information hiding in product development: the design Churn effect. *Res Eng Des* 14(3):145–161
- Zorriassatine F, Wykes C, Parkin R, Gindy N (2003) A survey of virtual prototyping techniques for mechanical product development. *Proc Ins Mech Eng Part B J Eng Manuf* 217(4):513–530

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.