The impact of product complexity on ramp-up performance

The Impact of Product Complexity on Ramp-Up Performance

Andreas Pufall, Jan C. Fransoo, Ad de Jong, Ton de Kok

Beta Working Paper series 389

BETA publicatie	WP 389 (working paper)
ISBN	
ISSN	
NUR	804
Eindhoven	June 2012

The Impact of Product Complexity on Ramp-Up Performance

Andreas Pufall[1,2], Jan C. Fransoo[2], Ad de Jong[2], A. G. (Ton) de Kok[2]

andreas.pufall@nokia.com, j.c.fransoo@tue.nl, a.d.jong@tue.nl, a.g.d.kok@tue.nl

1 Nokia GmbH
Product Creation Center Ulm
P.O. Box 2907
DE-89081 Ulm
Germany

2 Technische Universiteit Eindhoven
School of Industrial Engineering
P.O. Box 513
NL-5600 MB Eindhoven
Netherlands

Date: 21-Jun-2012

The Impact of Product Complexity on Ramp-Up Performance

Abstract

Fast product ramp-ups are crucial in consumer electronics because short product lifecycles prevail and profit margins diminish rapidly over time. Yet many companies fail to meet their volume, cost and quality targets and the ramp-up phase remains largely unexplored in new product and supply chain management research. This study identifies the key product characteristics that affect ramp-up performance using operational data from the cell phone industry. We investigate three research questions: (1) How to measure software and hardware complexity characteristics of consumer electronics products – and specifically cell phones? (2) To what extent drive product complexity characteristics manufacturing performance? and (3), in turn, to what extent drive manufacturing performance and complexity characteristics ramp up performance? The findings contribute to operations management literature in three ways: First, our model reflects the growing importance of software characteristics in driving hardware complexity, an aspect that prior empirical ramp-up studies have not yet addressed. Second, specific hardware and software complexity characteristics (i.e., component count, parts coupling and SW code size) primarily drive the performance of the manufacturing system in terms of final yield and effective capacity. And finally, effective capacity together with the novelty aspects of both software and hardware complexity (i.e., SW novelty and product novelty) are the key determinants of ramp-up performance.

Keywords

New product development, production ramp-up, product complexity

1 Introduction

New product development (NPD) is particularly challenging in the high-technology sector, increasingly characterized by shortening product lifecycles, rising market fragmentation, and rapid technological changes (Bowersox et al. 1999, Mallick and Schroeder 2005, Wildemann 2007). If firms want to succeed in this environment, they must be effective and efficient in their introduction of new products or product updates. According to Bowersox et al. (1999), new product introductions involve two major activities: product development (conceptualization, design, promotion, and pricing) and product launch (physical positioning in the market). Traditionally, the marketing literature has addressed positioning decisions (Cooper and Kleinschmidt 1995, Bowersox et al. 1999, Benedetto 1999) whereas the operations management literature has considered supply chain decisions (Clark and Fujimoto 1991, Tatikonda and Montoya-Weiss 2001).

In this study, we adopt an operations management perspective and focus on the final phase of the NPD process, namely, the ramp-up phase. This phase links product development to mass production, or as Wheelwright and Clark (1992, p. 8) detail: “In ramp-up the firm starts commercial production at a relatively low level of volume; as the organization develops confidence in its (and its suppliers’) abilities to execute production consistently and marketing’s abilities to sell the product, the volume increases. At the conclusion of the ramp-up phase, the production system has achieved its target levels of volume, cost and quality.” Yet many companies fail to meet their targets regarding product volume, cost, and quality. Schuh et al. (2005) show that 47% of automotive new product ramp-ups were neither technically nor economically successful. Kuhn et al. (2002) indicate that not a single company in their study claimed its production ramp-up was under control. Thus, the ramp-up phase remains a major challenge, even as it provides a significant opportunity for competitive advantages. In addition, the complex relationships that constitute the ramp-up phase have been investigated only partially and insufficiently developed (Kuhn et al. 2002). Therefore, there is strong motivation to gain a more thorough understanding of the influential factors that affect the ramp-up phase and how they relate to success or failure. Previous studies have identified several characteristics that affect ramp-up performance. Clark and Fujimoto’s (1991) global field study in the automotive industry reveals that the transition management between new and existing products (ramp-up scenario), the rate of production in terms of line speed, the number of products in the line and the operation time per day (operational pattern) as well as the manufacturing capabilities relate closely to superior product development and ramp-up performance. In addition, Pisano and Wheelwright (1995) reinforce the link between manufacturing process innovation, productive product launches and enhanced product functionalities. In their large German case study, Kuhn et al. (2002) confirm the importance of manufacturing and logistics capabilities and further identify the product, organization, cooperation and the tools used as crucial factors for ramp-up success. To analyze the types and sources of disturbances that affect manufacturing start-up phases, Almgren (2000) categorizes the different sources into four groups: product architecture, material flow, production technology, and work organization. A longitudinal study in the data storage industry also reveals organizational patterns and suggests that previous ramp-up experience, such as through product platforms, influences the ramp-up of new products (Terwiesch et al. 1999). Langowitz (1987) observes that the success of ramp-ups depends on the management of the development process and how well the requirements of the new product and factory capabilities fit together. Finally, van der Merwe (2004) proposes a conceptual model that supports the association between different types of novelty (product, personnel, supplier, and process) and learning types that drive ramp-up performance.

Despite these multiple studies that have identified a vast number of influential factors, we know very little about the quantitative and causal relationships between these factors and ramp-up performance. Krishnan and Ulrich (2001) argue that essentially no work has investigated the relationship between the rate of production ramp-up and product design decisions. In response, we use operational data to develop new quantitative measures for these characteristics and extend the current understanding of product attributes by including software-related elements to investigate three general research questions: (1) How to measure software and hardware complexity characteristics of consumer electronics products – and specifically cell phones? (2) To what extent drive product complexity characteristics manufacturing performance? and (3), in turn, to what extent drive manufacturing performance and complexity characteristics ramp up performance?

The remainder of the paper is organized as follows: in the next section, we provide an overview of the research domain and study environment. Subsequently, we present our conceptual model in section 3. After the formal presentation of our Hypotheses in section 4 and the illustration of our data and methodology in section 5, we present our results. In section 7, we discuss our results and provide managerial insights and conclude with some limitations of our study and implications for further research in section 8.

2 Research Setting

The unit of analysis for our study is a single cell phone, developed, manufactured, and sold by Nokia Corporation, a leading supplier of a wide range of cell phones, services, and software. Headquartered in Espoo, Finland, Nokia employs approximately 120,000 employees, maintains an R&D presence in 10 countries, and currently runs 9 cell phone manufacturing locations worldwide. Typically, one of its R&D centers develops new products, according to the fit between the individual capability of that center and existing product requirements, such as customer proximity, estimated sales volume, or innovativeness. After the center selection process, a cross-functional project management team of project managers from R&D, product marketing, sourcing, product validation, logistics, and manufacturing gets allocated to the project. Cell phone projects generally follow a highly structured, milestone-controlled development process: after the planning and concept development phase, design activities proceed until the point that physical prototypes can be manufactured. These prototypes emerge from a special production line that has equipment and other characteristics resembling those of the ramp-up line. The purpose of this procedure is to prove product functionality and to verify/fine-tune the production process - still in a controlled manufacturing environment. After the product has proven its functionality through several validation procedures and after the material supply has been guaranteed, the commercialization phase begins. In this phase, the new product gets introduced into the lead factory and the surrounding distribution network. It is typically called the ramp-up phase and it is defined within Nokia as the time between the project milestone “start of production” and the moment when production output switches from a push plan (based on sales estimates) to a pull plan that is consumer demand driven. In contrast with many other companies and industries, Nokia applies a “shut down” approach to the operational pattern, such that old products get completely ramped down before a new product is ramped up on an existing, converted manufacturing line.

3 Conceptual Model

Our conceptual model as shown in Figure 1 suggests that manufacturing and ramp-up performance depend on the level of product complexity, which we define in terms of software and hardware complexity. While our conceptualization of complexity is consistent with existing literature, it represents a refinement because it reflects the growing importance of software in a product development and ramp-up context, a topic that was generally ignored in prior empirical ramp-up studies (e.g., Langowitz 1987, Almgren 2000, van der Merwe 2004).

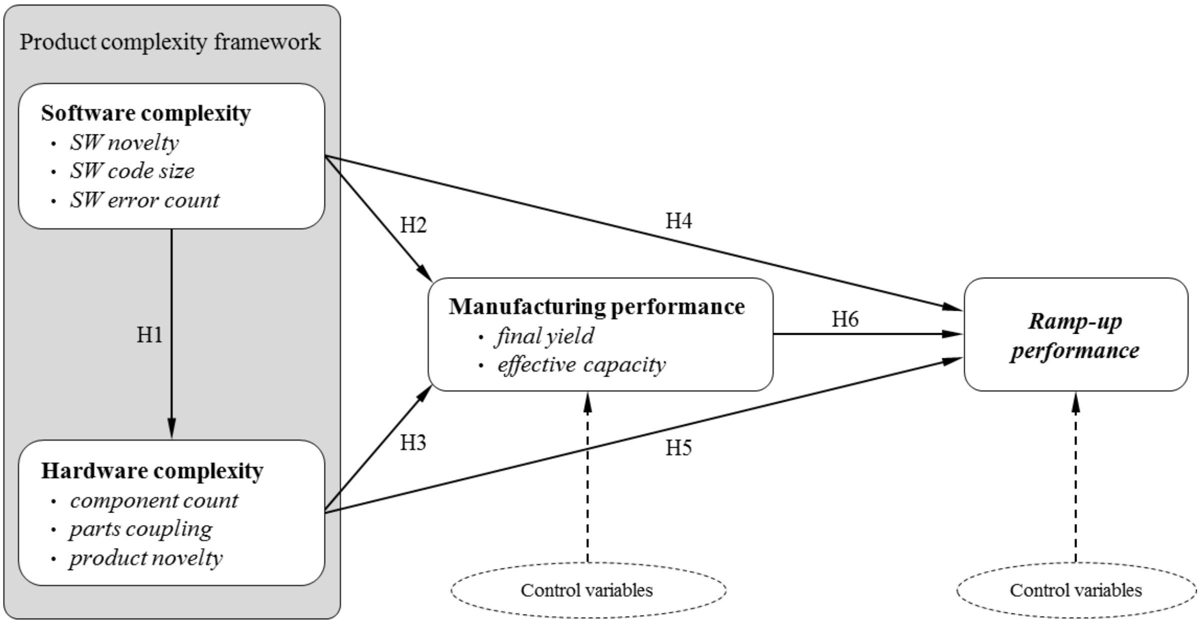

Figure 1. Conceptual Model

Before we state our Hypotheses we first introduce the variables that constitute our conceptual model.

3.1 Software Complexity Variables

With the term "software", we refer to the operating software and any application software under the direct control of the firm, which gets programmed into the logic board of the product during the production process and is entirely necessary for a successful launch. We do not include software modules or subroutines that are an inseparable part of any advanced component or application software from third-party suppliers. For example games, special ring tones, or other third-party applications can be introduced in subsequent software releases if they are not on a sufficiently mature level at ramp-up start. In contrast, in-house developed core-software elements must be available and error free for product launch, such as operating software functions (e.g., protocol stack routines). Errors in such functions may prevent regulatory approvals and potentially delay the ramp-up phase.

Generally, software complexity refers to the characteristics of the data structures and procedures within the software that make it difficult to understand and change (Curtis et al. 1979, Zuse 1991). Many software engineering studies rely on code and structure metrics as quantitative measures of software complexity. The former entail the individual system components (procedures and modules) and require detailed knowledge of their internal mechanisms, whereas the latter consider the product as a component of a larger system and focus on the interconnections of the system components (Kafura and Reddy 1987, Banker et al. 1998). However, previous research into multiple proposed software complexity metrics indicates high correlations among the various metrics (Banker et al. 1993, Munson and Koshgoftaar 1991). Our analysis has revealed three major groups that vary in different orthogonal dimensions and thus overcome this limitation: SW novelty, SW code size, and SW error count. Due to the fact that software development is standardized within Nokia, it allows for the collection of reliable data regarding these software characteristics.

First, *SW novelty* represents the number of new requirements/features in the software specification for each product and is derived from the requirements management database. Understanding and managing new functions, for which the behavior and interactions with other elements is not known in advance, adds uncertainty, risk, and effort to the team's responsibilities, which could provoke difficulties before and during the ramp-up phase. Krishnan and Zhu (2006) claim that adding more features usually increases complexity and reduces the software's ease of use. On the other hand, existing software code that gets used and tested across many products, all else being equal, should have greater design integrity and quality than new software code developed for a single, particular product. Therefore, we posit that the greater use of new software elements influences the integrity of the existing software structure and increases risks related to on-time readiness. In our study, SW novelty therefore refers to the number of new software features/requirements for the product that are not used by any other product. In other words, it is the number of new requirements/features in a cell phone that have not been included in previous products or the existing software baseline.

Our second software complexity variable measures the source *SW code size* in terms of executable lines of code as provided by the compiler log files. Source code size metrics, though common ways to describe software complexity, are particularly important in embedded systems that suffer from memory restrictions. Although software engineering literature often uses a lines of code measure, its problems are well known (Krishnan et al. 2000), especially related to the inaccurate and inconsistent definition of "a line of code" in various programming languages and the tools used to count the number of source lines. To ensure the consistency and accuracy of this measurement across products, we used a common analysis tool that measures the number of lines of executable code and excludes comment statements. According to Krishnan et al. (2000), counting executable statements offers a more accurate measure than counting the number of physical lines. Because the products in our sample share the same programming languages (i.e., a proprietary language for lower-level signaling functions and C/C++ for higher-level code) and are based on the same programming tools, our measure of SW code size is not biased by the programming language or environment.

Finally, and because SW novelty and SW code size do not sufficiently account for differences in the individual product configuration (both measures assume that software components have in-built complexities that are static and independent of their context) we include *SW error count* as a third variable to our software complexity framework. Even a small share of SW novelty and a small SW code size may lead to a disproportionate amount of development effort if the particular configuration results in a large number of errors due to interactions and side effects. Also, SW error count can be estimated *a priori* during the later development phases to predict the remaining development effort. Unlike other metrics (e.g., McCabe's (1976) cyclomatic complexity, Wood's (1986) component/coordination/dynamic dimensions, Halstead's (1977) effort metric) it can also be used by management as a means to monitor product maturity. Consumer electronics products must typically pass a series of standardized software acceptance tests, hence SW error count can reveal the progress of software development and the readiness for product launch. We measure it as the number of reported errors during the software acceptance/verification phase.

3.2 Hardware Complexity Variables

To quantify hardware complexity we consider products in physical terms and hence assume complexity to be a property of a product (Rodriguez-Toro et al. 2004). According to Novak and Eppinger (2001), it can be measured as (1) the number of product components to specify and produce, (2) the extent of interactions to manage between these components (parts coupling), and (3) the degree of product novelty. Please note that we consider hardware complexity at the macro level, that is, the first layer of abstraction, which is under managerial control and technically observable. We do not consider the internal structures of the lower levels (e.g., subsystems, advanced components) such as cameras, displays, or speakers.

Our first hardware complexity variable - *component count* - covers the total number of components in a complete cell phone, as reported from the product data management system (see Table 1). This definition does not include components that are inseparably embedded in advanced components, such as the single glass layers of display modules or the lens elements of camera components. Variety in the component count results from the growing diversity in the cell phone customer base which forces companies to offer tailored models with various functionality levels for different target groups. However, different functionalities cannot be integrated into the same architecture without altering the number of components needed. A product that offers dual display functionality, hands-free stereo audio, global positioning services (GPS), and sophisticated connectivity options must integrate more physical components into its architecture than a featureless counterpart. Adding more components to a product raises product complexity in terms of a more difficult manufacturing process (Boothroyd et al. 1987, Coughlan 1992), more complex supply logistics (Fisher et al. 1999) and greater verification effort (Novak and Eppinger 2001).

For our *parts coupling* variable, we note that modern cell phones exhibit diverse interdependencies among the embedded components. As Novak and Eppinger (2001) state, the more interconnected the parts in a system are, the more difficult it is to coordinate their development. To quantify the level of parts coupling, we use the report functionality of a circuit board design tool and count the number of signal networks across all electrical and electromechanical components in a product, (i.e., components that carry any electrical functionality like resistors, capacitors, integrated circuits, antennas, audio components). This group of components accounts for more than 70% of the total components in a product. Our rationale for this definition of parts coupling stems from discussions with R&D experts, who confirmed that the effective integration of components does not only require knowledge about the components but also about their simultaneous interactions. Many components are delivered fully functional and pretested, hence the key challenge of development lies in the mastering of coupling effects. Measuring the number of networks is a more reasonable approach to account for the difficulties that developers encounter than measuring the number of pairs (i.e., direct connections between electrical and electromechanical components). Development engineers must consider the electrical structure of the various subsystems as an arrangement of interlinked connections rather than a collection of individual point-to-point connections. The number of networks also is unbiased with regard to those aspects that increase pair count (e.g., test points) without adding interaction complexity.

Finally, the existing literature has conceptualized product novelty in several ways. Coughlan (1992) defines newness as the degree of similarity of a product to other members of its family, or the degree to which preexisting product parts get altered. Swink (1999) refers to newness as the percentage of new designs in the product. We build on these definitions though we use a richer operationalization: we define *product novelty* as the percentage material value of physical components in a cell phone that is new to the responsible development center, compared with previous products that have already been developed at this development center. Our observations have led us to conclude that this percentage material value offers a better operationalization than the percentage number of new parts – especially considering our macro perspective that considers certain advanced components (e.g., cameras, displays, speakers) as single components. Thus, definitions of product novelty that are based on the percentage of new parts regard each component's contribution to novelty as equal, even though new displays, cameras, or processors require considerably more effort during integration and testing phases and entail more supply risk during the ramp-up phase than simple parts like new screws, foams, or stickers. This variance in complexity within single components supports our use of their monetary value as a measure of product novelty. Based on the officially filed product development documentation and specification we were able to identify all lead components for each product. Together with the sourcing parts list we were consequently able to calculate the product novelty measure.

3.3 Manufacturing Performance Variables

Despite the many proposals on how to quantify manufacturing performance (Neely et al. 1995, White 1996, Slack et al. 2001, de Toni and Tonchia 2001), most of the ramp-up specific studies use capacity and/or final yield measures (Matsuo et al. 1997, Terwiesch et al. 1999, Hatch and Mowery 1998, Almgren 2000). We follow this approach and use *effective capacity* together with *final yield* as variables to measure manufacturing performance. This combination acknowledges that the actual output of any manufacturing system is only a fraction of the planned allocated capacity (see Figure 2) and the particular type of lost capacity may be of importance (e.g., yield losses may be different from other losses as they can be reworked and fed back into production).

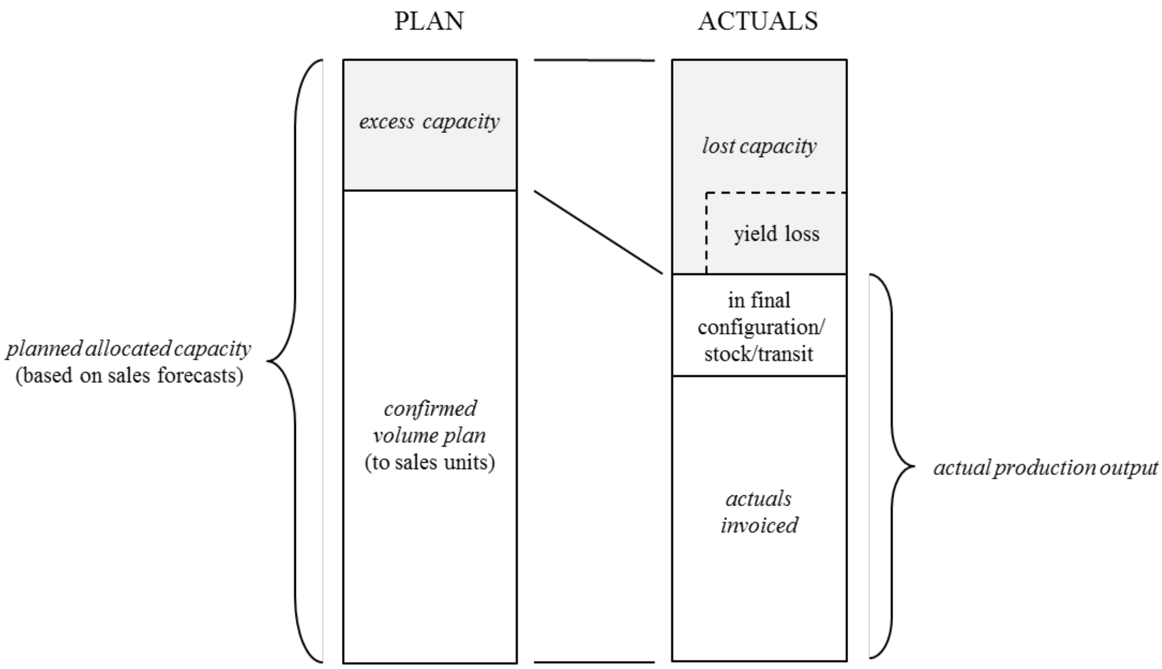

Figure 2. Determinants of Manufacturing and Ramp-up Performance

Final yield. In consumer electronics, manufacturing usually takes place on multistage production lines with different control or test phases. Typically, a first test phase takes place after the components are installed on the circuit board (i.e., commonly known as the logic board). The series of electronic tests at this point ensure that the circuit board is functional, that all parts are operational, and that all parts are correctly installed. Yield losses at this stage relate to soldering defects, material deficiencies and test system failures. After this stage, the assembly work begins and electromechanical parts are added to the circuit board before it is mounted between structural frames and undergoes a detailed functional test. Yield losses at both test phases identify product and process instabilities, which is why *final yield* frequently appears in manufacturing literature (Hatch and Mowery 1998, Terwiesch and Bohn 1998, Terwiesch et al. 1999, van der Merwe 2004, Keil et al. 2007).

Effective capacity is quantified as 1 minus the ratio of lost capacity to its planned allocated capacity. Although capacity measures are subject to criticism for its negative long-term implications (Slack et al. 2001, Goldratt and Cox 2004), effective capacity is sensitive to ramp-up specific disturbance factors that may result in various capacity losses, such as product and equipment readiness issues, product manufacturability concerns, material availability/quality problems, unscheduled engineering trials, or neglected operator training. Ultimately, these disturbance factors have a negative influence on effective capacity because they impede that the entire allocated capacity can be used to manufacture end products. Effective capacity and final yield are both based on data from a production database system and their calculations are summarized in Table 1.

3.4 Ramp-Up Performance

According to Mallick and Schroeder (2005), high-tech firms use their technology to create value for their customers and to capture value for their shareholders. Thus, any metric used to measure ramp-up performance in high-tech manufacturing should reflect the objective of value creation. In line with existing studies that focus on time, cost and quality (Kuhn et al. 2002, Schuh et al. 2005, Wildemann 2007); quantity, cost and quality (Almgren 2000); or missed targets for output, quality and delivery (Langowitz 1987), we posit that all activities surrounding the dependable delivery of products – provided they fulfill the set quality criteria – are significant drivers of customer value and hence ramp-up performance for high-tech products. Dependable sales volume deliveries are particular crucial before seasonal peaks (e.g., Christmas or the Chinese New Year) when strong consumer demand must be satisfied in a very short period of time (possibly at the expense of higher unit costs, compromises on inventory levels, or manufacturing effort) as lost sales and customer loyalty cannot be recaptured at a later phase. Hence, we measure *ramp-up performance* as the actual invoiced quantity during the ramp-up execution phase divided by the confirmed volume plan quantity for the same period. In other words, we measure sales volume fulfillment rather than absolute ramp-up speed. According to Voigt and Thiell (2005) the focus on pure ramp-up speed is economically inefficient, because quality and other cost drivers may accumulate and ultimately affect overall company competitiveness. Ramp-ups with expansion rates greater than planned may reveal strong output performance but do not necessarily contribute to profitability or increased value creation if they only fill outbound buffers. For the execution phase (T_{RU_EXE}) we chose a time horizon of 12 weeks since it reflects the (product-independent) short-term planning cycle that prevails at Nokia. During this time frame, capacity and most resource availability is considered fixed, because of equipment and material procurement lead time limitations. We detail this period in Figure 3.

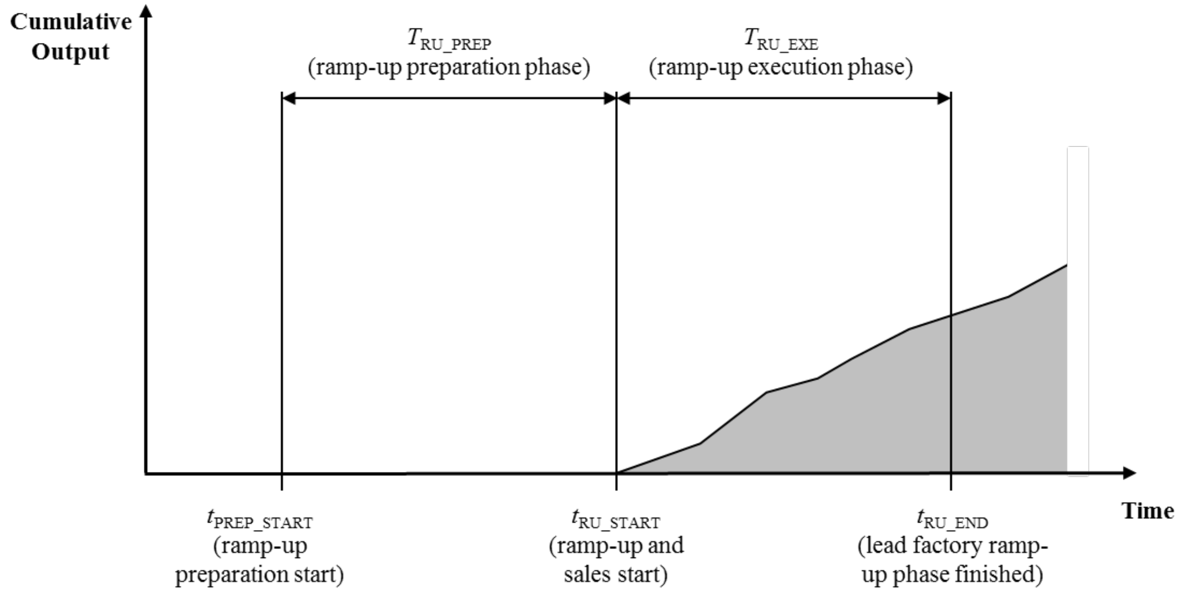

Figure 3. Ramp-Up Phase Time Parameters

3.5 Control Variables

First, we include a control variable that captures linear time effects. Consistent with learning curve studies, we define learning as the increase in ramp-up performance by a firm as its experience increases over time. We must differentiate between learning from experience and changes in ramp-up performance independent of experience, hence we include a *linear trend* (*LT*) control variable, which is the number of days between the ramp-up start of each product versus the ramp-up start of the first product in the study.

Second, empirical studies report influences of several factory characteristics on manufacturing and ramp-up performance (Hayes and Clark 1986, Langowitz 1987, Clark and Fujimoto 1991, Kuhn et al. 2002). Although the facilities in our study represent a relatively standardized and homogeneous capacity pool, we consider four variables that may account for effects in ramp-up performance.

Years in operation (*YiO*) measures the time a plant has been in operation prior to the ramp-up start of each product. This measure serves as a proxy for the accumulated experience level of any given plant. *Factory ownership* (*FO*) is a dummy variable, for which 1 indicates in-house facilities and 0 means contracted facilities. In our sample, three of the nine facilities were owned by contract manufacturers, which were responsible for manufacturing up to a generic product level, before the units were shipped to in-house facilities for the final configuration and distribution. This variable might explain differences in the internal learning curve, problem-solving capability and supply logistics. *Factory location* (*FL*) captures differences in work force cultures and supply network structures. The measure is another dummy variable, divided into facilities located in Asia (China and Korea = 1) and facilities located in Europe (Germany, Hungary, Finland = 0).

The relationship between the complexity variables and manufacturing/ramp-up performance might also be influenced by the extent to which management adjusts sales plans and sales forecasts during the ramp-up preparation phase. In order to control for these effects, we introduce *sales forecast change* (*SFC*) and *excess capacity* (*EC*). The former is the ratio between the sales forecast quantity at the start of the ramp-up period and the sales forecast quantity 12 weeks before the start of this period. The latter is similar to the construct from organizational theory (Nohria and Gulati 1996): the capacity and availability of materials in excess of the necessary minimum to produce a needed level of output. Thus, we calculate excess capacity (EC) as 1 minus the ratio between the confirmed volume plan and the planned allocated capacity – both captured at the start of the ramp-up period and calculated for the entire ramp-up execution period.

Table 1. Summary of Variables and Definitions

Hardware complexity variables

component count = number of components in a complete cell phone, as on a bill of materials parts list, down to the level of single parts that can be purchased separately

parts coupling = number of signal networks between all electrical and electromechanical components in the product

$$\textit{product novelty} = \frac{\text{material value of new physical components in the product}_{\text{average over period } T_{\text{RU\_EXE}}}}{\text{total material value of the product}_{\text{average over period } T_{\text{RU\_EXE}}}}$$

Software complexity variables

SW novelty = number of requirements/features in a cell phone, not been included in previous products or the existing software baseline

SW code size = source code size in terms of executable lines of code

SW error count = number of observed errors during the software acceptance/verification phase

Manufacturing performance and ramp-up performance variables

$$\textit{effective capacity} = 1 - \frac{\textit{lost capacity}_{\text{over period } T_{\text{RU\_EXE}}}}{\textit{planned allocated capacity}_{\text{at } t_{\text{RU\_START}} \text{ over period } T_{\text{RU\_EXE}}}}$$

$$final\ yield = \frac{passed\ units_{\text{at test phase 1 over period } T_{\text{RU\_EXE}}}}{all\ tested\ units_{\text{at test phase 1 over period } T_{\text{RU\_EXE}}}} * \frac{passed\ units_{\text{at test phase 2 over period } T_{\text{RU\_EXE}}}}{all\ tested\ units_{\text{at test phase 2 over period } T_{\text{RU\_EXE}}}}$$

Note:

$$all\ tested\ units_{\text{at test phase 1 over period } T_{\text{RU\_EXE}}} = planned\ allocated\ capacity - lost\ capacity$$

$$passed\ units_{\text{at test phase 2 over period } T_{\text{RU\_EXE}}} = actual\ production\ output$$

$$ramp\text{-}up\ performance = \frac{actuals\ invoiced_{\text{over period } T_{\text{RU\_EXE}}}}{confirmed\ volume\ plan_{\text{at } t_{\text{RU\_START}} \text{ over period } T_{\text{RU\_EXE}}}}$$

Control variables

LT = number of days between ramp-up start of each product versus ramp-up start of the first product in the study

YiO = number of years a plant was in operation until the ramp-up start of each product

$$FO = \begin{cases} 1 \text{ if in-house facility} \\ 0 \text{ otherwise} \end{cases}$$

$$FL = \begin{cases} 1 \text{ in case of Asian factories (China, Korea)} \\ 0 \text{ in case of European factories (Germany, Hungary, Finland)} \end{cases}$$

$$SFC = \frac{sales\ forecast\ quantity_{\text{at } t_{\text{RU\_START}} \text{ over period } T_{\text{RU\_EXE}}}}{sales\ forecast\ quantity_{\text{at } t_{\text{PREP\_START}} \text{ over period } T_{\text{RU\_EXE}}}}$$

$$EC = 1 - \frac{confirmed\ volume\ plan_{\text{at } t_{\text{RU\_START}} \text{ over period } T_{\text{RU\_EXE}}}}{planned\ allocated\ capacity_{\text{at } t_{\text{RU\_START}} \text{ over period } T_{\text{RU\_EXE}}}}$$

4 Hypotheses

In the previous discussion (see also Figure 1), we note that the co-design of software and hardware is a central system characteristic of cell phones or embedded systems in general (Wolf 1994). For example, the integration of personal navigation in cell phones requires not only the development of a large share of dedicated software code but also the inclusion of additional components into the product (e.g., GPS receiver with discrete circuitry and antenna). However, there is a shift towards software-based implementations as most of the new innovations in cell phones are software-related (e.g., augmented reality, games, video processing, social networking clients), since hardware release cycles are more expensive and time consuming. Thus, hardware modifications and extensions often emerge as a side effect when new software features – for example a social networking client – demand more processor power, memory size, or connectivity speed. Consequently, higher levels of software complexity are counterproductive for preserving hardware integrity as they may increase hardware complexity in terms of component count, parts coupling, or product novelty. We accordingly state our first Hypothesis:

HYPOTHESIS 1. *Higher levels of software complexity are associated with higher levels of hardware complexity.*

As outlined above, embedded software represents a core integration activity for cell phone projects and most observers acknowledge the difficulty of ensuring the completion of software-intensive projects in budget and on time (Austin 2001, Lindstrom and Jeffries 2004). For example, manufacturing cannot be executed as planned if the required software or customer specific configuration files are missing, for instance due to delayed regulatory or customer approvals (e.g., from large operators). In addition, cell phone production entails complex automatic test systems to calibrate wireless protocols, power management, or to control the manufacturing process. New or complex software features or interfaces can cause these tests to fail, resulting in reduced manufacturing output. Hence, we formally state:

HYPOTHESIS 2. *Greater software complexity is associated with lower manufacturing performance.*

Decisions about the number of components to be incorporated into a design and decisions about how much novelty to impose on a new product also relate closely to several important issues for operations. Ambitious products provide a fundamental source of difficulty for manufacturing (Langowitz 1987, Kuhn et al. 2002, van der Merwe 2004, Keil et al. 2007). Since complex product designs make specific demands on factories and since factories have unique sets of skills that they can use to meet those demands, initial manufacturing performance is a matter of accurate product–factory fit (Langowitz 1987). Typically, more complex products require more process steps and thus create more opportunities for process failure (Swink 1999). In addition, complex product design specifications frequently require more engineering change orders which may also affect performance in a negative way. Likewise, upstream supply operations face similar difficulties and affect manufacturing performance via material supply shortages and mismatches (Almgren, 2000).

HYPOTHESIS 3. *Greater hardware complexity is associated with lower manufacturing performance.*

Several studies have identified a relationship between product characteristics and ramp-up performance (Langowitz 1987, Almgren 2000, Kuhn et al. 2002, van der Merwe 2004, Schuh et al. 2005). Complex products – whether they involve hardware or software complexity – are less likely to accomplish customer acceptance because the lack of experience in the use of a new, complex product reduces the user's ability to describe its needs (Thomke and Bell 2001). This complicates project management as customers revise their requirements more often, request new customization options, or even find new errors after pretesting the new product. In addition, there is consensus that material problems (e.g., dimensional variations or delayed deliveries) and quality issues (visual defects or software variant difficulties) are more likely to occur in complex designs (Clark and Fujimoto 1991, Almgren 2000, Kuhn 2002). As a result, delivery commitments have to be lowered or shipments will lag behind planned schedules. Hence, we state:

HYPOTHESIS 4. *Higher levels of software complexity are associated with lower ramp-up performance.*
HYPOTHESIS 5. *Higher levels of hardware complexity are associated with lower ramp-up performance.*

Cell phones are manufactured with delayed customization (i.e., postponement), thus the final customization does not take place until real customer orders are known. As a result, manufacturing performance represents how well the generic part of a cell phone is manufactured and how well aggregate production plans are met. Ramp-up performance instead measures how well the generic products can be converted into customer-specific cell phones that are subsequently distributed and invoiced. While these items are distinct in nature, previous research states a relationship between superior manufacturing performance and successful ramp-ups (Clark and Fujimoto 1991, Wildemann 2007). In other words, product availability is a pre-condition for product sales which leads us to our last Hypothesis.

HYPOTHESIS 6. *Higher levels of manufacturing performance are associated with higher ramp-up performance.*

5 Data and Methodology

The data for our study pertain to 46 products that were developed at R&D centers in four countries between 2005 and 2008. The sampling and selection process covered a wide variety of price points, customer segments, form factors and total sales volume ranges. Our method of data collection was guided primarily by our conceptual model and employed multiple data sources, including project documentation systems, production databases, management information system reports, data archives and company reports. All operational definitions of the variables were additionally validated on the basis of interviews with several project managers and senior managers, as well as with a written questionnaire targeted toward the product ramp-up managers. These data provided additional insights into the many qualitative disturbance issues during the ramp-up phase. In addition, we collected longitudinal data over the course of four projects, through the efforts of one of the authors who is employed as a ramp-up manager by Nokia. Our unique database thus features highly reliable quantitative and qualitative information about the characteristics of each product; its development, production, and logistics process and the results of interviews and observations with key informants. Whenever possible, we triangulated the qualitative data with mandatory milestone review documents and expert opinions to confirm their accuracy and consistency. Table A-1 in the Appendix presents descriptive statistics and correlations for our variables. Correlations are generally as expected and moderate in magnitude. For confidentiality, we normalized the SW error count variable to have a mean equal to 1,000.

We use multiple linear regression models to test our Hypotheses. To enable comparison of effect sizes, we standardized all variables (mean = 0, variance = 1) before running the regression calculations. This is useful as our data is a mixture of different scales (e.g., component count uses pieces, linear trend uses days). The assumptions of our multiple regression models were tested by several statistical methods. First, all data panels were screened for abnormal observations to avoid bias in the regression calculations. Next, predicted values were plotted against standardized residuals to show a random scattered pattern, supporting the assumption of linearity and homoscedasticity. For each regression, we calculated variance inflation factors to rule out multicollinearity problems. Resulting variance inflation factors (≤ 7) indicated no significant multicollinearity effects for any of the models (Hair et al. 2006). Also, normality of the error term is supported by the appropriate histograms and normal probability plots.

6 Results

The analysis was divided into three stages. First, we used multiple regression models to test the effects of software complexity variables (i.e., SW novelty, SW code size and SW error count) on each of the three variables of hardware complexity separately (Table 2). In the second stage of the analysis, we used the regression results to test for Hypotheses 2 and 3 – the effect of complexity variables on the manufacturing performance variables (i.e., final yield and effective capacity). In the final stage, we employed multiple regression (Table 4) to test the combined effect of the complexity and manufacturing performance variables on ramp-up performance (Hypotheses 4-6). For brevity we only discuss the full models (including controls).

Table 2. Regression Results (H1)

	Dependent variables		
	component count	*parts coupling*	*product novelty*
Predictor variables			
SW novelty	0.198 (0.128)	0.020 (0.119)	0.734 *** (0.081)
SW code size	0.460 *** (0.120)	0.571 *** (0.111)	-0.006 (0.076)
SW error count	0.269 ** (0.127)	0.351 *** (0.118)	0.278 *** (0.081)
R-Sq(adj)	36.40%	45.40%	74.30%

Notes: N = 46, *** $p \leq .01$, ** $p \leq .05$, * $p \leq .10$, two-tailed tests.

The results of Table 2 largely support Hypothesis 1. Six out of nine possible relationships show strong and significant positive effects of software complexity variables on hardware complexity variables, where each software complexity variable significantly relates to at least one hardware complexity variable. We also observe the strongest relationship between SW novelty and product novelty (β = 0.734, p = 0.000). Overall, increasing levels of software complexity are associated with higher levels of hardware complexity.

The results of Table 3 provide partial support for Hypotheses 2 and 3, stating that increased software and hardware complexity are negatively associated with final yield and effective capacity. Component count provides the strongest effect on both, final yield (β = -0.410, p = 0.022; β = -0.519, 0.007) and effective capacity (β = -0.429, p = 0.016; β = -0.393, p = 0.031). We also find a significant negative effect of SW code size on final yield (β = -0.305, p = 0.078), indicating that final yield of cell phone manufacturing seems to be a function of the number of components and its SW code size. This is plausible as both variables are likely to increase the failure opportunities in production. For effective capacity we observe parts coupling in addition to component count to be strong and significant (β = -0.399, p = 0.029), revealing that manufacturing performance in the form of good output increases as component count and parts coupling decreases. All other variables do not significantly relate to final yield or effective capacity at the 0.10 level.

Table 3. Regression Results (H2 and H3))

	Dependent variables			
	final yield		*effective capacity*	
Predictor variables				
SW novelty	-0.271 (0.213)	0.148 (0.271)	-0.349 (0.211)	0.098 (0.262)
SW code size	-0.075 (0.143)	-0.305 * (0.168)	0.071 (0.141)	-0.126 (0.162)
SW error count	-0.260 * (0.147)	-0.188 (0.151)	-0.034 (0.146)	-0.007 (0.146)
component count	-0.410 ** (0.172)	-0.519 *** (0.181)	-0.429 ** (0.170)	-0.393 ** (0.175)
parts coupling	0.035 (0.182)	0.008 (0.182)	-0.342 * (0.180)	-0.399 ** (0.175)
product novelty	-0.005 (0.229)	-0.241 (0.240)	0.317 0.226)	0.118 (0.232)
Control variables				
LT		0.363 (0.223)		0.242 (0.215)
YiO		-0.298 (0.179)		0.042 (0.173)
FL		0.093 (0.194)		0.282 (0.187)
FO		0.065 (0.152)		-0.140 (0.147)
SFC		0.213 (0.128)		-0.136 (0.123)
R-Sq(adj)	46.4%	50.6%	47.6%	54.1%

Notes: N = 46, *** $p \leq .01$, ** $p \leq .05$, * $p \leq .10$, two-tailed tests.

Table 4 contains the results for ramp-up performance and provides the tests for Hypotheses 4–6. Contrary to our expectation, SW novelty appears to have a positive effect on ramp-up performance (β = 0.646, p = 0.027). Hence, we do not find support for Hypothesis 4. One possible explanation is the unpredictable implementation and testing effort of novel software features that frequently results in late project schedule slips. Factories appear to benefit from the extra waiting time for the approved software release, as this enables them to build up semi-finished product buffers and, consequently, achieve higher performance levels during the subsequent ramp-up phase.

In addition, product novelty has a negative effect on ramp-up performance (β = -0.447, p = 0.089) providing support for Hypothesis 5. New physical elements are more likely to cause material supply problems and product quality issues, which both result in ramp-up performance drops compared to proven ones. While the effect of final yield on ramp-up performance is not significant (β = -0.093, p =

0.608), the results show a strong and significant positive effect of effective capacity on ramp-up performance ($\beta = 0.836$, $p = 0.000$). Again, these mixed results provide partial support for Hypothesis 6 and yield some interesting insights. Advances in capacity management are rather likely to pay off during the ramp-up phase than investments in yield improvement activities.

Table 4. Regression Results (H4-H6)

	Dependent variable	
	ramp-up performance	
Predictor variables		
SW novelty	0.375 (0.246)	0.646 ** (0.278)
SW code size	0.105 (0.158)	-0.083 (0.177)
SW error count	0.197 (0.168)	0.027 (0.158)
component count	0.269 (0.214)	0.175 (0.217)
parts coupling	-0.032 (0.210)	-0.180 (0.197)
product novelty	-0.527 ** (0.258)	-0.447 * (0.255)
final yield	-0.075 (0.178)	-0.093 (0.179)
effective capacity	0.895 *** (0.180)	0.836 *** (0.178)
Control variables		
LT		0.260 (0.263)
YiO		-0.039 (0.195)
FL		0.103 (0.208)
FO		-0.191 (0.155)
SFC		0.263 * (0.137)
EC		0.393 ** (0.153)
R-Sq(adj)	35.1%	50.8%

Notes: N = 46, *** $p \leq .01$, ** $p \leq .05$, * $p \leq .10$, two-tailed tests.

Of the control variables, only SFC (β = 0.263, p = 0.063) and EC (β = 0.393, p = 0.015) have a significant positive relationship with ramp-up performance.

The effect of SFC suggests that when the demand for a product increases – compared to the fixed production plan at t_{RU_START} – management will do anything in their span of control to boost output up to material or capacity limitations, which will in turn lead to higher performance levels. On the other hand, management will respond with a decrease in output if demand weakens (to avoid excess inventories) with the consequence that ramp-up performance will drop.
The effect of excess capacity (EC) suggests that higher levels of planned allocated capacity (compared to the confirmed volume plan) dampens the negative impact of ramp-up disturbances (e.g., equipment breakdowns, material quality problems, customer rejections) but with the downside of creating idle capacity under steady or weak demand.

7 Discussion

The key objective of this study has been to investigate the effect of product complexity characteristics on manufacturing and ramp-up performance using operational data from the cell phone industry.

To begin with, the significant and directional coupling between software and hardware characteristics supports our view that most of the new innovations in cell phones are primarily enabled by software and by the way in which software and hardware designs are integrated throughout Nokia's product development process. As already pointed out in section 2, Nokia's product development approach is based on the premise that design activities are best divided into a number of sequential project "stages" separated by milestones reviews. After a requirements analysis, functionality is split into features that are implemented in software, in hardware or in a combination of both. In an iterative process, based on the fabrication of a series of prototypes, software/hardware integration is synchronized and feedback on whether the design meets customer requirements is gathered. As a new project proceeds through these successive prototype rounds, the design evolves in increasing levels of maturity, from early engineering samples to salable products that contain the final hardware. Finally, extensive testing and fine tuning activities take place in the course of which software releases are introduced in frequent intervals and tested on the final hardware. During that phase, product development managers focus mainly on software stability as most of the remaining errors arise from the realized software features or hardware problems that are corrected in software to save time and money (software release cycles are shorter and more flexible than hardware release cycles). This uniqueness of embedded systems and traditional cell phones stands in contrast to other products groups (e.g., personal computers, high-end smartphones), that show decoupled architectures and platform structures in the software and hardware development.

A second important finding is that the novelty variables of both software and hardware complexity are the most influential drivers of ramp-up performance. Interestingly, software novelty appears to be positively associated with ramp-up performance. Our explanation for this finding is in line with studies that found a positive relationship between increased software newness and the determinants of software development time (Callahan and Moretton 2001, Griffin 1997). The ongoing growth in software content, it's coding and testing effort as well as the flexibility of software to quick-fix detected hardware errors make software schedules increasingly unpredictable and vulnerable to late schedule slips. Hence, several studies acknowledge the difficulties of ensuring software-intensive projects to be completed within budget and on time (Austin 2001, Lindstrom and Jeffries 2004). Rather than suffering from delayed software readiness (as a result of higher SW novelty), firms may profit from it by starting the production gradually – despite rising inventory levels of semi-finished products – until the approved software release can be used for the re-programming of these product buffers and starting the regular ramp-up. Since the re-programming step is straightforward, quick and does not occupy any regular ramp-up production resources, it allows for higher output levels during the initial ramp-up phase. The practical significance of this result is that firms need to make a trade-off between the gains in ramp-up performance that are enabled through gradual production ahead of the delayed ramp-up start and the negative consequences of missed schedule adherence and hence delayed deliveries.

In contrast, we find that greater product novelty has a negative effect on ramp-up performance. Apparently, novel product designs increase the number of uncertain issues that development teams, suppliers and even customers must cope with. Hence, they require more training/learning effort by production engineers and operators (in-house and at suppliers) as well as by customers to achieve ramp-up performance levels similar to those of less novel designs. In other words, the more novel the product, the more learning effort is needed and the slower is the increase in manufacturing performance during ramp-up (van der Merwe 2004). Furthermore, we find that ramp-ups with large levels of product novelty are particularly slow at the beginning, forcing the ramp-up steepness to rise disproportionately towards the end of the ramp-up execution period in order to achieve the planned output levels. Thus, effective capacity may still reach planned levels but final configuration and distribution activities suffer due to the timely shifted and compressed availability of products for the final configuration and distribution stage..

Another important finding of our results shows that manufacturing performance has a strong impact on ramp-up performance. However, this effect is due to effective capacity and not due to final yield. The absence of a significant effect of final yield suggests that yield losses are compensated through repair activities and therefore have a negligible effect on the output. This is in line with our observations that repair resources are allocated to production lines on a need basis and most of these failures are easy to fix.

On the other hand the effect of effective capacity suggests that capacity losses apart from yield losses like unscheduled downtime, scheduled maintenance, setup changes and reduced speed are more disruptive in ramp-up environments. Since these losses cannot be absorbed by repair activities, subsequent final configuration and distribution activities may not proceed as planned, customer shipments are delayed and finally ramp-up performance decreases. More specifically, unscheduled downtime as the key contributor of effective capacity is the result of external (e.g., missing components or material) and internal (e.g., equipment downtime) factors.

For example, most of the external disturbances are related to the inability of suppliers to deliver the right material on time and in the required quantity. This frequently leads to line stops as buffer stocks are not available during the early ramp-up phase. Various reasons are described in the literature (e.g., Langowitz 1987, Terwiesch et al. 1999, Almgren 2000, Pfohl and Gareis 2000) but our results suggest that the key contributors to material issues are related to the number of components and their interactions (parts coupling). This is because material management is a complex process and the number of unique parts thereby drives complexity which in turn negatively affects performance (Fisher et al. 1999). It requires considerable resources to forecast and coordinate the timely arrival of the many parts that go into a cell phone product in the required quantity. This process remains error-prone and is likely to be exposed to more engineering changes the more components are involved.

With regard to capacity losses that are the result of internal factors, we find test system downtime as the most frequent source of disturbance during ramp-ups. That is because these systems are among the most complex in the factory and require the highest level of product specific adaptation and maintenance. For this reason, products with a large number of components and complex interactions are more likely to cause instabilities, failures and damages in these systems.

The practical significance of this result is that the careful management of product design, with an understanding of the effect that component count and parts coupling have on effective capacity instead of final yield, is highly relevant for ramp-up success.

Finally, the effect of our last control variable, excess capacity (EC) holds an important managerial implication. Recall that excess capacity represents the percentage difference between the planned allocated capacity and the confirmed volume plan. The former represents all of the materials and capacity that is reserved for the production of a particular product. Ramp-up teams use the input from sales teams that intend to sell the product to define this quantity. The latter represents the volume plan used by sales teams to confirm customer orders. Hence, the confirmed volume plan is a balance between material supply risks, production capacity risks, schedule risks and anticipated sales projections steered by management. Decisions, such as allowing for higher levels of excess capacity – given a certain level of planned allocated capacity – and being more restrictive with initial sales volumes, are therefore likely to improve ramp-up performance but at the expense of total output, cost, and thus profit. Finding the optimal level of excess capacity is linked to the managerial actions regarding incentives and rewards to product development teams. If management demands high levels of profits, it needs to design incentive systems that reward product development teams for achieved ramp-up performance but in relation to the chosen level of excess capacity. This relationship is complex and deserves further research attention with particular focus on the strategic priority of the firm with regard to output dependability against overall profit.

8 Conclusions

We have developed a set of regression models that relate quantitative product complexity characteristics – represented by software and hardware complexity variables – and manufacturing performance variables to ramp-up performance. With operational data from the cell phone industry, our models explain most of the variation in ramp-up performance. Beyond the growing importance of software characteristics in driving hardware complexity, we find that certain hardware and software characteristics (i.e., component count, parts coupling and SW code size) impact the performance of the manufacturing system in terms of final yield and effective capacity. Finally, we find that effective capacity together with the novelty aspects of both software and hardware complexity (i.e., SW novelty and product novelty) are the key determinants of ramp-up performance.

This study also highlights the importance of a novelty versus ramp-up performance trade-off and the relevance to distinguish between software and hardware novelty in order to properly deal with this trade-off. Because it is the main objective of the high-tech industry to achieve full-scale production and thereby time-to-volume targets, our study underscores the importance of the trade-off between implementing more product novelty (that may create surplus consumer attraction) and achieving ramp-up performance targets. Furthermore, advances in information and communication technologies will presumably lead to further growth in software novelty across products. Hence, effective software engineering with the focus on schedule adherence is becoming a central capability for launching new products quickly onto the market.

We contribute to the field of operations management by demonstrating the relevant product and manufacturing characteristics associated with ramp-up performance by offering a substantially enhanced and more detailed understanding of the ramp-up process and by validating the results of previous exploratory and qualitative studies. For managers, our findings underscore the importance of managing effective capacity instead of final yield and highlight the potential for firms to influence ramp-up performance through deliberate product design decisions. Another contribution is our application specific and quantitative definition of product complexity in the domain of cell phones. We are confident that our definition – which combines hardware and software characteristics – can be extended to other areas and industries. For example, products such as hi-fi systems, game consoles, cameras and flat screens share similar product characteristics with cell phones and even modern automobiles have some comparable properties.

Although we have attempted to build a comprehensive model with precise observations and argumentation based on existing literature, we also note some limitations. First, the relationships derived from the variables studied here capture only half of the overall variability in ramp-up performance. Additional factors may explain and contribute to ramp-up performance, such as product development lead times and late project schedule slips. Further research should identify and specify these factors in detail, particularly with regard to schedule performance. Also, the relationships obtained may not reflect the magnitude of their effects at certain firms. In particular, the magnitude of the effects of product complexity on ramp-up performance would be expected to be larger at firms that launch a smaller number of products per year but with progressive complexity upgrades.

Second, we identified excess capacity as a managerial decision variable that strongly relates to ramp-up performance. This raises the possibility to use this variable as managerial instrument to gauge performance against profit. Ideally, a newsvendor type model would guide management action to set the optimal level of excess capacity according to the strategic priority of the firm.

Finally, our conclusions are based on an analysis carried out within a single company; a wider analysis with different firms from within the consumer electronics industry would enhance our capability to generalize. Nevertheless, we believe our results are generalizable to the consumer electronics industry because our data (1) came from different geographical development centers with different cultural and managerial properties; (2) included a variety of customer groups, ranging from direct shipments to operator-exclusive agreements; and also (3) confirm existing models from other areas, such as the car industry.

Acknowledgements

We acknowledge the significant contribution of the Associate Editor, who provided excellent suggestions to improve the model, the analysis, and the paper.

Appendix A:

Table A-1. Descriptive Statistics and Correlations (Pearson) Between Variables

		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	*SW novelty*	1.00														
2	*SW code size*	0.14	1.00													
3	*SW error count*	0.36	0.08	1.00												
4	*component count*	0.36	0.51	0.38	1.00											
5	*parts coupling*	0.22	0.60	0.40	0.73	1.00										
6	*product novelty*	0.83	0.12	0.54	0.28	0.24	1.00									
7	*final yield*	-0.52	-0.32	-0.51	-0.62	-0.48	-0.49	1.00								
8	*effective capacity*	-0.32	-0.37	-0.28	-0.69	-0.63	-0.19	0.51	1.00							
9	*LT*	-0.48	0.47	0.03	0.11	0.34	-0.29	0.12	0.11	1.00						
10	*YiO*	-0.18	-0.28	-0.09	-0.45	-0.35	-0.24	0.15	0.26	0.09	1.00					
11	*FL*	-0.46	0.25	-0.31	-0.17	-0.06	-0.29	0.41	0.34	0.53	-0.12	1.00				
12	*FO*	0.15	0.09	-0.11	-0.15	-0.09	0.12	0.07	0.09	0.03	0.34	0.33	1.00			
13	*SFC*	-0.08	-0.08	-0.11	-0.07	-0.17	-0.15	0.16	-0.12	-0.27	0.20	-0.02	0.00	1.00		
14	*EC*	0.01	0.36	0.37	0.38	0.52	0.03	-0.37	-0.28	0.43	-0.15	-0.09	-0.18	-0.21	1.00	
15	*ramp-up performance*	-0.14	-0.08	-0.07	-0.21	-0.26	-0.16	0.16	0.58	0.23	0.10	0.17	-0.14	0.02	0.23	1.00
Mean		51.5	6.2M	1000	500.0	555.4	0.16	0.92	0.88	624.0	6.62	0.72	0.87	1.11	0.22	0.85
S.D.		61.5	0.4M	2185	97.2	134.3	0.19	0.04	0.40	295.9	3.08	0.46	0.34	0.43	0.21	0.28
Min		0	0.7M	-1214	268	287	0	0.83	0.33	0	2.80	0	0	0.36	-0.14	0.33
Max		242	20.2M	7215	694	842	0.73	0.98	2.36	1147	12.50	1	1	2.62	0.79	1.44

Note: Significance levels are omitted as the underlying data does not fulfill the requirements for parametric tests.

References

Almgren, H. 2000. Pilot production and manufacturing start-up: The case of Volvo S80. *International Journal of Production Research*, 38 (17), 4577-4588.

Austin, R. D. 2001. The effects of time pressure on quality in software development: An agency model. *Information Systems Research*, 12 (2), 195-207.

Banker, R. D., Datar, S. M., Kemerer, C. F., Zweig, D. 1993. Software complexity and maintenance costs. *Communications of the ACM*, 36 (11), 81-94.

Banker, R. D., Davis, G. B., Slaughter, S. A. 1998. Software development practices, software complexity, and software maintenance performance: A field study. *Management Science*, 44 (4), 433-450.

Benedetto Di, C. A. 1999. Identifying the key success factors in new product launch. *Journal of Product Innovation Management*, 16, 530-544.

Boh, E. F., Slaughter, S. A., Espinosa, J. A. 2007. Learning from experience in software development: A multilevel analysis. *Management Science*, 53 (8), 1315-1331.

Boothroyd, G., Dewhurst, P., Knight, W. 1987. *Product design for manufacture and assembly*, 2nd ed., Marcel Dekker Inc., New York.

Bowersox, D. J., Stank, T. P., Daugherty, P. J. 1999. Lean launch: Managing product introduction risk through response-based logistics. *Journal of Product Innovation management*, 16, 557-568.

Callahan, J., Moretton, B. 2001. Reducing software product development time. *International Journal of Project Management*, 19, 59-70.

Clark, K. B., Fujimoto, T. 1991. *Product development performance: Strategy, organization and management in the world auto industry*, Harvard Business School Press, Boston.

Cooper R. G., Kleinschmidt, E. J. 1995. Benchmarking the firm's critical success factors in new product development. *Journal of Product Innovation Management*, 12, 374-391.

Coughlan, P. D. 1992. Engineering change and manufacturing engineering deployment in new product development, in Susmann, G. I. (ed.) *Integrating Design and Manufacturing for Competitive Advantage*, Oxford University Press, New York, 157-177.

Curtis, B., Sheppard, S. B., Milliman, P., Borst, M. A., Love, T. 1979. Measuring the psychological complexity of software maintenance tasks with the Halstead and McCabe metrics. *IEEE Transactions on Software Engineering*, SE-5 (2), 96-104.

Fisher, M., Ramdas, K., Ulrich, K. 1999. Component sharing in the management of product variety: A study of automotive braking systems. *Management Science*, 45 (3), 297-315.

Goldratt, E. M., Cox, J. 2004. *The Goal*, 3rd ed., North River Press, Great Barrington.

Griffin, A. 1997. The effect of project and process characteristics on product development cycle time. *Journal of Marketing Research*, 34, 24-35.

Hair, J. F., Black, W. C., Babin, B. J., Anderson, R. E., Tatham, R. L. 2006. *Multivariate data analysis*, 6th ed., Pearson Education Inc., New Jersey.

Halstead, M. H. 1977. *Elements of software science*, Elsevier Science Publishing, New York.

Hatch, N. W., Mowery, D. C. 1998. Process innovation and learning by doing in semiconductor manufacturing. *Management Science*, 44 (11), 1461-1477.

Hayes R. H., Clark, K. B. 1986. Exploring the sources of productivity differences at the factory level, in Clark, K. B., Hayes, R. H., Lorenz, C. (eds.), *The Uneasy Alliance, Managing the Productivity-Technology Dilemma*, Harvard Business School, Boston.

Kafura, D., Reddy, G. R. 1987. The use of software complexity metrics in software maintenance. *IEEE Transactions on Software Engineering*, 13 (3), 335-343.

Keil, S., Lasch, R., Deutschländer, A., Schneider, G. 2007. Einfluss- und Erfolgsfaktoren auf den Ramp Up in der Halbleiterindustrie, in Otto, A. and Obermaier, R. (eds.), *Logistikmanagement,* Deutscher Universitäts-Verlag, Wiesbaden (in German).

Krishnan, M. S., Kriebel, C. H., Kekre, S., Mukhopadhyay, T. 2000. An empirical analysis of productivity and quality in software products. *Management Science*, 46, 745-759.

Krishnan, V., Ulrich, K. T. 2001. Product development decisions: A review of the literature. *Management Science*, 47 (1), 1-21.

Krishnan, V., Zhu, W. 2006. Designing a family of development-intensive products. *Management Science*, 52, 813-825.

Kuhn, A., Wiendahl, H. P., Eversheim W., Schuh, G. 2002. *Fast Ramp Up—Schneller Produktionsanlauf von Serienprodukten*, Verlag Praxiswissen, Dortmund (in German).

Langowitz, N. S. 1987. An exploration of production problems in the initial commercial manufacture of products. *Research Policy*, 17, 43-54.

Lindstrom, L., Jeffries, R. 2004. Extreme programming and agile software development methodologies. *Information Systems Management*, 21 (3), 41-60.

Mallick, D. N., Schroeder, R. G. 2005. An integrated framework for measuring product development performance in high technology industries. *Production and Operations Management*, 14 (2), 142-158.

Matsuo, H., McIvor, R., Martin, J. D., Ng, J. 1997. Profiting from Process Improvements in the New Semiconductor Manufacturing Environment. *Technology and Operations Review*, 1 (2), 1-16.

McCabe, T. J. 1976. A complexity measure. *IEEE Transactions on Software Engineering*, SE-2 (4), 308-320.

Munson, J. C., Khoshgoftaar, T. M. 1991. The use of software complexity metrics in software reliability modeling. *Proceedings of the International Symposium on Software Reliability Engineering*, Austin, TX.

Neely, A., Gregory, M., Platts, K. 1995. Performance management system design, a literature review and research agenda. *International Journal of Operations & Production Management*, 15 (4), 80-116.

Nohria, N., Gulati, R. 1996. Is slack good or bad for innovation? *Academy of Management Journal*, 39, 1245-1264.

Novak, S., Eppinger, S. D. 2001. Sourcing by design: Product complexity and the supply chain. *Management Science*, 47 (1), 189-204.

Pfohl, H. C., Gareis, K. 2000. Die Rolle der Logistik in der Anlaufphase. *Zeitschrift für Betriebswirtschaft*, 70. Jg., H. 11, 1189-1214 (in German).

Pisano, G. P., Wheelwright, S. C. 1995. The new logic of high-tech R&D. *Harvard Business Review*, (September-October), 94-105.

Rodriguez-Toro, C., Jared, G., Swift, K. 2004. Product-development complexity metrics: A framework for proactive-DFA implementation. *International Design Conference–Design 2004*, Cranfield University, United Kingdom.

Schuh, G., Kampker, B, Franzkoch, B. 2005. Anlaufmanagement, Kosten senken – Anlaufzeit verkürzen – Qualität sichern. *Werkstattstechnik online*, Jahrgang 95, H. 5, pp. 405-409 (in German).

Slack, N., Chambers, S., Johnston, R. 2001. *Operations Management*, 3rd ed., Pearson Education Ltd., Harlow.

Swink, M. 1999. Threats to new product manufacturability and the effects of development team integration processes. *Journal of Operations Management*, 17, 691-709.

Tatikonda, M. V., Montoya-Weiss, M. M. 2001. Integrating operations and marketing perspectives of product innovation: The influence of organizational process factors and capabilities on development performance. *Management Science*, 47 (1), 151-172.

Terwiesch, C., Bohn, R. E. 1998. Learning and process improvement during production ramp-up. *Information Storage Industry Centre Report*, 98-01.

Terwiesch, C., Chea, K. S., Bohn, R. E. 1999. An exploratory study of international product transfer and production ramp-up in the data storage industry. *Information Storage Industry Centre Report*, 99-02.

Thomke, S., Bell, D. E. 2001. Sequential testing in product development. *Management Science*, 47 (2), 308-323.

Toni de, A., Tonchia, S. 2001. Performance measurement systems: Models, characteristics and measures. *International Journal of Operations & Production Management*, 21 (1/2), 46-70.

van der Merwe, E. 2004. *A conceptual framework for ramp-up manufacturing*, PhD Thesis, Cambridge University, Cambridge.

Voigt, K. I., Thiell, M. 2005. Fast ramp-up—Handlungs- und Forschungsfeld für Innovations- und Produktionsmanagement", in Wildemann, H. (ed.) *Synchronisation von Produktentwicklung und Produktionsprozess: Produktreife – Produktanläufe – Produktionsauslauf*, TCW Verlag, München (in German).

Wheelwright, S. C., Clark, K. B. 1992. *Revolutionizing product development*, The Free Press, New York.

White, G. P. 1996. A survey and taxonomy of strategy-related performance measures for manufacturing. *International Journal of Operations & Production Management*, 16 (3), 42-61.

Wildemann, H. 2007. *Anlaufmanagement, Leitfaden zur Optimierung der Anlaufphase von Produkten, Anlagen und Dienstleistungen*, TCW-Verlag, München (in German).

Wolf, W. H. 1994. Hardware-Software Co-Design of Embedded Systems. *Proceedings of the IEEE*, 82 (7), 967-989.

Wood, R. E. 1986. Task complexity: Definition of a construct. *Organizational Behavior and Human Decision Processes*, 37, 30-82.

Zuse, H. 1991. *Software complexity, measures and methods*, De Gruyter Verlag, Berlin.

Working Papers Beta 2009 - 2012

nr.	Year	Title	Author(s)
389	2012	The Impact of Product Complexity on Ramp-Up Performance	Andreas Pufall, Jan C. Fransoo, Ad de Jong, Ton de Kok
388	2012	Co-location synergies: specialized versus diverse logistics concentration areas	Frank P.v.d. Heuvel, Peter W.de Langen, Karel H. v. Donselaar, Jan C. Fransoo
387	2012	Proximity matters: Synergies through co-location of logistics establishments	Frank P.v.d. Heuvel, Peter W.de Langen, Karel H. v.Donselaar, Jan C. Fransoo
386	2012	Spatial concentration and location dynamics in logistics:the case of a Dutch province	Frank P. v.d.Heuvel, Peter W.de Langen, Karel H.v. Donselaar, Jan C. Fransoo
385	2012	FNet: An Index for Advanced Business Process Querying	Zhiqiang Yan, Remco Dijkman, Paul Grefen
384	2012	Defining Various Pathway Terms	W.R. Dalinghaus, P.M.E. Van Gorp
383	2012	The Service Dominant Strategy Canvas: Defining and Visualizing a Service Dominant Strategy through the Traditional Strategic Lens	Egon Lüftenegger, Paul Grefen, Caren Weisleder
382	2012	A Stochastic Variable Size Bin Packing Problem With Time Constraints	Stefano Fazi, Tom van Woensel, Jan C. Fransoo
381	2012	Coordination and Analysis of Barge Container Hinterland Networks	K. Sharypova, T. van Woensel, J.C. Fransoo
380	2012	Proximity matters: Synergies through co-location of logistics establishments	Frank P. van den Heuvel, Peter W. de Langen, Karel H. van Donselaar, Jan C. Fransoo
379	2012	A literature review in process harmonization: a conceptual framework	Heidi Romero, Remco Dijkman, Paul Grefen, Arjan van Weele
378	2012	A Generic Material Flow Control Model for Two Different Industries	S.W.A. Haneya, J.M.J. Schutten, P.C. Schuur, W.H.M. Zijm
377	2012	Dynamic demand fulfillment in spare parts networks with multiple customer classes	H.G.H. Tiemessen, M. Fleischmann, G.J. van Houtum, J.A.E.E. van Nunen, E. Pratsini

376	2012	Improving the performance of sorter systems by scheduling inbound containers	K. Fikse, S.W.A. Haneyah, J.M.J. Schutten
375	2012	Strategies for dynamic appointment making by container terminals	Albert Douma, Martijn Mes
374	2012	MyPHRMachines: Lifelong Personal Health Records in the Cloud	Pieter van Gorp, Marco Comuzzi
373	2012	Service differentiation in spare parts supply through dedicated stocks	E.M. Alvarez, M.C. van der Heijden, W.H.M. Zijm
372	2012	Spare parts inventory pooling: how to share the benefits	Frank Karsten, Rob Basten
371	2012	Condition based spare parts supply	X.Lin, R.J.I. Basten, A.A. Kranenburg, G.J. van Houtum
370	2012	Using Simulation to Assess the Opportunities of Dynamic Waste Collection	Martijn Mes
369	2012	Aggregate overhaul and supply chain planning for rotables	J. Arts, S.D. Flapper, K. Vernooij
368	2012	Operating Room Rescheduling	J.T. van Essen, J.L. Hurink, W. Hartholt, B.J. van den Akker
367	2011	Switching Transport Modes to Meet Voluntary Carbon Emission Targets	Kristel M.R. Hoen, Tarkan Tan, Jan C. Fransoo, Geert-Jan van Houtum
366	2011	On two-echelon inventory systems with Poisson demand and lost sales	Elisa Alvarez, Matthieu van der Heijden
365	2011	Minimizing the Waiting Time for Emergency Surgery	J.T. van Essen, E.W. Hans, J.L. Hurink, A. Oversberg
364	2011	Vehicle Routing Problem with Stochastic Travel Times Including Soft Time Windows and Service Costs	Duygu Tas, Nico Dellaert, Tom van Woensel, Ton de Kok
363	2011	A New Approximate Evaluation Method for Two-Echelon Inventory Systems with Emergency Shipments	Erhun Özkan, Geert-Jan van Houtum, Yasemin Serin
362	2011	Approximating Multi-Objective Time-Dependent Optimization Problems	Said Dabia, El-Ghazali Talbi, Tom Van Woensel, Ton de Kok

361	2011	Branch and Cut and Price for the Time Dependent Vehicle Routing Problem with Time Window	Said Dabia, Stefan Röpke, Tom Van Woensel, Ton de Kok
360	2011	Analysis of an Assemble-to-Order System with Different Review Periods	A.G. Karaarslan, G.P. Kiesmüller, A.G. de Kok
359	2011	Interval Availability Analysis of a Two-Echelon, Multi-Item System	Ahmad Al Hanbali, Matthieu van der Heijden
358	2011	Carbon-Optimal and Carbon-Neutral Supply Chains	Felipe Caro, Charles J. Corbett, Tarkan Tan, Rob Zuidwijk
357	2011	Generic Planning and Control of Automated Material Handling Systems: Practical Requirements Versus Existing Theory	Sameh Haneyah, Henk Zijm, Marco Schutten, Peter Schuur
356	2011	Last time buy decisions for products sold under warranty	M. van der Heijden, B. Iskandar
355	2011	Spatial concentration and location dynamics in logistics: the case of a Dutch provence	Frank P. van den Heuvel, Peter W. de Langen, Karel H. van Donselaar, Jan C. Fransoo
354	2011	Identification of Employment Concentration Areas	Frank P. van den Heuvel, Peter W. de Langen, Karel H. van Donselaar, Jan C. Fransoo
353	2011	BOMN 2.0 Execution Semantics Formalized as Graph Rewrite Rules: extended version	Pieter van Gorp, Remco Dijkman
352	2011	Resource pooling and cost allocation among independent service providers	Frank Karsten, Marco Slikker, Geert-Jan van Houtum
351	2011	A Framework for Business Innovation Directions	E. Lüftenegger, S. Angelov, P. Grefen
350	2011	The Road to a Business Process Architecture: An Overview of Approaches and their Use	Remco Dijkman, Irene Vanderfeesten, Hajo A. Reijers
349	2011	Effect of carbon emission regulations on transport mode selection under stochastic demand	K.M.R. Hoen, T. Tan, J.C. Fransoo G.J. van Houtum
348	2011	An improved MIP-based combinatorial approach for a multi-skill workforce scheduling problem	Murat Firat, Cor Hurkens
347	2011	An approximate approach for the joint problem of level of repair analysis and spare parts stocking	R.J.I. Basten, M.C. van der Heijden, J.M.J. Schutten
346	2011	Joint optimization of level of repair analysis and	R.J.I. Basten, M.C. van der Heijden, J.M.J. Schutten

		spare parts stocks	
345	2011	Inventory control with manufacturing lead time flexibility	Ton G. de Kok
344	2011	Analysis of resource pooling games via a new extenstion of the Erlang loss function	Frank Karsten, Marco Slikker, Geert-Jan van Houtum
343	2011	Vehicle refueling with limited resources	Murat Firat, C.A.J. Hurkens, Gerhard J. Woeginger
342	2011	Optimal Inventory Policies with Non-stationary Supply Disruptions and Advance Supply Information	Bilge Atasoy, Refik Güllü, TarkanTan
341	2011	Redundancy Optimization for Critical Components in High-Availability Capital Goods	Kurtulus Baris Öner, Alan Scheller-Wolf Geert-Jan van Houtum
339	2010	Analysis of a two-echelon inventory system with two supply modes	Joachim Arts, Gudrun Kiesmüller
338	2010	Analysis of the dial-a-ride problem of Hunsaker and Savelsbergh	Murat Firat, Gerhard J. Woeginger
335	2010	Attaining stability in multi-skill workforce scheduling	Murat Firat, Cor Hurkens
334	2010	Flexible Heuristics Miner (FHM)	A.J.M.M. Weijters, J.T.S. Ribeiro
333	2010	An exact approach for relating recovering surgical patient workload to the master surgical schedule	P.T. Vanberkel, R.J. Boucherie, E.W. Hans, J.L. Hurink, W.A.M. van Lent, W.H. van Harten
332	2010	Efficiency evaluation for pooling resources in health care	Peter T. Vanberkel, Richard J. Boucherie, Erwin W. Hans, Johann L. Hurink, Nelly Litvak
331	2010	The Effect of Workload Constraints in Mathematical Programming Models for Production Planning	M.M. Jansen, A.G. de Kok, I.J.B.F. Adan
330	2010	Using pipeline information in a multi-echelon spare parts inventory system	Christian Howard, Ingrid Reijnen, Johan Marklund, Tarkan Tan
329	2010	Reducing costs of repairable spare parts supply systems via dynamic scheduling	H.G.H. Tiemessen, G.J. van Houtum

328 2010	Identification of Employment Concentration and Specialization Areas: Theory and Application	F.P. van den Heuvel, P.W. de Langen, K.H. van Donselaar, J.C. Fransoo
327 2010	A combinatorial approach to multi-skill workforce scheduling	Murat Firat, Cor Hurkens
326 2010	Stability in multi-skill workforce scheduling	Murat Firat, Cor Hurkens, Alexandre Laugier
325 2010	Maintenance spare parts planning and control: A framework for control and agenda for future research	M.A. Driessen, J.J. Arts, G.J. v. Houtum, W.D. Rustenburg, B. Huisman
324 2010	Near-optimal heuristics to set base stock levels in a two-echelon distribution network	R.J.I. Basten, G.J. van Houtum
323 2010	Inventory reduction in spare part networks by selective throughput time reduction	M.C. van der Heijden, E.M. Alvarez, J.M.J. Schutten
322 2010	The selective use of emergency shipments for service-contract differentiation	E.M. Alvarez, M.C. van der Heijden, W.H. Zijm
321 2010	Heuristics for Multi-Item Two-Echelon Spare Parts Inventory Control Problem with Batch Ordering in the Central Warehouse	B. Walrave, K. v. Oorschot, A.G.L. Romme
320 2010	Preventing or escaping the suppression mechanism: intervention conditions	Nico Dellaert, Jully Jeunet.
319 2010	Hospital admission planning to optimize major resources utilization under uncertainty	R. Seguel, R. Eshuis, P. Grefen.
318 2010	Minimal Protocol Adaptors for Interacting Services	Tom Van Woensel, Marshall L. Fisher, Jan C. Fransoo.
317 2010	Teaching Retail Operations in Business and Engineering Schools	Lydie P.M. Smets, Geert-Jan van Houtum, Fred Langerak.
316 2010	Design for Availability: Creating Value for Manufacturers and Customers	Pieter van Gorp, Rik Eshuis.
315 2010	Transforming Process Models: executable rewrite rules versus a formalized Java program	Bob Walrave, Kim E. van Oorschot, A. Georges L. Romme

314	2010	Getting trapped in the suppression of exploration: A simulation model	S. Dabia, T. van Woensel, A.G. de Kok
313	2010	A Dynamic Programming Approach to Multi-Objective Time-Dependent Capacitated Single Vehicle Routing Problems with Time Windows	
	2010		
312	2010	Tales of a So(u)rcerer: Optimal Sourcing Decisions Under Alternative Capacitated Suppliers and General Cost Structures	Osman Alp, Tarkan Tan
311	2010	In-store replenishment procedures for perishable inventory in a retail environment with handling costs and storage constraints	R.A.C.M. Broekmeulen, C.H.M. Bakx
310	2010	The state of the art of innovation-driven business models in the financial services industry	E. Lüftenegger, S. Angelov, E. van der Linden, P. Grefen
309	2010	Design of Complex Architectures Using a Three Dimension Approach: the CrossWork Case	R. Seguel, P. Grefen, R. Eshuis
308	2010	Effect of carbon emission regulations on transport mode selection in supply chains	K.M.R. Hoen, T. Tan, J.C. Fransoo, G.J. van Houtum
307	2010	Interaction between intelligent agent strategies for real-time transportation planning	Martijn Mes, Matthieu van der Heijden, Peter Schuur
306	2010	Internal Slackening Scoring Methods	Marco Slikker, Peter Borm, René van den Brink
305	2010	Vehicle Routing with Traffic Congestion and Drivers' Driving and Working Rules	A.L. Kok, E.W. Hans, J.M.J. Schutten, W.H.M. Zijm
304	2010	Practical extensions to the level of repair analysis	R.J.I. Basten, M.C. van der Heijden, J.M.J. Schutten
303	2010	Ocean Container Transport: An Underestimated and Critical Link in Global Supply Chain Performance	Jan C. Fransoo, Chung-Yee Lee
302	2010	Capacity reservation and utilization for a manufacturer with uncertain capacity and demand	Y. Boulaksil; J.C. Fransoo; T. Tan
300	2009	Spare parts inventory pooling games	F.J.P. Karsten; M. Slikker; G.J. van Houtum
299	2009	Capacity flexibility allocation in an outsourced supply chain with reservation	Y. Boulaksil, M. Grunow, J.C. Fransoo
298	2010	An optimal approach for the joint problem of level of repair analysis and spare parts stocking	R.J.I. Basten, M.C. van der Heijden, J.M.J. Schutten
297	2009	Responding to the Lehman Wave: Sales Forecasting and Supply Management during the Credit Crisis	Robert Peels, Maximiliano Udenio, Jan C. Fransoo, Marcel Wolfs, Tom Hendrikx
296	2009	An exact approach for relating recovering surgical patient workload to the master surgical schedule	Peter T. Vanberkel, Richard J. Boucherie, Erwin W. Hans, Johann L. Hurink, Wineke A.M. van Lent, Wim H. van Harten

295	2009	An iterative method for the simultaneous optimization of repair decisions and spare parts stocks	R.J.I. Basten, M.C. van der Heijden, J.M.J. Schutten
294	2009	Fujaba hits the Wall(-e)	Pieter van Gorp, Ruben Jubeh, Bernhard Grusie, Anne Keller
293	2009	Implementation of a Healthcare Process in Four Different Workflow Systems	R.S. Mans, W.M.P. van der Aalst, N.C. Russell, P.J.M. Bakker
292	2009	Business Process Model Repositories - Framework and Survey	Zhiqiang Yan, Remco Dijkman, Paul Grefen
291	2009	Efficient Optimization of the Dual-Index Policy Using Markov Chains	Joachim Arts, Marcel van Vuuren, Gudrun Kiesmuller
290	2009	Hierarchical Knowledge-Gradient for Sequential Sampling	Martijn R.K. Mes; Warren B. Powell; Peter I. Frazier
289	2009	Analyzing combined vehicle routing and break scheduling from a distributed decision making perspective	C.M. Meyer; A.L. Kok; H. Kopfer; J.M.J. Schutten
288	2009	Anticipation of lead time performance in Supply Chain Operations Planning	Michiel Jansen; Ton G. de Kok; Jan C. Fransoo
287	2009	Inventory Models with Lateral Transshipments: A Review	Colin Paterson; Gudrun Kiesmuller; Ruud Teunter; Kevin Glazebrook
286	2009	Efficiency evaluation for pooling resources in health care	P.T. Vanberkel; R.J. Boucherie; E.W. Hans; J.L. Hurink; N. Litvak
285	2009	A Survey of Health Care Models that Encompass Multiple Departments	P.T. Vanberkel; R.J. Boucherie; E.W. Hans; J.L. Hurink; N. Litvak
284	2009	Supporting Process Control in Business Collaborations	S. Angelov; K. Vidyasankar; J. Vonk; P. Grefen
283	2009	Inventory Control with Partial Batch Ordering	O. Alp; W.T. Huh; T. Tan
282	2009	Translating Safe Petri Nets to Statecharts in a Structure-Preserving Way	R. Eshuis
281	2009	The link between product data model and process model	J.J.C.L. Vogelaar; H.A. Reijers
280	2009	Inventory planning for spare parts networks with delivery time requirements	I.C. Reijnen; T. Tan; G.J. van Houtum
279	2009	Co-Evolution of Demand and Supply under Competition	B. Vermeulen; A.G. de Kok
278	2010	Toward Meso-level Product-Market Network Indices for Strategic Product Selection and (Re)Design Guidelines over the Product Life-Cycle	B. Vermeulen, A.G. de Kok
277	2009	An Efficient Method to Construct Minimal Protocol Adaptors	R. Seguel, R. Eshuis, P. Grefen
276	2009	Coordinating Supply Chains: a Bilevel Programming Approach	Ton G. de Kok, Gabriella Muratore
275	2009	Inventory redistribution for fashion products	G.P. Kiesmuller, S. Minner

		under demand parameter update	
274	2009	Comparing Markov chains: Combining aggregation and precedence relations applied to sets of states	A. Busic, I.M.H. Vliegen, A. Scheller-Wolf
273	2009	Separate tools or tool kits: an exploratory study of engineers' preferences	I.M.H. Vliegen, P.A.M. Kleingeld, G.J. van Houtum
272	2009	An Exact Solution Procedure for Multi-Item Two-Echelon Spare Parts Inventory Control Problem with Batch Ordering	Engin Topan, Z. Pelin Bayindir, Tarkan Tan
271	2009	Distributed Decision Making in Combined Vehicle Routing and Break Scheduling	C.M. Meyer, H. Kopfer, A.L. Kok, M. Schutten
270	2009	Dynamic Programming Algorithm for the Vehicle Routing Problem with Time Windows and EC Social Legislation	A.L. Kok, C.M. Meyer, H. Kopfer, J.M.J. Schutten
269	2009	Similarity of Business Process Models: Metics and Evaluation	Remco Dijkman, Marlon Dumas, Boudewijn van Dongen, Reina Kaarik, Jan Mendling
267	2009	Vehicle routing under time-dependent travel times: the impact of congestion avoidance	A.L. Kok, E.W. Hans, J.M.J. Schutten
266	2009	Restricted dynamic programming: a flexible framework for solving realistic VRPs	J. Gromicho; J.J. van Hoorn; A.L. Kok; J.M.J. Schutten;

Working Papers published before 2009 see: http://beta.ieis.tue.nl