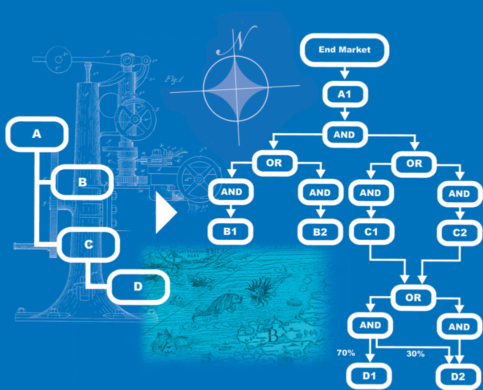
Department of Industrial Engineering and Management

An Effective Tool for Supply Chain Decision Support during New Product Development Process

Teemu Tynjälä



Aalto University

DOCTORAL DISSERTATIONS

Aalto University publication series
DOCTORAL DISSERTATIONS 73/2011

An Effective Tool for Supply Chain Decision Support during New Product Development Process

Teemu Tynjälä

Doctoral dissertation for the degree of Doctor of Science in Technology to be presented with due permission of the School of Science for public examination and debate in Auditorium AS1 in TUAS-building at the Aalto University School of Science (Espoo, Finland) on the 7th of October 2011 at 12 noon.

Aalto University
School of Science
Department of Industrial Engineering and Management

Supervisor
Professor Eero Eloranta

Instructor
Professor Eero Eloranta

Preliminary examiners
Professor emeritus Leo Ojala, Aalto University School of Science, Finland
Docent Tapani Savolainen, CAC-Research, Finland

Opponent
Professor Peter O'Grady, The University of Iowa, USA

Aalto University publication series
DOCTORAL DISSERTATIONS 73/2011

© Teemu Tynjälä

ISBN 978-952-60-4255-8 (pdf)
ISBN 978-952-60-4254-1 (printed)
ISSN-L 1799-4934
ISSN 1799-4942 (pdf)
ISSN 1799-4934 (printed)

Aalto Print
Helsinki 2011

Finland

The dissertation can be read at http://lib.tkk.fi/Diss/

Aalto University

Aalto University, P.O. Box 11000, FI-00076 Aalto www.aalto.fi

Abstract

Author
Teemu Tynjälä

Name of the doctoral dissertation
An Effective Tool for Supply Chain Decision Support during New Product Development Process

Publisher School of Science

Unit Department of Industrial Engineering and Management

Series Aalto University publication series DOCTORAL DISSERTATIONS 73/2011

Field of research Industrial Management

Manuscript submitted 29 April 2011 **Manuscript revised** 29 April 2011

Date of the defence 7 October 2011 **Language** English

☐ **Monograph** ☒ **Article dissertation (summary + original articles)**

Abstract

The global marketplace has transformed supply chain design into a discipline which requires business sense supported by mathematical expertise. Several methods have been introduced to support supply chain design, most notably mixed integer programming. The current methods are tailor-made for situations where a product's bill-of-material is fixed. However, this assumption does not hold during product development where several competing product designs exist. Therefore this research investigates the question of what is an effective way to support supply chain decisions during new product development. The study is divided into four research questions, corresponding to the articles from which the dissertation is compiled: (1) Does a product structure-driven method exist for modeling and analyzing supply chains? (2) If such a method is discovered, what is its mathematical formulation? (3) Is there evidence to support the theoretical and practical usability of such a method? (4) How can strategic supply chain decisions be validated?

Regarding question (1) the research finds that there is a shortage of methods that fulfill supply chain modeling and analysis requirements imposed by new product development process. During the research a Petri-net based method was constructed which satisfies these requirements. For question (2), the formal definitions of the constructed Petri net class are provided. Regarding question (3), the research finds that the created method and associated tool are useful aids when solving the question of the effect of demand variation and the number of product variants on the optimal supply chain. Furthermore, interviews with end users of the tool implementation provide evidence of the Petri net method's practical usefulness. Regarding question (4), the research finds that the validation of strategic supply chain decisions from companies' reporting systems is important, but it has not become a common practice due to the challenges in integrating various IT systems.

Keywords Supply chain management, decision support systems, Petri nets

ISBN (printed) 978-952-60-4254-1 **ISBN (pdf)** 978-952-60-4255-8

ISSN-L 1799-4934 **ISSN (printed)** 1799-4934 **ISSN (pdf)** 1799-4942

Location of publisher Espoo **Location of printing** Helsinki **Year** 2011

Pages 167 **The dissertation can be read at** http://lib.tkk.fi/Diss/

Aalto-yliopisto
Aalto-yliopisto, PL 11000, 00076 Aalto www.aalto.fi

Tiivistelmä

Tekijä
Teemu Tynjälä

Väitöskirjan nimi
An Effective Tool for Supply Chain Decision Support during New Product Development Process

Julkaisija Perustieteiden korkeakoulu

Yksikkö Tuotantotalouden laitos

Sarja Aalto University publication series DOCTORAL DISSERTATIONS 73/2011

Tutkimusala Teollisuustalous

Käsikirjoituksen pvm 29.04.2011 **Korjatun käsikirjoituksen pvm** 29.04.2011

Väitöspäivä 07.10.2011 **Kieli** Englanti

☐ **Monografia** ☒ **Yhdistelmäväitöskirja (yhteenveto-osa + erillisartikkelit)**

Tiivistelmä

Globalisaation ansiosta toimitusketjujen suunnittelu on muovautunut tieteeksi jossa vaaditaan sekä liiketoimintaosaamista että matemaattista kykyä. Toimitusketjujen matemaattiseen suunnitteluun on tarjolla useita metodeja, joista sekalukuoptimointi on saanut eniten huomiota. Nykyiset menetelmät soveltuvat parhaiten analyyseihin, joissa tuotteen rakenne pysyy vakiona. Edellinen oletus ei päde tuotesuunnittelun aikana, jolloin on olemassa monta kilpailevaa tuotearkkitehtuuria. Tämä tutkimus keskittyy löytämään metodologian tuotesuunnittelun aikaiseen toimitusketjujen analysointiin ja optimointiin. Tutkimus on jaettu neljään kysymykseen osajulkaisuja vastaavasti: (1) Onko olemassa tuoterakennelähtöisiä toimitusketjujen mallinnus- ja analysointimetodeja? (2) Jos tällainen metodi on olemassa, mikä on sen matemaattinen kuvaus? (3) Onko olemassa näyttöä ko. metodin teoreettisesta ja käytännön hyödystä? (4) Kuinka strategiset toimitusketjupäätökset voidaan osoittaa kelvolliseksi?

Liittyen kysymykseen (1), tutkimus osoitti, että nykyiset toimitusketjujen mallinnus- ja analysointimetodit eivät täytä kaikkia tuotekehitysprosessin asettamia vaatimuksia. Tässä työssä kehitettiin Petri-verkko perusteinen, tuoterakennelähtöinen metodi toimitusketjujen mallintamiseen ja analysointiin. Tämän Petri-verkon matemaattiset määrittelyt vastaavat tutkimuskysymykseen (2). Kehitettyä metodia sovellettiin toimitusketjuongelmaan, joka tutkii tuotevarianttien määrän ja loppukysynnän vaihtelun vaikutuksia optimaaliseen toimitusketjuun. Lähestymistavan käytännön hyötyä tutkittiin haastattelemalla metodiin perustuvan työkalun loppukäyttäjiä esimerkkiyrityksessä. Yhdessä nämä kaksi tapaustutkimusta vastaavat tutkimuskysymykseen (3). Liittyen kysymykseen (4), tutkimus havaitsi, että strategisten toimitusketjupäätösten kelvolliseksi osoittaminen on yrityksille tärkeää, mutta sitä ei tätä nykyä kyetä tekemään laajamittaisesti järjestelmäintegraation haasteiden vuoksi.

Avainsanat toimitusketjunhallinta, päätöksenteon tuen apuvälineet, Petri verkot

ISBN (painettu) 978-952-60-4254-1 **ISBN (pdf)** 978-952-60-4255-8

ISSN-L 1799-4934 **ISSN (painettu)** 1799-4934 **ISSN (pdf)** 1799-4942

Julkaisupaikka Espoo **Painopaikka** Helsinki **Vuosi** 2011

Sivumäärä 167 **Luettavissa verkossa osoitteessa** http://lib.tkk.fi/Diss/

Acknowledgements

This work began as a stroke-of-luck where 'a friend of a friend of a friend' at Nokia linked me to the *Methodologies and Tools for End-to-End Demand Supply Network Analyses* project. A project which originally started as a tool implementation grew subsequently to research work.

Many people at Nokia have been helping to get this thesis completed. The contribution has ranged from reading and commenting on manuscripts to obtaining raw data for analyses. Among the contributors are Dr. Ian Oliver, Mr. Vesa Luukkala, Dr. Mikko Rajala, Dr. Jari Ruokolainen, Mr. Mika Ojala, and Mr. Matti Torkkeli. My current superior Mr. Hannu Pahkala deserves thanks for allowing me to take time off work to finalize the thesis.

Professor Eero Eloranta deserves great thanks in guiding this research process to completion. Through his accurate and honest feedback, the articles and this cover story reached their final form.

Financially speaking, completing the thesis would not have been possible without the study grant from the Finnish Doctoral Program in Industrial Engineering and Management (DPIEM). Also, I wish to extend a long-overdue thanks to Tekniikan Edistämissäätiö (TES) for their grant given in 2001.

Most importantly, I would like to thank my family and friends for the great support they gave during the writing process. Naturally, thanks belong to my parents Asko and Saara, and my brother Miika. Special thanks belong to my lovely wife, Dr. Pirjo Tynjälä, who motivated me to complete this dissertation through her own example. She had just the right methods to pick up my spirits when times were tough.

Finally, science is but a small part of life and priorities should lie elsewhere. This is why I want to end this section with a quote from King Solomon.

> "Of making books there is no end, and much study wearies the body. Now all has been heard; here is the conclusion of the matter: Fear God and keep his commandments for this is the duty of all mankind." (Eccl. 12:12-13)

Espoo, September 2011
Teemu Tynjälä

Contents

List of original publications

This thesis is based on the following publications:

I Tynjälä, T. (2005). A formal, product structure driven design of optimized end-to-end demand supply chains. *Journal of Systemics, Cybernetics & Informatics, 3(1),* 7-11. Available online at www.iiisci.org/Journal/SCI/Past.asp

II Tynjälä, T. (2009). Supporting demand supply network optimization with Petri nets. In V. Sugumaran (Ed.), *Distributed artificial intelligence, agent technology, and collaborative applications,* (pp. 366-83). Hershey, PA: IGI Global.

III Tynjälä, T. & Eloranta, E. (2007). Investigating the effect of product variants, and demand distributions on the optimal demand supply network setup. *Production Planning & Control, 18(7),* 561-72.

IV Tynjälä, T. Validation in Supply Chain Decision Support Systems. Accepted for publication in *International Journal of Information Systems and Supply Chain Management*.

The publications are referred to in the text by their Roman numerals.

Abbreviations

ABC	Activity Based Costing
BI	Business Intelligence
BOM	Bill-of-Materials
(3D-)CE	(3 Dimensional) Concurrent Engineering
DC	Distribution Center
DCM	Demand Chain Management
DOS	Days-of-Supply
DSN	Demand Supply Network
DSS	Decision Support System
DW	Data Warehouse
EIS	Executive Information System
ERP	Enterprise Resource Planning
GDSS	Group Decision Support System
KPI	Key Performance Indicator
NPD	New Product Development
OM	Operations Management
PDSS	Personal Decision Support System
PDM	Product Data Management
SCS	JDA Supply Chain Strategist
SKU	Stock-Keeping Unit

1 INTRODUCTION

In this chapter, the motivation for the study is presented along with the research questions and scope. The format for the rest of the dissertation is also given.

1.1 Background and research environment

Managing the supply chain is a demanding discipline requiring business instinct and mathematical expertise. In the early days of mass production there was a limited amount of requirements set on products sold. The number of variants was small, and production could be done in a push mode, as the demand was high due to the cost advantage of mass production. In the very early days of manufacturing, suppliers were located in the same towns and villages which further eased the management of the supply chain.

Since then, almost everything has changed. Today, customers place heavy requirements on the quality and variety of manufactured products and are willing to change their manufacturer of choice quickly. Successful supply chains are forced to use on-shoring and off-shoring in an intelligent way to satisfy both the cost and response time requirements. The lifecycle of products, especially in consumer electronics industry, is growing shorter all the time (Fine, 2000; Fine et al., 2002). Today management of the supply chain is a continuous activity where the situation at hand may change every hour.

The literature proposes a few ways of solving the dilemma of creating a cost-efficient and responsive supply chain. One of the ideas is to construct two types of supply chains: lean and agile (Christopher & Towill, 2002). Lean supply chains attempt to cut all waste from a supply chain to enable the lowest total cost (Womack, Jones, & Roos, 1990; Womack & Jones, 1996). Agile supply chains, on the other hand, attempt to win orders through superior customer service (Nagel & Dove, 1991). The authors suggest that lean supply chains should be used to deliver commodity products and agile ones should deliver the highly profitable ones. The work of Fisher (1997) supports this observation. He claims that the type of product should dictate its supply chain. Two classes of products are specified: functional and innovative. According to Fisher, functional products should use lean supply chains and market-responsive products should be delivered through agile supply chains. These claims have been tested in practice and partly validated (Selldin & Olhager, 2007). The claim regarding functional products and lean supply chains was supported, but unequivocal evidence for connecting innovative products with responsive supply chains was not found.

In light of what has been said, it is common for companies to have multiple types of supply chains in use. Lean and agile supply chains can be used in combination (Naylor, Naim, & Berry, 1999). This parallel use may be based on the functional vs. innovative product classification. However, the supply chain of a single product may contain a lean part and an agile part. This is achieved through the use of an order penetration point (Olhager, 2003). The material flow (typically subassemblies and components) before the order penetration point is driven according to the lean paradigm, and the supply chain beyond the order penetration point is driven respecting the agile paradigm. Using this type of a supply chain requires product modularization into commodity components and order-specific varying parts. Although the demand volatility for specific end products is high, the demands for common components used in a product family may be estimated at a fairly high accuracy (Simchi-Levi, Kaminsky, & Simchi-Levi, 2004). If a product family uses a maximal number of common components, the forecasts for these components can be estimated surprisingly accurately. Thus the supply chain for subassemblies and components can be run according to lean principles. The agile supply chain paradigm is required beyond the order penetration point to quickly deliver the specific product variants to the end customers. Some researchers have also proposed that choosing the type of the supply chain depends on the lifecycle stage of a product (Aitken et al., 2005). During the beginning and the end of the lifecycle, the supply chain should be particularly agile. At the mid-stage of the lifecycle, the supply chain should be made as lean as possible.

Determining the optimal supply chain for a product is difficult, and it is made even more so because of trade protectionism, exchange rate fluctuations, and geopolitics (Christopher, Peck, & Towill, 2006). Duties and taxes are used to protect local manufacturing against import. In the worst case, duty and tax rates are changed several times per year, and the product that should have been competitive in the marketplace may become overpriced overnight. It is very difficult to make a supply chain decision for a 5-year horizon with such contingency factors. Economic boom and bust periods affect exchange rates and offshoring production in such an environment is always risky. The offshoring countries are facing ever-growing wage pressures, and the economic benefit of relocating production may diminish quickly. All of these unpredictable external factors are relevant for supply chain analyses, and make the design of a robust supply chain a challenge.

This discussion has thus far concentrated on the supply chain of a designed product. However, supply chain decisions are of paramount importance during new product development (Krishnan & Ulrich, 2001). It is during this time that various internal and external product designs are tried and tested, and the corresponding supply chains should be rated according to the end-to-end costs and anticipated response times. Design choices typically lead to the consideration of many component supplier alternatives. Occasionally a choice will have to be made between cost-effective suppliers which are located in countries prone to political unrest, and more expensive suppliers located in stable countries. A method is needed to place a monetary figure on such choices. The anticipated demands for a product may also change drastically during its design. For instance, if there is a key component supplier whose available capacity cannot support an up-swing in demand, the supply chain is in trouble. The claim of the present work is that end-to-end supply chain analyses should be done throughout the product design process, and such a method should enable quick re-analyses if changes take place in any variable mentioned. After the supply chain decision has been taken, there is also a need to do a follow-up on the quality of the decision. If during the early product delivery phase it is found that the supply chain costs are prohibitively higher than what was predicted, a supply chain redesign should be done immediately.

In light of the phenomena discussed it is quite evident that support for supply chain decision making is needed more than ever. In fast-moving businesses a wrong supply chain decision can have quick and severe consequences on a firm's profitability. The inspiration for the current research began during the "Methodologies and tools for end-to-end demand supply network analyses" project at Nokia. The requirement at that time was to find a methodology to support product-specific demand supply network analyses during product creation. This requirement, which was then company specific, started a process where the question was explored to determine whether it constitutes a generic research problem and what are the best solutions for it. In the following, Nokia will be referred to as the *case company*.

1.2 Research objectives

The objective of this dissertation is to examine the methods and tools for supply chain decision making during new product development. First a set of desirable attributes for an effective supply chain modeling method is derived, and an examination of the currently available methods and tools is carried out. If the research finds that no current tools fulfill all the requirements, the task of the present work is to construct a suitable one. The rationale for carrying out such research is inadequacy in research concerning the optimal manner in which supply chain optimization and simulation should be done during new product development (Fine, Golany, Naseraldin, 2005; Chiu, Gupta, & Okudan, 2009). The use of optimization and simulation methods in solving distribution planning problems has been done for decades (Geoffrion & Powers, 1995), but these studies have omitted the inherent requirements of new product development.

The second objective of the dissertation is to examine the validation of strategic supply chain decision support systems. Supply chain budget variations are among companies' most important key performance indicators (KPI), which are used as a key justification for this part of the research (Gunasekaran, Patel, & McGoughey, 2004). A research gap in this area has also been acknowledged, e.g., by Jonsson, Kjellsdotter, and Rudberg (2007).

1.3 Research approach and dissertation structure

The encompassing question focused on in this dissertation is the following: "What is an effective way of supporting supply chain optimization during new product development?" A three-part process is used to answer this main research question. The first focuses on understanding the inherent requirements imposed by the new product development process on supply chain optimization. As a result, a list of five requirements is formed. Through literature study it is found that the current methods do not support all of these requirements and a justification for constructing a new method is obtained. Articles I and II encompass the first part of the research process, and they answer two questions that are derived from the main research question, “does a product-driven formalism for optimizing supply chains exist?” and “what is its mathematical formulation?”, respectively.

The second part of the research process answers a research question “does evidence exist to support the theoretical usefulness and practical usability of the constructed method?” Two case studies are carried out regarding this question. In the first, the newly constructed formalism is applied to solve a supply chain management problem concerning the effect of the number of product variants and demand fluctuations on the optimal supply chain. In the second, the results of method and tool validation through end user interviews are presented. The second part of the research process encompasses article III.

The final part of the research process addresses the question: “how can strategic supply chain decisions be validated?” It posits that the problem of strategic supply chain decision support system validation has not been adequately investigated. It proceeds to carry out a detailed case study inside the case company, and a set of comparative case studies based on interviews with other companies.

The thesis is divided into chapters as follows. Chapter 2 presents the common literature review for the entire thesis. Chapter 3 presents the results from articles I and II. Chapter 4 presents the results of article III. Chapter 5 presents the findings of article IV. Chapter 6 presents the discussion of results, and a short summary of the entire thesis is provided in Chapter 7.

2 COMMON LITERATURE REVIEW

This section presents a common literature review for the entire thesis. It describes the questions that must be solved during new product development, and the concepts involved in supply chain design.

2.1 New product development

New product development (NPD) is one of the key processes a successful company must repeatedly carry out to succeed in the marketplace (Krishnan & Ulrich, 2001; Fine, Golany, & Naseraldin, 2005; Jiao, Simpson, & Siddique, 2007; Chiu, Gupta, & Okudan, 2009). At a high level, new product development refers to a process that transforms "a market opportunity and a set of assumptions about product technology into a product available for sale" (Krishnan & Ulrich, 2001, p. 1). Most of the key functions of a company are involved in product development: Marketing considers the fit of the new product into the company's portfolio and its position among competitors; Project organization considers the resourcing of teams to develop the new product; Engineering function considers the available technologies for the product; Finally, operations management designs the supply chain structure and factory processes to manufacture the new product.

Concept development refers to the feature list and the price point of a new product (Krishnan & Ulrich, 2001). A product concept may be fulfilled by several technologies provided by multiple suppliers. Supply chain considerations enter into the picture at this point of product design. Specifically, new product development process considers supply chain-specific questions (Krishnan & Ulrich, 2001, p. 5)

- Which components will be designed and which will be selected from existing supplier offerings?
- Who will design the components?
- Who will produce the components and assemble the product?
- Which variants of the product will be offered?
- What is the configuration of the physical supply chain, including the location of the decouple point?

New product development is challenging from the supply chain design point of view, because three major entities are varying concurrently: product architecture, manufacturing processes, and supplier base (Fine, Golany, & Naseraldin, 2005). With the product architecture fixed, there still exist numerous supply chain options as the manufacturing company may give a number of choices for allowed manufacturing sites, and component suppliers. Supply chain analyses should be performed both, within a single product architecture, and between competing product architectures. Thus, new product development and supply chain management naturally intertwine.

2.2 Supply chain management and its selected problem areas

Before the notion of Supply Chain Management became widely accepted, it had to differentiate itself from two related concepts. At the beginning of the industrial era, an individual company was the focal point of analysis. This gave rise to the formation of the term Operations Management (OM). OM is defined as "design, execution and control of a firm's operations that convert its resources into desired goods and services, and implement its business strategy" (BusinessDictionary, 2011). OM targets mainly the internal efficiency of a firm's operations, for instance, production planning and scheduling. As the use of external suppliers and distributors became more widespread, the term Logistics Management was born. The Council of Supply Chain Management Professionals defines it as follows:

> Logistics Management is that part of Supply Chain Management that plans, implements, and controls the efficient, effective forward and reverse flow and storage of goods, services and related information between the point of origin and the point of consumption in order to meet customers' requirements. (CSCMP, 2011)

Logistics management added the perspectives of upstream and downstream inventory and transportation management to the company-centric view held by Operations Management. Logistics management recognizes three flows in the supply chain: material flow, information flow, and financial flow. However, it does not address the strategic development and improvement of supply chains.

Supply Chain Management (SCM) is defined as the "systematic, strategic coordination of the traditional business functions and the tactics across these business functions within a particular company and across businesses within a supply chain, for the purposes of improving the long-term performance of the individual companies and the supply chain as a whole" (Mentzer et al., 2001). Supply chain management works hand-in-hand with the long-term strategy of a company. For instance, if the company's strategy changes to include a new product business, it is the job of SCM to determine and build the needed manufacturing sites, supplier channels and distributor channels. Similarly, SCM has the mandate to continuously monitor the health of the supply chain, and make interventions, such as building or buying additional production capacity, as needed.

After SCM had been embraced by the industry, yet another definition was coined, namely Demand Supply Network management (Hoover et al., 2001). Demand supply network management is the combination of Supply Chain Management and Demand Chain Management (DCM). DCM is concerned with the manner in which the demand signal from end customers is cascaded to the component suppliers through the focal company. Similarly, it suggests optimal placements of order penetration points to fit customer needs. For the present study, the notion of Demand Supply Network management is too broad. The problems considered in this dissertation belong to the domain of Supply Chain Management, which will be described in more detail.

From SCM perspective, the typical components in a company's supply chain include (Goetschalckx, Vidal, & Dogan, 2002):

1) The manufacturing plants
2) Zero, one or more distribution echelons with distribution centers
3) The customers

4) The suppliers of components and raw materials
5) Recycling centers for used products and returned packaging containers
6) The transportation channels that link all the previous components

Generally speaking, a manufacturing company makes decisions in three areas: procurement, manufacturing and distribution (Guedes, 1994). Procurement decisions are made about the number and location of component suppliers, and the optimal component inventories at manufacturing locations. The key points of decisions regarding manufacturing strategy concern the number and location of needed manufacturing sites, the allocation of products to each factory, and the technological capability level at each factory. Detailed distribution planning decisions deal with the number and size of depots, the assignment of SKU's to the depots, and the inventory levels and the allocation of demand geographies to each depot.

Solving these strategies gives rise to three types of problems in supply chain management: the facility location problem, the network flow / demand allocation problem, and the distribution planning problem (Guedes, 1994). The first attempts to find the best locations for manufacturing plants, supplier locations, and distributor locations while taking into account the market regions and anticipated customer demands. The second problem type takes an existing network together with its capacity and other constraints, and attempts to allocate the production and distribution of SKU's in an optimal way to satisfy customer demand. Finally, the distribution planning problem is formulated to allocate the right products to each depot, and set inventory targets in a way that transportation fleets may be utilized effectively to achieve the required customer service levels. The three subproblems and their key input parameters are discussed next.

2.2.1 Facility location problems

The facility location problem has been known since the beginning of the 20th century. Weber defined this problem as: "The question of the location of industries is part of the general problem of the local distribution of economic activities. In each economic organization and in each stage of technical and economic evolution there must be a "somewhere" as well as a "somehow" of production, distribution, and consumption" (Weber, 1909).

In general, facility location problems are concerned with finding the optimal **number**, **size**, **location**, and **service area** of facilities that minimize the costs to serve the customers (Guedes, 1994). The solution may be made based on two principles: all geographical points on a 2-dimensional plane are possible depot locations, or second, the possible depot locations are given as a finite set. The latter one has been more prominent in literature (Melo, Nickel, & Saldanha-da-Gama, 2009). Solutions to the facility location problem may additionally suggest space allocations between products in facilities, production requirements in facilities, high-level sourcing requirements, and high-level distribution strategy.

The facility location problem is one of the most strategic questions a company has to solve, as it involves decisions on adding new facilities and possibly relocating existing ones. The granularity of the data used in facility location problems is typically coarse and deals with anticipated demands and possible supplier and manufacturing locations five to ten years into the future. Demands are specified on a major geographical region level (e.g. North America, South East Asia and Pacific), and the location of suppliers and manufacturing facilities is described on the country level. It is generally assumed that the problem is solved for the case of multiple facility echelons, and multiple commodities (Van Roy, 1989; Kaufman, Eede, & Hansen, 1977). A simplified illustration of the facility location problem is given in Figure 1.

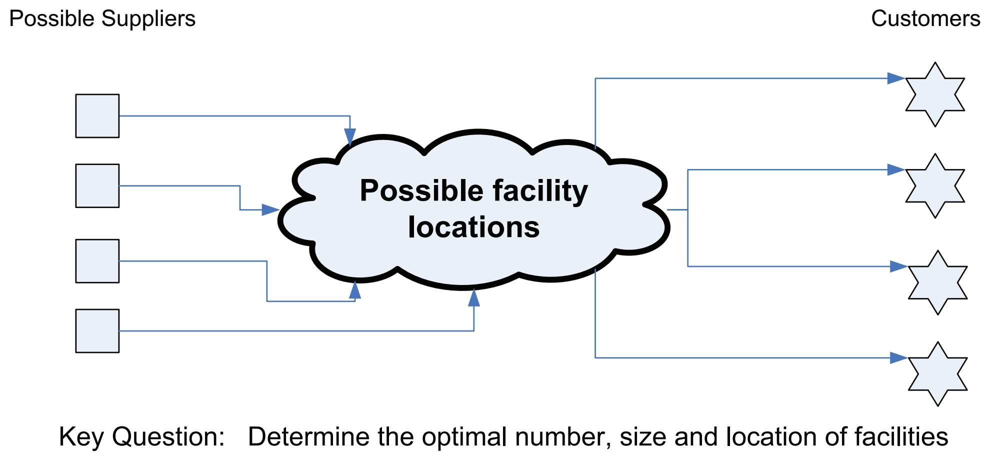


Figure 1 *Illustration of the facility location problem*

2.2.2 Network flow / demand allocation problems

Once the number, size and location of a focal company's facilities are fixed, the problem of satisfying customer demand with an existing constrained network is formulated as the network flow / demand allocation problem (Cohen & Lee, 1989). The network flow / demand allocation problem addresses the questions of "which markets to serve with what quantities from the existing distribution facilities?", "what is the optimal allocation of products and volumes to each production plant?", and "which supplier plants are to be used as component sources?". A network flow / demand allocation problem may be formulated to solve, for instance, the optimal way of satisfying global customer demand for a product family in the next 12 months.

The input data granularity for a network flow / demand allocation problem is finer than in facility location problems. The possible supplier and factory locations are given using precise cities, and the customers are specified on a country level. Figure 2 illustrates the network flow / demand allocation problem.

Network flow / demand allocation is the key problem setting used during new product development as supplier choices, factory capabilities, factory capacities, and anticipated demands vary.

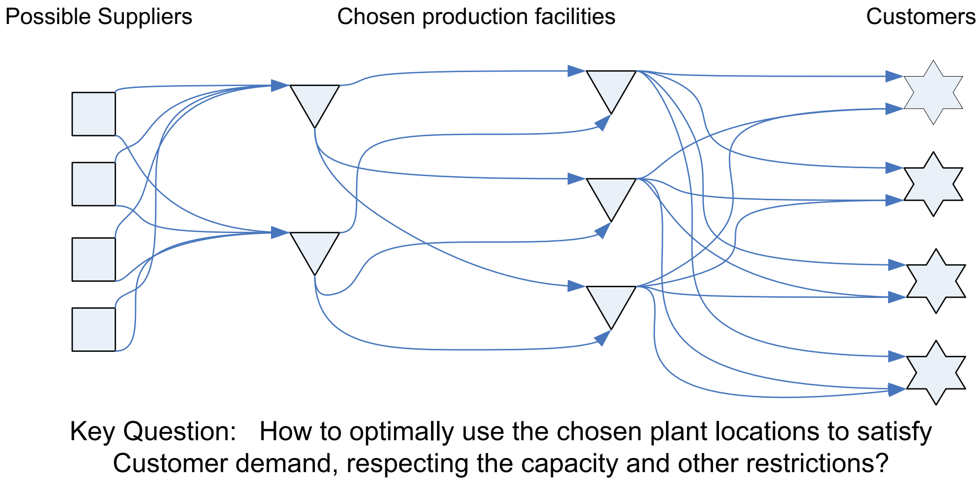


Figure 2 *Illustration of the network flow / demand allocation problem*

2.2.3 Distribution planning problems

Distribution planning problem considers the number, size and location of depots to serve the customers optimally (Geoffrion, Graves, & Lee, 1978). It concentrates on optimally solving the "last link" of the supply chain. It suggests which SKUs should be stocked in each depot, and what inventory levels at depots achieve the best balance between inventory holding costs and customer service level. Solutions to this problem also suggest which customers should be served from each depot and which transportation modes should be employed.

The input data for the distribution planning problem is very fine grained. Demand data per SKU may be given for each account customer in daily or weekly timeframes. Accurate hour-by-hour transport schedules are used by algorithms to compute maximal delivery consolidations at each depot. Distribution planning takes place in an operational time frame, and a solution algorithm may be run every week to adjust the inventory levels and transportation plans. Figure 3 shows a schematic for the distribution planning problem.

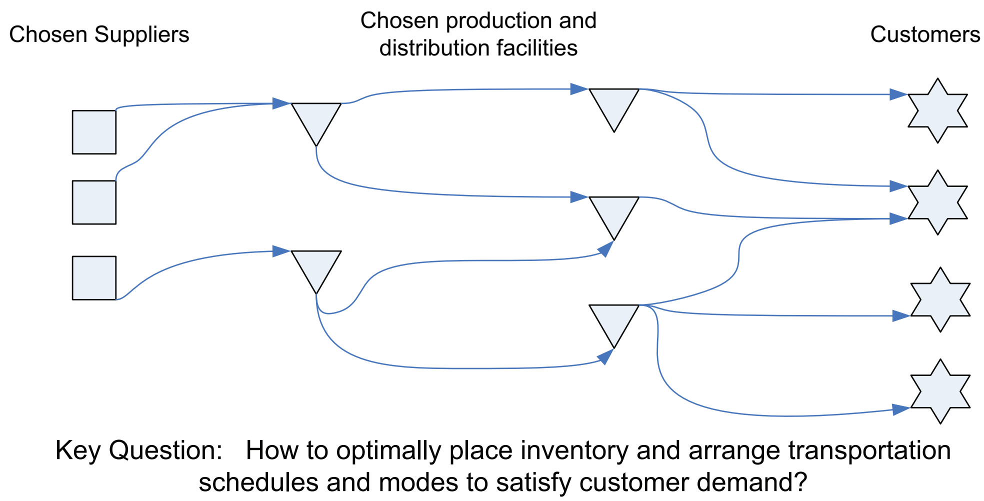


Figure 3 *Illustration of the distribution planning problem*

2.2.4 Key inputs to supply chain problems

The body of knowledge regarding solution methods to the above supply chain problems is extensive (e.g. Melo, Nickel, & Saldanha-da-Gama, 2009; Goetschalckx & Fleischmann, 2008; Meixell & Gargeya, 2005; Goetschalckx, Vidal, & Dogan, 2002; Vidal & Goetschalckx, 1997; Thomas & Griffin, 1996; Geoffrion & Powers, 1995; Guedes, 1994; Cohen & Lee, 1989; Van Roy, 1989). The first two problems, facility location and network flow / demand allocation, may be solved with similar methods, most notably mixed integer programming. The last problem, detailed distribution planning, may be approached with discrete event simulation and simulation optimization in addition to mixed integer programming.

Early authors solved the above three problems for a single time period and a single product (Guedes, 1994). Many tools posed restrictions on the maximum number of distribution echelons and the use of Bill-of-Materials (ibid.). Today nearly all solution methods are able to solve problems with arbitrary distribution echelon structure, multiple commodities, and multiple time periods (Melo, Nickel, & Saldanha-da-Gama, 2009).

A list of input parameters and constraints that are used to solve the three supply chain problems is described in Figure 4 (Melo, Nickel, & Saldanha-da-Gama, 2009; Meixell & Gargeya, 2005; Goetschalckx, Vidal, & Dogan, 2002; Vidal & Goetschalckx, 1997).

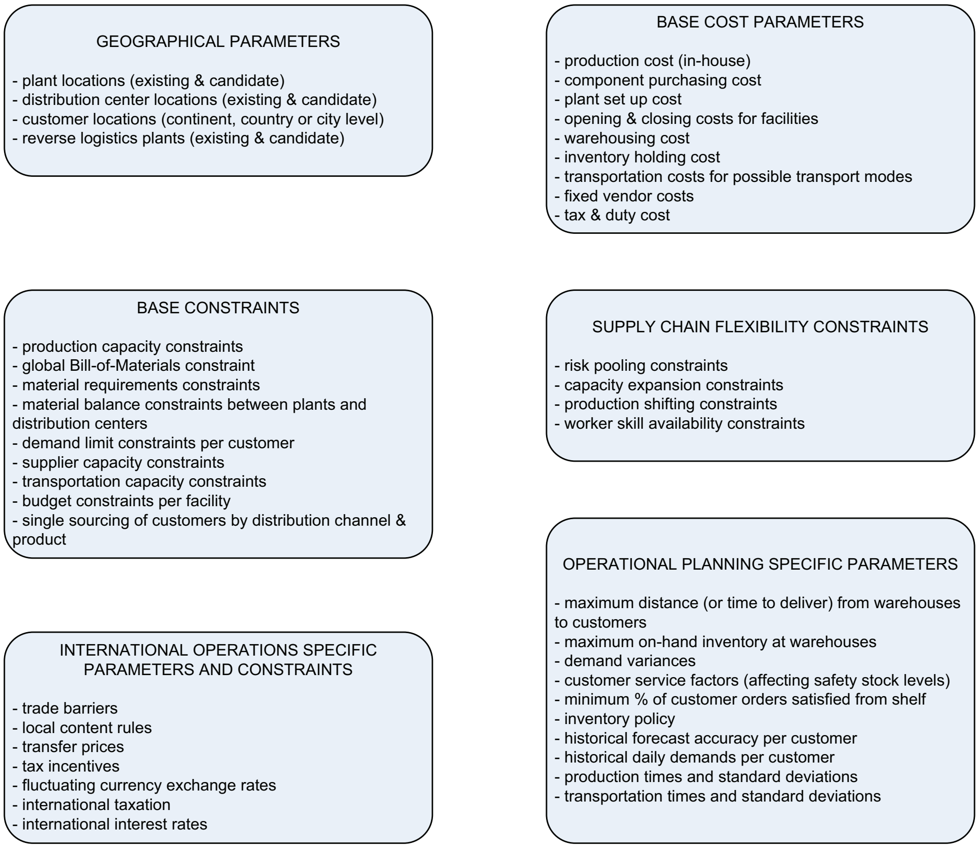


Figure 4 *Classification of parameters used in solving supply chain problems (Melo, Nickel, & Saldanha-da-Gama, 2009; Meixell & Gargeya, 2005; Goetschalckx, Vidal, & Dogan, 2002; Vidal & Goetschalckx, 1997)*

The facility location problem and network flow / demand allocation problem use selected parameters and constraints from all the boxes except the bottom right. For most cases the geographical parameters, base cost parameters and base constraints shown in the figure form an adequate input data set. In more precise network flow problems, additional elements from supply chain flexibility parameters and international operation-specific parameters are added. The operational planning-specific parameters depicted in the bottom right of the figure are used almost exclusively to solve the distribution planning problem.

For the present study's perspective of optimally supporting supply chain decisions during new product development, it is important to find a flexible methodology and tool for solving the network flow / demand allocation problem. The quest for such a method is presented in the next chapter.

3 AN EFFECTIVE METHOD AND TOOL FOR SUPPLY CHAIN ANALYSES DURING NEW PRODUCT DEVELOPMENT

The quest for discovering an effective method and tool for supply chain analyses during product development starts with a literature-based discussion of required attributes. Following this, an enquiry into extant literature for solution methods and tools is presented. The research does find a gap between the required attributes and the existing solution methods, and proceeds to develop a new one.

In the following discussion it is important to distinguish between the notions of *method* and *tool*. In this discourse method refers to formalism, such as mixed integer programming, which forms the basis of supply chain models. A *tool* is (typically) a software artifact that employs a certain method to solve supply chain problems. Similar division between method and tool has been employed in, e.g., Ingalls and Kasales (1999).

3.1 Desirable characteristics of modeling methods for facility location and network flow / demand allocation problems to be used during new product development

The cost effect of new product development is significant before the new products are launched in the marketplace. In fact, 70% of the product program's cost is committed during the design stage (Andersen, Khler, & Lund, 1986). A successful product will quickly achieve break-even status after its launch, but an unsuccessful one may never achieve this. An unsuccessful launch will cut the development budget of the next product generation, and a possible downward spiral of a company may follow. Therefore, making the right choices in product design, manufacturing process and supply chain is of paramount importance during new product development.

Traditionally, new product development was a long sequential process (Fine, Golany, & Naseraldin, 2005). The product design was selected primarily based on marketing targets and engineering constraints. The chosen design was subsequently passed on to production planning organization. Production planning organization developed the internal manufacturing processes for the product subject respecting the target cost minimization and plant utilization plans. Finally, the production plan decisions were passed on to the logistics organization as constraints, which largely dictated the supplier choices. This traditional process was not only long and rigid, but also produced sub-optimal results. Concurrent Engineering (CE) (Prasad, 1996; Lee & Sasser, 1995; Huang, 1996) was invented to overcome these problems. CE dictates that "product and process decisions are made in parallel as much as possible and that production considerations be incorporated into the early stages of product design" (Fine, Golany, & Naseraldin, 2005; p. 390). The possible benefits from CE are reduced product development time, reduced product cost, increased quality, reduced product lead times, reduced supply chain problems, and smoother product introductions (Blackhurst, Wu, & O'Grady, 2005). By the same token, CE complicates the design problem as "it requires joint optimization of a more complex objective with a larger set of constraints" (Fine, Golany, & Naseraldin, 2005).

The CE approach incorporates only the product design and process decisions. However, it was noticed that this was insufficient, because the concept of supply chain was missing. Thus the notion of 3D-CE (Fine, 2000; Fine et al., 2002; Fine, Golany, & Naseraldin, 2005) was created. 3D-CE aims at the parallel decision-making of product design, manufacturing process, and supply chain configuration. Moreover, it suggests that the design of a new product should be carried out in a set-based manner (Fine, Golany, & Naseraldin, 2005). In this approach, designers in the three decision domains will communicate among each other about a set of designs rather than a specific one. This set of possible design choices becomes narrower as time passes, until the final design choice is left.

CE and 3D-CE place a lot of requirements on modeling methodologies. Fine, Golany, and Naseraldin (2005) argue that the literature up to that point consisted only of qualitative frameworks for making decisions in the three domains, which prompted them to create a goal-programming-based approach. Similarly, Blackhurst, Wu and O'Grady (2005) called for the development of a modeling method which can concurrently support the representation of product, process, and supply chain tradeoffs. Even though the current work approaches a variant of the CE problem, i.e., finding an effective modeling method and tool for simultaneously representing various product designs with their possible supply chain configurations, these arguments remain valid.

In the past decade, the concept of supply chain risk management has become very relevant due to, e.g., the 9/11 terrorist attacks (Giunipero et al., 2008), the ash cloud of 2010, and Japan's major earthquake of 2011. Therefore, the planning for supply disruptions, and the use of multi-sourcing has become commonplace (Norrmann & Jansson, 2004; Blackhurst, Wu, & O'Grady, 2007). Moreover, the possibility of such risks in the supply chain means that the complete state space of the supply chain options should be analyzed for feasibility and cost. Concentrating on just one supply chain may prove to be a fatal mistake.

From the previous discussion, it is possible to derive the desired attributes for an effective supply modeling method to be used during new product development. First, the set-based approach of designing products (Fine, Golany, & Naseraldin, 2005) provides a hint that a modeling method for describing possible supply chains should be product structure-driven. This provides the needed flexibility when several product architectures are considered (Krishnan & Ulrich, 2001). Secondly, an effective supply chain modeling method should allow for the refinement and re-use of supply chain analyses made from coarse-grained product structures. This is an immediate consequence of the concurrent engineering and set-based approach of designing products (Huang, 1996; Fine, Golany, & Naseraldin, 2005). The first iterations of analyses are typically completed using coarse-grained product structures. Some elimination of design choices is possible already at this level. The remaining product designs are refined before subsequent analyses are performed. Thirdly, the modeling method should be robust to allow the inclusion of various cost and other parameters for decision-making. Firms are very different in nature, and the manner in which, e.g., their products' prices erode differs significantly (Helo, 2004). Fourthly, complex supply policies (e.g. multi-sourcing) should be easily representable in the models as they are one of the key approaches to mitigating supply chain risks (Simchi-Levi, Kaminsky, & Simchi-Levi, 2004; Blackhurst, Wu, & O'Grady, 2007). Finally, the modeling method should generate and analyze the complete state space of supply chain options, when this number is moderate, e.g., below 10.000 options. This requirement is also in line with mitigating supply chain risk through the analysis of various supply chain options (Blackhurst, Wu, & O'Grady, 2007).

Now we are ready to look at the existing methods and tools that have been used to solve the selected supply chain problems outlined in chapter 2.2 and examine how they fulfill the requirements imposed in the present chapter.

3.2 Current solution methods for selected supply chain problems

The literature introduces several possible solution methods to the supply chain problems introduced in chapter 2.2 (Eom et al., 1998). Optimization and simulation methods are the most common ones, but recently a number of hybrid and emerging methods from such diverse areas as control theory and Petri nets have surfaced. In the following discussion, the solution methods are divided into four classes: Optimization methods, simulation methods, heuristics methods, and hybrid / emerging methods.

3.2.1 Optimization methods

Optimization methods have been used in solving logistics problems since the 1950s after the discovery of the simplex method by Dantzig (Geoffrion & Powers, 1995; Dantzig, 1951). The simplex algorithm allows the efficient solving of a linear program where the objective function and the constraints are represented with linear expressions. The variables are assumed to be continuous. A few years later, after the discovery of simplex, total cost analysis emerged in logistics network analysis (Lewis, Culliton, & Steele, 1956). The same authors were also the pioneers in transportation cost and inventory cost tradeoff analyses (ibid.). The key algorithms in optimization were developed during the period before 1980 (Geoffrion & Powers, 1995). Among the major developments were the formulation of mixed integer programs which allow integer and continuous variables to co-exist in objective and constraint functions (Padberg, 1999). Benders decomposition algorithm opened an avenue for solving very large mixed integer programs, which are the most relevant in supply chain management (Benders, 1962; Geoffrion & Graves, 1974).

As was demonstrated in chapter 2.2.4, the set of possible parameters employable in solving supply chain management problems is large. Additionally, the objective functions in such problems differ significantly (Melo, Nickel, & Saldanha-da-Gama, 2009). A list of examples is given below (Meixell & Gargeya, 2005; p.537):

- Maximize after tax profit

- Minimize material/labor/transportation/utility costs
- Minimize production, shutdown, and startup costs
- Minimize cost and/or weighted activity time
- Minimize fixed and variable costs
- Minimize shortage/overage costs
- Maximize utility (revenues – costs) for manufacturers and retailers

The actual formulation of a mathematical program consists of the objective function, and a set of constraints. The following is a verbal description of a mathematical program which minimizes the total supply chain cost (Goetschalckx, Vidal, & Dogan, 2002):

Minimize: Total cost = Supply cost + Fixed manufacturing cost + Variable manufacturing cost + Fixed facility operating cost + Variable facility operating cost + Warehousing cost + Cycle inventory cost at the facilities + Pipeline inventory cost + Inventory carry-over cost + Transportation cost

Subject to: Customer demand satisfaction;
Conservation of flow at facilities;
Conservation of flow at suppliers;
Supplier capacity;
Facility capacity;
Machine capacity;
Linkage constraints between machines and facilities

A complete example of one of the first mathematical formulations of a supply chain problem, attributed to Geoffrion and Graves (1974, pp. 822-23), is shown below.

$$\textbf{Minimize}_{x\geq 0;\ y,z=0,1} \sum_{ijkl} c_{ijkl}\, x_{ijkl} + \sum_k \left[f_k z_k + v_k \sum_{il} D_{il}\, y_{kl}\right],$$

Subject to

(1) $\sum_{kl} x_{ijkl} \leq S_{ij}$ all ij
(2) $\sum_{j} x_{ijkl} = D_{il}\, y_{kl}$ all ikl
(3) $\sum_{k} y_{kl} = 1$ all l
(4) $\underline{V}_k z_k \leq \sum_{il} D_{il}\, y_{kl} \leq \overline{V}_k z_k$ all k

i,j,k,l	Indices of commodities, plants, distribution centers, and customer zones
S_{ij}	Supply (production capacity) for commodity *i* at plant *j*
D_{il}	Demand for commodity *i* in customer zone *l*
$\underline{V}_k, \overline{V}_k$	Minimum, maximum allowed total annual throughput for a DC at site *k*
f_k	Fixed portion of possession and operating costs for a DC at site *k*
v_k	Variable unit cost of throughput for a DC at site *k*
c_{ijkl}	Average unit cost of producing and shipping commodity *i* from plant *j* through DC *k* to customer zone *l*

x_{ijkl} A variable denoting the amount of commodity *i* shipped from plant *j* through DC *k* to customer zone *l*

y_{kl} A 0-1 variable that will be 1 if DC *k* serves customer zone *l*, and 0 otherwise

z_k A 0-1 variable that will be 1 if DC is acquired at site *k*, and 0 otherwise

Equation (1) stipulates that the demand (served through all DC sites to all customer zones) for a commodity *i* from plant *j* cannot exceed the supply. Equation (2) says that all customer zone demand will be satisfied by some plants. Equation (3) says that every customer zone is served from one DC only. Finally, equation (4) specifies the upper and lower bounds on throughput volume of DC sites.

The above formulation was the first of its kind to solve a multi-commodity distribution problem for a single echelon case. It was solved using Benders decomposition algorithm. The idea is to first find a lower bound of the solution by choosing a configuration (i.e. which DC zones are open, and which customer zones are served from which DC's) that satisfies constraints (1)-(4), and minimize the right side of the objective function sum (i.e. plant specific costs) and the transportation cost constraints from Benders' algorithm's earlier iterations. Following this, the transportation problem for the configuration (the left side of the objective function sum) is minimized. When the optimums of the two subproblems are added together, an upper bound of the solution to the original problem is obtained. Subsequently, the solution of the transportation problem is added as a new constraint to the master problem by employing dual variables, and the whole process is re-run to find a solution with a lower total cost. The work of Graves and Geoffrion spurred a considerable amount of research effort into supply chain problems. Yet, as recent authors acknowledge, using the Benders decomposition algorithm requires significant technical expertise on the part of the user. Decomposing arbitrary mixed integer programs into a "master problem" and "transportation subproblem" is not trivial and deciding "Benders cuts" (i.e. plant and DC configurations above) poorly may result in very slow convergence toward the optimum (Goetschalckx, Vidal, & Dogan, 2002).

The formulation as above is deterministic where each parameter has an exact value. Stochastic optimization methods have since been developed (e.g. Santoso et al., 2005 provide an overview). An example of a stochastic optimization problem is: "Design a supply chain so that the investment costs and expected operating costs is minimized". This problem is mathematically stated as (c, y and ξ are vectors):

$$\mathrm{Min}_y \quad f(y) := c^T y + \mathbf{E}[Q(y,\xi)]$$

Here, y is a vector where each component takes on a value of 0 or 1, c is the vector of investments costs, and $Q(y,\xi)$ is the optimal value of the following problem (the anticipated future operating costs)

$$\mathrm{Min}_{x,z} \quad q^T x + h^T z$$

Subject to:

(5) $Nx = 0$

(6) $Dx + z \geq d$

(7) $Sx \leq s$

(8) $Rx \leq My$, and x is a vector containing positive real numbers

In this equation x denotes the flow through each facility, and z represents the unserviced demand. ξ is a random vector $\xi(\mathbf{q}, \mathbf{d}, \mathbf{s}, \mathbf{M})$ that contains the probabilistic distributions of processing and transportation costs (**q**), demands (**d**), supply levels (**s**), and capacities at plants (**M**). A realization of ξ is a vector where values for each random component are sampled from their respective distributions. The cost component $h^T z$ corresponds to the cost of unmet demand. Matrices N, D, and S achieve summations on the left side, and R represents the resource requirements at plants.

This stochastic optimization problem is solved by employing the law of large numbers, which permits the substitution of the following expression in the place of the master problem:

$$\mathrm{Min}_y \quad \{ f'(y) := c^T y + N^{-1} \sum_{n=1..N} Q(y, \xi^{\mathbf{n}}) \}$$

In this formulation, N subproblems are generated for optimizing the expected value of future operating costs. In the generation of the subproblems, variable cost components are sampled from their respective probability distributions. An expectation of the stochastic expression is given by taking the average of N such subproblems. Santoso et al. solved this formulation using a variant of the Benders decomposition algorithm (Santoso et al., 2005). The article by Santoso et al. provided evidence that the stochastic formulation of the problem gave better solutions than a corresponding deterministic problem (where q, d, s, and M were assigned the mean values from their corresponding probability distributions). However, it is more difficult to solve stochastic optimization models than deterministic optimization models which limit their applicability to difficult problems.

In summary, optimization has made it possible to solve very large supply chain problems efficiently. Optimization guarantees the best solution for a given parameter set. Starting from the seminal work of Geoffrion and Graves (1974), the field has experienced rapid growth and it is now possible to model multi-commodity, multi-period distribution problems with international aspects such as transfer pricing, tax incentives, and fluctuating currency exchange rates. The discovery of solution methods for stochastic optimization problems have also made it possible to account for randomness in input parameters. Many supply chain optimization tools such as JDA Supply Chain Strategist, and CAST V10, have been brought to the marketplace and the use of optimization methods in business is currently increasing. However, the technical expertise required to solve very large mixed integer programs remains an obstacle to the more widespread use of optimization techniques in industry (Goetschalckx & Fleischmann, 2008). Moreover, optimization methods are unable to generate complete state spaces of supply chain solutions, but rather converge to the optimal result. Thus the fifth requirement of section 3.1 is not satisfied.

3.2.2 Simulation methods

Discrete simulation methods have been developed to enable the modeling of the dynamic behavior of a supply chain. Forrester discovered the bull-whip effect in 1961 (Forrester, 1961) and raised the importance of studying customer service levels and optimal inventory targets in the supply chain. The ability to handle uncertainty and maintain good responsiveness in the supply chain is a key characteristic of world-class supply chains (Christopher & Towill, 2005). Discrete simulation is currently viewed as the most realistic method of analyzing logistics networks (Guedes, 1994; Bowersox & Closs, 1989; Powers, 1989). It allows the specification of uncertainties in, e.g., customer demand, transportation lead time, production lead time, and production yield rate (Peidro et al., 2009; Kleijnen, 2005; Bennett, Tipi, & Riddalls, 2000; Ingalls, 2008). Such a model may be simulated to determine the resulting customer service level contingent on the network uncertainties. Discrete simulation does not guarantee optimality. The modeler chooses a certain network structure, and explores in a "what if" manner the effect of input parameters (e.g. inventory policies) on customers' perceived service levels (Bowersox & Closs, 1989; Kleijnen, 2005).

Discrete simulation is performed on user-defined network structures. A user decides the skeleton (e.g. geographical position of plants, and the possible transportation links) of the supply chain, and uses the random number generators to model, e.g., the incoming orders, manufacturing equipment failures, and transportation delays (Ingalls, 2008). During the simulation run the simulation software can collect statistics about the utilization of manufacturing facilities and machines, the waiting times of customers, and incurred penalty costs.

Discrete simulation may be employed to solve the distribution strategy problem introduced in chapter 2.2. The network to be simulated may resemble the one shown in Figure 5. This network is made up of nodes (representing manufacturing facilities, buffers and the customers) and arcs (which denote transportation links).

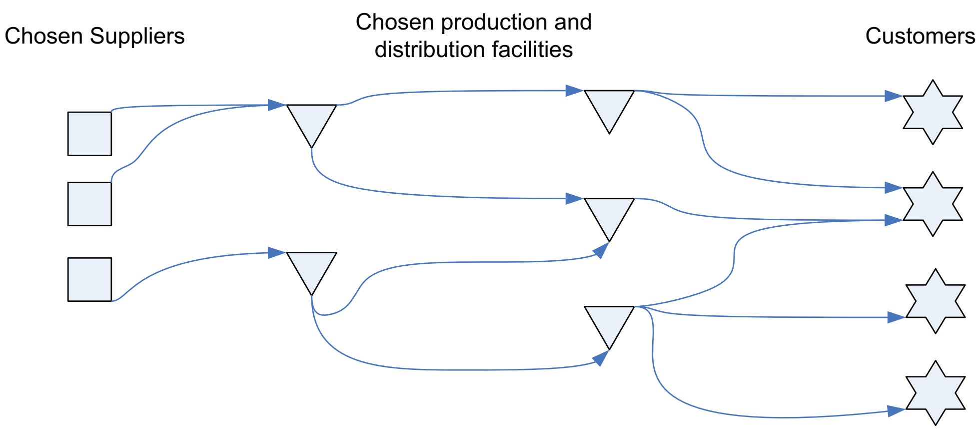


Figure 5 *A distribution network, which may be analyzed via simulation for its dynamic properties*

A simulation model consists of certain fixed elements (Ingalls, 2008). *Entities* are objects that cause changes in the state of the simulation. An example entity is a new customer order. Entities have *attributes*, which are unique to the entity in question: if an entity is an order, its inherent attributes are the identity of the customer, the time at which the order was placed, and the type and the amount of articles ordered. The nodes also have attributes – these are, for instance, the anticipated manufacturing time in a factory, and the associated assembly cost. Similarly, the arcs contain attributes – arcs represent transportation links and are characterized by a cost, a lead time, and a lead time standard deviation. *Activities* represent the process and the logic of the supply chain. An example activity in a supply chain is the production of a batch of items. Two special activities are *delays* and *queues*. For instance, the execution of a production order by a manufacturing cell is modeled as a delay activity reflecting the needed assembly time. Similarly, a trolley containing the materials of the next production batch is placed in a queue waiting for the production of the current batch to be completed. Activities and entities co-operate to generate *events*. *Events* are conditions that occur at a certain time, and change the system state. For instance, the completion of a production order in a supply chain is an event. Finally, each of the nodes (e.g. manufacturing cell) contains a certain number of *resources*, such as qualified personnel, that perform activities.

Randomness in simulation models is achieved by the use of *pseudo-random number generators*. These generate, e.g., the timestamps according to which new orders appear, and influence the quantity to be ordered in each simulated order. Random number generators are used to simulate the breakdown of machines and other events causing disruption in the supply chain. *Global variables* control, e.g., the maximum queue lengths, and the number of resources at each node. The simulation packages track the values of global variables throughout the simulation.

Simulation is typically repeated tens or hundreds of times, each time with a different pseudo-random parameter set. The main statistics are collected for each run, and when the confidence intervals of the main statistics become narrow enough, the simulation stops.

Advances in computation capacity have made the execution of large simulation models possible in a "what-if" manner. Its main drawback is that optimality is not guaranteed. The user is responsible for selecting the network structure to be simulated. From a practical point of view, it is non-trivial to choose the simulation length correctly – a simulation model may require a certain "warm-up time" before it reaches its steady state (Kleijnen, Bettonvil, & Persson, 2004). Simulation also requires a large amount of development time as there is not a standard, succinct language for model formulation (Peidro et al., 2009; Bennett, Tipi, & Riddalls, 2000). Finally, customizing simulation software to fit a particular purpose in a single company may require a large programming effort (Vamanan et al., 2004).

Simulation methods are not relevant for the present research, as they are used for exploring the dynamic properties of a single supply chain. Regarding the requirements specified in chapter 3.1, simulation methods offer poor support for product structure-driven modeling of supply chains, and the refinement and re-use of earlier supply chain analyses. Moreover, they are unable to generate full state space of supply chain options.

3.2.3 Heuristic methods

Logistics problems have also been solved using heuristic methods (Ballou, 1989). Heuristics are rules-of-thumb that give a good solution to a problem quickly, but do not guarantee optimality (ibid.). Many heuristics have been computerized, and with increasing computation power they are able to find representative (not necessarily optimal) solutions to problems which are intractable from an optimization point of view (Geoffrion & Powers, 1995). If heuristics are applied in solving, e.g., the facility location problem, the solution possesses the following three properties (Guedes, 1994, p. 37).

(1) For **n** depots the objective function is convergent and has a unique sub-optimal solution that depends on the initial position (i.e. a required user input to the heuristic is an initial guess of where the depots should be located)
(2) The initial location affects/biases the final solution
(3) A good first guess does not guarantee a good final solution

The user is responsible for guessing the initial placement of depots, and thus he/she affects the quality of the solution greatly. Accordingly the use of heuristics should not be recommended for novices, whereas the technique may prove to be very helpful for experienced users.

In their simplest form, heuristics are sentences which establish relationships between variables. The following list gives relationships that guide the optimal placement of warehouses (Ballou, 1989; p.126).

- The most likely sites for warehouses are those that are in or around the centers of greatest demand
- Customers that should be supplied directly from source points and not through a warehousing system are those that can purchase in full vehicle-load quantities
- A product should be warehoused if the differential in transportation costs between inbound and outbound movement justifies the cost of warehousing
- Items in a product line that are best managed by just-in-time, rather than statistical inventory control procedures are those that show the least variability in their demand and lead time patterns
- The next warehouse to add to a distribution system is the one that shows the greatest cost savings
- The most expensive customers from a distribution standpoint are those that purchase in small quantities and are located at the end of the transportation lanes
- Economical transportation loads are built by consolidating small volume loads into full-vehicle loads beginning from the most remote customers on the distribution network and combining loads along a line to the transportation origin point

One of the earliest solutions to a facility location problem for single and multi-commodity cases where depot location choices were given as discrete options was a heuristic (Eilon & Watson-Gandy, 1970).

(1) Select initial set of potential facility locations (exceeding the number of expected depots)
(2) Allocate each customer to the cheapest facility (e.g. by considering inbound, warehousing, and delivery cost)

(3) Calculate new facility locations using, e.g., the center of gravity algorithms
(4) Allocate each customer to the cheapest facility using previous throughput level
(5) Go to step (3) and repeat until no further reduction can be made
(6) Apply the drop routine:
    a. Remove the smallest facility in the system
    b. Allocate customer served by this facility to the cheapest facility
    c. Cost this option
    d. If the cost is lower than that obtained in step (5), go to step (3) and repeat (4) and (5) until no further improvement in the cost function is obtained
    e. Repeat the drop routine until no improved solution is found

With increasing computation power, the use of heuristics to solve supply chain problems has decreased, and in contrast the application of more complex Operations Research algorithms has increased (Geoffrion & Powers, 1995). Some tools still offer the possibility to use both heuristics and mixed integer programming (e.g. CAST V10). The rationale has been the speed of obtaining solutions using heuristic methods. In practice, heuristics should be used today only for obtaining candidate solutions to otherwise intractable problems. Heuristics are still relevant in the simulation optimization community where they are used to expedite the convergence of solution algorithms (Fu, 2008).

Regarding the modeling requirements of chapter 3.1, heuristic methods are unable to generate the complete state space of supply chain options, and because they offer poor support for product structure-driven modeling of supply chains, and the refinement and re-use of earlier supply chain analyses. Moreover, heuristics do not guarantee optimality.

3.2.4 Hybrid and emerging methods

Recently, there has been increasing interest in hybrid and emerging methods for solving supply chain problems (Lee & Kim, 2002). The combination of optimization and simulation was suggested already by House and Karrenbauer (1978). Since then, reported hybrid methods for supply chain management include, e.g., genetic algorithms, fuzzy logic, and analytical hierarchy processes – each combined with optimization methods (Truong & Azadivar, 2003). Emerging methods in supply chain management refer to techniques that have been known in other fields of science, but have only recently been applied in SCM. Among these are control theoretical methods, queueing networks, and Petri nets. The present section describes several hybrid and emerging methods for supply chain management.

Simulation optimization attempts to merge the capabilities of optimization and simulation to solve problems (Fu, 2008; Almeder & Preusser, 2007; Fu, 2002; April et al., 2003; April et al., 2005; Truong & Azadivar, 2003; Azadivar, 1999; Andradottir, 1998). Pure optimization is able to analytically solve plant location and network flow problems, but can do this only for static cost parameters – network dynamics is excluded. Discrete simulation, on the other hand, is able to explore the dynamic properties of a chosen demand supply network setup, but does not aid the user in picking the optimal network structure in the first place. Simulation optimization techniques attempt to solve, for instance, the optimal number of transportations per week from warehouses such that the network will still perform well in the presence of uncertain demand.

A simplified picture of the operation of simulation optimization is given in Figure 6. A metaheuristic optimizer selects a parameter set for a supply chain that satisfies given constraints. In a supply chain there may be constraints on the allowed number of warehouses, and the maximum number of workers per warehouse. Once the parameters are selected, discrete simulation is run on the network to explore its dynamic properties. The iteration between optimization and simulation is repeated until the value of the objective function converges. The objective function may concern supply chain cost minimization, whereas a constraint may stipulate that customer service must remain above a predefined level, e.g., 95%.

Simulation optimization has two subproblems: the estimation of the optimal parameter set and the estimation of the output performance function (Fu, 2002). A sample problem for simulation optimization is given in (April et al., 2005). They consider the operation of a hospital emergency room which follows a certain process. The objective is to minimize the expected total asset cost subject to several constraints:

(1) Average cycle time for critical patients is at most 2.4 hours
(2) No. of nurses is between 1 and 7
(3) No. of physicians is between 1 and 3
(4) No. of patient care technicians is between 1 and 4
(5) No. of administrative clerks is between 1 and 4
(6) No. of ER rooms is between 1 and 20

The first subproblem deals with the determination of the optimal number of nurses, physicians, patient care technicians, administrative clerks, and the rooms. The second subproblem involves the estimation of the asset cost. Discrete simulation is employed in two contexts: First, to ensure that the first constraint, i.e., critical patient cycle time, is satisfied and second, to estimate the asset cost. The problem has 7 x 3 x 4 x 4 x 20 = 6720 theoretical resourcing possibilities. However, the simulation optimization method is able to find the optimal answer in just 100 iterations of Figure 6.

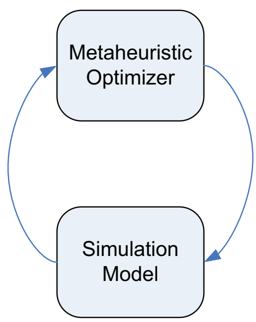


Figure 6 *Simplified idea of simulation optimization (April et al., 2005)*

At least one reported work on simulation optimization also considers the strategic decisions in a supply chain – i.e., facility location, stocking locations, supplier selection, production capacity, distribution strategy and transportation mode (Truong & Azadivar, 2003). They use the combination of genetic algorithms and mixed integer programming in a metaheuristic optimizer to decide the suppliers, locations, inventory holding positions, and transportation modes. Simulation is performed to check the resulting customer service level of the solution. From the present work's perspective, the modeling method of Truong & Azadivar (2003) is location- and production phase-driven rather than product structure-driven, and no proof was found related to their tool's deployment in a commercial setting.

The recent research in simulation optimization has dealt with improving the metaheuristic optimization algorithms (Fu, 2008). Among the candidates are stochastic approximation via gradient methods, response surface methods, random search, and sample path optimization (Fu, 2008; April et al., 2005). From the present work's perspective, most of the applications of simulation optimization are in the operative questions of supply chain management – i.e., optimal staffing (people and machines) to minimize asset costs and still achieve target customer service levels – which render the technique irrelevant in the current discussion.

Optimization has also been combined with qualitative methods, such as Analytical Hierarchy Processes (AHP) (Wang, Huang, & Dismukes, 2004; Dotoli et al., 2005). The objective of this hybrid technique is to allow the consideration of qualitative and quantitative variables simultaneously. One of the typical application areas is the supplier selection question. Suppliers have quantitative parameters such as available capacity, price per component, and transportation cost from facilities. Additionally they possess qualitative parameters such as delivery reliability and production flexibility. AHP compares all suppliers pairwise for the qualitative (this task requires expert judgment) and quantitative aspects. The supplier with the best weighted score in AHP is selected as the best candidate. Alternatively, a certain AHP score may be set as a threshold for acquiring a short list of possible suppliers. Mathematical programming techniques may be used in conjunction with AHP to acquire the best supplier group, if no single supplier has enough capacity to fulfill all the demand. AHP methodology is not relevant for the scope of the current work, as it is assumed that the list of supplier candidates is available, and no qualitative variables are considered.

Queueing networks (Suri et al., 1995; Leung & Suri, 1990; de Treville & van Ackere, 2006) have been used to simulate, e.g., flexible manufacturing systems in factories. Queueing network analysis, which is an emergent method in supply chain management, provides "rapid prototypes" of a system's dynamic behavior. Queueing networks provide quick estimates of production rate, average lead times, machine utilizations, and queue lengths based on the well-known Little's law. The accuracy of these estimations compared to detailed simulation models has been within 10% (de Treville & van Ackere, 2006). Queueing networks may be used to give quick approximations of lead times in supply chains. This is achieved by considering the interplay of three factors: bottleneck utilization (e.g. warehousing or transportation), lot sizes, and demand variability. As such, the authors suggest that queueing theory could be used first to approximate a plausible parameter set in a manufacturing facility, and detailed simulations may be built subsequently to fine-tune the values (ibid.). It has been reported that the construction of a queueing network model for a certain supply chain required 4 hours, whereas its detailed simulation model took 21 days to build (Suri et al., 1995; p. 134-35). Regarding the modeling requirements in chapter 3.1, queueing networks support product structure-driven modeling of supply chains poorly, and are unable to generate the complete state space of supply chain options.

A rather novel emerging technique for analyzing supply chains comes from control theory (e.g. Lin et al., 2004). A supply chain is modeled as a control-theoretical PI system to discover its dynamic properties (ibid.). Stability analyses of the supply chain (e.g. the magnitude of bullwhip effects with certain ratios of demand variability and inventory level) are carried out after the supply chain's transfer function is determined in the z-plane. Control theoretical methods are similar to queueing network methods, i.e., providing quick analytical ways of assessing the dynamic properties of a single supply chain. Regarding the modeling requirements of chapter 3.1, they support product structure-driven modeling of supply chains and the inclusion of various cost parameters poorly, and are unable to generate the complete state space of supply chain options.

3.2.4.1 Petri nets

Petri nets were originally developed in the 1960s to analyze problems in particle physics (Jensen, 1996). Since then, they have been applied in many technical areas, most notably in the verification of hardware and software protocols (ibid.). Petri nets have been considered for application in logistics and workflow modeling since the 1990s (van der Aalst, 1992; van der Aalst, 1998; van der Aalst 2000a; van der Aalst 2000b; Desel & Erwin, 2000; Salimifard & Wright, 2001; Artigues & Roubellat, 2001; van der Aalst & ter Hofstede, 2005; Russell & ter Hofstede, 2009). An example of a specialized Petri net is given in Figure 7. Petri nets are structured as a network of places (often referred to as 'conditions' and drawn as circles) and transitions (often referred to as 'activities' and drawn as squares). If a token (solid black circle) exists in every place leading to a transition, it may fire and insert a token into each of the places coming out of the transition. Petri nets are applicable to the modeling of manufacturing systems, and transportation networks. Time can also be handled in Petri net models in deterministic (Barthomieu & Diaz, 1991) and stochastic manners (Ajmone Marsan, Donatelli, & Neri, 1990; Alves, Maciel, & Lima, 2008).

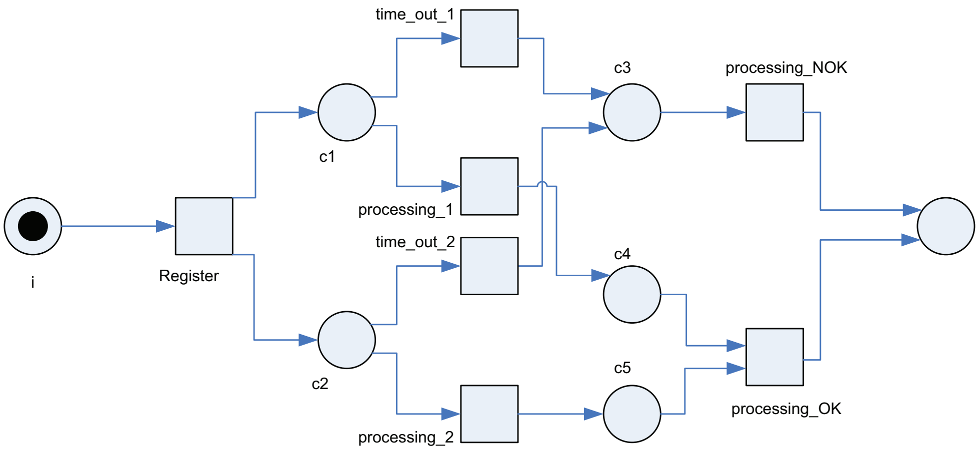


Figure 7 *An example of a Workflow Net (van der Aalst, 2000b)*

The network in Figure 7 is a workflow net (van der Aalst, 2000b). A workflow net contains two special places, *i* and *o*. The former denotes the beginning of a process and the latter the end of it. The example depicts a process for sending two questionnaires, and waiting for corresponding answers. The process' logic is such that if both questionnaires are received in time, processing goes as planned and 'processing_OK' fires, otherwise 'processing_NOK' will fire. Van der Aalst (2000b) defines the concepts of soundness for workflow nets, which can be stated as follows:

(1) For any valid state of the network, it is possible to reach the output place o
(2) When there is a token in place o, all the other places must be empty
(3) It is possible to fire every transition of the network in some execution path (i.e. no dead transitions exist)

A sound workflow net guarantees that it is always possible to reach the end of the process (1), every activity may be carried out as part of some execution (3), and once the process stops, all resources must be released (2). A problem with the net in the previous picture is that when 'transition time_out_1' and 'processing_2' fire, a single token reaches the output place o, but one token (i.e. a resource) will remain unreleased in place c5. Therefore requirement (2) above is violated. Van der Aalst (2000b) describes algorithms which can detect violations of the soundness property from a network's structure. Workflow nets may be simulated for performance with tools such as ExSpect (van der Aalst et al., 2000; ExSpect, 1999). Simulation of workflow nets is comparable to the discrete simulation discussed earlier. A benefit of using a Petri net formalism for specifying the business process is that the validity of a process structure can be established through soundness check before a simulation run takes place.

Desel and Erwin (2000) introduce process nets, as depicted in Figure 8. They employ the hierarchical modeling capability of Petri nets to decompose certain activities into more precise descriptions (e.g. Produce_chassis). The contribution of Desel and Erwin is the integration of mathematical programming and Petri net process description. Time and cost parameters may be attached to each place and transition as shown. In Figure 8 it is possible to produce body type A or B. The net annotations show that the processing time and cost will be different in each case. A relevant management question concerns the optimal placement of work-in-process buffer such that the least inventory cost will be incurred. This particular network has two possible 'runs' – one which produces a car with body type A and the other which produces a car with body type B. If body type B is selected, the optimal solution of work-in-process buffer is obtained via the mathematical program below.

Minimize $10*x1 + 10*x2 + 60 + 10*x3 + 10*x4 + 80 + 20*x5 + 30 + 10*x6$

Where $x2 + 120 + x3 = x4 + 100 + x5$

The constraint is derived from the time delay between 'Start' and 'Assemble' transitions. The time spent in the upper process (delay and process time) must equal the time spent in the lower process (delay and process time). The solution to this linear program suggests that the optimal place for a buffer is 'Parts_chassis_available', and chassis production should start 20 time units after body production has been started. When this occurs the body and chassis arrive simultaneously to the 'Assemble' activity, and expensive buffering costs are avoided there. It is thus possible to determine the possible executions of a process using Petri nets and subsequently optimize inventory placement using mathematical programming.

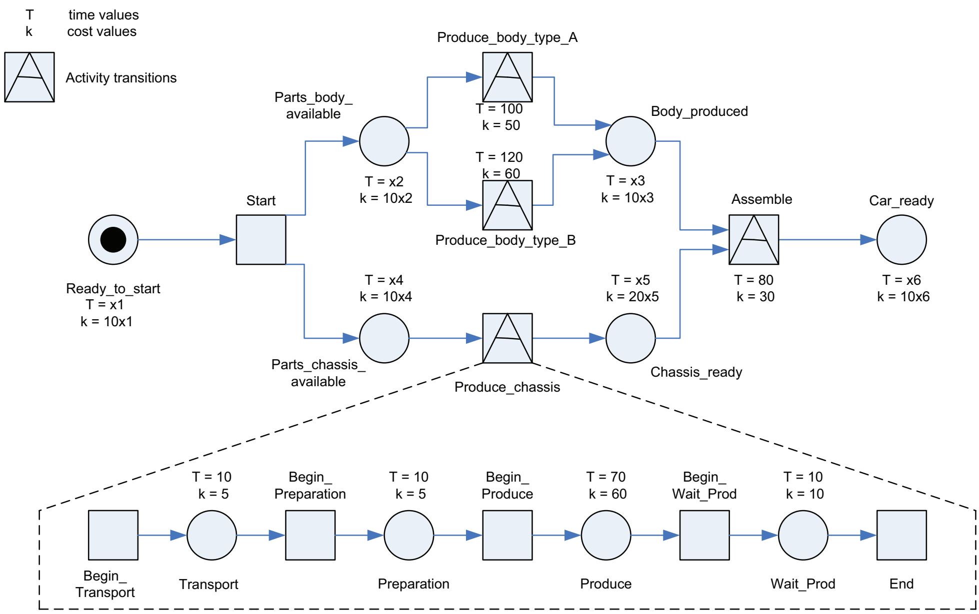


Figure 8 *A hierarchical business process description (Desel & Erwin, 2000)*

Petri nets have been applied as modeling aids in the Concurrent Engineering approach (Wu & O'Grady, 1999; Blackhurst, Wu, & O'Grady, 2005; Blackhurst, Wu, & O'Grady, 2007; Zhang, Lu, & Wu, 2009). The developed methodologies are able to analytically estimate the effect of certain order sizes, inventory re-order points and material and information lead times on the effectiveness of supply chains. Blackhurst et al. (2005) indicate that their methodology should be combined with a way of optimizing supply chain configurations on a high level. Zhang et al. (2006) present a Petri net method for optimizing supply chain configurations, where decisions are made recursively at each supplier tier based on local information only. This approach may lead to suboptimal configurations. Supply chain disruptions have also been studied using Petri nets and reachability analysis (Wu, Blackhurst, & O'Grady, 2007).

The existing Petri net techniques have primarily investigated the dynamic properties of single supply chains subject to changes in key parameters such as order sizes and lead times. Wu et al. (2007) apply reachability analysis to determine the operations which are impacted if a disruption happens somewhere in the existing supply chain. However, reachability analysis may be applied in a different manner to solve the research problem of the current work.

3.3 Current tool support for network flow / demand allocation problems

This section presents the current state of supply chain optimization tools for solving the network flow / demand allocation problem. Supply chain optimizers are normally divided into two parts: the user interface and the solver engine, as in Figure 9. This section briefly discusses the available optimization engines, and then proceeds to describe three generic supply chain optimization packages: JDA Supply Chain Strategist, CAST V10 from Barloworld, and Supply Chain Guru from Llamasoft. Last, it gives an account of the reported methods and tools that have been applied specifically in supporting supply chain analyses during new product development.

As the focus of the present dissertation is on solving the facility location and network flow / demand allocation problems, tool landscape for simulation, heuristic, emerging and hybrid methods are not described. These methods, with the exception of heuristics, were primarily used for exploring dynamic properties of single supply chains. Heuristic methods, on the other hand, provide sub-optimal results and are highly dependent on user-provided starting values.

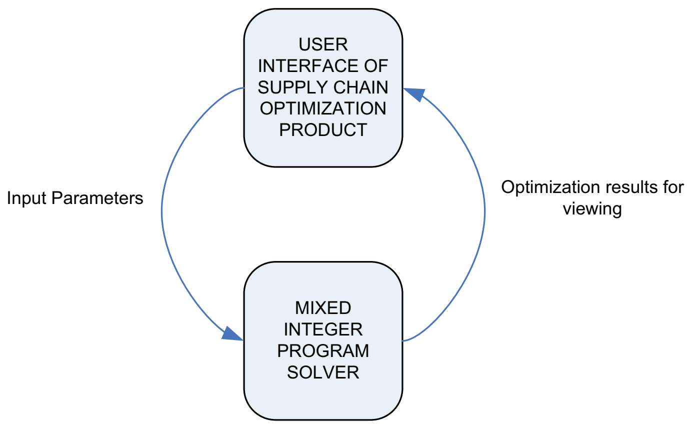


Figure 9 *Typical structure of a supply chain optimization product*

3.3.1 Optimization engines

Powerful optimization engines are needed to solve the supply chain problems introduced in chapter 2.2. Currently there are several linear and mixed integer programming solvers available in the public and private domains (e.g. LP_Solve, CVXOPT, glpk, CBC, CPLEX, Lingo, MOSEK, Xpress-Mosel MATLAB, Mathematica). Of these products at least CPLEX, Lingo and Xpress-Mosel have been used as external solvers in supply chain optimization packages. The three products can solve linear, mixed integer, network flow, quadratic, and quadratically constrained programming problems. Lingo additionally solves stochastic and non-linear programs.

CPLEX (IBM, 2011) is a solver originally developed by ILOG and later acquired by IBM. It has a limited command line interface suitable for entering small models in interactive mode. CPLEX is typically used by entering models in text files of a specified format. Solver functionalities may also be called directly from C++, Java, or .NET applications. This capability is employed in JDA Supply Chain Strategist, which uses CPLEX as its internal optimization engine. An example of the interactive language of CPLEX is shown below:

```
maximize  x1 + 2 x2 + 3 x3
subject to -x1 +  x2 + x3 <= 20
            x1 - 3 x2 + x3 <= 30
bounds
  0 <= x1 <= 40
  0 <= x2
  0 <= x3
end
```

Lingo (Lindo Systems, 2011) has a window-based user interface for specifying and solving mathematical programs. It contains a more verbose declarative input language which allows, e.g., insertions of comments for documentation purposes. Lingo solver functions may be called from Excel, C++, Java, and .NET applications. TG Optima, investment optimization software, uses Lingo as its core engine, mainly because of its stochastic optimization capabilities. Lindo System’s product pages reported several cases where the Lingo mathematical programming environment had been used in a supply chain context in such places as Bridgestone (car tire distribution) and Proctor & Gamble. In these applications the mathematical programming had been done directly in Lingo’s own language without a supply chain-centric user interface. An example of a mathematical program specified in Lingo’s declarative language follows.

```
!Maximize revenues – costs;
MAX = (2.70 - .02)*RS + (2.82 - .02)*MS1 + (2.80-.02)*MS2
      -   (2.70+.01)*RB – (2.90+.01)*PB;
!Sources = uses for Regular and Premium;
[SEQUR]     5000 + RB = RS + .5*(MS1 + MS2)
[SEQUP]     PB = .5 * (MS1 + MS2)
!Upper limits on amount we can sell;
[UL1]           MS1 <= 6000;
[UL2]           MS2 <= 8000;
```

Mosel (FICO, 2011) is a declarative mathematical programming language similar to a high-level programming language. Dash optimization (acquired since by FICO) has developed several solvers that optimize problems given in the Mosel language. Solver modules exist for standard linear and mixed integer programming, quadratic programming, mixed integer quadratic programming, and non-linear programming. CAST V10 supply chain optimizer uses Mosel as its internal solver. An example of a mathematical program written in the Mosel language is shown below. The resemblance between Mosel and a high-level programming language is evident. *mmxprs* below refers to one of the available solver modules.

```
Model simple

uses "mmxprs"
declarations
   a: mpvar
   b: mpvar
end-declarations

profit := a + 2*b
first:=      3*a + 2*b <= 400
second:=      a + 3*b <= 200

maximize(profit)
writeln("Profit is ", getobjval)
end-model
```

Adoption of supply chain optimization software is influenced more by user friendliness than the technical superiority of a software package (Geoffrion & Powers, 1995). Accordingly, the development of supply chain optimization packages has split into two fronts. The companies specializing in optimization engines develop faster algorithms, whereas the companies developing complete supply chain optimization packages focus on creating intuitive user interfaces.

The following subsections discuss in more detail the specifics of three popular generic supply chain optimization software packages – Supply Chain Strategist, CAST V10, and Supply Chain Guru. Special attention is paid to the data architecture and modeling capabilities of each product. Following this, a literature review is given regarding methods and tools that have been specifically applied for supply chain decision making during new product development.

3.3.2 Supply Chain Strategist

JDA Supply Chain Strategist (SCS) (i2 Technologies, 2009) is among the popular supply chain optimization packages. According to the taxonomy presented in chapter 3.2, it is a *tool* that makes use of *optimization* methods. It solves two types of objective functions in a multi-period, multi-commodity, capacitated case

- Cost minimization while serving committed demand
- Profit maximization while serving forecasted demand (the optimizer may choose not to satisfy some demand points where cost-to-serve is high)

SCS is based on eight modeling entities: Facilities, processes, products, periods, transportation modes, shipment sizes, service levels and demand regions. From the eight modeling entities, several relationship tables are formed to specify, for instance, which products are manufactured in a certain facility during a specified period. The relationship tables hold the pertinent parameters for optimization. The list of the possible relationship tables is presented next:

- Process at Facility (which processes may occur at which facilities)
- Process Component (fixed bill of materials for the processes)
- Facility in Period (which facilities are open during each period)
- Product in Period (which products are produced during each period)
- Product at Facility (which facilities are allowed to handle which products)
- Product at Facility in Period (which products are allowed to be handled in which facilities during each period)
- Process at Facility in Period (which processes are allowed to occur in which facilities during each period)
- Demand Requirement (what is the demand for each product during every period, and what the required service levels and shipment sizes are)
- Transportation Mode in Period (which transportation modes are available during each period)
- Transportation Mode Component (how much each kilometer of transportation adds to, e.g. carbon emissions)
- Interfacility Link in Period (which transportation links are available between the facilities during each period)
- Service Link in Period (which transportation links are available between the facilities and the customers during each period)

The data input to SCS happens through tables accessible in the user interface. Table parameters may be directly manipulated in the UI, or data may be collected to separate Excel sheets which are imported to SCS. Due to the large number of input data tables, SCS is not very approachable to a new user, but it offers a great deal of flexibility to the experienced modeler. The optimization results may be visualized in a world map, but setting up the map views is a rather complex process for novices.

From the perspective of the present work, SCS is a tool that is too versatile. This versatility translates into a plethora of data tables which have to be updated if changes in product architecture or supply chain structure take place. In particular, if the bill-of-materials (product modularization and possibly the suppliers used) changes, one has to make cumbersome amendments to the following tables: product, facility, product in period, product at facility, product at facility in period, process component, process at facility in period, interfacility link in period. On the other hand, SCS is well-suited to solving complex network flow problems, where multiple products are sourced from tens of facilities to serve hundreds of customers.

3.3.3 CAST V10

CAST V10 (Barloworld, 2008) began as a transport network optimization tool in the late 1980s and has since evolved into an end-to-end supply chain optimization tool. The current version (V10) solves multi-commodity, capacitated, single-time period network flow problems. CAST employs two solution methods: a network strategy heuristic, and a mixed integer program optimizer (Xpress-Mosel). Therefore CAST is a *tool* which employs *optimization* and *heuristic* methods according to the classification in chapter 3.2.

The main entities of the CAST model are Customer, Depot, Supplier, Vehicle, and Product groups. 'Customer' specifies the geographical locations of demand, quantity to be delivered and the anticipated number of deliveries. 'Supplier' itemizes the geographical locations of suppliers, quantity of components to be delivered, and the number of collections. 'Vehicle' includes the possible transport lanes and associated capacities and tariff structures. 'Product Group' contains the components and the assembled products that flow through the network. Input is entered through wizard-like windows which make the tool approachable for new users.

CAST differentiates itself from other supply chain optimization packages in having very accurate road and grid networks of the world's countries. As such, it is able to estimate required transportation times with great precision. A second point of differentiation is the inclusion of isochrone modeling. The tool is able to graphically show the area that is serviceable from a certain depot in a specified amount of transportation time. Thirdly, CAST includes the cost center of gravity and volume center of gravity heuristics as solution options of locating new facilities. CAST is more user-friendly when compared to SCS, but is restricted to solving only the single-period optimization case.

From the present work's perspective, CAST is able to model a simple "component-assembled product" relationship. Unfortunately, the components and assembled products must be declared separately for each location where manufacturing operations may be carried out. This results in similar cumbersome model alterations, as in the case of SCS.

3.3.4 Supply Chain Guru

Supply Chain Guru (Llamasoft, 2007) is a more recent supply chain optimization software package that was first introduced to the market in 2003. Its goal is to provide a solution where a single network model can be optimized and simulated without requiring user involvement in model conversion. Supply Chain Guru uses Xpress-Mosel as its mixed integer program solver (as CAST V10), and its simulation functionality is carried out by ProModel. According to the taxonomy in chapter 3.2, Supply Chain Guru is a *tool* that employs *optimization* and *simulation* methods.

The main entities in the Supply Chain Guru data model are Products, Sites, Demand, Inventory Policy, Sourcing Policy and Transportation Policy. The input parameters are given in tabular format, and both optimization and simulation parameters are stored in the same table for a particular model entity. For instance, a Demand table contains the aggregate demand figures for optimization purposes, and the interarrival times of orders and their size distributions for simulation. The Inventory Policy table specifies how products are stored at a site, when the replenishment orders are generated, and what the replenishment quantities are. The Sourcing Policy table specifies for each product whether it is made at the site, or acquired externally. It also stipulates which supply site should receive replenishment orders according to the decided policies – two examples of policies are single source (i.e. the component is always sourced from the same location), and maximal inventory (i.e. the component is sourced from the location which has the most inventory). Transportation Policy records parameters telling when and how products are shipped, how much the shipping costs, what the lead times between the sites are, under what circumstances shipments may be expedited, and which transportation modes are to be used in certain lanes.

From the current work's perspective, Supply Chain Guru can handle multi-level Bill-of-Materials at all sites. However, the specification remains location-oriented. If, for example, the Bill-of-Materials changes, the user must also change the Inventory Policy and Sourcing Policy of the respective locations. Compared with SCS and CAST V10, Supply Chain Guru still has the important advantage of being able to both optimize and simulate a single network specification.

3.3.5 Reported methods and tools for supporting network flow / demand allocation problem solving during new product development

The literature describing the specific quantitative methods and tools for supporting supply chain decisions, particularly network flow / demand allocation, is not very wide. In fact, Fine, Golany and Naseraldin (2005) state that before 2005 most of the tradeoffs between product, process and supply chain design were specified via qualitative frameworks. As was stated earlier, one of the most important decisions during new product development regards the product architecture (Krishnan & Ulrich, 2001). Typically the goal of designing a product family's architecture is to employ a maximal number of common parts, as this regulates the variation in component demand even in the presence of large variations in the demand for a single product (Simchi-Levi, Kaminsky, & Simchi-Levi, 2004).

Design for Logistics approaches have used "rules-of-thumb" to suggest maximal common platforms for product families. These suggest a compromise between inter-product differentiation and the use of common parts enabling better component forecasts (Martin, Hausman, & Ishii, 1998; Ericsson & Erixon, 2000). A case in point is Hewlett-Packard, which reported on the deployment of a simple web-based 'rough-cut' analysis method which analyzes the effect of the number of SKUs, product-internal modularization, postponement strategy, and component lead times on inventory carrying costs (Cargille & Bliss, 2001; Cargille & Fry, 2006). These analyses were not performed on the entire supply chain. Rather, if the rough-cut methods produced inconclusive results, a team of supply chain specialists at Hewlett-Packard was summoned to perform a detailed optimization and simulation study (ibid.). The common thread in the works published before 2005 is that simple rules-of-thumb were used to estimate the approximate effect of product structures on supply chain cost. Reports on commercial deployments of these methodologies were also scarce, with the exception of Hewlett-Packard.

Fine, Golany and Naseraldin (2005) were among the first to produce a quantitative method for analyzing tradeoffs in product design, process design and supply chain configuration. They employed a weighted goal programming approach, where the user decides the weights of component yield rates, supplier lead times, product modularity, and the dependence on single suppliers to be considered in evaluation of goal objectives. Their approach produces a single solution for the preferred product design, process design and supply chain configuration. The method does not produce the complete state space of supply chain options, but requires iterative runs with varying weights of decision variables. Another approach for applying mixed integer programming to consider product structure and supply chain configuration simultaneously was published by Lamothe, Hadj-Hamou and Aldanondo (2006). These authors developed a systematic methodology for translating market requirements into product features, which is translated to a Generic Bill-of-Materials (GBOM). GBOM is a structure which specifies product structure options with AND-OR logic. A number of candidate BOMs may be generated from a GBOM. The rest of their mathematical program formulation specifies the possible locations in which each component may be manufactured, and the relevant transportation links. Their formulation considers manufacturing and transportation costs, but does not include inventory carrying cost, duties or taxes. Lamothe, Hadj-Hamou and Aldanondo (2006) report that the proposed model has been implemented as a C++ application which calls the CPLEX solver to perform optimization. Their article does not indicate definitively if the method has been deployed commercially. However, a case study involving a car manufacturer is presented.

Since the publication of these two works, the literature has expanded to include, e.g., taboo search techniques for solving the original formulation of Lamothe et al. (El Hadj Khalaf, Agard, & Penz, 2009). Despite these developments Khan and Creazza (2009) conclude that more integration between product development and supply chain decisions is needed.

From the current work's perspective, the works of Fine et al. (2005) and Lamothe et al. (2006) are important uses of optimization for the concurrent determination of product design and supply chain structure, but they lack the possibility of generating the entire solution space of supply chain options. The early rules-of-thumb reported in Cargille & Bliss (2001), and Cargille and Fry (2006) suffered from the omission of the complete supply chain in tradeoff calculations.

3.4 Synthesis of present methods and tools for solving selected supply chain problems during new product development

Chapter 3.1 specified five literature-based requirements for methods and tools which solve network flow / demand allocation problems during new product development. They are: (1) possibility to model supply chains in a product structure-driven manner, (2) possibility to refine and reuse supply chain analyses made for coarse product structures,

(3) possibility to include various parameters in supply chain decision-making, (4) possibility to model complex supply policies, and (5) possibility to generate the complete state space of supply chain options, when this number is moderate (below 10.000).

In chapter 3.2 it may be inferred that generic optimization models are able to satisfy the requirements except for requirement (5). Simulation methods are unable to satisfy requirement (5) and support requirements (1) and (2) poorly. Heuristic methods possess the same characteristics. Finally, the reported hybrid and emerging methods do not fully support requirements (1), (2), and (3), and are unable to satisfy requirement (5). Whereas optimization methods always converge to the optimum result, thus violating requirement (5), hybrid and emerging methods may be developed so that each of the requirements is satisfied.

Table 1 shows in detail how the methods and tools specifically targeted at solving network flow / demand allocation problems during new product development satisfy the above requirements. Methods are referred to by their publication details, whereas the tools are identified by their proper names. The early works dealing with part commonality indices are not explicitly mentioned, but are assumed to be included in the citation of Cargille and Fry (2006). Similarly, the work of El Hadj Khalaf, Agard, and Penz (2009) is not included as it presents an alternative solution method to the formulation of Lamothe et al. (2006) with the same solution characteristics.

Table 1 Satisfaction of new product development specific modeling requirements by current methods and tools

	Req (1)	**Req (2)**	**Req (3)**	**Req (4)**	**Req (5)**
(Cargille & Fry, 2006)	NO	NO	PARTLY	NO	NO
(Fine et al., 2005)	YES	PARTLY	YES	YES	NO
(Lamothe et al., 2006)	YES	PARTLY	PARTLY	YES	NO
SCS	NO	PARTLY	PARTLY	YES	NO
CAST V10	NO	PARTLY	PARTLY	YES	NO
Supply Chain Guru	NO	PARTLY	PARTLY	YES	NO

Table 1 shows that none of the current methods and tools satisfy all of the requirements presented in chapter 3.1. Interestingly Requirement (1) is supported by mathematical programming formulations, but the supply chain optimization packages do not contain an applicable user interface. Each of the studied supply chain optimization packages adheres to the general pattern of supply chain model building where materials are attributes of geographical location entities (Guedes, Saw, & Waller, 1995; pp. 46-47). Moreover, the tools do not support the addition of extra user-specified parameters to the decision model, which can be taken into account during model optimization. Also, the re-use and refinement of existing supply chain analyses for coarse product structures is straightforward only in special cases. As for the methods included in Table 1, none support the generation of complete state space of supply chain options as specified in Requirement (5). The model of Cargille and Fry (2006) performs limited cost tradeoff analyses between the number of SKUs, component lead times, and inventory levels, but does not consider the view of the entire supply chain. The formulation of Lamothe et al. (2006) omits inventory carrying cost, duties, and taxes from their model.

In summary, the present literature study found that the current methods and tools for the network flow / demand allocation problem do not optimally support the requirements inherent in new product development. Therefore, the task of the present research is to construct a suitable one. The next section lays the groundwork for this task by presenting the pertinent methodological considerations.

3.5 Methodological considerations – constructive research

The previous literature review indicates that there is a need to construct a new method or a tool to support network flow / demand allocation problems during new product development. Constructive research (Kasanen, Lukka, & Siitonen, 1993) provides the best fit to the situation at hand. The next subsections present a generic overview of constructive research, and a discussion about how this research method is applied to the present case.

3.5.1 Constructive research in general

The constructive research approach solves problems through the construction of models, diagrams, plans and organizations (Kasanen, Lukka, & Siitonen, 1993; p. 243). The prime concern in constructive research approach is to produce a new artifact which solves a concrete problem, and to demonstrate that the construction works in practice. The constructive research approach has often been criticized for being too close to technical consulting (Labro & Tuomela, 2003). However, there are notable differences between the two areas. Labro & Tuomela (2003, p. 410) list some of these characteristics:

- In consulting, it is common to apply an existing technique to a different context with marginal alterations.
- In consulting, the generalizability of a method to a wider contexts is not of concern
- In consulting there is no incentive to connect findings to theory because there is no felt obligation to publish findings

Labro and Tuomela (2003) further discuss the differences between action research and constructive research. Whereas the creation of a new artifact is of prime concern in the constructive research approach, the aim of action research is to make planned interventions to social situations to carry out experimentally made changes (Labro & Tuomela, 2003; pp. 412-13). Lukka, one of the originators of the constructive research approach noted that action research involves academicians picking up a new management practice and diffusing it (Lukka, 2000; in Labro & Tuomela, 2003; pp. 412-13). Action research is an effective method of establishing closer ties between industry practitioners and academicians. Without similar efforts practitioners may engage in uninformed action, and academicians may produce theory without application. However, the innovative work of creating a new construct is better supported by constructive research.

The main steps in carrying out constructive research are as follows (Kasanen, Lukka & Siitonen, 1993; p. 246).

1. Find a practically relevant problem which also has research potential.
2. Obtain a general and comprehensive understanding of the topic.
3. Innovate, i.e., construct a solution idea.
4. Demonstrate that the solution works.
5. Show the theoretical connections and the research contribution of the solution concept.
6. Examine the scope of applicability of the solution.

The success of a new construct is established through three levels of market test: weak, semi-strong and strong (Kasanen, Lukka & Siitonen, 1993; p. 253). This phase is a key part of constructive research as "there is no lack of formal optimization models which supposedly solve managerial control problems but which no one is using in practice" (ibid., p. 253). The truth value of a new construct should be established in the 'market for innovations' where end users have the possibility of accepting or rejecting a new development. These market tests are defined as follows (Kasanen, Lukka, & Siitonen, 1993; p. 253):

> Weak market test: Has any manager responsible for the financial results of his or her business unit been willing to apply the construction in question in his or her actual decision making?
>
> Semi-strong market test: Has the construction become widely adopted by companies?
>
> Strong market test: Have the business units applying the construction systematically produced better financial results than those which are not using it?

According to Kasanen et al. (1993), even the weak market test is quite demanding and many new constructs will fail it. On another note, Labro and Tuomela (2003) criticize the weak market test of being too coarse in granularity. They suggest a classification where the weak market test is divided into finer detail. They argue that it is important to know whether the new construct is used ad hoc, in parallel with old systems, or whether it is the only one in use. They also differentiate between situations where a construct is used by one person, one team, a strategic business unit, a division, or the entire organization (Labro & Tuomela, 2003; p. 431). This more accurate notion of a weak market test is useful in comparing the penetration of new constructs, as the semi-strong market test requires wide adoption by companies which usually takes years.

In summary, constructive research fits perfectly into the present situation as a new construct is created for supporting network flow / demand allocation problems during new product development. This construct must be shown to work in theory and in practice.

3.5.2 Constructive research applied to the present case

The present discourse has shown that finding an effective method or tool for solving the network flow / demand allocation problem during new product development is a question with research potential. The simultaneous treatment of many possible product designs, several possible component suppliers and manufacturing facilities demands a lot from the chosen modeling method. Literature study showed that the present methods and tools do not fulfill the key requirements put forth by earlier authors. Therefore, the creation of a new construct is justified.

Two research questions were born out of the literature study, which are answered by articles I and II, respectively:

1) Is there a product structure-driven method of analyzing supply chains, which is also capable of generating full state spaces of supply chain options?
2) If so, what is its exact mathematical formulation?

The following section describes the new construction, and demonstrates how it fulfills the modeling requirements presented in chapter 3.1. Finally, chapter 3.7 concludes with a brief discussion of results.

3.6 DSN Setup Tool – A Petri-net based tool for supply chain decision-making during new product development

During this research a Petri net-based method was created for supporting supply chain optimization during new product development. As described in chapter 3.2, Petri nets are a hybrid and emerging method for supply chain management. However, in the current literature there were some capabilities of Petri nets, especially dealing with the exploration of complete state spaces, which were not employed in the publications. Therefore, the current research ventured to contribute to hybrid and emerging methods for supply chain management by demonstrating how these capabilities may be used for benefit.

Petri nets have been used since the 1960s to model parallel and distributed digital systems (Jensen, 1996). Moreover, they have recently been used very actively to simulate business processes (Russell & ter Hofstede, 2009). They have not been used to compute demand supply network options because the space and time limitations of the algorithms have been seen as insurmountable (van der Aalst, 1992). However, Petri nets and reachability analysis can be used for supply chain analyses during new product development, because the number of supply chain possibilities at that point is rather small. The number of options is typically measured in hundreds and a few thousands because demand points are given only for major geographical regions and product structures are specified using the main subsystems. In these instances, reachability analysis remains tractable. The following sections describe the Petri net foundations of the created method, and its implementation in the case company.

3.6.1 DSN Setup Tool – description of the underlying Petri net formalism

This subchapter describes the underlying Petri net formalism which forms the core of the DSN Setup Tool. DSN, a specific process of the case company, involves the selection of suppliers and factories to produce upcoming products. DSN Setup Tool is used in solving the associated network flow / demand allocation problems. The skeleton which defines the types of nodes, arcs, and cost elements that may be present in the DSNnet is termed the *DSNnet_skeleton*. A net which respects the definitions of a particular DSNnet_skeleton is termed a *DSNnet*. A DSNnet with an initial marking is called a *DSNnet_system*. The following provides the definitions of a DSNnet_skeleton, DSNnet and DSNnet_system. DSNnets are analyzed via reachability analysis which is described in detail in article II, and Appendix A of the present work. Pictorial examples which aid in understanding DSNnets are presented in chapter 3.6.3.

Definition 1

DSNnet_skeleton = { NodeTypes ∪ { AND_node, OR_node }, ArcTypes,

ParameterPool, ParameterMap }, where

NodeTypes = types of nodes in analysis (e.g. customer, manufacturing, buffer)

ArcTypes = types of arcs in analysis (e.g. transportation arcs)

ParameterPool = collection of all cost parameters used in the analysis

$\text{ParameterMap} \subseteq \text{NodeTypes} \times 2^{\text{ParameterPool}} \cup \text{ArcTypes} \times 2^{\text{ParameterPool}}$

Definition 2 – DSNnet

DSNnet = { Nodes, Arcs, F, DSNnet_skeleton, Typing, Valuation}, where

Nodes = set of Nodes (Petri Net places, color = Real)

Arcs = set of Arcs (Petri Net transitions, transition guards are TRUE)

$F \subseteq Nodes \times Arcs \cup Arcs \times Nodes$, each arc's inscription is X, a Real number variable

DSNnet_skeleton = as defined above

$Typing \subseteq Nodes \times DSNnet\_skeleton.NodeTypes \cup Arcs \times DSNnet\_skeleton.ArcTypes$

Valuation = **V**(Nodes, Arcs, Typing) **–** a function that assigns relevant parameter values to each node and arc according to its type

Definition 3 – DSNnet_system

DSNnet_system = { DSNnet, $Nodes_0$, $M_0 : Nodes_0 \mapsto Real$ }

DSNnet_system has a valid DSNnet structure, a set of initial nodes ($Nodes_0$), and initial marking (M_0) that maps a single token of type Real to each initial node (this value represents a customer's average daily demand for a certain product).

The firing rule for DSNnet_system is that of High Level Petri Nets.

3.6.2 DSN Setup Tool – description of the tool and its environment

DSN Setup Tool as implemented in the case company consists of a web-based user interface, and a database for storing input parameters and analysis results. A rough schematic of the tool as seen by the end user is depicted in Figure 10. The DSN Setup Tool generates the complete state space of possible supply chain options on the basis of the input parameters specified.

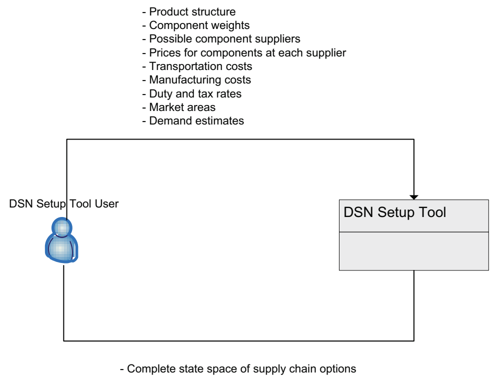


Figure 10 *High level schematic of DSN Setup Tool use case*

Figure 11 provides a representation of data used in the DSN Setup Tool. Transportation costs, duty rates, customer hierarchies and supplier hierarchies are stored as master data in a database. Product structures may be imported from a Product Data Management (PDM) system, but are more commonly specified manually through the user interface. Estimates of the manufacturing costs at facilities may be imported from the Activity Based Costing (ABC) tool. The parameters entered through the user interface include component weights, supplier options for components including prices, and market regions with product lifetime volumes. The modeling requirements of chapter 3.1 are discussed next.

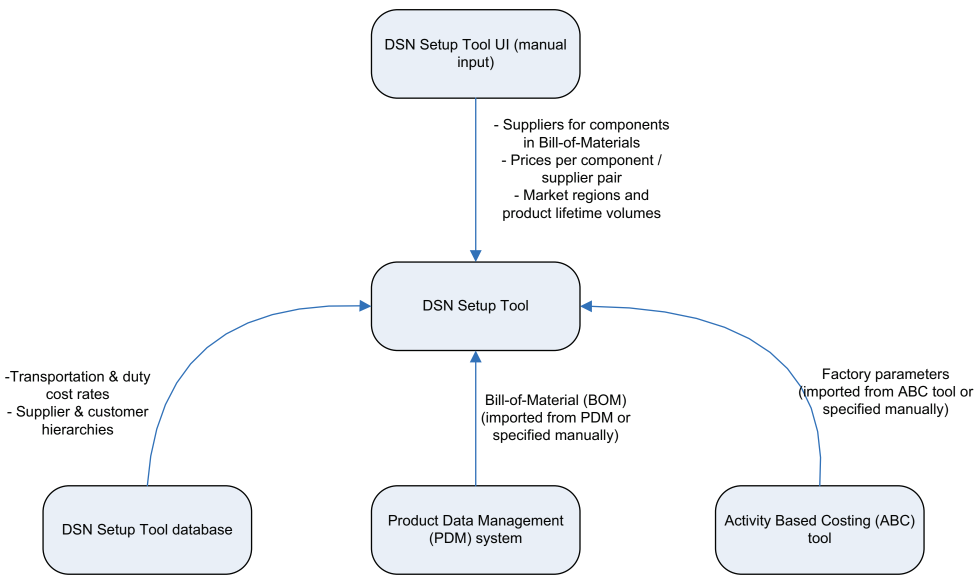


Figure 11 *System integration of DSN Setup Tool*

3.6.3 DSN Setup Tool – exhibiting the fulfillment of modeling requirements pertaining to network flow / demand allocation problems during new product development

This section discusses each requirement for supply modeling methodology specified in subsection 3.1. Each of the requirements will be treated separately, and it is shown how DSNnets, and DSN Setup Tool in particular, may be used to satisfy them.

i) Possibility to model supply chains in product structure-driven manner

DSNnets allow the construction of supply chain models in the following manner:

1) A product structure with arbitrary number of tiers is constructed
2) Sourcing options for each component are specified
3) Node and arc specific data (e.g. manufacturing costs, investment costs, transportation costs) is specified
4) Reachability analysis is run to generate every possible demand supply network.

Figure 12 demonstrates the principle of product structure-driven modeling. On the left is a simple product structure, which is delivered to one end market. Component A has a single supplier, and components B and C have two possible suppliers each. The associated DSNnet is shown on the right. The information that is entered into the nodes and arcs includes, e.g. end market demand, component weights, transportation cost per kg in the used transportation legs, manufacturing line investments, and manufacturing cost of components. This information comes partly from master data stored in a database, and partly through user input. The particular example below translates into four possible supply chains, which are generated together with cost estimates using the reachability analysis algorithm.

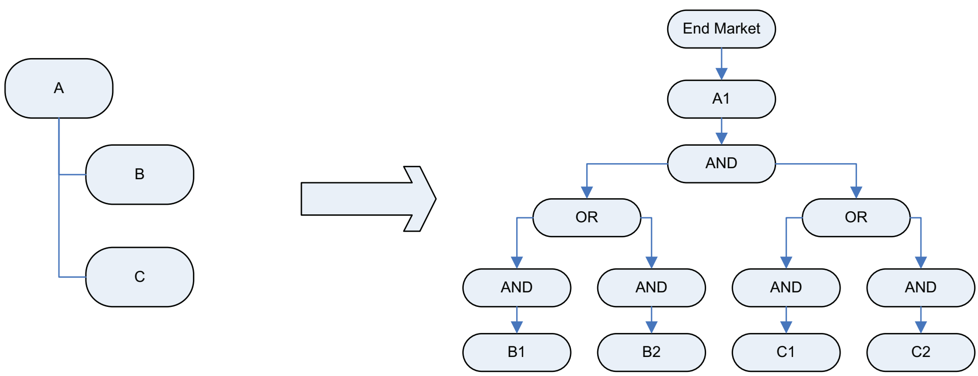


Figure 12 *Translating a product structure with constraints to a DSNnet*

A reachability analysis routine (Jensen, 1996) is run separately for each end customer included in a DSNnet. Following this, the results are aggregated. Figure 13 shows an example DSNnet with three end markets. There are 4 supply chain options for each market. In total, the DSNnet reachability analysis algorithm will produce $4^3 = 64$ supply chain options.

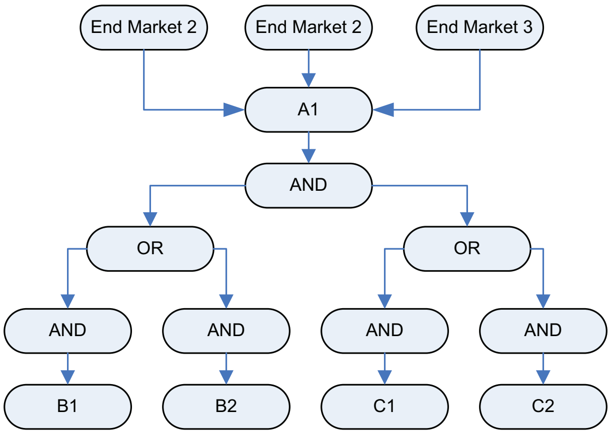


Figure 13 *A DSNnet containing three end markets*

ii) Possibility to refine and reuse supply chain analyses made for coarse product structures

Figure 14 demonstrates how DSNnet methodology enables the re-use of supply chain analyses carried out for coarse-grained product structures. Figure 14 is a refinement of Figure 12 where component C has received a substructure, component D. Component D has two supply options: Either everything is sourced from supplier D2, or 70% of the volume is acquired from supplier D1 and 30% from D2. In Figure 14 it is evident that only the bottom right part of the DSNnet has changed. The DSNnet of Figure 12 may be re-used by adding just the component D with the corresponding supply options.

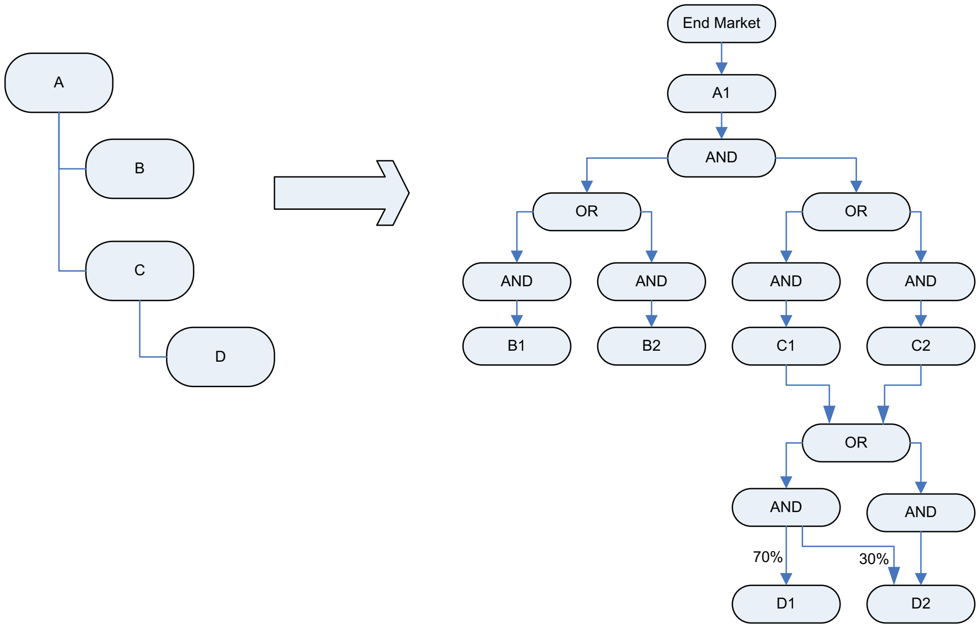


Figure 14 *Forming a DSNnet for a refined product structure*

iii) Possibility to include various parameters in supply chain decision making (and extend the parameter set if needed)

The highest-level specification of a DSNnet is a 'skeleton' which describes the pertinent parameters which are used in computing the demand supply network cost. It is possible to have a different DSNnet skeleton for a company depending on its needs – e.g. inventory carrying cost is not a great concern to a brick and nail manufacturer, whereas it is of substantial interest to high tech companies (Helo, 2006). A DSNnet skeleton also describes the Node Types and Arc Types which are allowed in a valid DSNnet. Typically there are Manufacturer Nodes, Buffer Nodes, Customer Nodes, and Transportation Arcs. Figure 15 shows the DSNnet skeleton used in the DSN Setup Tool implemented in the case company.

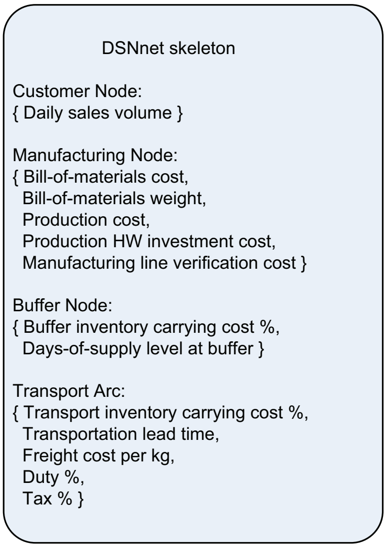


Figure 15 *DSNnet skeleton used in the case company*

The addition of new cost parameters is relatively simple in the DSN Setup Tool. The tool implementation gives users the possibility to employ four user-defined direct and indirect cost elements. No additional programming is required to take these into use. However, if the cost functions associated with the new cost elements are complex (i.e. if a cost element is neither a pure indirect cost nor a pure direct cost), additional programming is required.

iv) Possibility to model complex supply policies (e.g. multi-sourcing cases)

Figure 16 provides evidence of the possibilities for DSNnets in modeling complex sourcing policies. The figure contains a simple parent-child product structure, which is supplied to one customer. Component A is sourced in an 80 / 20 split from suppliers A1 and A2, or everything is acquired from A2. Component B is sourced in a 50 / 50 split from suppliers B1 and B2, or alternatively B2 is used as the only supplier.

More generally, DSNnets allow the modeling of arbitrary supply splits through the use of AND and OR nodes. A topmost OR node is needed under each component node to allow for several sourcing options. Under each OR is an AND node. The AND node allows the possibility to do an N-way supplier split. If an AND node has only one child node, it represents the single-sourcing case.

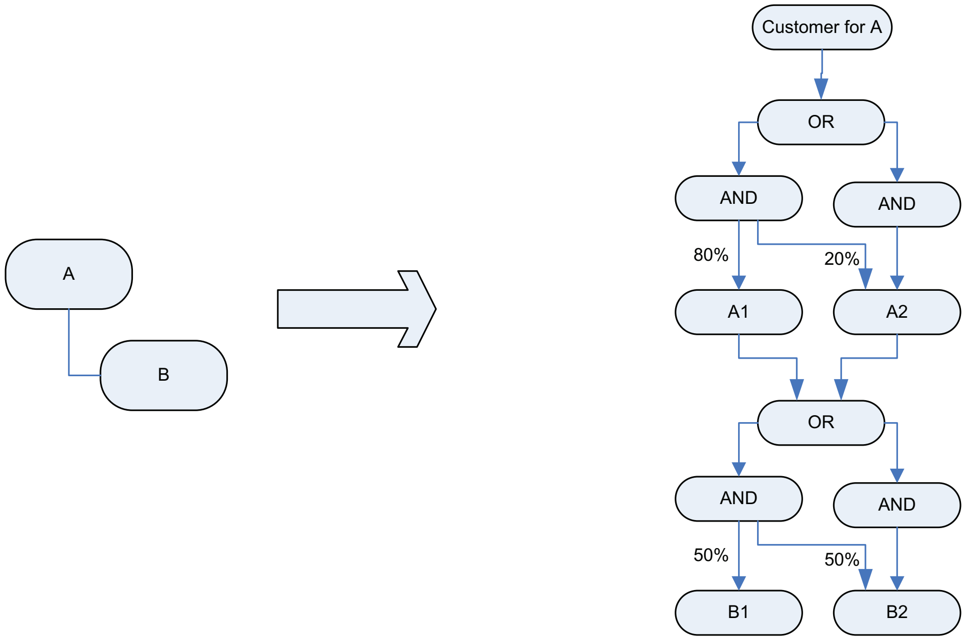


Figure 16 *Depiction of generic sourcing modeling with DSNnet*

v) Possibility to generate the complete state space of supply chain options when this number is moderate (e.g. below 10.000)

The reachability analysis algorithm detailed in article II produces the complete state space of all demand supply network options with associated costs. The performance figures from the same article have suggested that generating circa 1000 demand supply network options takes approximately 5 minutes. Elaborating on this, 10,000 demand supply network options are generated in 50 minutes. During tool development, a case with 12,000 options was executed. This optimization run lasted a little over one hour. The performance is clearly worse than in optimization methods which converge very quickly to the optimal solution (without computing the entire state space). In practice the logistics professionals in the case company have very seldom analyzed cases with more than 100 options. For these typical cases the performance is fast.

3.7 An effective method and tool for supply chain analyses during new product development – a summary

The literature study performed in this chapter presented a need for construction of a novel method and a tool for solving network flow / demand allocation problems that better addresses the needs imposed by new product development. The research proceeded to construct the DSN Setup Tool based on DSNnets (i.e. a Petri net formalism). It has been shown that DSNnets and the DSN Setup Tool in particular satisfy the five requirements laid out for an effective modeling method.

What is required next is to show how DSNnets and the DSN Setup Tool may be applied in solving real-life supply chain management problems. Moreover, evidence should be obtained about the practical usefulness of the DSN Setup Tool through end user interviews. These issues will be addressed in the next chapter.

4 ESTABLISHING THEORETICAL AND PRACTICAL USEFULNESS OF DSNNETS AND THE DSN SETUP TOOL

The previous chapter described the reasons for constructing a new modeling method for supply chain analyses during new product development. It also showed that the five requirements for an effective modeling method were satisfied by DSNnets and the DSN Setup Tool. One of the requirements of the constructive research approach is to demonstrate the practical usefulness of a new artifact (Kasanen, Lukka, & Siitonen, 1993). In this chapter, the market test-based validation will be extended by somewhat more analytical, deeper, field studies. The approach is inductive rather than deductive because the latter requires a much longer time frame to complete. A case study approach was selected for this part of research as it suits such field studies very well.

In the present chapter case study methodology will first be described in general terms. Following this, an application of DSNnets to solve a business problem is presented as in article III. Finally, a summary of the end user interviews addressing the practical usefulness of the DSN Setup Tool is given.

4.1 Methodological considerations – case study research

Case study research is a useful research method when answering "how" or "why" types of research questions (Yin, 2003a, p. 1; Eisenhardt, 1989). Yin (2003a, p. 13-14) defines a case study in two parts:

"1. A case study is an empirical inquiry that
- Investigates a contemporary phenomenon within its real-life context, especially when the boundaries between phenomenon and context are not clearly evident

2. The case study inquiry
- Copes with the technically distinctive situation in which there will be many more variables of interest than data points
- Relies on multiple sources of evidence, with data needing to converge in a triangulating fashion
- Benefits from the prior development of theoretical propositions to guide data collection and analysis"

The five components of a case study research are (Yin, 2003a; p. 21):

1. a study's questions
2. its propositions, if any
3. its unit(s) of analysis
4. the logic linking data to the propositions; and
5. the criteria for interpreting the findings

Case studies have been used particularly in the social sciences but lately also in business research (Yin, 2003a; p.1). One of the criticisms towards positivist world-view has been its focus on being able to find atoms of knowledge and find causal relationships between them in a 'mathematical' manner (Susman & Evered, 1978; p. 582; in Lanning, 2001; p. 41). In many instances this requires that an investigator has control over behavioral events. In most real-life cases, and especially in social sciences, it is not possible to control behavioral events. In these situations, a quantitative experimentation may not be the best way to proceed in research, and there is a need to deeply understand some representative cases of a phenomenon for description and theory building. It is particularly in these situations that case study research methodology is of use (Yin, 2003a; p. 5).

Stake (1995, pp. 2-4; in Lanning, 2001; p. 47) has divided the types of case studies into three categories: intrinsic, instrumental and collective case studies. In intrinsic case studies, the object is to understand a single unit of study without concern for the study's generalizability. In instrumental case studies, the study's object is used to answer a research question or to solve a problem. In effect, the case study is the "instrument" for problem solving. In collective case studies, several instrumental case studies are combined into one entity. Kasanen et al. (1991, p. 315; in Lanning, 2001; p. 47) also note that case studies may be either descriptive (describing, analyzing, explaining and understanding) or normative (modeling, guiding and suggesting).

Yin divides case studies into three subtypes: exploratory, descriptive and explanatory (Yin, 2003a; p.1). In exploratory case studies, the data collection is undertaken before the actual research questions are formed (Yin, 2003b; p. 6). The objective of an exploratory case study is to create theories for further testing, and thus there will be a rationale and a general direction in each inquiry even though the actual research question arises only after the data collection has begun. In descriptive case studies, the research is targeted to describing certain phenomena through, e.g., the viewpoints of competing rival theories (Yin, 2003b; pp. 22-27). Explanatory case studies attempt to form a cause and effect relationship between events. In contrast to the descriptive case studies, explanatory case studies use rival theories to explain phenomena and decide which rival theory explains the situation the best. In the absence of a suitable theory in literature, the explanatory case study attempts to form a new explanation.

In terms of research setup, a case study may have a single-case or multiple-case design (Yin, 2003a; p. 40). In a single-case study, there is a single object of study which is thoroughly researched. In multiple-case studies, there are several units of study which makes it possible to apply, e.g., replication logic for theory testing and validation (Yin, 2003a; p. 47).

Yin (2003a) uses case studies primarily for theory testing, whereas Eisenhardt (1989) employs case studies for theory building. In the current work, the motivation is theory testing. The initial hypotheses are that DSNnets and DSN Setup Tool are useful in 1) solving theoretical supply chain problems as well as in 2) supporting practical supply chain decisions. The first hypothesis is tested by applying DSNnet method to solving a theoretical supply chain management problem. The second hypothesis is addressed through interviews with DSN Setup Tool end users.

4.2 Establishing the theoretical usefulness of DSNnets and the DSN Setup Tool – application to a generic supply chain management problem

The usability of DSNnets was tested in Article III by solving the question "What is the optimal demand supply network for a varying product in the presence of multiple value-adding layers, several variation options and multiple consumers with fluctuating demand distributions?" In particular, the general problem may be pictured as in Figure 17. Namely, there a variable number of customers whose demand patterns may change. Furthermore, arbitrary number of supplier tiers, either the focal company's own manufacturing sites or external suppliers, exists. The supply policies may also be arbitrary between each successive tier. The number of product (or semi-finished product) variants at each tier is also a variable.

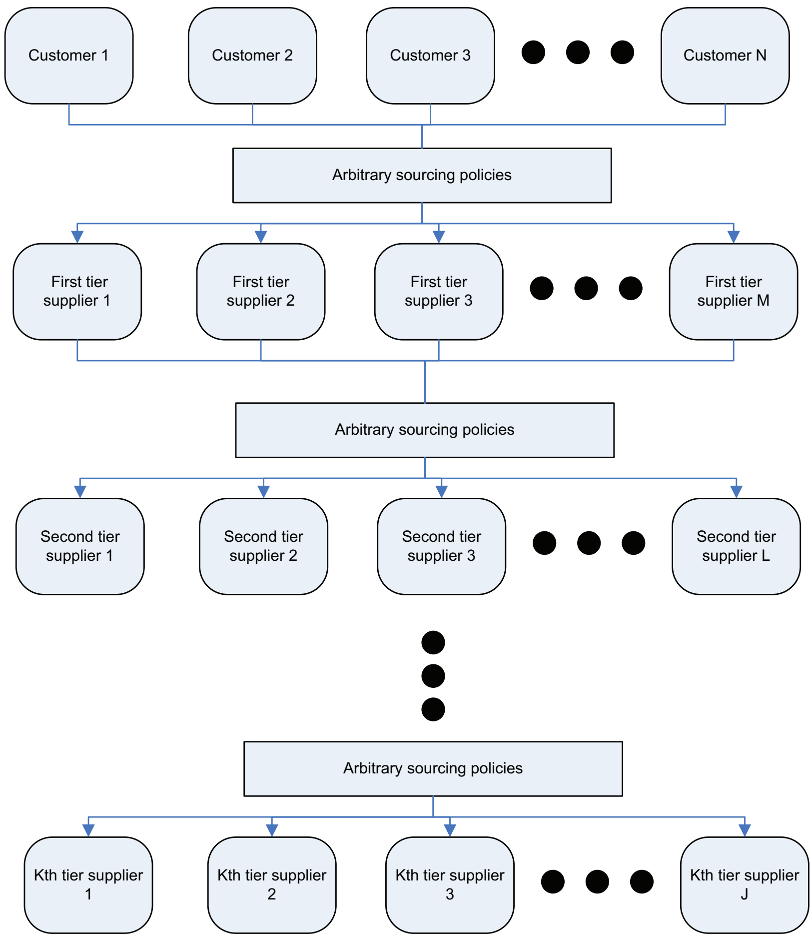


Figure 17 *Generic problem setting for Article III*

This generic research problem was operationalized as total logistics cost analysis for a two-tier product structure with several supply chain options. There was a common "Kernel Product" from which one to five variants of the "Complete Product" were assembled. The costs which were taken into account in the analyses were transportation cost, buffer inventory carrying cost, and transport inventory carrying cost. The detailed numerical values for product weights, transportation costs and transportation times may be found in article III.

The DSNnet denoting the operationalized research problem is pictured in Figure 18. There were three factory choices for "Complete Product" and two for "Kernel Product". In addition, there were three end customers, which displayed two different demand patterns. The lifetime volume of "Complete Product", 1 million pieces, was distributed between the end customers A, B, and C in ratios 80-10-10, or 40-30-30. The total number of supply chain options is $(2 \times 3)^3 = 216$. However, due to tax and duty restrictions (not modeled as cost elements, but known in practice), only three supply chain options were seen as viable.

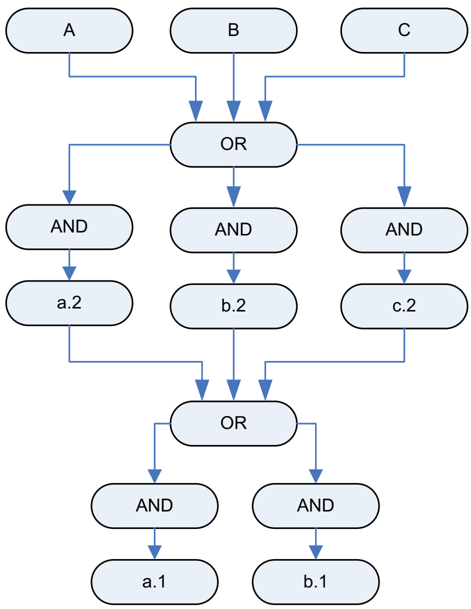


Figure 18 *DSNnet for Article III study*

In Option 1, there were two factories used to produce "Kernel Product", and three factories assembled the "Complete product". "Kernel Product" factory b.1 ships the semi-finished product to "Complete Product" factory c.2, where a seven-day buffer is kept to safeguard against demand fluctuation. Option 1 is described in Figure 19.

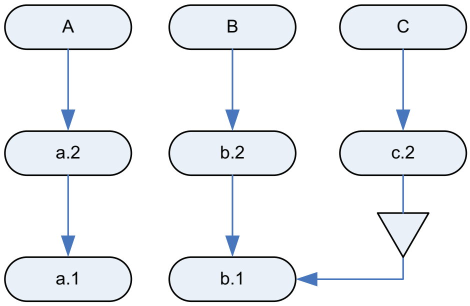


Figure 19 *DSNnet Option 1 in Article III*

In Option 2, only one "Kernel Product" factory is utilized, whereas three "Complete Product" factories remain operational. Two of the "Complete Product" factories maintain a seven day buffer for semi-finished goods. Option 2 is depicted in Figure 20.

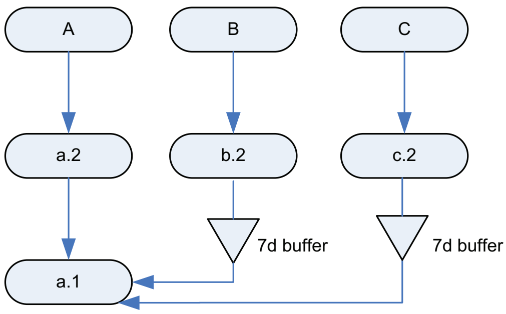


Figure 20 *DSNnet Option 2 in Article III*

In Option 3, two "Kernel Product" and two "Complete Product" factories are used for production. One of the "Complete Product" factories delivers goods to two end customers. In this scenario, no semi-finished good buffers are needed, but a higher transportation cost is incurred. It is also assumed that the transportation lead time between "Complete Product" factory a.2 to end customer C is acceptable. Figure 21 shows Option 3.

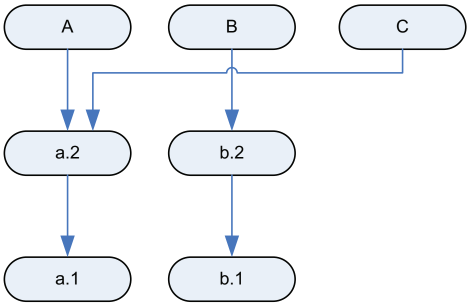


Figure 21 *DSNnet Option 3 in Article III*

The three candidate networks were analyzed for total cost with one to five product variants. The effect of variants was taken into account using Square-Root Law (Maister, 1976; Eppen, 1979; Zinn et al., 1989). The effective days-of-supply figure in the reachability analyses was treated as DOS√**n**, with **n** indicating the number of variants. The computed numerical results are presented in Article III, but a summary is shown in Table 2.

Table 2. Answers to Article III research question

Demand Distribution	**Number of "Complete Product" variants**	**Optimal supply chain**
80-10-10	1	Option 1
80-10-10	2	Option 1
80-10-10	3	Option 3
80-10-10	4	Option 3
80-10-10	5	Option 3
40-30-30	1	Option 1
40-30-30	2	Option 1
40-30-30	3	Option 3
40-30-30	4	Option 3
40-30-30	5	Option 3

Results show that increasing the number of product variants pushes the optimal demand supply network to Option 3 where "Complete Products" are shipped directly from factory a.2 to market area C. Local final assembly is more cost efficient in case of 1 and 2 product variants. The effect of skewed (80-10-10) versus evenly (40-30-30) distributed demand was that the savings were larger in the latter.

The study performed in article III demonstrated that in the presence of reasonable transportation costs, centralized production of end product variants is more cost efficient than localized final assembly. In this case the net effect of inventory carrying cost from several variants outweighs the increased transportation costs. The results are very sensitive to transport cost fluctuations. The study also demonstrated how the square root law may be integrated into DSNnets. In summary, the study showed that DSNnets may be used to solve a theoretical supply chain problem.

4.3 Establishing the practical usefulness of DSNnets and DSN Setup Tool – interviews with end users

The constructive research approach requires that new constructs are tested with market tests to assess their practical utility. In this section, the results of the weak market test applied to the DSN Setup Tool at the case company are presented. The case company is a corporation consisting of two major divisions, mobile phones and networks. Case study interviews were carried out with a total of 20 DSN Setup Tool end users, who came from the two major divisions.

The common thread in the interviews was to find an answer to the question “Has any manager responsible for the financial results of his or her business unit been willing to apply the construction in question in his or her actual decision making?” To answer this question, the end users of the DSN Setup tool were asked these two questions:

1. What problems does the DSN Setup Tool solve in your current work?
2. Comment on the impact of the tool in your organization

Table 3 shows the answers to Question 1. The specific problems solved by the DSN Setup Tool are briefly described. Plant focus analysis was the most common subproblem solved with the DSN Setup Tool. Plant focus analysis determines which factories of the case company and component suppliers should be used to build the supply chain for a product. Three interviewees said that the DSN Setup Tool offers “proof for business associates”. Analyses may be done on-line with the upper management present, to answer possible what-if questions. In variant analyses, operations and logistics managers consider the profitability of introducing a new phone variant (e.g. a new color variant). One interviewee from the networks division performed analyses on the optimal number of warehouses and their level of inventory. There was one respondent who no longer used the DSN Setup Tool. The reason was the unavailability of accurate input and master data for analyses. Some of the comments from the interviewees regarding question 1 were: “The tool is like a giant calculator that summarizes all possible demand supply network scenarios with associated costs, and provides a way for fact-based decision making” (an operations and logistics manager from the mobile phones subdivision). “The Petri net tool gives direction to decision-making for global setups, especially for products whose lifetime volumes are between 5 million and 30 million pieces. In these situations there is a need for a good understanding of possible supply chains, and the supplier base.” (product design site manager, mobile phones subdivision).

Table 3. Interviewee's answers to Question 1 "What problems does the DSN Setup Tool solve in your current work?"

Reply to Question 1	Number of respondents
Product structure analyses	1
Plant focus analyses	7
Variant analyses	2
Decision support in general	10
Proof for business associates	3
Solving optimal number of warehouses and their inventories in a specific country	1
Not used anymore	1

The answers to the second question are given in Table 4. The impact of the DSN Setup Tool on the respondents' organizations was seen collectively as positive. There were two individuals who did not see benefits from the tool. One of the people was the user who experienced problems in acquiring correct data for analyses. The other was a person who did not get support from DSN Setup Tool key users, and did not know all of the tool's capabilities. The other interviewees saw benefits in many areas. The networks division of the case company utilized the DSN Setup Tool extensively. The tool was used in making two distribution hub open / close decisions, in analyzing network element deliveries to end customers, and in the division's five-year strategy scenario analysis. In the mobile phones division one of the greatest benefits of the DSN Setup Tool was the mindset change in people to focus on end-to-end logistics costs rather than a single factory's costs. The product programs have reportedly saved money with the tool. In fact, one senior operations and logistics manager from mobile phones division commented: "The tool has lowered logistics costs inside the company, and the tool's benefits are already tens of times the amount of money that was spent in developing it." The other benefits mentioned included better collaboration between operations, logistics and sourcing organizations, better quality of decisions, better visibility to decisions, and better product introductions.

Table 4. Interviewees' answers to Question 2: "Comment on the impact of the tool in your organization

Reply to Question 2	Number of respondents
Positive effect in general	10
Collaboration between operations, logistics and sourcing organizations	2
Logistics cost savings	3
Mindset change to focus on end-to-end logistics costs	8
No impact	2
Visibility to results	1
Better product introductions	1
Better quality of decisions	1
Analytical support in two hub open / close decisions	1
Decision support in end customer delivery analysis	1
Scenario building in 5 year strategy process	1

4.4 Summary of the theoretical and practical usefulness of DSNnets and the DSN Setup Tool

In summary, this chapter presented evidence for DSNnets' and DSN Setup Tool's theoretical and practical usefulness. An instance of a generic supply chain problem concerning the combined effect of the number of end product variants and end customer demand fluctuations on the optimal supply chain choice was solved using DSNnets. The practical usefulness of the DSN Setup Tool was examined via interviews with 20 end users at the case company. The results of the interviews suggest that DSNnets and the DSN Setup Tool pass the weak market test. The tool is used in actual day-to-day work, and has been beneficial for operations and logistics managers in designing supply chains for new products. With the present evidence it may be concluded that DSNnets and the DSN Setup tool demonstrate theoretical and practical usefulness in supply chain management.

5 HOW CAN SUPPLY CHAIN DECISIONS FOR NEW PRODUCTS BE FOLLOWED UP?

Once a modeling methodology is constructed and proved to be of use while designing a product, its value in practice will be realized only if the delivered results can be validated in actual day-to-day business. The actualized supply chain costs for new products should be monitored to ensure that they remain within a certain tolerance of the planned costs. If this is not the case, a supply chain re-design should happen.

In this chapter, the question of a supply chain decision support system validation from a company's reporting systems is dealt with. First a generic literature review regarding decision support systems and supply chain management is given which points to the need for further study of decision support system validation. This is followed by the actual study, as reported in Article IV. The results are given in two parts – a detailed quantitative case study dealing with the case company, and a set of multiple case studies involving interviews with six other companies. The study proposes a generic IT landscape that ameliorates the process of following up actualized versus planned supply chain costs. For clarity, the DSN Setup Tool is considered a Decision Support System (DSS) in the following.

5.1 Decision support systems and supply chain management

Decision support systems (DSS) are computer-based interactive systems that support human decision makers through the use of data and models (Eom et al., 1998; p. 109; Eom & Lee, 1990b). Each of the mathematical formulations introduced in chapter 3.2 may be used as a solver inside a decision support system. To illustrate, the supply chain optimization packages discussed in chapter 3.3 constitute decision support systems. For instance, Supply Chain Guru is a decision support system that allows users to apply either optimization or simulation to supply chain problems. Decision support systems may contain several solution modules – optimization, discrete simulation, heuristics, or knowledge base. The choice of which 'solver' is to be used is done either by the user or by the decision support system. Figure 22 shows a schematic composition of a decision support system. In Supply Chain Guru, the user interface is the Windows-based application the user interacts with. The data base is Access based, and the model base contains Xpress-Mosel for mixed integer programs and ProModel for discrete event simulation.

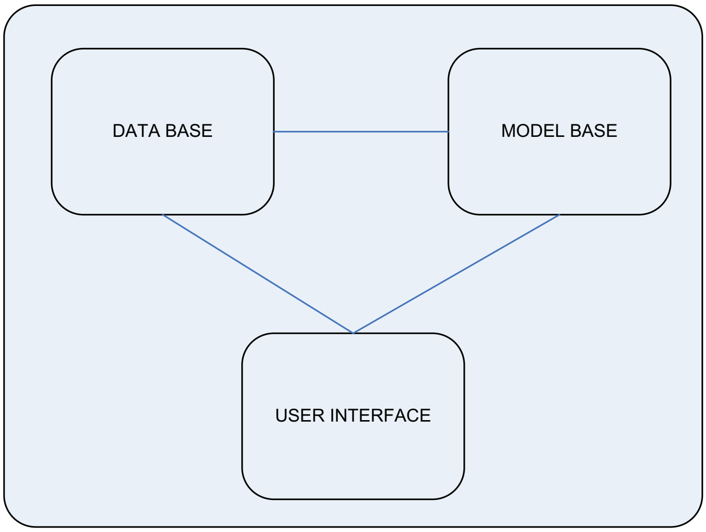


Figure 22 *Schematic composition of decision support systems*

The DSS field has been categorized into seven types of systems: File drawer systems, data analysis systems, analysis information systems, accounting models, representational models, optimization models and suggestion models (Alter, 1980; cited in Eom et al., 1998; p. 115). *File drawer systems* provide on-line access to particular data items without analyses. *Data analysis systems* provide on-line data retrieval, manipulation, and display of current and historical data by means of pictures and summaries. *Analysis information systems* are capable of providing business intelligence from transactional systems and combine internal data with external data via statistical packages. *Accounting models* attempt to estimate consequences of actions on financial statements such as estimate-of-income statements and balance sheets. *Representational models* include partly non-definitional models which include simulation models. *Optimization models* generate optimal solutions based on constraints. Finally, *suggestion models* guide a decision maker to an optimal solution in a structured way similar to "wizards" in commercial software packages. According to Eom et al. (1998), 38% of published DSS's are optimization models, 25% are representational models and 17% are analysis information system models.

DSS's have also been categorized according to the number and kinds of people (roles) that are using them (Arnott & Pervan, 2005). A personal decision support system (PDSS) is a small-scale system developed for one manager, or a small number of independent managers for a single-type of decision task (ibid., p. 68). In fact, the DSN Setup Tool developed as part of this research is a PDSS. A group decision system (GDSS) attempts to support a number of decision makers jointly responsible for a decision but who are, e.g., located on different continents (Benbasat & Nault, 1990; p. 204). Executive Information Systems (EIS) provide reporting through organizational structures to the upper management. With EIS, management is able to drill up and down organizational structures for performance metrics (Arnott & Pervan, 2005, p. 71). Currently the term EIS has changed to include both Data Warehousing (DW) and Business Intelligence (BI) (Arnott & Pervan, 2005; March & Hevner, 2007). A data warehouse (DW) supports all managerial levels in decisions through acquisition of raw data, both internal and external, for analysis purposes (Arnott & Pervan, 2005). Business Intelligence (BI) refers to the business knowledge that is obtained when algorithmic analyses are applied to the data gathered in a Data Warehouse (March & Hevner, 2007).

Logistics and distribution problems have been a key application area of DSS research right from its inception (Eom & Lee, 1990a, p.67). In the first 18 years of DSS research, 33.1% of articles published dealt with marketing, transportation and logistics area (ibid.). Decision support system research typically contains detailed descriptions of a decision problem, the support a decision maker needs, and the system delivering the support. Operations management and logistics still remain the key areas in which the decision support system study is carried out (Eom et al., 1998). However, decision support system research faces challenges of practical relevance as it has not produced enough practical research on important topics such as data warehousing and business intelligence (Arnott & Pervan, 2005). Also, it has been recognized that decision support systems generally are haphazardly validated (Kleijnen, 1995).

The research community of Supply Chain Management long since recognized the importance of supply chain performance (Lee & Billington, 1992; Beamon, 1999; Gunasekaran, Patel, & Tirtiroglu, 2001). Many authors indicate the need to define and follow metrics in two dimensions: financial versus non-financial, and strategic versus tactical versus operational (Gunasekaran, Patel, & Tirtiroglu, 2001). Beamon suggests three performance measures necessary for any performance management system: resources, output and flexibility (Beamon, 1999). Strategic-level decisions require support from financial metrics, tactical decisions can be supported by both financial and non-financial metrics, and operational decisions should be supported via non-financial metrics (Gunasekaran, Patel, & Tirtiroglu, 2001). Typically there are too many possible metrics to choose from, and the issue is to determine the relevant ones, and to establish processes for follow-up (Gunasekaran & Kobu, 2007). In a field study involving top company executives, a group of researchers found that the level of customer-perceived value was the most important performance metric followed closely by variance in supply chain budget (Gunasekaran, Patel, & McGoughey, 2004). A follow-up work synthesized a list of 27 key performance indicators where the most important financial performance indicators were inventory cost, obsolescence cost, overhead cost, return on investment, selling price, stock out cost, transportation cost, value added and warranty costs (Gunasekaran & Kobu, 2007).

Therefore, supply chain cost is an important performance metric, and the accuracy of decision support tools that estimate supply chain cost should be validated. Unfortunately there is almost no published research about this problem (Jonsson, Kjellsdotter, & Rudberg, 2007, p. 817). Empirical evidence about supply chain decision support system use and expected benefits is missing (ibid.). One reason for the lack of publications may be the criticality of such systems to companies' competitive success. Another is the challenge in application landscapes. Many companies have developed supply chain decision support tools apart from the enterprise resource planning (ERP) systems, which presents an integration challenge that is often solved only by manual effort (ibid., p. 818, 824).

Both decision support system and supply chain research recognize the fact that validation of models is not generally done properly (Adelman, 1991; Kleijnen, 1995; Jonsson, Kjellsdotter, & Rudberg, 2007; Gunasekaran & Kobu, 2007). Two generic methods have been reported in literature to measure the effectiveness and validity of a decision support system (Adelman, 1991). First, to establish that a DSS produces a good prediction of reality, historical input data is fed into the model and output is compared with the real output value observed in history. If there is a significant correlation between the DSS's prediction of output and the observed output, the model is acceptable. Secondly, the impact of a DSS on an organization is studied via an experiment where two groups of decision-makers participate. In a pre-test, individuals in both groups are given the same set of problems, and they must make decisions unaided. In a post-test, one group is allowed to use a DSS to solve problems whereas the other group must make decisions unaided. If the individuals in the former group consistently choose better solutions, there is some evidence that the decision support system might have organizational impact.

In validating supply chain decision support systems in particular, the role of application landscape is central. Cost predictions are made using granular product structures in separate planning systems, but business execution data is collected in enterprise resource planning systems. Therefore, the role of a correct IT application landscape is central. Supply chain decision support system validation is also a longitudinal activity. Therefore, data warehousing technologies with precise data architecture mappings between product structures in planning and selling phases are required. Validating a decision support system is important as it adds to the credibility and increases the probability of organizational acceptance of the system (Kleijnen, 1995).

5.2 Considerations on how strategic supply chain decisions may be followed-up

A key requirement of constructive research methodology is the assessment of validity and reliability. Internal validity refers to a concrete causal relationship between independent and dependent variables. In the case of DSNnet methodology, assessing internal validity can be translated to the problem of verifying that the actualized product costs are within a tolerable range of those predicted. This was the problem setting for article IV where the research question was formulated as: "Are there challenges in validating a company's supply chain management decision support system with official reports? If yes, what are some ways of solving the issue?" This research question was operationalized as an in-depth quantitative case study inside the focal company, and a small set of multiple case studies involving interviews with other companies.

5.2.1 Single case study – validating DSN Setup Tool results with actual values in the case company

Verifying the actual costs with predicted costs for a single product inside the case company was a very manual exercise. The reason is visible in the current application landscape surrounding DSN Setup Tool shown in Figure 23. It reveals that transportation costs, duty rates, tax rates, supplier hierarchies and customer hierarchies are stored in a DSN Setup Tool database. Automatic integrations exist between the Product Data Management (PDM) system, the Activity Based Costing (ABC) tool and DSN Setup Tool. The user of a DSN Setup Tool specifies the product structure, possible suppliers for individual components with cost, and the market regions with lifetime product volumes through the User Interface. However, four key systems containing actualized product cost information remain unintegrated: Material Cost Planning (MCP) system, Demand Supply Planning system, Days-of-Supply (DOS) reporting and Cost Reporting (CR) system.

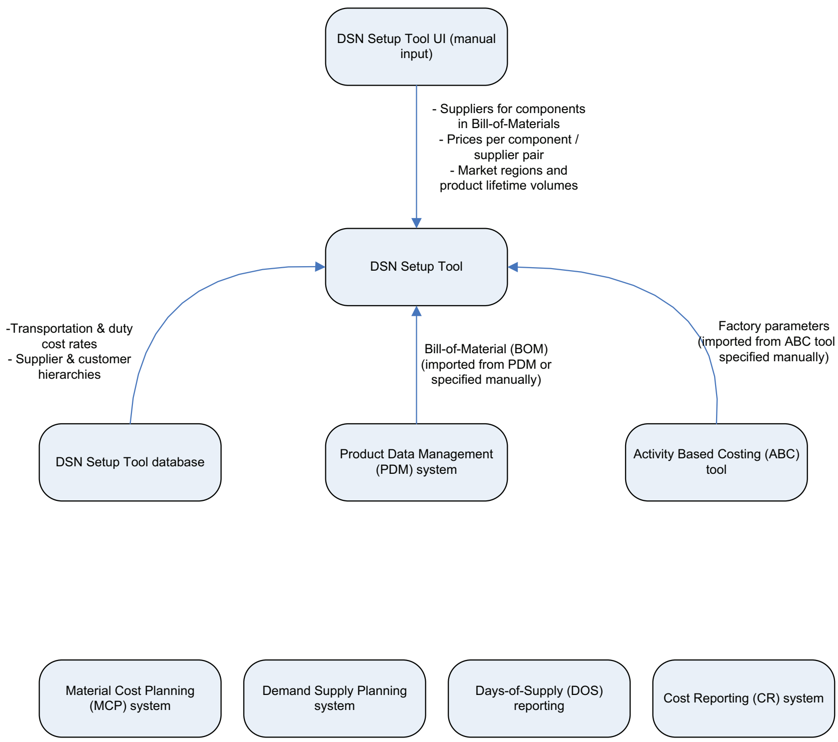


Figure 23 *Application landscape relevant to DSN Setup Tool*

The actual process for cost comparison is given in Figure 24. The research discovered that step 2 of the process was rendered complicated due to the fact that the Material Cost Planning (MCP) tool stored only the billing address of the supplier company. The actual supplier plants where the components were shipped from had to be inquired from some 20 logistics managers worldwide. This required one week to complete. A second point of difficulty emanated from the unreliable component weight data in the PDM system. For 15 out of 199 components, the deviation in weight was more than 5%. Because of this, each component had to be weighed separately for analysis purposes.

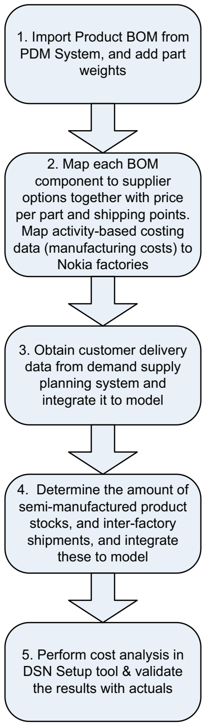


Figure 24 *Process of validating DSN Setup Tool cost predictions with actual costs*

The output of the analysis was that the actual end-to-end logistics cost for the chosen product was 4.6% lower than the DSN Setup Tool prediction. Almost all the deviation was due to Bill-of-Materials cost. The MCP tool reported component costs for a pre-selected variant of the product, whereas the CR tool reported costs based on actualized production volumes. Such a large deviation was possible as the sales package contents (hence, the costs) for the same phone model can vary greatly from country to country. All in all, the user of the MCP tool had not selected the representative variant optimally for this particular phone model. Nevertheless having a cost prediction within 5% of the actual cost can be considered as fairly good performance.

In summary, the research found that a large-scale predicted versus actual cost follow-up for the entire product portfolio at the case company would be infeasible. The biggest hindrance was the presence of many unintegrated systems, requiring significant manual work to perform the cost follow-up. The research also found some suggestions that can remedy the situation. First, the component weight inaccuracies in the PDM system could be overcome by having an electronic scale integrated into PDM for the automatic input of component weights. This would eliminate the typographical errors that can easily occur in cases of very small component weights. Secondly, the MCP tool should record not only the billing address of the suppliers, but also the possible shipping locations. This would mean that each factory of the focal company producing a particular product should manage its own Bill-of-Materials structure with shipping suppliers' locations. Finally, the system landscape given in Figure 25 automates the information exchange for cost follow-up. First, the MCP system should receive the Bill-of-Materials directly from the PDM system complete with accurate component weight information. As MCP would manage the shipping points for components, it should send the Bill-of-Materials with component prices, correct weights and shipping points automatically to the DSN Setup Tool. Days-of-supply levels should be integrated in a similar manner. The CR system should automatically transfer production volumes and factory-to-factory shipment volumes to the DSN Setup Tool. Finally, a data warehouse should collect up-to-date transport cost information and send updated transport lane prices to the DSN Setup Tool database.

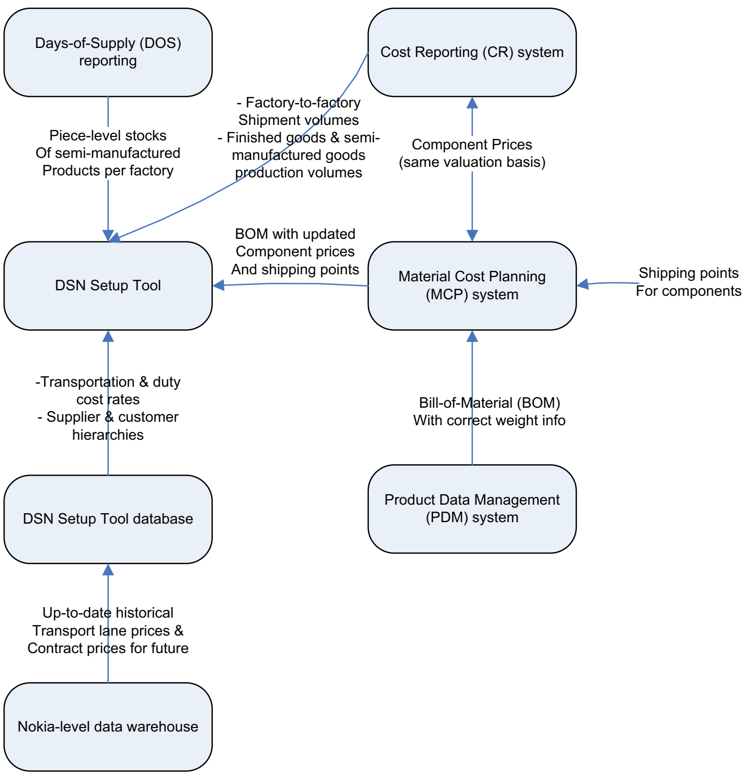


Figure 25 *Suggested application landscape for DSN Setup Tool cost validation*

In fact, the suggested application landscape for the case company is an instance of a generic application landscape that enables supply chain decision follow-up. The generic framework proposed in article IV is shown in Figure 26. In this picture, the key inputs for supply chain cost prediction are automatically transferred to a supply chain DSS. The supply chain decision with a cost prediction is stored in a data warehouse and analytics platform. The finance and reporting platform is also made aware of the decision taken. The execution of the supply chain is monitored by the data warehouse and analytics platform based on actual supply chain costs. If a significant deviation from the plan is noticed, a supply chain redesign request is escalated to the DSS. It is evident that this framework requires a well-structured data architecture and seamless system integration to work properly. Further discussion on this topic is included in chapter 5.2.3.

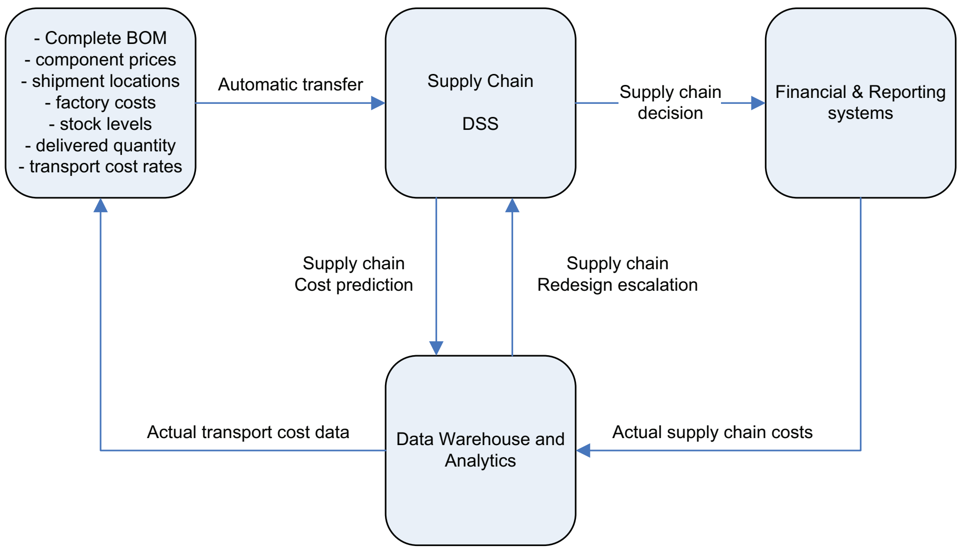


Figure 26 *Generic application landscape supporting strategic supply chain decision follow-up*

5.2.2 Multiple case studies – interviews with other companies

The findings from the first case study suggested that there are clear challenges in validating strategic demand supply network decisions with reporting systems. The first case study suggested a modified application landscape as a way of enabling strategic supply chain decisions.

The result created a need to validate the importance of strategic supply chain decision follow-up with other companies. A set of six small interviews were conducted with the other companies' logistics managers. Three questions were asked: namely, if a strategic supply chain decision support system was used in the company; whether its results were validated; and what were seen as the key challenges in using supply chain decision support systems and validating the results. The companies, labeled A through F, are described in the following.

Company A is a global player in computing hardware, software and consulting industry. It has a headcount of over 100,000 employees worldwide. Company B is a global supplier of process industry machinery and systems. It has a global workforce of 30,000 people. Company C is a global supplier of pharmaceuticals, active pharmaceutical ingredients and diagnostic tests. It employs circa 3,500 people. Company D is a pharmaceutical distributor operating in Scandinavia. It has a workforce of 4,400 people. Company E is a global supplier of industrial machinery, specializing in metal cutting and mining. It employs circa 50,000 people worldwide. Company F is a pharmaceutical distributor operating in Scandinavia with circa 500 employees.

The results of the interviews are summarized in Table 5. The interviews revealed that DSSs are used in four (out of six) companies. Companies B and C did not use any DSS to support strategic supply chain decisions. Company B relied on transfer pricing rules and Company C on supplier strategy. Companies D and F used DSSs for operational-level decisions. Only Companies A and E used DSSs to support strategic supply chain decisions.

Validation proved to be difficult for the interviewed companies. Only Companies D and F were able to validate their (operational level) decisions through accurate tracking of product margins and logistics costs. Company A was unable to validate the supply chain decisions because its DSSs are stand-alone and not integrated into ERP. Company E could not validate its supply chain decisions because the company has many non-integrated ERPs and departmental silo-thinking prevents open information sharing. In conclusion, the use of DSSs for operational decisions was found to be more commonplace than their use for strategic supply chain decisions. However, none of the interviewed companies was able to validate strategic supply chain decisions with actual day-to-day execution data.

5.2.3 Cross case analysis

The answer to the first question addressed by article IV, "are there challenges in validating a company's supply chain management decision support system with official reports?", is a resounding yes. The first case study dealing with Nokia revealed that the system performing strategic supply chain decisions is not well integrated into the systems performing day-to-day business operations. The process of validating a single product's predicted supply chain cost with actual data required two weeks of work, and is infeasible for the case company's entire product portfolio. Most of the work involved finding out actual shipment points for the 199 components that formed the product in question. There were also challenges with the quality of master data regarding component weights. Finally, the material cost planning tool predicted the cost of a different sales package variant than the actual cost reporting system.

Table 5. Interview results concerning strategic supply chain decision follow-up (other companies)

Company	Is DSS used for strategic supply chain decisions?	Are the results of strategic supply chain decisions validated?	What are the key challenges in using DSSs and validation?
Company A	Yes	No	DSS is not integrated into company ERP, employees are unable to use DSSs efficiently, reporting does not support end-to-end process view
Company B	No	No	Supply chain decisions are based on (inaccurate) transfer prices, there are no reporting capabilities for measuring logistics costs
Company C	No	No	The company is unable to change a supply chain quickly (supplier strategy dictates the supply chain)
Company D	No (but used for operational level decisions)	No (but operational level decisions are validated)	The company can't influence the strategic supply chains of drug manufacturers
Company E	Yes	No	Employees are not yet taught to use the DSS, integration into ERP system does not exist, departmental silo-thinking hinders openness
Company F	No (but used for operational level decisions)	No (but operational level decisions are validated)	The company is unable to influence manufacturers

The multiple case studies based on interviews strengthened the findings of the first case. Of the six companies interviewed, only two (Companies A and E) were using strategic supply chain design tools. In these cases the systems were stand-alone. In terms of validation, neither of the two companies was systematically able to validate strategic supply chain decisions with actual day-to-day execution data. For one company the main reason was the fact that DSSs were stand-alone. For the other, a heterogeneous application landscape as well as departmental silo-thinking proved to be obstacles. The limited number of interviewed companies with strategic supply chain decision support tools is a limitation of this part of the research. A large survey study should be conducted before strong conclusions about supply chain decision support system validation in industry can be stated.

Comparing the single case study with the multiple case studies, a common theme regarding the inability of validating strategic supply chain decisions is the presence of unintegrated IT systems. Therefore Figure 26 presented a generic system landscape that enables strategic supply chain decision follow-up. In it the key inputs for supply chain cost prediction are automatically transferred to the DSS. The quality of the supply chain decision is constantly monitored, and if a significant deviation from the plan is noticed, a supply chain redesign request is escalated to the DSS. In order for the setup to work efficiently the input data quality must be high. For instance, weighing scales integrated directly into a PDM system can be used to ameliorate the quality of component weight data. Most importantly, the planning structures and reporting structures used in the company's central systems must be compatible with each other. The supply chains should be planned in both strategic and tactical timeframes in such a manner that they may be directly compared with actual cost reports. An important nuance in this challenge is that a strategic planning window is measured in months and years, whereas the tactical one is measured in weeks and months. A practical data architecture challenge is to design a way in which the transition from a strategical to tactical time horizon is made, that will enable iterations between the two planning horizons.

6 DISCUSSION

The following subsections present the theoretical and practical implications of the current work, its limitations and possible topics of further research.

6.1 Theoretical and practical implications

This work contributes to three areas. First, it contributes to decision support for supply chain decisions during early product development, which has not been adequately studied (Chiu, Gupta & Okudan, 2009). The early decision support methods dealt with rules-of-thumb for deciding, e.g., the optimal number of stock keeping units in light of the anticipated inventory carrying costs (Cargille & Bliss, 2001). These methods did not employ end-to-end supply chain modeling. Since then, more advanced methods for deciding product family bill-of-materials together with the optimal supply chain have appeared starting with Fine, Golany, & Naseraldin (2005) and Lamothe, Hadj-Hamou, & Aldanondo (2006). Fine et al. (2005) present a weighted goal programming approach to investigate the product design, required manufacturing processes, and the configuration of the supply chain in parallel. The work of Lamothe et al. (2006) used a custom made C++ code with a CPLEX optimizer to solve a case study for a manufacturer. Optimization engines such as CPLEX converge toward the optimum result and do not provide the results for the entire state space of possible networks. However, in real-world supply chain optimization the management is interested in more than just the optimum result. In fact, Goetschalckx and Fleischmann (2008) reviewed the earlier literature and found evidence of a need to explore hundreds of supply chain variants during a real-life optimization case. One of the reasons is to better manage and plan for supply chain disruptions (Norrmann & Jansson, 2004). This presents a need for a technique that explores the complete state space of demand supply network options when the number of possibilities is reasonable. During early product design, where the bill-of-materials is often on a subsystem level, the number of options regarding supply chain options reaches from tens to a few thousand. In these cases, Petri net reachability analysis, as presented in this work, is still a tractable method, and outputs all supply chain options with associated costs.

The constructed Petri net-based method and tool for supply chain analysis during new product development contributes to the hybrid and emerging techniques for SCM. The dissertation also provides case-based evidence of the theoretical and practical useability of hybrid methods. In fact, DSNnets and the DSN Setup Tool were found to provide good decision support for supply chain problems in the mobile phones division and the networks division of the case company. Especially the networks division made extensive use of the DSN Setup Tool's capabilities. DSNnets belong to the class of coarse modeling methods (Lehtonen, 1999). These are methods that enable repetitive zoom-and-focus analyses for supply chains where high level (i.e. coarse) models quickly point out problem areas, which are further examined by focused analyses. The strength of coarse modeling methods is that they can rapidly provide a result which is relatively accurate, without having to perform detailed and time-consuming optimization and simulation model construction (de Treville & van Ackere, 2006). DSNnets may similarly be constructed for various levels of product structure granularity depending on the need for detail. At a high level, the use of DSNnets for deciding a supply chain configuration closely resembles the use of queueing networks, another coarse modeling method, to discover the dynamic properties of a single supply chain (Suri et al., 1995).

The second field where this dissertation contributes is decision support system validation. Decision support systems validation has not been optimally done historically, which has contributed to, e.g., resistance in tool deployments (Kleijnen, 1995). In the field of supply chain management, in particular, a need was voiced for more studies in estimating the impact of using decision support systems (Jonsson, Kjellsdotter, & Rudberg, 2007). When decision support system validation is done correctly, tools gain credibility and the management obtains a way of following up each decision. The IT application architecture framework developed in article IV provides a way in which, e.g., data warehousing may be integrated with supply chain decision support to improve input data quality and enable decision quality follow-up.

The third field where this dissertation contributes is the application of Petri net theory, and the diffusion of such methods to industrial use. Petri nets have been used extensively in modeling the dynamic behavior of systems (e.g. van der Aalst, 2000b; van der Aalst, 1998; Desel & Erwin, 2000). However, work has not been carried out in applying Petri nets to the static optimization of supply chains. When the anticipated state space is reasonable in size, reachability analysis is suitable to compute all possible supply chain options. The advantages of modeling with Petri nets include the ability to do hierarchical modeling (adding detail to a coarse grained product structure), and the ability to present complex supplier relationships (e.g. multi-source cases) more simply than by using enumeration techniques. Finally, as a large amount of research into specifying business processes with Petri nets for dynamic analysis already exists (van der Aalst & ter Hofstede, 2005; Russell & ter Hofstede, 2009), results of static optimization with Petri nets may be converted into a dynamic Petri net for simulation purposes. Moreover, several research papers have been published focusing on the use of Petri nets for modeling concurrent engineering (Wu & O'Grady, 1999; Blackhurst, Wu, & O'Grady, 2005; Blackhurst, Wu, & O'Grady, 2007). DSNnets may readily be combined with these methods.

6.2 Validity of results

Yin (2003a) introduces the concept of validity, and gives four subtypes. Relevant to the present discussion are internal validity and external validity. Internal validity ensures that there is a causal relationship between independent and dependent variables. External validity investigates the populations to which the obtained results may be generalized.

The validity of the results produced by the newly created method was internally validated in the case company by parallel analyses with existing cost analysis methods. This ensured that the new method produces results that are mathematically correct. The old costing methods were retired after the DSN Setup Tool gained widespread acceptance in the company. The results of the end user interviews in chapter 4.3, i.e., weak market test, also provide evidence to the internal validity of the new method. Chapter 5 investigated internal validity from another perspective – i.e., it enquired into how strategic supply chain decisions may be followed up from various systems to assess decision quality. This translated into IT application landscape considerations that suggest intelligent ways of connecting planning systems with enterprise resource planning and cost reporting systems. The reported IT application landscape in chapter 5 contributes as a method which may be used to establish internal validity in the company.

External validity can be dealt with only partly, as it has been recognized as one of the typical shortcomings of small case studies (Yin, 2003a). The six interviews carried out with external companies in chapter 5.2.2 contribute to this aspect. These interviews qualitatively established the use of strategic supply chain decision tools and validation in the companies, and explored possible hindrances to their use. As an outcome, it was found that strategic supply chain planning was done quite rarely, and in case it was done, the results were not validated.

The number of interviews was small, and contributed only little to establishing the external validity of the study results. The inability to obtain more interviews was influenced partly by the fact that the study was done by a person employed in a commercial company. Some people did not respond to the invitation to an interview because of concerns over data confidentiality. If an academic institution were to make such an investigation, the turnout of participants would very likely be better. Nevertheless, the evidence obtained from the few interviews provided some support to the importance of the present investigation, and provided minor evidence of its external validity.

6.3 Recommendations for further research

Further research could be done in at least four areas that build on this work. First of all, cross-company investigations into supply chain decision support system validation should be carried out. The small number of external companies that accepted the interviews in article IV was the greatest limitation of the current study's generalizability. It would be worthwhile to longitudinally follow a company that systematically follows up its strategic demand supply network choices and see if it achieves, e.g., better financial performance. Also, the types of system-supported approaches could be investigated to find the recommended solution.

Secondly, the IT architecture supporting supply chain decision support system validation (Article IV) is expensive to build, so studies into the typology of supply chain management problems could be done. It is instructive to find the types of supply chain problems where a heuristic will provide an adequate answer, and where a system-supported supply chain cost follow-up is needed.

Third, a Petri Net Markup Language (PNML) (Billington et al., 2003; Kindler, 2004) transfer format can be developed for the DSNnet. This will enable the (statically) optimized Petri nets to be transferred to dynamic Petri net simulation tools for performance analysis. In this way, an optimization-simulation iterative cycle may be established for Petri nets in a tool-independent manner. Such a transparent transfer language between mixed integer programming and discrete event simulation tools does not exist (Azadivar, 1999).

Finally, recent literature has shown interest in enabling supply chains to operate in a decision support system-centric way (Ivanov, Sokolov, & Kaeschel, 2010). This advance in theory should be followed up with practical evidence. The case study in article IV dealing with a case company's IT architecture that can support such decision follow-up is a contribution toward this aim.

7 SUMMARY

This dissertation searched for an effective methodology and a tool to support supply chain decisions during new product development. During new product development it is very customary for multiple product structure options to exist, and supply chain implications must be studied for each alternative. It is also known that during real-life supply chain optimization studies, a large number of alternatives, not just the optimal one, are of interest to the management. This led our efforts to searching for a method that supports supply chain analyses in a product structure-driven manner and generates the complete state space of network options.

In the past, supply chain decisions regarding, for instance, the number of SKUs, were made with simple rules-of-thumb. In these cases the whole network perspective was not analyzed. When the methods involving weighted goal programming and mixed integer programming surfaced in the mid 2000s, they were developed as stand-alone tools with no description of commercial deployments. The basic question any manager involved in a product development program wants to ask is: 'If I input my product structure, and the possible suppliers for each component, what are all of my possible demand supply network options?'. This simple question posed by Nokia logistics managers was the starting point of the entire work, which resulted in a product structure-driven Petri net-based methodology for supply chain optimization. The created construct was tested in its ability to solve a theoretical supply chain problem. The practical utility of the construct was examined through end user interviews.

Once a decision support system is constructed, the validity of its results must be proven. The available literature regarding this problem is limited despite the recognition of the problem's importance. Thus, the present dissertation contributed by conducting an in-depth study in the case company regarding the IT enablers that make it possible to follow the accuracy of strategic supply chain decisions.

In the future, it is instructive to study how companies that embark on the road of strategic supply chain decision follow-up will fare against their competition in financial terms. Furthermore, the types of system architectures that enable such follow-up in companies should be studied cross-industry to discover the best practices. Petri net markup language can be utilized to construct generic converters between the DSNnet and several available dynamic Petri net formalisms. Finally, there is growing interest in being able to execute supply chains in a decision support system-driven manner. For such systems to become a reality IT infrastructure must be able to support the validation efforts across different supply chain decision support systems.

APPENDIX A

The traditional Petri Net reachability analysis algorithm computes all reachable system states. In DSNnet formalism, reachability analysis differs in two important points:

1. The algorithm computes all possible paths of a system (valid DSNnet structures are directed and acyclic, guaranteeing the absence of infinite paths)
2. The algorithm allows for several initial states, and aggregates the separate reachability graphs to a single result

The result of the reachability analysis is a list of complete paths in the DSNnet, each associated with a cost. The optimum path is the one with the lowest cost. In DSNnet context, a reachability graph is an array of arrays (matrix) where each component array (matrix column) is one demand supply network setup with its cost.

The pseudocode for the algorithm is presented next. The abbreviation RG is short for "reachability graph", and arrays are indexed in C language style from 0 to array_size-1. The pseudocode uses four helper functions: *append_node*, *append_RG*, *aggregate_RG* and *add_per_item_costs*. *Append_node* appends a node (first argument) to all paths in the reachability graph (second argument). *Append_RG* joins two reachability graphs to form a single reachability graph – i.e., the appending of RG1 with 5 paths and RG2 with 3 paths results in a single RG with 8 paths. *Aggregate_RG* takes the Cartesian product of two reachability graphs, where each path in the resulting reachability graph has a cost equal to the sum of the two constituents. The Cartesian product of RG1 with 5 paths and RG2 with 3 paths has 15 paths. Finally, *add_per_item_costs* adds the costs of an arc or a node (the first argument) to each path in the reachability graph (the second argument).

Main DSNnet analysis routine

The main analysis routine is given in Algorithm 1. The investment costs are computed last because several customers can source their products from the same suppliers (manufacturing nodes). The total volumes for each supplier are known when the second for-loop has been executed. The second for-loop – aggregation of individual customers' reachability graphs – is also the source of the algorithm's computational complexity. Assume that we have *C* customers and *P* is the maximum of the number of DSN setup options for a customer. Then the size of the state space (and computation time) grows exponentially as $O(P^C)$.

```
RG_main (DSNnet_system) returns all DSN setups with cost {
    RG[] = array of new Reachability Graphs;
    total_RG = new Reachability Graph;
    for each initial customer node do
        RG[i] = RG_1_customer(customer_node[i]);
    end
    for i = 0..number of customer nodes -1 do
        total_RG = aggregate_RG( total_RG, RG[i]);
    end
    for each path in total_RG do
        investment_cost = compute_investments(path);
        add investment_cost to the path's cost;
    end
    return total_RG; }
```

Algorithm 1: Main Analysis Routine

Reachability Analysis Routine for a Single Customer Node

Algorithm 2 presents the algorithm that computes the reachability graph for one customer. It follows the traditional reachability analysis algorithm with the addition of AND and OR nodes.

```
RG_1_customer(Start Node with volume marking, Start RG) returns RG {
    RG[] = array of empty Reachability Graphs;
    theChildNodes[] = Array for children of AND and OR nodes;
    theChildArcs[] = Array for arcs leading to theChildNodes;
    Add volume marking to Start Node's total volume;
    if Start Node has 0 children then                              CONTINUES
```

```
        append_node(Start RG, Start Node);
        Start RG = add_per_unit_costs(Start RG, Start Node, NULL);
        return Start RG;
    else if Start Node has 1 childNode then
        childArc = DSN Arc between Start Node and childNode;
        append_node(Start RG, Start Node);
        Add volume marking from Start Node to childNode and childArc;
        Start RG = RG_1_customer(childNode, Start RG);
        Start RG = add_per_unit_costs(Start RG, Start Node, childArc);
        return Start RG;
    else if Start Node is AND then
        total_RG = new Reachability Graph;
        append_node(Start RG, Start Node);
        Add the volume marking to theChildNodes[] and theChildArcs[];
        for each theChildNodes[i] do
            RG[i] = RG_1_customer(theChildNodes[i], RG[i]);
        end
        for i = 0..number of Child Nodes-1 do
            total_RG = aggregate_RG(total_RG, RG[i]);
        end
        total_RG = aggregate_RG(Start RG, total_RG);
        return total_RG;
    else if Start Node is OR then
        total_RG = new Reachability Graph;
        append_node(Start RG, Start Node);
        Add the volume marking to theChildNodes[] and theChildArcs[];
        for each theChildNodes[i] do
            RG[i] = RG_1_customer(theChildNodes[i], RG[i]);                CONTINUES
```

```
        end
        for i = 0..number of Child Nodes-1 do
            total_RG = append_RG(total_RG, aggregate_RG(Start RG, RG[i]));
        end
       return total_RG;
}
```

Algorithm 2: Reachability analysis routine for a single customer

Computation of Investment Costs

Algorithm 3 presents the computation of investment costs. This part of the reachability analysis algorithm determines production volume-dependent investment costs in the supply chain. For instance, if the capacity of a single manufacturing line is 7500 pieces per day, this algorithm first computes the number of required manufacturing lines based on daily demand, and then multiplies the result by the investment cost of establishing a single manufacturing line.

```
compute_investments(one DSN setup) returns InvestmentCost {
    for each manufacturing node in setup do
        determine total volume throughput;

        determine number of manufacturing lines;
        InvestmentCost = InvestmentCost + (no. of manuf. lines * investment per line);
        end
        return InvestmentCost;
    }
```

Algorithm 3: Computation of investment costs

REFERENCES

Adelman, L. (1991). Experiments, quasi-experiments, and case studies: A review of empirical methods for evaluating decision support systems. *IEEE Transactions on Systems, Man, and Cybernetics, 21*(2), 293-301.

Aitken, J., Childerhouse, P., Christopher, M., & Towill, D. (2005). Designing and Managing Multiple Pipelines. *Journal of Business Logistics, 26*(2), 73-96.

Ajmone Marsan, M., Donatelli, S., & Neri, F. (1990). GSPN models of Markovian multiserver multiqueue systems. *Performance Evaluation, 11*(4), 227-40.

Almeder, C. & Preusser, M. (2007) A toolbox for simulation-based optimization of supply chains. In Henderson, S.G., Biller, B., Hsieh, M.-H., Shortle, J., Tew, J.D., & Barton, R.R. (Eds.), *Proceedings of the 2007 Winter Simulation Conference* (pp. 1932-39). New York: ACM.

Alter, S. (1980). *Decision Support Systems: Current practice and continuing challenges*. Reading, Massachusetts: Addison-Wesley.

Alves, G., Maciel, P., & Lima, R. (2008). Stochastic evaluation of supply chains and replenishment policies with Petri net components. *International Journal of Services Operations and Informatics, 3*(3-4), 221-41.

Andersen, M., Khler, S., & Lund, T. (1986). *Design for Assembly*. London, UK : IFS Publications.

Andradottir, S. (1998) A review of simulation optimization techniques. In Medeiros, D.J., Watson, E.F., Carson, J.S., & Manivannan, M.S. (Eds.), *Proceedings of the 1998 Winter Simulation Conference* (pp. 151-158). New York: ACM.

April, J., Better, M., Glover, F., Kelly, J.P., & Laguna, M. (2005) Enhancing Business Process Management with Simulation Optimization. *BPTrends, January 2005*. Retrieved January 15, 2008, from http://www.bptrends.com/resources_publications.cfm?publicationtypeID=DFC61D66-1031-D522-3EBDAB1F65A451AA .

April, J., Glover, F., Kelly, J.P., & Laguna, M. (2003) Practical introduction to simulation optimization. In Chick, S., Sanchez, P.J., Ferrin, D., & Morrice, D.J. (Eds.), *Proceedings of the 2003 Winter Simulation Conference* (pp. 71-8). New York:ACM.

Arnott, D. & Pervan, G. (2005). A critical analysis of decision support systems research. *Journal of Information Technology*, *20*, 67-87.

Artigues, C. & Roubellat, F. (2001). A Petri net model and a general method for on and off-line multi-resource shop floor scheduling with setup times. *International Journal of Production Economics, 74,* 63-75.

Azadivar, F. (1999). Simulation optimization methodologies. In Farrington, P.A., Nembhard, H.B., Sturrock, D.T., & Evans, G.W. (Eds.), *Proceedings of the 1999 Winter Simulation Conference* (pp. 93-100). New York : ACM.

Ballou, R. (1989). Heuristics: Rules of thumb for logistics decision making. *Journal of Business Logistics, 10*(1), 122-32.

Barloworld (2008). *CAST User Guide.* Barloworld Optimus Ltd, UK.

Barthomieu, B. & Diaz, M. (1991). Modeling and verification of time dependent systems using time Petri nets. *IEEE Transactions on Software Engineering, 17*(3), 259-73.

Beamon, B. (1999). Measuring supply chain performance. *International Journal of Operations and Production Management, 19*, 275-92.

Benbasat, I. & Nault, B. (1990). An evaluation of empirical research in managerial support systems. *Decision Support Systems, 6*, 203-26.

Benders, J. (1962). Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, *4*, 238-52.

Billington, J., Christensen, S., van Hee, K., Kindler, E., Kummer, Ol, Petrucci, L., Post, R., Stehno, C., & Weber, M. (2003). The Petri Net Markup Language: Concepts, Technology and Tools. *Lecture Notes in Computer Science, 2679*, 483-505.

Blackhurst, J., Wu, T., & O'Grady, P. (2005). PCDM: a decision support modeling methodology for supply chain, product and process design decisions. *Journal of Operations Management, 23,* 325-43.

Blackhurst, J., Wu, T., & O'Grady, P. (2007). A network-based decision tool to model uncertainty in supply chain operations. *Production Planning & Control, 18*(6)*,* 526-35.

Bowersox, D.J. & Closs, D.J. (1989). Simulation in Logistics: A review of present practice and a look to the future. *Journal of Business Logistics, 10(1),* 133-48.

BusinessDictionary (2011). *Operations Management - Definition*. Retrieved March 23, 2011, from http://www.businessdictionary.com/definition/operations-management.html

Cargille, B. & Bliss, R. (2001). How supply chain analysis enhances product design. *Supply Chain Management Review, 5(5),* 64-74.

Cargille, B. & Fry, C. (2006). Design for supply chain: Spreading the word across HP. . *Supply Chain Management Review, 10(5),* 34-41.

Chiu, M., Gupta, S., & Okudan, G. (2009). Integration of supply chain decisions at the conceptual design stage: A repository enabled decision tool. In *Proceedings of the ASME 2009 International Design Engineering Technical Conferences (IDETC 2009)* (pp. 351-59). San Diego, CA : ASME.

Christopher, M. & Towill, D. (2002) Developing Market Specific Supply Chain Strategies. *International Journal of Logistics Management, 13*(1), 1-14.

Christopher, M., Peck, H., & Towill, D. (2006). A Taxonomy for Selecting Global Supply Chain Strategies. *International Journal of Logistics Management, 17*(3), 277-87.

Cohen, M. & Lee, H. (1989). Resource deployment analysis of global manufacturing and distribution networks. *Journal of Manufacturing Operations Management, 2*, 81-104.

CSCMP (2011). *Logistics Management - Definition*. Retrieved March 23, 2011, from http://cscmp.org/aboutcscmp/definitions.asp

Desel, J. & Erwin, T. (2000). Modeling, simulation and analysis of business processes. *Lecture Notes in Computer Science, 1806,* 129-41.

Dotoli, M., Fanti, M., Meloni, C., & Zhou, M. (2005). A multi-level approach for network design of integrated supply chains. *Integrated Journal of Production Research, 43*(20)*,* 4267-87.

Eilon, S. & Watson-Gandy, C. (1970). Models for determining depot location. *International Journal of Physical Distribution, 1,* 5-15.

Eisenhardt, K. (1989). Building theories from case study research. *Academy of Management Review, 14*(4)*,* 532-50.

El Hadj Khalaf, R., Agard, B., & Penz, B. (2009). Product and supply chain design using a taboo search. In *Proceedings International Conference on Industrial Engineering and Systems Management (IESM 2009).*

Eom, H. & Lee, S. (1990a). A survey of decision support system applications (1971 – April 1988). *Interfaces, 20*(3), 65-79.

Eom, H. & Lee, S. (1990b). Decision support system applications research: A bibliography (1971-1988). *European Journal of Operations Research, 46*, 333-42.

Eom, S., Lee, S., Kim, E., & Somarajan, C. (1998). A survey of decision support system applications (1988-1994). *Journal of the Operational Research Society, 49*, 109-20.

Eppen, G.D. (1979). Effects of centralization on expected costs in a multi-location newsboy problem. *Management Science, 25*(5), 498-501.

Ericsson, A. & Erixon, G. (2000). *Controlling Design Variants – The power of modularity*. New York: ASME Press.

ExSpect (1999). *ExSpect 6.4 User Manual*. Retrieved January 21, 2008, from www.exspect.com

FICO (2011). *Xpress Optimization Suite product pages*. Retrieved February 2, 2011, from http://www.fico.com/en/Products/DMTools/Pages/FICO-Xpress-Optimization-Suite.aspx

Fine, C.H. (2000). Clockspeed-based strategies for supply chain design. *Production and Operations Management*, *9*(3), 213-21.

Fine, C.H., Vardan, R., Pethick, R., & El-Hout, J. (2002). Rapid-Response Capability in Value-Chain Design. *MIT Sloan Management Review*, *Winter 2002*, 69-75.

Fine, C., Golany, B., & Naseraldin, H. (2005). Modeling tradeoffs in three-dimensional concurrent engineering: a goal programming approach. *Journal of Operations Management*, *23*, 389-403.

Fisher, M. (1997). What is the right supply chain for your product?, *Harvard Business Review*, *75*(2), 105-16.

Forrester, J. (1961). *Industrial Dynamics*. Cambridge, MA: MIT Press.

Geoffrion, A. & Graves, G. (1974). Multicommodity distribution system design by Benders decomposition. *Management Science*, *20*(5), 822-44.

Geoffrion, A., Graves, G., & Lee, L. (1978). Strategic Distribution Systems Planning: A Status Report. In Hax, A. (*Ed.*), *Studies in Operations Management* (pp. 179-204). New York : American Elsevier.

Giunipero, L., Hooker, R., Joseph-Matthews, S., Yoon, T., & Brudvig, S. (2008). A decade of scm literature: past, present and future implications. *Journal of Supply Chain Management*, *44*(4), 66-86.

Geoffrion, A. & Powers, M. (1995). Twenty Years of Strategic Distribution System Design: An Evolutionary Perspective. *Interfaces*, *25*(5), 105-27.

Goetschalckx, M., Vidal, C., & Dogan, K. (2002). Modeling and design of global logistics systems: A review of integrated strategic and tactical models and design algorithms. *European Journal of Operational Research, 143,* 1-18.

Goetschalckx, M. & Fleischmann, B. (2008). Strategic network design, in Stadtler, H. & Kilger, C. (eds.) *Supply chain management and advanced planning – Concepts, models, software, and case studies* (pp. 118-32). Berlin, Germany: Springer.

Guedes, A.P. (1994). An integrated approach to logistics strategy planning using visual interactive modelling and decision support. [PhD Thesis]. Cranfield University.

Guedes, A.P., Saw, R.J.,& Waller, A.G. (1995). Logistics strategy planning: Modeling and decision support techniques for the 1990s. *The International Journal of Logistics Management, 6*(1)*,* 37-50.

Gunasekaran, A., Patel, C., & Tirtiroglu, E. (2001). Performance measures and metrics in a supply chain environment. *International Journal of Operations & Production Management, 21*(1/2)*,* 71-87.

Gunasekaran, A., Patel, C., & McGaughey, R. (2004). A framework for supply chain performance measurement. *International Journal of Production Economics, 87,* 333-47.

Gunasekaran, A. & Kobu, B. (2007). Performance measures and metrics in logistics and supply chain management: A review of recent literature (1995-2004) for research and applications. *International Journal of Production Research, 45*(12)*,* 2819-40.

Helo. P. (2004). Managing agility and productivity in the electronics industry. *Industrial Management & Data Systems, 104*(7)*,* 567-77.

Hoover, B., Eloranta, E., Holmström, J., & Huttunen, K. (2001). *Managing the Demand-Supply Chain: Value Innovations for Customer Satisfaction.* New York, NY : Wiley.

House, J. & Karrenbauer, J.J. (1978). Logistics System Modeling. *International Journal of Physical Distribution and Materials Management*, *8*, 189-99.

Huang, G. (1996). *Design for X: Concurrent Engineering Imperatives.* London, UK : Chapman & Hall.

I2 Technologies (2009). *Supply Chain Strategist version 6.3.2 Getting Started Guide*.

IBM (2011). *CPLEX product introduction*. Retrieved February 2, 2011, from http://www-01.ibm.com/software/integration/optimization/cplex-optimizer/

Ingalls, R. & Kasales, G. (1999). CSCAT: the Compaq supply chain analysis tool. In Farrington, P., Nembhard Black, H., Sturrock, D., & Evans, G. (Eds.), *Proceedings of the 1999 Winter Simulation Conference* (pp. 1201-06). New York : ACM.

Ingalls, R. (2008). Introduction to simulation. In Mason, S., Hill, R., Mönch, L., Rose, O., Jefferson, T., & Fowler, J. (Eds.), *Proceedings of the 2008 Winter Simulation Conference* (pp. 17-26). New York : ACM.

Ivanov, D., Sokolov, B., & Kaeschel, J. (2010). A multi-structural framework for adaptive supply chain planning and operations control with structure dynamics considerations. *European Journal of Operational Research, 200*, 409-20.

Jensen, K. (1996). Coloured Petri Nets: Basic Concepts, Analysis Methods and Practical Use. 2nd ed. Vol. 1. *EATCS Monographs on Theoretical Computer Science.* London, UK: Springer.

Jiao, J., Simpson, T., & Siddique, Z. (2007). Product family design and platform-based product development: a state-of-the-art review. *Journal of Intelligent Manufacturing, 18*, 5-29.

Jonsson, P., Kjellsdotter, L., & Rudberg, M. (2007). Applying advanced planning systems for supply chain planning: Three case studies. *International Journal of Physical Distribution & Logistics, 37*(10), 816-34.

Kasanen, E., Lukka, K., & Siitonen, A. (1991). The Constructive Research Approach in Economics (in Finnish). *The Finnish Journal of Business Economics*, *40*, 301-329.

Kasanen, E., Lukka, K., & Siitonen, A. (1993). The constructive approach in management accounting research. *Journal of Management Accounting Research, 5*(3)*,* 243-64.

Kaufman, L., Eede, M.V., & Hansen, P. (1977). A Plant and Warehouse Location Problem. *Operational Research Quarterly*, *28*(3), 547-54.

Khan, O. & Creazza, A. (2009). Managing the product design-supply chain interface – Towards a roadmap to the "design centric business". *International Journal of Physical Distribution & Logistics Management, 39*(4), 301-19.

Kindler, E. (2004) High-level Petri Nets – Transfer Format. *Working Draft of the International Standard ISO/IEC 15909 Part 2 – Version 0.5.0.* Retrieved January 30, 2008, from http://www2.informatik.hu-berlin.de/top/pnml/about.html

Kleijnen, J. (1995). Verification and validation of simulation models. *European Journal of Operational Research, 82*, 145-62.

Kleijnen, J., Bettonvil, B., & Persson, F. (2004). Finding the important factors in large discrete-event simulation: sequential bifurcation and its applications. In Dean, A. & Lewis, S. (Eds.), *Screening.* New York: Springer-Verlag.

Kleijnen, J. (2005). Supply chain simulation tools and techniques: a survey. *International Journal of Simulation & Process Modelling, 1*(1/2), 82-89.

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

Labro, E. & Tuomela, T-S. (2003). On bringing more action into management accounting research: process considerations based on two constructive case studies. *European Accounting Review,* 12(3), 409-42.

Lamothe, J., Hadj-Hamou, K., & Aldanondo, M. (2006). An optimization model for selecting a product family and designing its supply chain. *European Journal of Operational Research, 169*, 1030-47.

Lanning, H. (2001). Planning and implementing change in organisations – a construct for managing change projects. [PhD Thesis]. Helsinki University of Technology.

Lee, C.Y. (1993). The Multiproduct Warehouse Location Problem: Applying a Decomposition Algorithm. *International Journal of Physical Distribution and Logistics Management, 23*(6), 3-13.

Lee, H. & Billington, C. (1992). Managing supply chain inventory: Pitfalls and opportunities. *Sloan Management Review, 33*(3), 65-73.

Lee, Y. & Sook, H. (2002). Production-distribution planning in supply chain considering capacity constraints. *Computers & Industrial Engineering, 43*, 169-90.

Lee, H. & Sasser, M. (1995). Product universality and design for supply chain management. *Production Planning & Control*, 6(3), 270-77.

Lehtonen, J.-M. (1999). Supply chain development in process industry. [PhD Thesis]. Helsinki University of Technology.

Leung, Y.-T., Suri, R. (1990). Performance evaluation of discrete manufacturing systems. *Control Systems Magazine: IEEE, 10(4),* 77-86.

Lewis, H.T., Culliton, J.W., & Steele, J.D. (1956). *The Role of Air Freight in Physical Distribution,* Boston : Harvard University.

Lin, P.-H., Wong, D., Jang, S.-S., Shieh, S.-S., & Chu J.-Z. (2004). Controller design and reduction of bullwhip for a model supply chain system using z-transform analysis. *Journal of Process Control, 14*, 487-99.

Lindo Systems (2011). *Lingo product introduction*. Retrieved February 2, 2011, from http://www.lindo.com/index.php?view=article&catid=15%3Aarchives-home&id=86%3Aapplication-archives&option=com_content&Itemid=34

Llamasoft (2007). *Supply Chain Guru User Guide version 3.0*. Llamasoft, Inc.

Lukka, K. (2000). The key issues of applying the constructive approach to field research, in Reponen, T. (ed.) *Management Expertise for the New Millenium: In Commemoration of the 50th Anniversary of Turku School of Economics and Business Administration* (pp. 113-28). Turku, Finland : Turku School of Economics.

Maister, D.H. (1976). Centralization of inventories and the 'square root law'. *International Journal of Physical Distribution and Logistics Management, 6*(3), 124-34.

March, S. & Hevner, A. (2007). Integrated decision support systems; A data warehousing perspective. *Decision Support Systems, 43*, 1031-43.

Martin, M., Hausman, W., & Ishii, K. (1998). Design for variety. In Ho, T..-H. & Tang, C. (Eds.) *Product Variety Management, Research Advances*. Norwell, MA. : Kluwer Academic Publishers.

Melo, M., Nickel, S., & Saldanha-da-Gama, F. (2009). Facility location and supply chain management – A review. *European Journal of Operational Research, 196*, 401-12.

Mentzer, J.T. & Kahn, K.B. (1995). A framework of logistics research. *Journal of Business Logistics, 16*(1), 231-50.

Mentzer, J., DeWitt, W., Keebler, K., Min, S., Nix, N., & Smith C. (2001). Defining supply chain management. *Journal of Business Logistics, 22*(2), 1-25.

Nagel, R. & Dove, R. (1991). *21st Century Manufacturing Enterprise Strategy*. Betlehem, PA : Lehigh University.

Naylor, J.B., Naim, M.M., & Berry, D. (1999). Leagility: Interfacing the Lean and Agile Manufacturing Paradigm in the Total Supply Chain, *International Journal of Production Economics*, *62*, 107-18.

Norrmann, A., & Jansson, U. (2004). Ericsson's proactive supply chain risk management approach after a serious sub-supplier accident. *International Journal of Physical Distribution & Logistics Management, 34*(5), 434-56.

Olhager, J. (2003). Strategic positioning of the order penetration point. *International Journal of Production Economics, 85*, 319-29.

Padberg, M. (1999). *Linear Optimization and Extensions* (2^{nd} ed.). Berlin: Springer-Verlag.

Peidro, D., Mula, J., Poler, R., & Lario, F-C (2009). Quantitative models for supply chain planning under uncertainty: a review. *International Journal of Advanced Manufacturing Technologies, 43,* 400-20.

Powers, R.F. (1989). Optimization Models for Logistics Decisions. *Journal of Business Logistics, 10(1),* 106-21.

Prasad, B. (1996). *Concurrent Engineering Fundamentals: Integrated Product and Process Organization* (Vol 1). Prentice-Hall.

Riddalls, C., Bennett, S., & Tipi, N. (2000). Modelling the dynamics of supply chains. *International Journal of Systems Science*, *31*(8), 969-76.

Russell, N. & ter Hofstede, A. (2009). Surmounting BPM challenges: the YAWL story. *Computer Science – Research and Development*, *23*(2), 67-79.

Salimifard, K. & Wright, M. (2001) Petri net-based modelling of workflow systems: An overview. *European Journal of Operations Research, 134,* 664-676.

Santoso, T., Ahmed, S., Goetschalckx, M., & Shapiro, A. (2005). A stochastic programming approach for supply chain network design under uncertainty. *European Journal of Operational Research, 167,* 96-115.

Selldin, E. & Olhager, J. (2007). Linking products with supply chains: testing Fisher's model. *Supply Chain Management: An International Journal, 12*(1), 42-51.

Simchi-Levi, D., Kaminsky, P., & Simchi-Levi, E. (2003). *Designing & Managing the Supply Chain – Concepts, Strategies, & Case Studies* (2^{nd} ed.). New York, NY: McGraw-Hill.

Simchi-Levi, D., Kaminsky, P., & Simchi-Levi, E. (2004). *Managing the supply chain: the definitive guide for the business professional.* New York, NY: McGraw-Hill.

Stake, R. (1995). *The Art of Case Study Research*. San Francisco: Sage Publications.

Spencer, T., Brigandi, A., Dargon, D., & Sheehan, M. (1990). AT&T's telemarketing site selection system offers customer support. *Interfaces*, 20(1), 83-96.

Suri, R., Diehl, G.W.W., de Treville, S., & Tomsicek, M.J. (1995). From CAN-Q to MPX: Evolution of Queueing Software for Manufacturing. *Interfaces, 25(5),* 128-150.

Susman, G.I. & Evered, R.D. (1978). The Assessment of the Scientific Merits of Action Research. *Administrative Science Quarterly, 23*, 582-602.

de Treville, S, & van Ackere, A. (2006). Equipping students to reduce lead times: The role of queueing-theory-based modeling. *Interfaces, 36*(2), 165-73.

Truong, T.H. & Azadivar, F. (2003). Simulation-Based Optimization for Supply Chain Configuration Design. In Chick, S., Sanchez, P.J., Ferrin, D., & Morrice, D.J. (Eds.), *Proceedings of the 2003 Winter Simulation Conference* (pp. 1268-75). New York : ACM.

Tynjälä, T. (2005). A formal, product structure driven design of optimized end-to-end demand supply chains. *Journal of Systemics, Cybernetics & Informatics, 3(1),* 7-11. Available online at: www.iiisci.org/Journal/SCI/Past.asp

Tynjälä, T. & Eloranta, E. (2007). Investigating the effect of product variants, and demand distributions on the optimal demand supply network setup. *Production Planning & Control, 18*(7)*,* 561-72.

Tynjälä, T (2009). Supporting demand supply network optimization with petri nets. In Sugumaran, V. (Ed.) *Distributed Artificial Intelligenve, Agent Technology, and Collaborative Applications* (pp. 366-83). Hershey, PA: IGI Global.

Tynjälä, T. (2011). Validation in Supply Chain Decision Support Systems. Accepted for publication in *International Journal of Information Systems and Supply Chain Management (IJISSCM).*

Vamanan, M., Wang, Q., Batta, R., & Szczerba, R. (2004). Integration of COTS software products ARENA & CPLEX for an inventory/logistics problem. *Computers & Operations Research, 31*, 533-47.

Van Der Aalst, W.M.P. (1992). Timed coloured Petri nets and their application to logistics. [PhD Thesis]. University of Eindhoven.

Van der Aalst, W.M.P. (1998). The application of Petri nets to workflow management. *Journal of Circuits, Systems and Computation, 8(1),* 21-66.

Van der Aalst, W.M.P. (2000a). Loosely coupled interorganizational workflows: modeling and analyzing workflows crossing organizational boundaries. *Information & Management, 37,* 67-75.

Van der Aalst, W.M.P. (2000b). Workflow verification: Finding control-flow errors using Petri-net based techniques. *Lecture Notes in Computer Science, 1806,* 161-83.

Van der Aalst, W.M.P., de Crom, P.J.N., Goverde, R.R.H., van Hee, K.M., Hofman, W.J., Reijers, H.A., & van der Toorn, R.A. (2000). ExSpect 6.4: An executable specification tool for hierarchical colored Petri nets. *Lecture Notes in Computer Science, 1825,* 455-464.

Van der Aalst, W.M.P., & ter Hofstede, A.H.M. (2005). YAWL: Yet another workflow language. *Information Systems, 30*(4)*,* 245-275.

Van Roy, T.J. (1989). Multi-level Production and Distribution Planning with Transportation Fleet Optimization. *Management Science, 35*(12), 1443-53.

Vidal, C. & Goetschalckx, M. (1997). Strategic production-distribution models: A critical review with emphasis on global supply chain models. *European Journal of Operational Research, 98*, 1-18.

Vidal, C. & Goetschalckx, M. (2001). A global supply chain model with transfer pricing and transportation cost allocation. *European Journal of Operational Research, 129*, 134-18.

Wang, G., Huang, S.H., Dismukes, J.P. (2004). Product-driven supply chain selection using integrated multi-criteria decision-making methodology. *International Journal of Production Economics*, *91,* 1-15.

Weber, A. (translation by Friedrich, C.J., 1929) (1909). *Theory of Location of Industries*. Chicago : The University of Chicago Press.

Womack, J., Jones, D., & Roos, D. (1990). *The Machine that Changed the World.* New York : Macmillan.

Womack, J.P. & Jones, D.T. (1996). *Lean Thinking*, New York : Simon and Schuster, 1996.

Wu, T. & O'Grady, P. (1999). A concurrent engineering approach to design for assembly. *Concurrent Engineering: Research and Applications, 7*(3), 231-43.

Wu, T., Blackhurst, J., & O'Grady, P. (2007). Methodology for supply chain disruption analysis. *International Journal of Production Research, 45*(7), 1665-82.

Yin, R. (2003a). *Case Study Research: Design and Methods* (3rd ed.). Thousand Oaks, CA: SAGE Publications.

Yin, R. (2003b). *Applications of Case Study Research* (2nd ed.). Thousand Oaks, CA: SAGE Publications.

Zhang, X., Xiao, Y., Jiao, J., & Helo, P. (2006). Modeling supply chain configuration based on colored petri nets. *Control, Automation, Robotics and Vision, 2006. ICARCV'06. 9th International Conference on* (pp. 1-6). Piscataway, NJ : IEEE.

Zhang, X., Lu, Q., & Wu, T. (2009). Petri-net based application for supply chain management: An overview. *Industrial Engineering and Engineering Management, 2009. IEEM 2009. IEEE International conference on* (pp. 1406-10). Piscataway, NJ : IEEE.

Zinn, W., Levy, M., & Bowersox, D.J. (1989). Measuring the effect of inventory centralization/decentralization on aggregate safety stock: 'the square root law' revisited. *Journal of Business Logistics, 10*(1), 1-13.

Aalto-DD 73/2011

ISBN 978-952-60-4255-8 (pdf)
ISBN 978-952-60-4254-1
ISSN-L 1799-4934
ISSN 1799-4942 (pdf)
ISSN 1799-4934

Aalto University
School of Science
Department of Industrial Engineering and Management
www.aalto.fi

BUSINESS +
ECONOMY

ART +
DESIGN +
ARCHITECTURE

SCIENCE +
TECHNOLOGY

CROSSOVER

DOCTORAL DISSERTATIONS