ESSAYS ON THE INTERFACE OF SUPPLY CHAIN AND PROJECT MANAGEMENT

BY XIN XU

A dissertation submitted to the

Graduate School—Newark

Rutgers, The State University of New Jersey

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

Graduate Program in Supply Chain Management

Written under the direction of

Yao Zhao

and approved by

Newark, New Jersey

October, 2014

ABSTRACT OF THE DISSERTATION

Essays on the Interface of Supply Chain and Project Management

by Xin Xu

Dissertation Director: Yao Zhao

This thesis focuses on the interface of project and supply Chain Management. Supply chain decisions (e.g., material planning, network design, supply management) and project management decisions (e.g., resource planning, expediting, and project scheduling) are intertwined in many firms. The objective of this thesis is to construct and analyze new models and methods that can help firms integrate supply chain and project management. Specifically, we addressed the following issues: (1) Joint optimization of inventory and project planning decisions for recurrent projects subject to random material delays (Chapter 2), often found in construction industries. (2) Designing and managing the development chain for one-of-a-kind R&D projects with an extensive workload outsourced (Chapters 3-4), representing the recent trend in the aerospace and defense industries.

In Chapter 2, we study a new class of problems – recurrent projects with random

material delays, at the interface between project and supply chain management. Recurrent projects are those similar in schedule and material requirements. We present the model of project-driven supply chain (PDSC) to jointly optimize the safety-stock decisions in material supply chains and the crashing decisions in projects. We prove certain convexity properties which allow us to characterize the optimal crashing policy. We study the interaction between supply chain inventory decisions and project crashing decisions, and demonstrate the impact of the PDSC model using examples based on real-world practice.

In Chapter 3, we study incentive and coordination issues in development chains. Collaboration and partnership are the way of life for large complex projects in many industries. While they offer irresistible benefits in market expansion, technological innovation, and cost reduction, they also present a significant challenge in incentives and coordination of the project supply chains. In this chapter, we study strategic behaviors of firms under the popular loss-sharing partnership in joint projects by a novel model that applies the economic theory of teamwork to project management specifics. We provide insights into the impact of collaboration on the project performance. For a general project network with both parallel and sequential tasks where each firm faces a time-cost trade-off, we find an inherent conflict of interests between individual firms and the project. Depending on the cost and network structure, we made a few surprising discoveries, such as, the Prisoners' Dilemma, the Supplier's Dilemma, and the Coauthors' Dilemma; these dilemmas reveal scenarios in which individual firms are motivated to take actions against the best interests of the project and exactly how collaboration can hurt. As remedy, we enhance collaboration by a set of new provisions into a "fair sharing" partnership and prove its effectiveness in aligning individual firms' interests with that of the project.

In Chapter 4, we extend the model in Chapter 3 in two directions. First, we extend the discrete-time model to a continuous-time model and show that the Coauthor's Dilemma still holds and thus the project will never be finished earlier under the loss-sharing partnership than the centralized control system. Second, we consider stochastic task durations and find that the uncertainty increases the probability of project delay.

Acknowledgements

My first and foremost appreciation goes to my advisor, Professor Yao Zhao, under whose guidance this thesis was written, for his invaluable research advice, encouragement and support during my entire graduate study at Rutgers University. His rigorous and passionate attitude to research has truly inspired me. He patiently guided me through my entire Ph.D. study and also left me enough freedom to explore on my own. At the early stage of my graduate study, he encouraged me to attend academic conferences, introduced me to other colleagues and helped me expand my academic network. He provided comments and suggestions not only on research but also on many other things like career path and development, tips to improve work efficiency, and so on. I also would like to thank him particularly for the help he offered in my job search.

I am deeply grateful to my dissertation committee members for their insightful comments and suggestions on my dissertation. Professor Xiaowei Xu provided valuable comments on the game theoretical model. Professor Michael Katehakis offered two excellent courses and provided journal review opportunity for me. Professor Arash Azadegan provided insightful suggestions on my career development and opened the door of empirical research for me which greatly inspired my research. Finally, I would like to thank Professor Wenqiang Xiao from NYU for his time and effort to serve on my committee.

I also would like to express my gratitude to all the faculty members and staffs in

Supply Chain Management Department. In particular, I would like to thank Professor Lei Lei for her broad help throughout my years at Rutgers. I would like to thank Professor Weiwei Chen for his comments and suggestions on my career. Finally, I would like to thank Professor Tae-Kyun Kim for his recommendations on time management skills.

I am also grateful to many friends and fellow students who have helped me during these years at Rutgers. In particular, I would like to thank Anh Ninh and Hui Dong for provocative research discussions and presentation practices. I would like to thank Andy Johnson for his support and help in the Special Olympics project. I would also like to thank all my friends.

Finally, my thanks go to my family: my wife, my parents and my 7-month old boy. This thesis would not have been possible without the support and the love from my wife and my parents.

Dedication

To My Wife, My Son and My Parents

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Over the last three to four decades, advances in technology and the networked economy have led to the evolution of the business models in many project driven industries, from the one-company-does-all approach to a more collaborative and decentralized one on a global basis. While this has brought tremendous benefits in the areas of market expansion, technological innovations and cost reduction, it also led to significant challenges in the coordination and management of the project (driven) supply chains. Indeed, with the outsourced work accounting for 50% or more of the revenue for projects in these industries, supply chain management has never been so important.

The problems studied in this thesis share a common feature: project management and supply chain management decisions are intertwined. In practice, they are typically managed as projects without taking the supply chain perspective into account. In the literature, the connections and interactions between the projects and their supply chains are yet fully recognized and understood. We studied these problems by an interdisciplinary approach, which combines mathematical modeling with real-life examples and if possible data. Our objective is to develop new insights and solutions beyond conventional wisdom, and reconcile the discovery with practice to demonstrate its economic or social impact.

This thesis covers two research areas:

1. Recurrent projects subject to random material delays (found in the construction industry)

2. One-of-a-kind development projects with an extensive workload outsourced (found in the aerospace and defense industry).

1.1 Recurrent Projects With Random Material Delivery

Our research on recurrent projects was motivated by recent issues in the construction resource management. Today, construction projects frequently spend a significant portion of their budgets (more than 50 percent) on materials sourced from an extended supply network. However, projects and their material supply chains are often managed in separation despite the fact that they are tightly coupled, for instance, lean supply practice often leads to long and variable material lead-times which may ruin the project schedule and result in expediting cost and/or delay penalties.

We have been working alongside Intercontinental Construction Management (ICM) Inc; a middle sized construction management firm which specializes in military buildings. Structural steel is the most expensive material used in all ICM projects, which is subject to a long and unpredictable lead time. When the supplies are delayed behind the schedule, ICM has to struggle to expedite the rest of the project activities because of its project management practice, which involves treating each project as a separate and unique entity. However, material supply is not an issue specific to a project but an ongoing concern, as it is required by all the projects. To resolve issues of this kind, we identified a new class of problem - recurrent projects with random material delays. We constructed a modeling framework to plan for supply chain and project operations jointly so that we could achieve an overall efficiency. The key idea is to plan material supplies not only for confirmed projects (project-based management) but also for potential projects yet confirmed (supply-based management).

Specifically, Chapter 2 provides a modeling framework, namely, the project-driven supply chain (PDSC) model, to jointly optimize supply chain safety-stock and project crashing decisions for recurrent projects with random material delays. In a nutshell, PDSC model is a multi-stage mathematical model where one makes supply chain inventory decisions in the first stage, and then makes crashing decisions dynamically as material delays are realized in subsequent stages. The objective is to minimize the total safety-stock and project cost per unit of time. We prove certain convexity properties of the cost function and characterize the optimal crashing policy for each project. We also study the interaction between supply chain inventory decisions and project crashing decisions.

Applying the model back to ICM, we show that a certain amount of planned inventory, if placed in the right locations within the supply chain, could reduce and stabilize the schedule of the projects and greatly improve the company's overall performance. Finally, we conduct an extensive numerical study to generate insights on when PDSC model may provide significant savings.

1.2 One-of-a-kind Projects: Incentives and Coordination

Collaboration is the way of life in today's new product development projects in the aerospace and defense industries. For instance, product development programs (e.g., Boeing 787, Airbus 380, and China 919) outsourced a significant amount of workload in design and/or fabrication to a global network of suppliers. While outsourcing offers irresistible benefits in market expansion, utilization of the best-in-class technologies, and cost/cycle-time reduction, it poses a significant challenge in managing project supply chains, which are increasingly more complex, more risky and involve more organizations with diverse strategic motivations.

As witnessed by the repetitive delays of the Dreamliner and other programs in commercial aviation, supply chain management becomes a center of gravity for managing large development programs of this kind. Inspired by these events, the objective of this research stream is to understand how to manage the development chain, more specifically, how to coordinate multiple companies' efforts in a joint development project. These companies may develop different subsystems in parallel or develop subsystem and integrate systems sequentially depending on the project network structure.

In Chapter 3, we applied the economic theory of teamwork to project operations to study strategic behaviors of firms under various partnerships (incentive schemes) for joint development projects where tasks are performed by different companies and each company's objective is to maximize its own benefit but not that of the project. We provide insights into the impact of these partnerships on project performance. In particular, we found that collaboration can hurt! Depending the project network and cost structure, we discovered many innovative results on exactly how collaboration hurts and the remedies, such as the Prisoners' Dilemma in a Project Management setting, the Supplier's Dilemma, and the Coauthors' Dilemma. The study opens up a new stream of research at the interface of project and supply chain management and produces many interesting and novel results on how to design and manage partnerships in projects jointly developed by multiple companies.

In this Chapter 4, we extend the discrete time deterministic model of Chapter 3 to a continuous-time model and to include stochastic task durations, where we show the Coauthor's Dilemma holds in the continuous-time model, and uncertainty in task durations increases the probability of project delay.

Both supply chain management and project management have an extensive literature. However, they rarely interact. This thesis build new models to capture the interaction between the supply chain and project operations, and develop methods for their joint optimization. On one hand, it integrates recent advances in stochastic multi-echelon inventory theory with project management (for the recurrent projects in construction industries). On the other hand, it extends the project management literature to include supply chain planning and incentive alignment (for the one-of-a-kind projects in aerospace and defense industries).

Chapter 2

Project-Driven Supply Chains: Integrating Safety-Stock and Crashing Decisions for Recurrent Projects

2.1 Introduction

Today, complex projects frequently spend a significant portion of their budgets on material supplies sourced from an extensive supply network. The success of these projects depends critically on both the project and the supply chain operations. Examples can be found in construction and airspace/defense industries in particular, and in engineering-procurement-construct (EPC) industries in general.

While some of these projects are unique and one of a kind, many of them are routine and recurrent, i.e., they share similar schedule and material supplies. Indeed, as companies standardize their processes and components to streamline their operations, recurrent projects are becoming increasingly popular in practice. In these cases, the project and supply chain operations are intertwined because the demand for the supply chain is driven by projects' material requirement and project progresses are constrained by random material delays. Despite the extensive literature of project and supply chain management, project operations and supply chain operations are rarely studied jointly in academia, and they are almost always managed separately in industry. This chapter explores ways to improve the efficiency of recurrent projects by integrating project and supply chain operations.

Specifically, we study a new class of problems - recurrent projects with random material delays where the future projects' occurrence and material requirement cannot be fully predicted in advance. The project and supply chain decisions are naturally coupled because under-stocked supply chains lead to excessive material delays which may increase project crashing (i.e., expediting) and delay costs. To track down these problems, we develop a modeling framework to integrate project crashing and supply chain safety-stock decisions. Our objective is to validate a new approach - carrying safety stock for materials used in recurrent projects, in performance improvement for projects and their material supply chains.

One motivating example is the Intercontinental Construction Management (ICM) Inc., a construction management firm specialized in military buildings (the name is disguised to protect proprietary information). Although each construction project has a unique goal, they are recurrent - similar in schedule and material requirement. Structural steel is the most expensive material used in all projects, which is subject to a long and random lead time. Because projects are awarded through a bidding process, ICM cannot fully anticipate the occurrence of future projects and their material requirement, and thus cannot order materials before projects are awarded. Consequently, structural steel may be delayed behind schedule, in which case, ICM has to crash (i.e., expedite) the remaining tasks as needed to meet the due date.

ICM has never looked at the safety-stock of structural steel as a part of the solution because it assigns each project to a project manager who manages the project as a separate and unique entity. ICM's practice does not stand alone - project and supply chain are rarely managed jointly in practice because of the distinct approaches to manage them. Projects are often managed on a one-for-one basis while supply chains are often managed continuously across demand from multiple projects. Such a one-for-one (or project-based) approach fails to take advantage of the similarity across recurrent projects.

Recurrent projects are widely seen in many industries, e.g., construction and build-to-order manufacturing, where standard components are combined in different ways to build complex and customized products (e.g., houses, ships, aircrafts). In addition to ICM, other examples of recurrent projects can be found in Walsh et al. (2004), Brown et al. (2004) and Elfving et al. (2010). Specifically, Walsh et al. (2004) presents a case study of a food company that is frequently engaged in projects of expanding an existing or adding a new facility. The key concern is on the critical material of stainless steel components that are used in all projects and are subject to the longest and most variable lead time. Brown et al. (2004) presents the case study of Quadrant Homes Inc. which follows a standard schedule in construction of residential houses. Elfving et al. (2010) conducts an empirical study on 180 projects by one Finnish construction company in five years and shows that many projects share make-to-stock standardized materials. Finally, Schmitt and Faaland (2004) shows that in addition to houses, projects of constructing airplanes and ships can be recurrent.

The industry reports of Kerwin (2005) and Xu and Zhao (2010) further confirm the popularity of recurrent projects in the construction industry. Kerwin (2005) shows that home-builders like Pulte Homes Inc. offer only the most popular floor plans to boost efficiency in fulfilling tens of thousand of new home orders in a year. Xu and Zhao (2010) surveys an important trend in construction industry – prefabricated housing, where houses have limited variety and are assembled by prefabricated materials. All these cases are characterized by repeating projects with limited variety and their supply

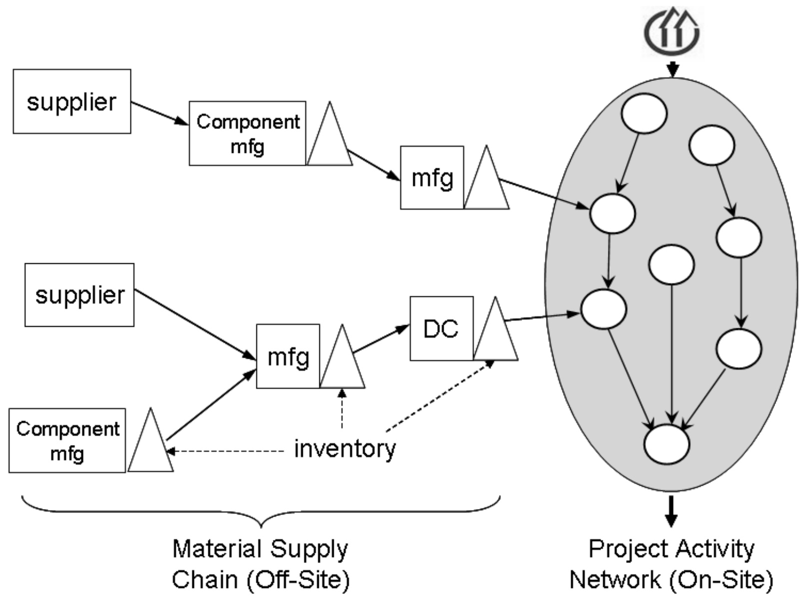


Figure 2.1: Recurrent projects with standardized material requirements.

chains of standardized materials (see Figure 2.1 for an illustration).

One key issue of the recurrent projects is the coordination of material delivery and project progress which is critical for on-time and on-budget completion of the projects. This is especially true when materials account for a significant portion of the budget (e.g., 65% for residential houses – Somerville (1999)), and when material availability and delivery lead times are random, as illustrated by the examples of ICM and Walsh et al. (2004), and confirmed by an investigation of time waste in construction (Yeo and Ning 2002) which reveals that the site work-force spends a considerable amount of time waiting for approval or for materials to arrive on site.

One way to improve the efficiency of recurrent projects is to explore the similarity among these projects and plan for them on a continuing basis rather than for each project separately. Specifically, the supply-based options, such as inventory management and strategic alliance, have been considered to improve material supply management for recurrent projects. We refer the reader to Walsh et al. (2004), Brown et al.

(2004), and Elfving et al. (2010) for case studies, and Tommelein et al. (2003), Tommelein et al. (2009) for conceptual frameworks. These studies focus only on material supplies rather than the integration between project-based and supply-based decisions. However, the supply chain decisions are coupled with the project decisions: if materials are in shortage and the project due date cannot accommodate the delays, project tasks will have to be expedited which can be very costly, and the project is subject to delay. On the other hand, holding inventory to guarantee immediate availability of all materials may not be wise. While one can minimize the project crashing and delay costs, the supply chain inventory cost can be unaffordably high.

In this chapter, we develop an integrated approach to optimize supply chain and project decisions simultaneously. Specifically, we provide a modeling framework, namely, the project-driven supply chain (PDSC) model, to jointly optimize supply chain safety-stock and project crashing decisions for recurrent projects with random material delays. In a nutshell, the PDSC model is a multi-stage mathematical model where we make supply chain inventory decisions in the first stage, and then make crashing decisions dynamically as material delays are realized in subsequent stages. The objective is to minimize the total safety-stock and project cost per unit of time. We prove certain convexity properties of the cost functions and characterize the optimal crashing policy for each project. We study the interaction between supply chain inventory decisions and project crashing decisions. We also apply the PDSC model to a real-world example and demonstrate its potential by comparing to the current practice. Finally, we conduct an extensive numerical study to generate insights on when the PDSC model may provide significant savings.

The chapter is organized as follows: after reviewing the related literature in §2.2, we present the modeling framework in §2.3. Then, we provide an analysis for the optimal crashing policy in §2.4, and study the interaction between inventory and crashing decisions in §2.5. In §2.6, we apply the model to ICM and quantify its impact. In §2.7, we conduct a numerical study to generate managerial insights. Finally, we conclude the chapter in §2.8.

2.2 Literature Review

The project-driven supply chain (PDSC) model in particular and the integration of supply chain and project management in general are related to the literature of project management, supply chain management, and their interfaces such as project scheduling and material ordering (PSMO) and construction supply chain management. We shall review related literature in each area.

Project Management Literature. The project management literature focuses primarily on the planning and execution of a single project, which includes the classic results of critical path method (CPM), time-costing analysis (TCA), project evaluation and review techniques (PERT) and resource constrained project scheduling (RCPS). We refer to Ozdamar and Ulusoy (1995), Pinedo (2005) and Józefowska and Weglarz (2006) for recent surveys. The time-cost analysis (TCA) or crashing analysis is a well developed technique in the project management literature to balance the duration and budget of a project. Most work on RCPS focuses on non-consumable and reusable resources such as machine and labor. For consumable resources (e.g., materials), the standard approach is to assume fixed lead times and then model material procurement processes as tasks.

Supply Chain Management Literature. There is a vast literature of supply chain inventory management mainly grown out of applications in the manufacturing industry. We refer to Zipkin (2000), Porteus (2002) and Axsater (2007) for comprehensive reviews. For inventory placement/positioning models in general structure supply chains, we refer to Axsater (2007), Graves and Willems (2003) and Simchi-Levi and Zhao (2011) for recent reviews. Graves and Willems (2005) presents a general model to optimize stock decisions and supply chain configurations simultaneously for new product introductions. All of these works focus on material supply chains without considering the project decisions and their interactions.

Our work is related to three models of stochastic inventory systems. The first model, see Hadley and Whitin (1963), assumes full backorder and that the system fulfills demand as soon as on-hand inventory becomes available. The second model, see Hariharan and Zipkin (1995), assumes also full backorder but that the system fulfills demand only on or after it is due. Clearly, it is possible to hold inventory and demand (not due) simultaneously in the second model. The third model, see Graves and Willems (2000), assumes guaranteed service-time (unsatisfied demand is filled by extraordinary measures other than on-hand inventory) and that the system fulfills demand only on or after it is due. Graves and Willems (2003) calls the first two models "stochastic service-time" models, and the third one the "guaranteed service-time" model.

In this chapter, we cannot assume the guaranteed service-time model because the random material delay (due to stock-out and/or random processing times) is a necessary part of the problem. We shall use the stochastic service-time models – both the first and the second models for different stages of the material supply chain.

Interface – Project Scheduling and Material Ordering (PSMO). One approach to incorporate consumable resources in project management is the PSMO model which jointly plans for project schedule and material order quantities. This approach is based on the observation that a project may repetitively require the same material over time. Given the project schedule, the timing and size of material requirement are known, which serve as input to optimize material order quantities so as to balance the fixed ordering cost and inventory holding cost. Clearly, the project scheduling and material ordering decisions are coupled, and the question is how to jointly optimize both sets of decisions for a project. Aquilano and Smith (1980) initiates this approach by considering joint CPM and MRP planning with constant task durations. Smith-Daniels and Aquilano (1984) and Smith-Daniels and Smith-Daniels (1987) present various extensions. More recently, Dodin and Elimam (2001) considers varying task duration, early reward/late penalty and quantity discount.

The PDSC model complements the PSMO model by taking uncertainty and the safety-stock issue into account. While the PSMO model focuses on the cycle stock issues for a single project facing economies of scale in ordering, the PDSC model focuses on the safety-stock issues for recurrent projects subject to random material delays. While the PSMO model jointly optimizes project schedule and material order quantities, the PDSC model jointly optimizes project crashing decisions and material safety-stock levels.

Interface – Construction Supply Chain Management. The literature of construction management has traditionally focused on the management of individual projects (Tommelein et al. 2003). Since middle 1990s, the supply chain management concepts and methodologies have been introduced into this field and gained substantial attention. However, supply chain management is still relatively new in the construction industry (OBrien et al. 2002, Tommelein et al. 2003), and most published results focus on qualitative and conceptual frameworks (Vrijhoef and Koskela 2000, Vaidyanathan and Howell 2007) or on case studies (Walsh et al. 2004, Brown et al. 2004). There is a lack of rigorous mathematical modeling that integrates the issues of projects and material supply chains, and resolves them jointly. We refer to OBrien et al. (2002) for a survey on construction supply chain management.

In this literature, Walsh et al. (2004), Brown et al. (2004) and Elfving et al. (2010) are mostly related to our work. In particular, Walsh et al. (2004) use simulation to determine the proper positions of safety-stock in the supply chain of stainless steel components, independently of any specific project, to reduce and stabilize the random lead time. These papers focus on supply chain operations only without considering the project scheduling issues.

2.3 The Modeling Framework

In this section, we present the model of Project-Driven Supply Chain.

2.3.1 Preliminaries

We consider recurrent projects that share the same schedule and the same type of materials which may be subject to random lead times. The starting times of the projects and the amount of each material required are random. The model is depicted in Figure 2.2 where each project consists of a set of tasks that are conducted sequentially. Each task (except the first one) requires a material which is provided by a supply chain. Specifically, there are $n + 1$ tasks and task i ($i > 0$) requires a material which is supplied by supply chain i. Task i can start only after the required material becomes

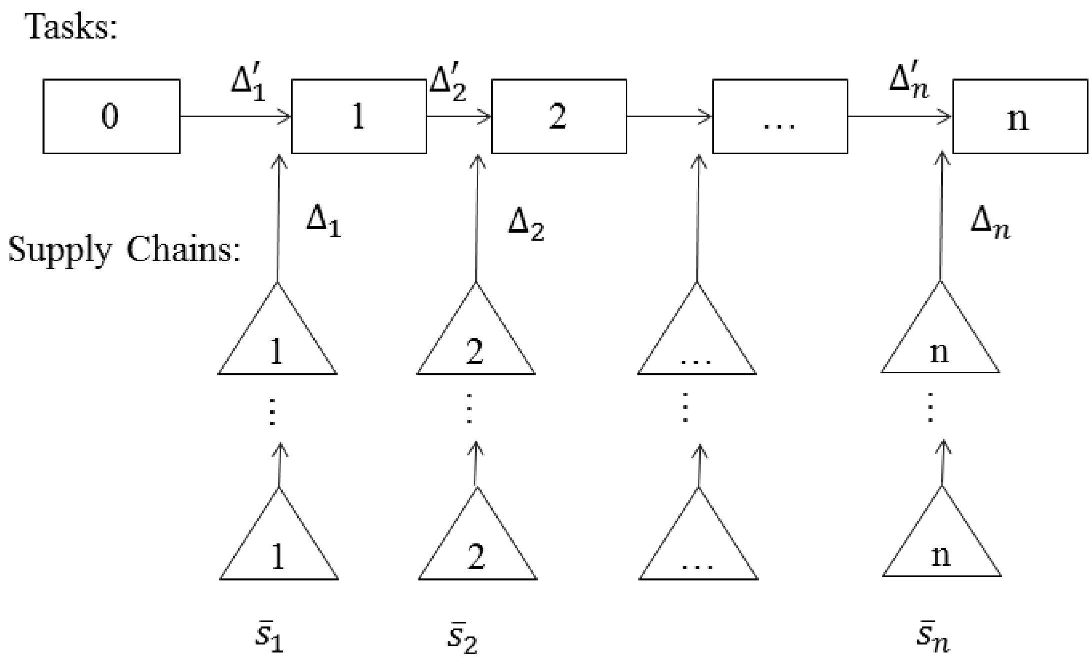


Figure 2.2: The model of project driven supply chain.

available and task $i-1$ completes. Orders for all materials are placed as soon as the project starts. All projects share a standard schedule. We point out that the model remains mathematically identical if task i is replaced by a group of tasks.

Assumption 2.1 *We make the following assumptions on the project and supply chain operations, as well as their interface:*

- **Project operations***: (1) each task has a known duration and a known crashing cost function. (2) The project delay penalty function is known. (3) There is no reward for completing the project earlier than the due date. (4) No task can start prior to the planned starting time by the standard schedule. (5) The standard schedule assumes no crashing for each task and no time buffer between consecutive tasks.*

- **Supply chain operations***: (1) every stage in the supply chains operates under a periodic-review base-stock policy. (2) Processing times at all stages of the supply* *chain are sequential. (3) Unsatisfied demand is fully backordered at each stage. (4) Orders are fulfilled on a first-come-first-serve basis (FCFS) at each stage. (5) For any supply chain, delivery is made as soon as inventory becomes available in all stages except the last one where no early delivery can be made to projects on-site. (6) The delivery lead time from the last stage of a supply chain to projects on-site is negligible.*

- **Interface***: (1) the occurrences of projects are independent, and the amount of each type of material required are independent across projects. (2) All needed materials for one task of a project must be delivered in one set. (3) Material delivery status cannot be updated along with time. (4) Supply chains for materials required at different tasks operate independently. (5) Material inventory cannot be held at projects on-site.*

While most assumptions are common sense, a few require more explanation. Project assumption (4) is standard in construction industry as equipment and personnel are typically not available prior to the planned schedule. Project assumption (5) is based on the fact that tasks i $(0 < i \leq n)$ are critical, and the observed practice that no time buffer is planned for material delays. Supply chain assumption (5) and Interface assumption (5) are true when the project sites have limited space – often applicable in construction. Interface assumption (2) is true when there are significant economies of scale in transportation. We make Supply chain assumption (6) and Interface assumption (3) for simplicity. Relaxations are discussed at the end of this section.

Given the standard schedule, the project-driven supply chain (see Figure 2.2) operates as follows: task 0 is always on schedule because it is not subject to random material delays. Once task 0 is completed and material 1 is delivered, task 1 starts immediately. If material 1 is ready before the completion time of task 0, we hold it in inventory and pay a holding cost before delivering it at task 0's completion. Otherwise, material i is delivered immediately once it is ready. Depending on the material delay beyond the standard schedule, task 1 may be expedited (but not earlier than the completion time planned by the standard schedule) and incurs a crashing cost. Similar events take place for subsequent tasks and supply chains. Finally, if task n is delayed beyond the planned due date, a penalty is charged.

2.3.2 The Mathematical Model

To construct the model, we define the following notation:

- z_i: the duration crashed for task $i = 1, 2, ..., n$.
- Z_i: an upper bound on z_i.
- D_i: the amount of material i required at task i for a project. Define $\bar{D} = \{D_1, D_2, ..., D_n\}$. Let d_i ($\bar{d}$) be the realization of D_i ($\bar{D}$, respectively).
- $\bar{s}_i$: the base stock levels in the supply chain i, a vector.
- Δ_i: the time when material i is ready subtracts the planned starting time of task i. Let δ_i be its realization. If $\delta_i > 0$, it refers to the material delay of supply chain i beyond the planned starting time of task i; otherwise, it indicates how much earlier material i is ready than the planned schedule. Note that Δ_i depends on $\bar{s}_i$ and D_i.
- $\Delta'_i \geq 0$: the delay of task $i-1$'s completion beyond the planned starting time of task i, $i = 1, 2, \ldots, n$. Let δ'_i be the realization of Δ'_i.

- $W_i((\Delta_i^{'} - \Delta_i)^+)$: the inventory holding cost of material i at the last stage of supply chain i if this material is ready before the completion time of task $i-1$.

- $H_i(\bar{s}_i)$: the annual inventory holding cost in supply chain i excluding $W_i(\cdot)$.

- $C_i(z_i)$: crashing cost function of task i.

- Δ': project delay. Let δ' be its realization.

- $\Pi(\Delta')$: penalty cost if a project is delayed by Δ'.

- λ: the average number of projects completed in one year.

We construct a multi-stage mathematical model to determine the optimal base-stock levels for all supply chains and the crashing decisions for all tasks of projects. In the first stage, we set the base-stock levels for the supply chains. In subsequent stages, we consider each project and determine the optimal crashing policy for each task as material delays are realized over the current and subsequent tasks. Our objective is to minimize the annual supply chain and project cost.

Specifically, we utilize the flow-unit method (see Axsater 1990, Zipkin 1991, Zhao and Simchi-Levi 2006) by keeping track of a specific project and its material requirement. To identify the optimal crashing policy given a set of inventory decisions, we use a dynamic programming model where we let $G_i(\delta_i^{'}; \bar{s}_i, \bar{s}_{i+1}, ..., \bar{s}_n, d_i, d_{i+1}, \ldots, d_n)$ be the minimum expected cost incurred from task i on to the end of the project, including both supply chain and project costs, given that the delay from task $i-1$ is δ_i', the base-stock levels in supply chains i, $i+1, ..., n$ are $\bar{s}_i$, $\bar{s}_{i+1}, ..., \bar{s}_n$, and the amount of materials required from stages i to stage n are $d_i, d_{i+1}, ..., d_n$.

Consider $i = n$, because the probability distribution of Δ_n depends on $\bar{s}_n$ and d_n,

$$\begin{aligned} G_n(\delta'_n; \bar{s}_n, d_n) &= E_{\Delta_n}[W_n((\delta'_n - \Delta_n(\bar{s}_n, d_n))^+)] + E_{\Delta_n}[\min_{0 \le z_n \le Z_n} \{C_n(z_n) + \\ &\quad \Pi((\max\{\delta'_n, \Delta_n(\bar{s}_n, d_n)\} - z_n)^+)\}], \end{aligned} \tag{2.1}$$

where the notation E_{Δ_n} indicates that the expectation is taken with respect to Δ_n.

Similarly, for any $i = 2, 3, ..., n-1$, we have,

$$\begin{aligned} & G_i(\delta'_i; \bar{s}_i, \bar{s}_{i+1}, ..., \bar{s}_n, d_i, d_{i+1}, \ldots, d_n) \\ = \; & E_{\Delta_i}[W_i((\delta'_i - \Delta_i(\bar{s}_i, d_i))^+)] + E_{\Delta_i}[\min_{0 \le z_i \le Z_i} \{C_i(z_i) + \\ & G_{i+1}((\max\{\delta'_i, \Delta_i(\bar{s}_i, d_i)\} - z_i)^+; \bar{s}_{i+1}, ..., \bar{s}_n, d_{i+1}, ..., d_n)\}], \end{aligned} \tag{2.2}$$

and the transition function $\Delta'_{i+1} = (\max\{\delta'_i, \Delta_i\} - z_i)^+$.

For $i = 1$, because task 0 is not subject to any random material delay, $\Delta'_1 = 0$ and we drop Δ'_1 from the notation. Following a similar logic as Eqs. (2.1)-(2.2), $G_1(\bar{s}_1, \bar{s}_2, ..., \bar{s}_n, \bar{d})$ can be written as,

$$\begin{aligned} & G_1(\bar{s}_1, \bar{s}_2, ..., \bar{s}_n, \bar{d}) \\ = \; & E_{\Delta_1}[W_1((-\Delta_1(\bar{s}_1, d_1))^+)] + E_{\Delta_1}[\min_{0 \le z_1 \le Z_1} \{C_1(z_1) + \\ & G_2((\max\{0, \Delta_1(\bar{s}_1, d_1)\} - z_1)^+; \bar{s}_2, ..., \bar{s}_n, d_2, ..., d_n)\}]. \end{aligned} \tag{2.3}$$

Consider all projects conducted per unit of time, our first-stage optimization problem is,

$$\begin{aligned} \min_{\bar{s}_1, \bar{s}_2, ..., \bar{s}_n} \quad & \textstyle\sum_{i=1}^n H_i(\bar{s}_i) + \lambda E_{\bar{D}}[G_1(\bar{s}_1, \bar{s}_2, ..., \bar{s}_n, \bar{D})] \\ s.t. \quad & \bar{s}_i \ge 0, \forall i = 1, 2, ..., n. \end{aligned} \tag{2.4}$$

In this model, we first determine the optimal crashing decisions upon each scenario of material delays, then we optimize the inventory decisions accordingly.

2.4 Convexity and The Optimal Crashing Policy

In this section, we characterize the optimal crashing policy when given the inventory decisions.

Assumption 2.2 *We make the following assumptions:*

1. $C_i(\cdot)$ *and* $\Pi(\cdot)$ *are convex and increasing.*

2. $W_i(\cdot)$ *is convex and increasing.*

The convexity of the crashing cost, $C_i(\cdot)$, of a single task is justified by Nahmias (2004) Chapter 9. The crashing cost of a set of tasks is convex in the time crashed because one would first crash the cheapest tasks. The assumption on the delay penalty $\Pi(\cdot)$ and inventory holding cost $W_i(\cdot)$ includes the commonly seen practice of linear functions as a special case.

The following observations are straightforward, we omit the proof.

Observation 2.1 $f(x,y) = (\max\{x, x_0\} - y)^+$ *is jointly convex in* $(x,y) \in R^2$*; if* $g(z)$ *is convex and increasing in* z *and* $f(x,y)$ *is jointly convex in* (x,y)*, then* $g(f(x,y))$ *is also jointly convex in* (x,y)*.*

By Observation 2.1, $(\max\{\delta'_i, \delta_i\} - z_i)^+$ is jointly convex in (δ'_i, z_i). By Assumption 2.2 and Observation 2.1, $\Pi((\max\{\delta'_n, \delta_n\} - z_n)^+)$ is jointly convex in (δ'_n, z_n). These results are summarized in the following observation.

Observation 2.2 $(\max\{\delta'_i, \delta_i\} - z_i)^+$ *is jointly convex in* (δ'_i, z_i) *for* $1 \leq i \leq n$*.* $\Pi((\max\{\delta'_n, \delta_n\} - z_n)^+)$ *is jointly convex in* (δ'_n, z_n)*.*

Proposition 2.1 $G_i(\delta'_i; \bar{s}_i, ..., \bar{s}_n, d_i, ..., d_n)$ *is convex and increasing in* δ'_i *for* $i > 1$*.*

Proof. We use induction by first considering $i = n$. By Assumption 2.2, $C_n(z_n)$ is convex in z_n. By Zipkin (2000) Proposition A.3.10 and Observation 2.2,

$$\min_{0 \le z_n \le Z_n} \{C_n(z_n) + \Pi((\max\{\delta'_n, \delta_n\} - z_n)^+)\}$$

is convex in δ'_n. By Eq. (2.1),

$$\begin{aligned} G_n(\delta'_n; \bar{s}_n, d_n) &= E_{\Delta_n}[W_n((\delta'_n - \Delta_n(\bar{s}_n, d_n))^+)] + \\ & E_{\Delta_n}[\min_{0 \le z_n \le Z_n} \{C_n(z_n) + \Pi((\max\{\delta'_n, \Delta_n(\bar{s}_n, d_n)\} - z_n)^+)\}]. \end{aligned}$$

To show $G_n(\delta'_n; \bar{s}_n, d_n)$ is convex in δ'_n, we only need $W_n((\delta'_n - \delta_n)^+)$ to be convex in δ'_n, which is true by Assumption 2.2. $G_n(\delta'_n; \bar{s}_n, d_n)$ is also increasing in δ'_n because $W_n((\delta'_n - \delta_n)^+)$ is increasing in δ'_n (Assumption 2.2), and $\Pi((\max\{\delta'_n, \delta_n\} - z_n)^+)$ is increasing in δ'_n.

Now we consider $1 < i < n$. By the induction assumption, G_{i+1} is convex and increasing in δ'_{i+1}. By Eq. (2.2),

$$\begin{aligned} & G_i(\delta'_i; \bar{s}_i, \ldots, \bar{s}_n, d_i, ..., d_n) \\ = \; & E_{\Delta_i}[W_i((\delta'_i - \Delta_i(\bar{s}_i, d_i))^+)] + \\ & E_{\Delta_i}[\min_{0 \le z_i \le Z_i} \{C_i(z_i) + G_{i+1}((\max\{\delta'_i, \Delta_i(\bar{s}_i, d_i)\} - z_i)^+; \bar{s}_{i+1}, \ldots, \bar{s}_n, d_{i+1}, \ldots, d_n)\}]. \end{aligned}$$

By Observations 2.1-2.2 and the induction assumption,

$$G_{i+1}((\max\{\delta'_i, \delta_i\} - z_i)^+; \bar{s}_{i+1}, ..., \bar{s}_n, d_{i+1}, ..., d_n)$$

is increasing in δ'_i and jointly convex in δ'_i and z_i. Following the same logic as for $i = n$, we can show that $G_i(\delta'_i; \bar{s}_i, \ldots, \bar{s}_n, d_i, ..., d_n)$ is increasing and convex in δ'_i. The proof is completed. $\square$

Theorem 2.1 *There exists a threshold level, z_i^* (depending on δ_i' and δ_i), for task i ($0 < i \leq n$), such that the optimal crashing policy for task i, is to crash its duration by z_i^* if feasible, otherwise not crash if $z_i^* < 0$ or crash Z_i if $z_i^* > Z_i$.*

Proof. We first consider $i = 1$. By Eq. (2.3),

$$\begin{aligned} G_1(\bar{s}_1, \bar{s}_2, ..., \bar{s}_n, \bar{d}) \quad &= \quad E_{\Delta_1}[W_1((-\Delta_1(\bar{s}_1, d_1))^+)] + E_{\Delta_1}[\min_{0 \leq z_1 \leq Z_1} \{C_1(z_1) + \\ &\quad\quad G_2((\max\{0, \Delta_1(\bar{s}_1, d_1)\} - z_1)^+; \bar{s}_2, ..., \bar{s}_n, d_2, \ldots, d_n)\}]. \end{aligned}$$

By Assumption 2.2, Proposition 2.1 and Observation 2.1, $C_1(z_1) + G_2((\max\{0, \delta_1\} - z_1)^+; \bar{s}_2, ..., \bar{s}_n, d_2, ..., d_n)$ is convex in z_1. Let its global minimum be achieved at z_1^*, thus it is optimal to crash task 1 by z_1^* if $0 \leq z_1^* \leq Z_1$, otherwise not crash if $z_1^* < 0$ or crash Z_1 if $z_1^* > Z_1$.

For $1 < i < n$, it follows by Eq. (2.2) and the above logic that there exists a z_i^* (depending on δ_i', δ_i) achieving the global minimum for $C_i(z_i) + G_{i+1}((\max\{\delta_i', \delta_i\} - z_i)^+; \bar{s}_{i+1}, ..., \bar{s}_n, d_{i+1}, ..., d_n)$, and the optimal policy is to crash task i by z_i^* if feasible, otherwise not crash if $z_i^* < 0$ or crash Z_i if $z_i^* > Z_i$. The same logic applies to $i = n$. $\Box$

We now discuss some relaxations of Assumption 2.1. Project assumption (5) can be relaxed by allowing time buffers between consecutive critical tasks. To accommodate this generality, we must introduce additional notation on the starting times of the standard and crashed schedule rather than relying only on their difference. Supply chain assumption (6) can be relaxed without changing the model and results by properly adding a constant to Δ_n. We can relax Interface assumption (3) by assuming that material delivery status can be updated as project processes. While Proposition 2.1 and Theorem 2.1 remain true, the optimal crashing policy of a task shall depend on the material delivery status for all subsequent tasks.

Finally, we should point out that the cost function

$$\sum_{i=1}^{n} H_i(\bar{s}_i) + \lambda E_{\bar{D}}[G_1(\bar{s}_1, \bar{s}_2, ..., \bar{s}_n, \bar{D})]$$

is generally not convex in the base-stock levels. An example is given in §2.6.

2.5 Interaction of Supply Chain and Project Decisions

In this section, we study the interaction between supply chain inventory decisions and project crashing and delay decisions. Our key result is that if we reduce the inventory levels for material k which feeds task k and thus increase the delay of this material, Δ_k, with probability one (w.p.1), then the cumulative time crashed for all tasks preceding task k will decrease w.p.1 to match the delayed material k; while for each task succeeding task k, the time crashed will increase w.p.1 to catch up with the delayed schedule. The project delay will also increase w.p.1. More specifically, we derive the following results:

- By increasing the delay of material k (feeding task k) w.p.1, we show that the cumulative time crashed for all tasks prior to task k decreases w.p.1. (Proposition 2.2). We prove it by first showing that the right derivatives of the optimal cost functions for all tasks prior to $k+1$ decrease for each cumulative delay (Lemma 2.1), and then we show that the cumulative delays for all tasks prior to k increase w.p.1 (Lemma 2.2).

- By increasing the delay of material k (feeding task k) w.p.1, we also show that the time crashed for task k and all the subsequent tasks increases w.p.1 (Proposition 2.3). We prove it by first showing that the time crashed at task k increases w.p.1 (Lemma 2.3), and then we show that the cumulative delays for task k and all subsequent tasks as well as the project delay increase w.p.1 (Lemma 2.4).

As we decrease the inventory levels, $\bar{s}_k$, for material k, we assume that the delay of this material, Δ_k, increases w.p.1. This assumption holds for many inventory systems, such as single-echelon, multi-echelon serial and distribution systems, under quite general conditions (see Simchi-Levi and Zhao (2011) for a review). Given this monotonic relationship between Δ_k and $\bar{s}_k$, studying the impact of $\bar{s}_k$ on the project decisions is equivalent to studying the impact of Δ_k on these decisions. Thus, for the ease of exposition, we drop the notation of demand and stock levels in the subsequent analysis. We consider the right derivatives in the following proofs.

Impact On Tasks Preceding Task k

We first study the impact of a stochastically greater material k delay, Δ_k, on the optimal cost functions.

Lemma 2.1 *For any $j \leq k$, $G'_j(\delta'_j)$ decreases for each δ'_j as Δ_k increases w.p.1.*

Proof. When $j = k$, $G_k(\delta'_k) = E_{\Delta_k}[W_k((\delta'_k - \Delta_k)^+) + \min_{0 \leq z_k \leq Z_k}[C_k(z_k) + G_{k+1}((\max(\delta'_k, \Delta_k) - z_k)^+)]]$. Because it is not optimal to set $z_k > \max(\delta'_k, \Delta_k)$, we remove $()^+$ from G_{k+1} for simplicity in the rest of this section. The results still hold without this simplification.

Suppose Δ_k increases to $\Delta_k + E$, where E is a non-negative random variable and its realization is ϵ, let $\widetilde{G}_k(\delta'_k) = E_{\Delta_k+E}[W_k((\delta'_k - \Delta_k - E)^+) + \min_{0 \leq z_k \leq Z_k}[C_k(z_k) + G_{k+1}(\max(\delta'_k, \Delta_k + E) - z_k)]]$. Consider a sample path of $\Delta_k = \delta_k$ and $E = \epsilon$, we have three cases:

1. $\delta'_k < \delta_k$. $\widetilde{g}_k(\delta'_k) = \min_{0 \leq z_k \leq Z_k}[C_k(z_k) + G_{k+1}(\delta_k + \epsilon - z_k)]$ and $g_k(\delta'_k) = \min_{0 \leq z_k \leq Z_k}[C_k(z_k) + G_{k+1}(\delta_k - z_k)]$. Since neither of them is a function of δ'_k, $\widetilde{g}'_k(\delta'_k) = g'_k(\delta'_k) = 0$.

2. $\delta_k \le \delta'_k < \delta_k + \epsilon$. $\widetilde{g}_k(\delta'_k) = \min_{0 \le z_k \le Z_k}[C_k(z_k) + G_{k+1}(\delta_k + \epsilon - z_k)]$ and $g_k(\delta'_k) = W_k(\delta'_k - \delta_k) + \min_{0 \le z_k \le Z_k}[C_k(z_k) + G_{k+1}(\delta'_k - z_k)]$. Here $\widetilde{g}'_k(\delta'_k) = 0$ and $g'_k(\delta'_k) \ge 0$ ($g_k(\delta'_k)$ is convex increasing in δ'_k by Proposition 2.1), so $\widetilde{g}_k{}'(\delta'_k) \le g'_k(\delta'_k)$.

3. $\delta'_k \ge \delta_k + \epsilon$. $\widetilde{g}_k(\delta'_k) = W_k(\delta'_k - \delta_k - \epsilon) + \min_{0 \le z_k \le Z_k}[C_k(z_k) + G_{k+1}(\delta'_k - z_k)]$. Because $W'_k(\delta'_k - \delta_k - \epsilon) \le W'_k(\delta'_k - \delta_k)$ for each δ'_k ($W_k(\cdot)$ is convex increasing by Assumption 2.2) and the second terms of $\widetilde{g}_k(\delta'_k)$ and $g_k(\delta'_k)$ are the same, so $\widetilde{g}'_k(\delta'_k) \le g'_k(\delta'_k)$.

By Assumption 2.2, it is easy to see that for any δ'_k, the right derivatives, $g'_k(\delta'_k)$ and $\widetilde{g}'_k(\delta'_k)$, exist for all sample paths; furthermore, the right derivatives, $g'_k(\delta'_k)$ and $\widetilde{g}'_k(\delta'_k)$, are bounded from above. By Rubinstein and Shapiro (1993), Lemma A2, p. 70, the sample path derivatives are unbiased. Thus, $G'_k(\delta'_k)$ decreases for each δ'_k when Δ_k increases w.p.1.

To prove the same result for task $j < k$, we use induction by making the following induction assumption: for a given j ($j < k$), $G'_j(\delta'_j)$ decreases for each δ'_j when Δ_k increases w.p.1. Then, we consider $G'_{j-1}(\delta'_{j-1})$, where

$$\begin{aligned} G_{j-1}(\delta'_{j-1}) = E_{\Delta_{j-1}}[W_{j-1}((\delta'_{j-1} - \Delta_{j-1})^+) + \min_{0 \le z_{j-1} \le Z_{j-1}} [C_{j-1}(z_{j-1}) + \\ G_j(\max(\delta'_{j-1}, \Delta_{j-1}) - z_{j-1})]] \end{aligned} \tag{2.5}$$

Consider a sample path of $\Delta_{j-1} = \delta_{j-1}$, we first note that $W_{j-1}((\delta'_{j-1} - \delta_{j-1})^+)$ does not depend on Δ_k, so we omit it in the following analysis. For the second part of G_{j-1}, we replace z_{j-1} by $\max(\delta'_{j-1}, \delta_{j-1}) - \delta'_j$ and ignore the boundary of z_{j-1} for the ease of exposition (the same result holds with the boundaries), we arrive at

$\min_{\delta'_j}[C_{j-1}(\max(\delta'_{j-1}, \delta_{j-1}) - \delta'_j) + G_j(\delta'_j)]$. Let δ'^*_j be the δ'_j that minimizes this expression, δ'^*_j should satisfy the first order condition (FOC):

$$C'_{j-1}(\max(\delta'_{j-1}, \delta_{j-1}) - \delta'^*_j) = G'_j(\delta'^*_j).$$

Note that $C_{j-1}(\cdot)$ and $G_j(\cdot)$ are both increasing convex functions (by Assumption 2.2 and Proposition 2.1) and $G'_j(\delta'_j)$ decreases for each δ'_j when Δ_k increases w.p.1 by the induction assumption, we conclude that the optimal δ'^*_j increases when Δ_k increases w.p.1.

We now take the derivative of $C_{j-1}(\max(\delta'_{j-1}, \delta_{j-1}) - \delta'^*_j) + G_j(\delta'^*_j)$ with respect to δ'_{j-1} and use the FOC for δ'^*_j to arrive at,

$$d(C_{j-1}(\max(\delta'_{j-1}, \delta_{j-1}) - \delta'^*_j) + G_j(\delta'^*_j))/d\delta'_{j-1} = \begin{cases} 0, & \text{if } \delta'_{j-1} < \delta_{j-1} \\ C'_{j-1}(\delta'_{j-1} - \delta'^*_j), & \text{otherwise .} \end{cases}$$

Because the optimal δ'^*_j increases when Δ_k increases w.p.1, $C'_{j-1}(\delta'_{j-1} - \delta'^*_j)$ decreases for each δ'_{j-1} when Δ_k increases w.p.1. Consequently, $g'_{j-1}(\delta'_{j-1})$ decreases for each δ'_{j-1} when Δ_k increases w.p.1. Using a similar analysis and by Rubinstein and Shapiro (1993), Lemma A2, p. 70, we can show that the sample path derivatives are unbiased. Thus, we conclude that $G'_{j-1}(\delta'_{j-1})$ decreases for each δ'_{j-1} when Δ_k increases w.p.1. By induction, we have proven that $G'_j(\delta'_j)$ decreases for each δ'_j for all $j \leq k$ when Δ_k increases w.p.1. $\square$

We then study the impact of a stochastically greater Δ_k on the cumulative delays for the tasks preceding task k. For the ease of exposition, we ignore the boundaries on z_k in the following analysis. The results stay the same if we include these boundaries.

Lemma 2.2 Δ'^*_j *increases w.p.1 for any* $j \leq k$ *if* Δ_k *increases w.p.1.*

Proof. For task 1, we consider a sample path of $\Delta_1 = \delta_1$.

We must have $\min_{\delta_2'}[C_1(\max(\delta_1', \delta_1) - \delta_2') + G_2(\delta_2')]$. By definition, $\delta_1' = 0$. By Lemma 2.1, $G_2'(\delta_2')$ decreases for each δ_2' as Δ_k increases w.p.1, thus it follows from the first order condition (FOC) that the optimal $\delta_2'^*$ increases (hence, $\Delta_2'^*$ increases w.p.1) as Δ_k increases w.p.1 because $C_1(\max(\delta_1', \delta_1) - \delta_2')$ is convex decreasing in δ_2' and $G_2(\delta_2')$ is convex increasing in δ_2'.

To prove the same result for tasks $1 < j \leq k$, we use induction and make the following induction assumption: $\Delta_j'^*$ $(j < k)$ increases w.p.1 as Δ_k increases w.p.1. Consider task j and a sample path of $\Delta_j'^* = \delta_j'^*$ and $\Delta_j = \delta_j$, we must have:

$$\min_{\delta_{j+1}'}[C_j(\max(\delta_j'^*, \delta_j) - \delta_{j+1}') + G_{j+1}(\delta_{j+1}')]$$

By Lemma 2.1, $G_{j+1}'(\delta_{j+1}')$ decreases for each δ_{j+1}' as Δ_k increases w.p.1. Meanwhile, $C_j'(\max(\delta_j'^*, \delta_j) - \delta_{j+1}')$ decreases in δ_{j+1}'. By the induction assumption, $\max(\delta_j'^*, \delta_j)$ increases as Δ_k increases w.p.1, and so $C_j'(\max(\delta_j'^*, \delta_j) - \delta_{j+1}')$ increases for each δ_{j+1}' as Δ_k increases w.p.1. It follows from the FOC that the optimal solution $\delta_{j+1}'^*$ also increases as $\Delta_k$increases w.p.1. This concludes the induction and yields the desired result. □

We are now ready to study the impact of a stochastically greater Δ_k on project crashing decisions.

Proposition 2.2 $z_1^* + z_2^* + \cdots + z_{k-1}^*$ *decreases w.p.1 if* Δ_k *increases w.p.1.*

Proof. We consider a sample path of $\Delta_i = \delta_i$ for all $i \leq k$. By definition, we have

$$\delta_1 - z_1^* + (\delta_2 - \delta_2'^*)^+ - z_2^* + (\delta_3 - \delta_3'^*)^+ - \cdots - z_{k-1}^* + (\delta_k - \delta_k'^*)^+ = \max(\delta_k', \delta_k).$$

Rewrite the equation,

$$\begin{aligned} z_1^* + \cdots + z_{k-1}^* &= \delta_1 + (\delta_2 - \delta_2^{'*})^+ + \cdots + (\delta_k - \delta_k^{'*})^+ - \max(\delta_k^{'*}, \delta_k) \\ &= \delta_1 + (\delta_2 - \delta_2^{'*})^+ + \cdots + (\delta_{k-1} - \delta_{k-1}^{'*})^+ - \delta_k^{'*}. \end{aligned}$$

Because $\delta_2^{'*}, \ldots, \delta_k^{'*}$ increase as Δ_k increases w.p.1, we conclude, from above equation, $z_1^* + z_2^* + \cdots + z_{k-1}^*$ decreases for any sample path, and thus decreases w.p.1 as Δ_k increases w.p.1. □

Impact On Task k and Succeeding Tasks

We first focus on the time crashed at task k.

Lemma 2.3 *z_k^* increases w.p.1 if Δ_k increases w.p.1.*

Proof. We consider a sample path of $\Delta_i = \delta_i$ for all $i \leq k$. At task k, the optimization problem is $\min_{z_k}[C_k(z_k) + G_{k+1}(\max(\delta_k^{'*}, \delta_k) - z_k)]$, and the FOC is,

$$C_k'(z_k) = G_{k+1}'(\max(\delta_k^{'*}, \delta_k) - z_k).$$

Because $C_k(z_k)$ is a convex increasing function on z_k, $C_k'(z_k)$ is positive and increasing on z_k. We also note that because $G_{k+1}(\cdot)$ is a convex increasing function, then $G_{k+1}'(\max(\delta_k^{'*}, \delta_k) - z_k)$ is positive and decreasing on z_k. By Lemma 2.2, it follows by the assumption that δ_k increases, $\max(\delta_k^{'*}, \delta_k)$ shall increase as Δ_k increases w.p.1. As a result, $G_{k+1}'(\max(\delta_k^{'*}, \delta_k) - z_k)$ increases for each z_k as Δ_k increases w.p.1, and thus the optimal z_k^* increases for any sample path as Δ_k increases w.p.1. □

We then study the cumulative delays of task k and all subsequent tasks as well as the project delay.

Lemma 2.4 *$\Delta_j^{'*}$ (for any $j > k$) and the project delay $\Delta^{'*}$ increase w.p.1 if Δ_k increases w.p.1.*

Proof. For task k, we consider a sample path of $\Delta_i = \delta_i$ for all $i \leq k$. The optimization problem is $\min_{\delta'_{k+1}}[C_k(\max(\delta'^*_k, \delta_k) - \delta'_{k+1}) + G_{k+1}(\delta'_{k+1})]$. We note that $C_k(\max(\delta'^*_k, \delta_k) - \delta'_{k+1})$ is a convex decreasing function in δ'_{k+1} and $G_{k+1}(\delta'_{k+1})$ is a convex increasing function in δ'_{k+1}. By Lemma 2.2, $\max(\delta'^*_k, \delta_k)$ increases as δ_k increases, consequently, the optimal δ'^*_{k+1} increases when δ_k increases.

For task $j > k$, we use induction and assume that, for $j > k+1$, Δ'^*_j increases if Δ_k increases. Using a sample path of $\Delta_i = \delta_i$ for all $i \leq j$, we consider the optimization problem at task j, $\min_{\delta'_{j+1}}[C_j(\max(\delta'^*_j, \delta_j) - \delta'_{j+1}) + G_{j+1}(\delta'_{j+1})]$. We first note that $C_j(\max(\delta'^*_j, \delta_j) - \delta'_{j+1})$ is a convex decreasing function in δ'_{j+1} and $G_{j+1}(\delta'_{j+1})$ is a convex increasing function in δ'_{j+1}. By the induction assumption, $\max(\delta'^*_j, \delta_j)$ increases as δ_k increases, consequently, the optimal δ'^*_{j+1} increases when δ_k increases. This concludes the induction for tasks $j > k$.

The project delay $\Delta' = \Delta'_{n+1}$ if we extend the definition of Δ'_j to $n+1$. Because the penalty function is also a convex increasing function in Δ', by a similar proof, we can show that the optimal project delay increases w.p.1 when Δ_k increases w.p.1. $\square$

Finally, we study the impact of a stochastically greater Δ_k on the crashing decisions.

Proposition 2.3 *z^*_j increases w.p.1 for all $j > k$ if Δ_k increases w.p.1.*

Proof. For task j ($j > k$), we consider a sample path of $\Delta_i = \delta_i$ for all $i \leq j$ and the optimization problem

$$\min_{z_j}[C_j(z_j) + G_{j+1}(\max(\delta'^*_j, \delta_j) - z_j)].$$

We note that $C_j(z_j)$ is convex increasing in z_j and $G_{j+1}(\max(\delta'^*_j, \delta_j) - z_j)$ is convex decreasing in z_j. By Lemma 2.4, $\max(\delta'^*_j, \delta_j)$ increases when δ_k increases, and $G'_{j+1}(\max(\delta'^*_j, \delta_j) - z_j)$ increases for each z_j as $\max(\delta'^*_j, \delta_j)$ increases. As a result, the optimal z_j^* increases if δ_k increases. Thus, z_j^* increases w.p.1 for all $j \geq k$ if Δ_k increases w.p.1. □

2.6 An Illustrating Example

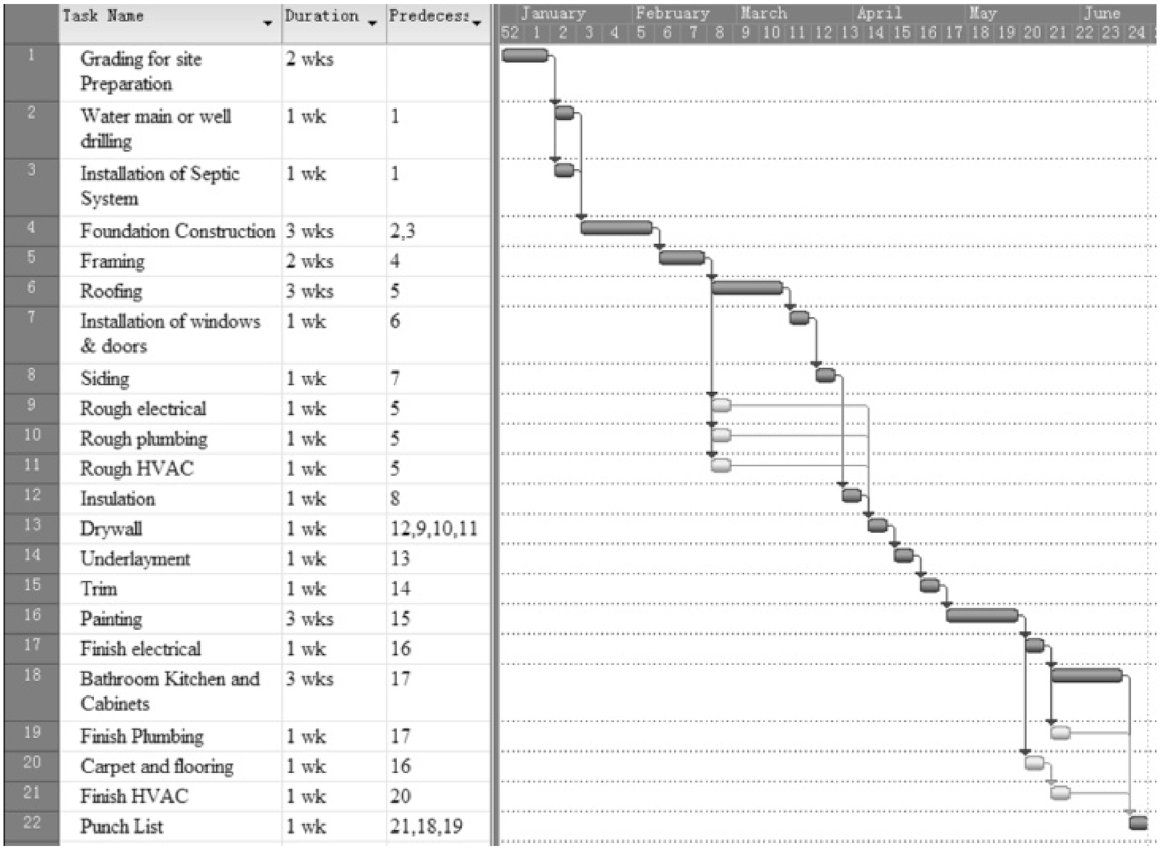
In this section, we apply the PDSC model and analysis to a real-world example, ICM, to demonstrate its potential.

2.6.1 ICM Overview

ICM is a U.S. based construction management firm that keeps internally only design, engineering, bidding, project planning and management functions. The company follows a standard bidding process for each project, and the outcome of which is not predictable. Once a project is awarded, ICM assigns it to a project manager who puts the plan into action by securing subcontractors (for labor) and ordering materials from suppliers. The project manager oversees the entire project execution and is not connected in any formal way to other project managers. We refer the reader to Shah and Zhao (2009) for a detailed case study.

Project operations. All construction projects follow a standard schedule (see Figure 2.3) where the tasks in darker color are on the critical path. The total duration of a project is 25 weeks.

The structural steel is needed for Task 5 (Framing) at the beginning of the 7th week. One project only requires one set of structural steel at this point in time. If this material is delayed, the tasks that can be crashed (expedited) to bring the schedule back on track are 5 (Framing), 6 (Roofing), 16 (Painting), and 18 (Bathroom, Kitchen and Cabinets). Crashing a task reduces its duration but must maintain the total labor



	Task Name	Duration	Predecessors
1	Grading for site Preparation	2 wks	
2	Water main or well drilling	1 wk	1
3	Installation of Septic System	1 wk	1
4	Foundation Construction	3 wks	2,3
5	Framing	2 wks	4
6	Roofing	3 wks	5
7	Installation of windows & doors	1 wk	6
8	Siding	1 wk	7
9	Rough electrical	1 wk	5
10	Rough plumbing	1 wk	5
11	Rough HVAC	1 wk	5
12	Insulation	1 wk	8
13	Drywall	1 wk	12,9,10,11
14	Underlayment	1 wk	13
15	Trim	1 wk	14
16	Painting	3 wks	15
17	Finish electrical	1 wk	16
18	Bathroom Kitchen and Cabinets	3 wks	17
19	Finish Plumbing	1 wk	17
20	Carpet and flooring	1 wk	16
21	Finish HVAC	1 wk	20
22	Punch List	1 wk	21,18,19

Figure 2.3: Standard project schedule at ICM.

hours. Crashing cost comes from the over-time wage which is 50% higher than the regular wage. The delay penalty per week for a project is 1% of the project revenue (less materials).

Structural steel supply chain. Structural steel is the most expensive material and is used in all projects. The structural steel supply chain consists of three stages: producer, service center and fabricator. The producer manufactures standardized shapes. The service center serves as a warehouse before fabricator. The fabricator customizes the structural steel according to engineering drawings. We refer to Figure 2.4 for a detailed description. These companies are the only government authorized suppliers in proximity and thus cannot be switched. Currently, the entire structural steel supply chain makes

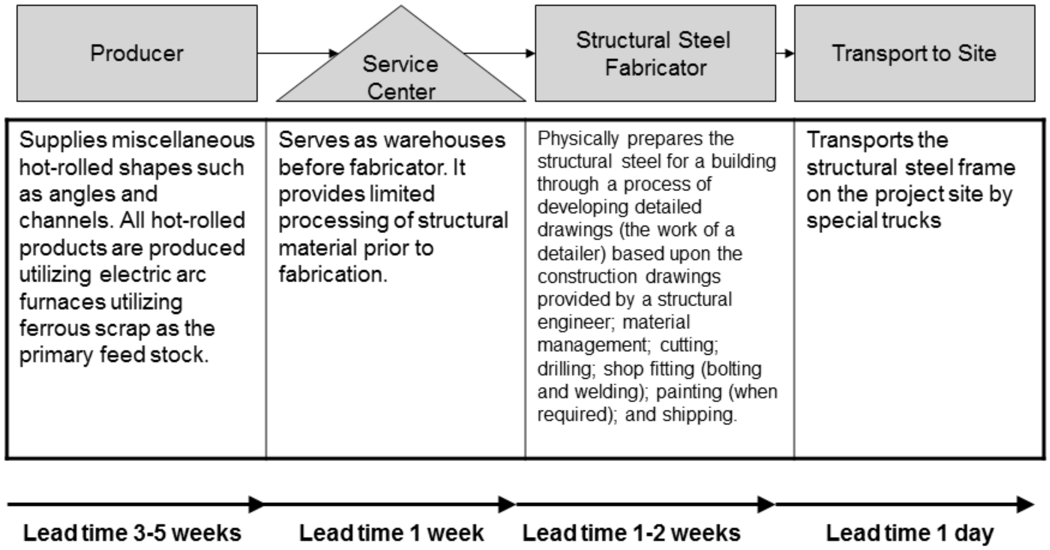


Figure 2.4: The structural steel supply chain.

to order and the total lead time ranges from 5 to 8 weeks.

Because projects may start at different times and require different amount of structural steel, ICM's weekly material requirement is highly sporadic and possibly zero if no projects require framing at a certain week.

2.6.2 PDSC Model for ICM

Assumption 2.1 holds for ICM. In addition, we need,

Assumption 2.3 *For ICM's projects and its structural steel supply chain, we assume*

- *Project operations: (1) There is at most one project requiring structural steel at any given period. (2) Project delay penalty is a linear function of the period delayed.*

- *Supply chain operations: (1) Safety-stock can only be held at the service center in the form of standardized shapes; (2) All stages in the structural steel supply chain share an identical review period – one week.*

The assumption on projects is justified by the real-world data and the fact that ICM receives about 20 projects a year. Safety-stock cannot be held at the fabricator because customized steel is project-specific and thus very risky to hold before the project is awarded.

For the structural steel supply chain, we assume the standard sequence of events at each stage (see, e.g., Hadley and Whitin (1963)): At the beginning of a period, the stage receives replenishment, then reviews inventory and places orders if appropriate. At the end of the period, all demand is realized, deliveries are made, and all costs are calculated for the period.

The PDSC model of §2.3 can be applied to ICM as follows: we group all tasks prior to Task 5 (where the structural steel is required) as task 0. We also group Task 5 and all subsequent tasks as task 1. The structural steel supply chain is denoted as supply chain 1. We define the following additional notation for the structural steel supply chain:

- D_1^t, d_1^t: the demand for structural steel at the t^{th} week, and its realization. Note that each project only requires one set of structural steel at one point in time and thus this demand process is formed by material requirement of consecutive projects. By the same nature of lead time demand, we define $D_1[t, t+k] = \sum_{i=0}^{k} D_1^{t+i}$ for $k \geq 0$. If $k < 0$, then $D_1[t, t+k] = 0$. We also define $D_1(l)$ be demand during l periods of time.

- L: the total lead time of the service center that includes the processing times at the producer and the service center.

- S: the base-stock level at the service center.

- X: the service time provided by the service center.
- Y: the processing time at the fabricator.
- h_s, h_f: The annual inventory holding cost per unit at the service center and fabricator respectively.
- T: the due date – the time when structural steel is needed for a project since the beginning of the project.

By definition,

$$\Delta_1 = X + Y - T. \tag{2.6}$$

By Eq. (2.3), $G_1(\cdot)$ for ICM is

$$\begin{aligned} G_1(S, d_1) &= E_{\Delta_1}[W_1((-\Delta_1(S, d_1))^+)] + \\ & E_{\Delta_1}[\min_{0 \le z_1 \le Z_1} \{C_1(z_1) + \pi(\max\{0, \Delta_1(S, d_1)\} - z_1)^+)\}], \end{aligned} \tag{2.7}$$

where the first term is the inventory holding cost at the Fabricator for materials that are ready before they are needed at a project.

For ICM, the first-stage optimization problem is,

$$\begin{aligned} \min_{S} \quad & H_1(S) + \lambda E_{D_1}[G_1(S, D_1)] \\ s.t. \quad & S \ge 0, \end{aligned} \tag{2.8}$$

where $H_1(S)$ is the annual inventory holding cost at the service center.

We now characterize how the base-stock level, S, determines the distribution of the service time (X) at the service center, the random material delay (Δ_1) to projects, and the cost of the supply chain. Define $R_k(d_1^t)$ to be the probability that demand d_1^t (> 0) is satisfied within the period $t + k$ at the service center. For a constant lead time L, $R_k(d_1^t) = \Pr\{S - D_1[t-L+k,t] \geq 0 | d_1^t > 0\}$ for $k \leq L$ (Hausman et al. 1998). For stochastic and sequential lead time L (see Zipkin 2000, chap. 7), we let the probability of $L = L_i$ be $\Pr\{L = L_i\}$, then $R_k(d_1^t) = \sum_i \Pr\{S - D_1[t - L_i + k, t] \geq 0 | d_1^t > 0\} \Pr\{L = L_i\}$ for $0 \leq k \leq \max_i\{L_i\}$. Let $X(d_1^t)$ be the service time of d_1^t (> 0), then $R_0(d_1^t) = \Pr\{X(d_1^t) = 0\}$ and $R_k(d_1^t) = \Pr\{X(d_1^t) \leq k\}$ for $k > 0$. Consequently, $\Pr\{X(d_1^t) = 0\} = R_0(d_1^t)$ and $\Pr\{X(d_1^t) = k\} = R_k(d_1^t) - R_{k-1}(d_1^t)$ for $k > 0$.

With the distribution of $X(d_1^t)$ (for $d_1^t > 0$), we can calculate the distribution of Δ_1 by Eq. (2.6). The expected inventory holding cost at the fabricator for a project that requires d units of structural steel can be calculated by,

$$E_{\Delta_1}[W_1((-\Delta_1(d))^+)] = E_{\Delta_1}[W_1((T-X-Y)^+)] = h_f \times d \times E[(T-X(d)-Y)^+]. \quad (2.9)$$

The net inventory at the end of period t at the service center is $N(t) = S - D_1[t-L,t]$, and the on-hand inventory at the service center is $I_s(t) = N(t)^+$. In steady state, the annual inventory holding cost at the service center is

$$H_1(S) = h_s E[I_s] = h_s E[(S - D_1(L+1))^+]. \quad (2.10)$$

2.6.3 Impact of the PDSC Model

Using historical data, we construct an empirical distribution for the weekly demand of structural steel, D_1^t, as shown in Table 2.1. Here we use the approximation of discrete demand by rounding up, for instance, all values in $[41, 50]$ to 50.

With $S = 0$ at the service center, the structural steel supply chain can deliver an order to project on-site at the beginning of 7^{th}, 8^{th}, 9^{th}, and 10^{th} week due to the review period and the sequence of events of the structural steel supply chain. Thus it may cause a delay of 1, 2, or 3 weeks because it is needed at the beginning of 7^{th} week.

Value	Frequency	Probability	**Value**	Frequency	Probability
0	32	0.604	**60**	4	0.075
10	0	0	**70**	1	0.019
20	0	0	**80**	2	0.038
30	1	0.019	**90**	0	0
40	7	0.132	**100**	2	0.038
50	4	0.075	**110**	0	0

Table 2.1: The empirical distribution of D_1^t.

We assume that the processing time at the producer has an equal chance to be 3, 4, and 5 weeks, and the processing time at the fabricator has an equal chance to be 1 and 2 weeks. The monthly inventory holding cost is \$16.6 per ton, and thus the annual inventory holding cost $h_s = h_f =$ \$199.2 per ton per year.

For ICM, the average number of construction projects conducted annually, λ, is 20. $T = 6$ weeks. The crashing cost function is piece-wise linear where,

$$C_1(z_1) - C_1(z_1 - 1) = \begin{cases} \$4,500 & z_1 = 1, \\ \$6,500 & z_1 = 2, 3, 4, 5, \\ \$8,000 & z_1 = 6, 7. \end{cases} \tag{2.11}$$

The project delay penalty per week, $\pi = \$13,700$. Clearly the delay penalty is much more expensive than the crashing costs and thus should be avoided.

We now compare the PDSC model to the current practice of ICM which holds zero inventory ($S = 0$) in the structural steel supply chain. Figure 2.5 summarizes the numerical result, where the vertical axis stands for the annual operating costs while the horizontal axis presents the base-stock level at the service center, S. We make the following observations:

- The total cost function is not convex in S. It first increases slightly (due to the bulky material requirement of projects, a small amount of inventory does not reduce lead time but adds to cost) and then decreases as S increases before it

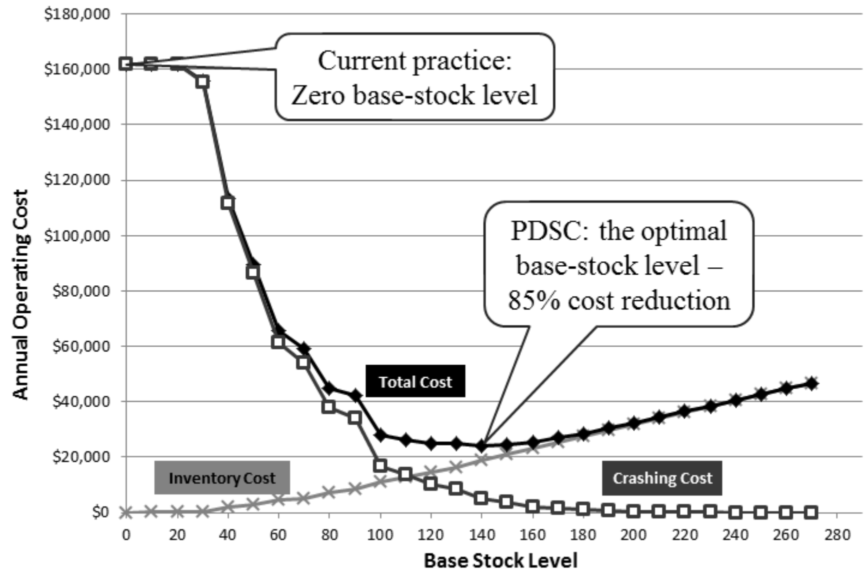


Figure 2.5: The annual operating costs.

reaches the global minimum at $S = 140$. Then it slowly increases as S further increases.

- The components of the total cost behave as we expected in §3.1: as we increase the base-stock level from $S = 0$, the inventory cost increases slowly but the project crashing cost decreases sharply, and thus the total cost decreases sharply. Beyond $S = 140$, the project crashing cost almost reaches zero but the inventory cost becomes significant, and so further increasing the base-stock level becomes un-economical. Comparing to the current practice with $S = 0$ (ignoring material supply in project decisions), the PDSC model (integrated supply chain and project decisions) brings the cost down from \$161,667 to \$24,164, with a 85.1% saving.

- The delay of structural steel, Δ_1, at the optimal base-stock level, $S = 140$, is much smaller (stochastically) than Δ_1 at $S = 0$, see Table 2.2. Thus the safety-stock at the service center reduces the random material delays and stabilizes the schedule of the projects, such benefits far outweigh the holding cost of the safety-stock.

	$P\{\Delta_1 =$ 0 week$\}$	1	2	3	$E[\Delta_1]$
$S=0$	0.167	0.333	0.333	0.167	1.5
$S=140$	0.954	0.037	0.008	0	0.054

Table 2.2: Delays of structural steel.

2.7 Numerical Study

In this section, we conduct an extensive numerical study to gauge the potential savings of the PDSC model by solving various environments that companies may face in practice. We use ICM as a base case and conduct a sensitivity analysis with respect to its parameters, such as demand variability, inventory holding cost, lead time and due date.

Demand Variability and Project Arrival Rate λ. We first study the impact of demand variability on the savings. We define $\tilde{D}$ to be the nominal weekly demand which follows a normal distribution. Since demand cannot be negative, we set the weekly demand distribution to be discrete truncated normal where $P\{D_1(1) = 0\} = P\{\tilde{D} \leq 0\}$, and $P\{D_1(1) = k\} = P\{\tilde{D} \leq k\} - P\{\tilde{D} \leq k - 10\}$ for $k = 10, 20, \ldots$. In the numerical study, we set $E[\tilde{D}] = 30$ tons and change the standard deviation of $\tilde{D}$ from 3 tons to 48 tons while keeping everything else the same as in §2.6.

Consistent to intuition, our numerical results (not reported here) shows that when the demand variability increases, more inventory is needed to buffer against demand uncertainty and to reduce material delays down to the same level. Thus, the optimal base stock level and the total cost increase. Given the cost of the benchmark, $S = 0$, remains unchanged, the percentage saving decreases as demand variability increases.

We also studied the impact of project arrival rate, λ, on the percentage savings. Here we set $E[\tilde{D}] = 30$ tons and the standard deviation of $\tilde{D}$ to be 15 tons but vary

Inventory cost factor	Optimal base-stock level	% Cost increment at $S = 150$ over the optimal cost
1.4	140	2.99%
1.3	140	2.39%
1.2	140	1.7%
1.1	140	0.91%
1	150	0
0.9	150	0
0.8	150	0
0.7	150	0
0.6	160	1.34%

Table 2.3: Robustness of the optimal base-stock level.

λ from 10 to 50. Our numerical study (not reported here) shows that the percentage savings increase as λ increases. Intuitively, it is more beneficial to integrate inventory and project decisions if projects occur more frequently.

Inventory Cost. Second, we study the impact of inventory holding cost on the savings from the PDSC model. We set the coefficient of variation of the nominal demand to be 0.8 and change the inventory holding cost per ton per week (at the service center and fabricator) according to \$3.82 $\times$ an inventory cost factor which varies from 0.6 to 1.4. The result (not reported here) shows that when holding inventory becomes more expensive, the percentage savings decrease. The result is intuitive because more expensive inventory cost means less valuable the option of holding inventory, and so the smaller the percentage saving from integrating safety-stock and crashing decisions.

We also study the sensitivity of the optimal base-stock level with respect to the inventory holding cost. To test the robustness of the solution, we compare the cost of a benchmark base-stock level, $S = 150$, to the optimal cost at various inventory holding costs for which this benchmark base-stock level is not optimal. Table 2.3 shows that when the holding cost increases, the optimal base-stock level tends to decrease

Lead Time Range	(3,5)	(4,6)	(5,7)	(6,8)	(7,9)
% Cost Savings	85.6%	89.1%	93.2%	94.9%	93.7%

Table 2.4: Impact of the lead time.

but not by much. Even if the optimal base-stock level differs from the benchmark, the cost of benchmark solution is only slightly higher than the optimal cost. In summary, these studies show that inventory holding cost may have a significant impact on the percentage savings of the PDSC model but much less an impact on the optimal base-stock level.

Lead Time. To study the impact of lead time on the effectiveness of the PDSC model, we assume that the lead time for the service center follows a uniform distribution with different ranges, see Table 2.4. We set the coefficient of variance of the nominal demand to be 0.8, inventory holding cost per ton per week to be \$3.82, and everything else remains unchanged.

The impact of lead time is more complex than that of demand variability and inventory holding cost because the latter only affects the supply chain inventory cost but the former affects both the project crashing and supply chain inventory costs. Table 2.4 shows that as the lead time increases, the % saving tends to increase but not always. This is true because as lead time increases, the project crashing cost may increase significantly in all solutions, and thus lead to smaller percentage savings.

Due Date. Finally, we study the impact of the project due date, T, on the effectiveness of the PDSC model. For this purpose, we set the lead time for the service center to be uniformly distributed from 3 to 8 weeks, the coefficient of variance of the nominal demand to be 0.8, and inventory holding cost per ton per week to be \$3.82. We choose a wider ranged lead time than previous studies because it allows us to study a wider

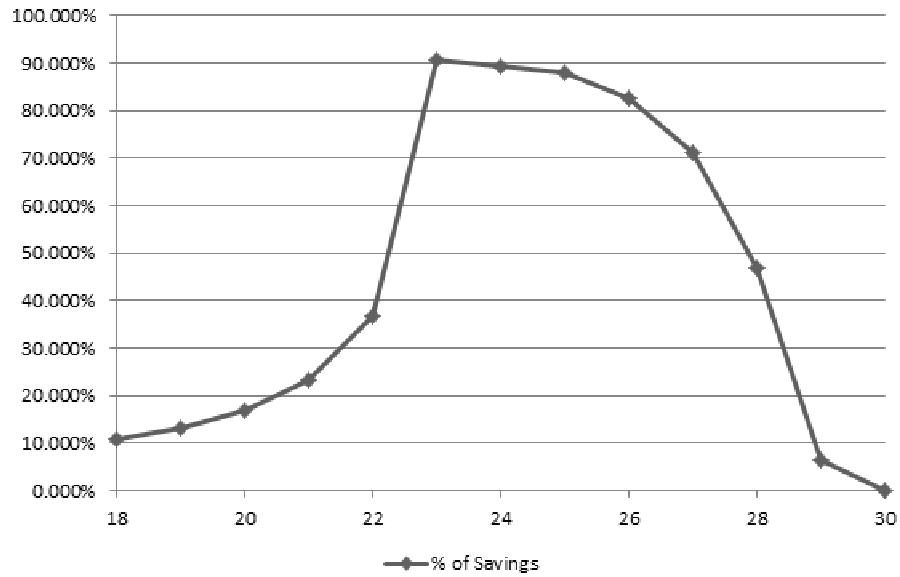


Figure 2.6: The impact of project due date.

range of the due date.

In the standard schedule of ICM, task 0 (task 1, see definition in §2.6.2) has a duration of 6 (19) weeks. So the project duration is 25 weeks. Because task 0 cannot be expedited and task 1 can be expedited by seven weeks at most, the shortest duration of the project is 18 weeks without considering material delays. Considering the worst case of material delay and no expediting of any task, the longest duration of the project is 30 weeks.

Figure 2.6 illustrates the following insight,

- When the project due date is very tight (approaching the shortest duration), the saving of the PDSC model diminishes because the high penalty cost and the tight due date force us to crash all tasks to their minimum in all solution approaches. In this case, safety-stock only helps to balance the project delay penalty and inventory holding cost.

- When the project due date is very loose (approaching the longest duration), the saving again diminishes. This is true because there is plenty of time to accommodate material delays, and thus it is not necessary to consider material supply while planning for projects.

- When the due date is in between the shortest and longest durations, the savings can be quite significant because one has the flexibility to coordinate safety-stock and crashing decisions so as to balance the inventory and project costs. Specifically, the percentage saving reaches its peak value at 23 weeks. We note that if the due date is less than 23 weeks, the percentage saving is low because the total cost is high due to project delay penalty paid in some events. If the due date is greater than 23 weeks, we can manage not to pay delay penalty in any event, which reduces the total cost and increases the percentage savings.

In summary, the saving of integrating safety-stock and crashing decisions reaches its peak value when the due date is moderate, which is most likely the case in practice.

2.8 Conclusion

In this chapter, we study a new class of problems – recurrent projects with random material delays, at the interface between supply chain and project management. We present the model of project-driven supply chains to jointly optimize safety-stock and project crashing decisions. We prove certain convexity properties for the model which facilitate fast computation of the optimal crashing decisions. We also study the dependence of the project crashing decisions on the supply chain inventory decisions. We demonstrate the impact of the model by a real-world example and study its sensitivity with respect to several system parameters. The model, although ignoring the economies of scale in production as well as material/project customization, captures the trade-off between supply chain operations and project operations for recurrent projects under uncertainty. It demonstrates that a certain amount of material inventory, if placed at the right location(s) of the supply chain, can help stabilizing the schedule of projects and reducing system-wide cost substantially.

One variation of the problem not considered in this chapter is recurrent projects with different schedules. For instance, the same material may be required by different types of projects at different tasks. While the general trade-off and idea of the PDSC model still apply, we must modify the model to account for different project types. In this chapter, we only test the effectiveness of the model in examples with one critical material. It would be interesting to see the impact of the PDSC model in practice with multiple materials. Finally, the structure of the projects considered in this chapter is simplified into a sequence of critical tasks. In reality, projects can be more complex with parallel non-critical paths. The development and application of the PDSC model in these areas remain as our future research directions.

Chapter 3

Incentives and Coordination in Project-Driven Supply Chains

3.1 Introduction

Over the last three to four decades, advances in technology and the networked economy have led to the evolution of the business models in many project driven industries, from the "one-firm-does-all" approach to a more collaborative one on a global basis. Examples can be found in book publishing, commercial aerospace, and engineering-procurement-construction (EPC) industries. While projects in these industries vary significantly in content and scale, they share the following commonalities: First, they require diverse knowledge and expertise; Second, they demand a significant investment of time and/or capital up front. The significant up front investment mandates market expansion a necessity for success.

The book publishing industry is popularized by books with many coauthors. Using textbooks on operations management as an example, a simple search of the key-word "operations management" on Amazon.com in September 2013 returns 48 textbooks which are the most relevant (definition: (1) production & operations section (2) hard-cover (3) four stars & up). Among them, 17 (35.42%) are single authored, 19 (39.58%)

have two authors, and the rest have three or more authors. Thus, coauthored books account for a majority (about 65%) of the most relevant textbooks on operations management. Replicating the search on “supply chain management” and “marketing science” returns similar results.

In the commercial aerospace industry, suppliers are playing an increasingly important role in the development of new aircrafts. Recent examples are Boeing 787 Dreamliner, Airbus 380, China Comac C919 and Airbus 350. In particular, the Boeing 787 Dreamliner outsourced 65% of the development work to more than 100 suppliers from 12 countries (see Horng and Bozdogan (2007) and Exostar (2007)). Tier 1 suppliers design and fabricate 11 major subassemblies, Boeing integrates and assembles the airplane. To manage the relationship with the suppliers, Boeing made the suppliers stakeholders of the program by establishing a collaborative partnership (similar to the coauthorship) where the suppliers are responsible for the non-recurring development cost of their tasks and must wait until the completion of the project to get paid (see Xu and Zhao 2011).

In the EPC industries, the $150 billion international space station (ISS) is a representative example where the design and construction of ISS are spread out to fifteen countries around the world. The elements of ISS are not assembled on the ground but launched from different countries and mated together on orbit. Each country invests a huge amount of money into its elements and takes the responsibility of their maintenance. Five countries are the principals (partners) of ISS due to their significant contributions.

As we can see, collaboration and partnership are everywhere, especially in large complex projects. By definition (Macmillan Dictionary), collaboration is “the action of working with someone to produce or create something". In the project management context, we define collaboration the basic form precisely as follows: the workload of a project, for instance, different tasks, is spread out to multiple players (firms) where each player is fully responsible for the financial needs of its own tasks until the completion of the project and share the revenue (or the credit or the utility) when the project is completed. This definition is consistent to the coauthorship in book publishing, the collaborative partnership of the Boeing 787 Dreamliner program, and the agreement among multiple countries for the International Space Station. For the ease of exposition, we call the financial arrangement of this kind of collaboration, the "loss-sharing partnership", as the loss due to a project delay is shared among all players. We also call the supply chain created by spreading the workload of a project among multiple firms "a project-driven supply chain".

Collaboration and partnership offer significant benefits to projects: First, they allows the project to utilize the best in-class expertise and knowledge. For instance, authors with different expertise can combine their domain knowledge in a single book. Second, a collaborative partnership allows multiple players to share the up front investment and thus make a costly project that is infeasible for any individual player feasible, as in the ISS project. Thirdly, collaboration and partnership are essential to market expansion. As witnessed in the Boeing 787 Dreamliner program, the suppliers are the stakeholders of the program and thus are motivated to sell the plane in their own countries and keep the customers waiting despite the significant delay of the program.

Collaboration (and partnership) is one way to outsource the workload of a project, subcontracting is another. Collaboration (and the "loss-sharing " partnership) differs from subcontracting because in the latter, suppliers get paid when their tasks are completed and certified. Thus in subcontracting, a supplier's interests are tied only to its tasks, whereas in collaboration, its interests are tied to the project. This difference is important because collaboration provides a much stronger incentive than subcontracting to the players to expand the market (so everyone gets more) and keep customers waiting until the final completion of the project (so everyone loses less).

Although the benefits are irresistible, collaboration (and partnership) poses a significant challenge in the incentive and coordination of joint projects (or project-driven supply chains); in the economics terms, collaboration may suffer the externalities. To see this intuitively, let's consider a simple example (see Figure 3.1) where a project has five tasks and four participating companies. It is easily seen that firm B can only start its task after firm A completes its tasks, and has to watch out for firm D's completion time to determine its own task duration. Thus each company's cost and schedule depend not only on its own effort but also on the efforts of other companies working on other parts of the same project. In this way, collaboration introduces gaming issues to project management where the ultimate goal of each firm is to optimize its own benefit even if doing so harms the interests of the project.

Although the economics and supply chain literatures study externalities and gaming issues extensively, they rarely consider project management specifics, e.g., project networks, cost structures and time-cost trade-off. In this chapter, we combine the game theoretical models of the economics and supply chain literatures with operational specifics drawn from the project management literature to study strategic gaming behaviors of firms under loss sharing partnership in joint (i.e., collaborative) projects. Our objective is to provide insights into the following issues: (1) What is the performance of

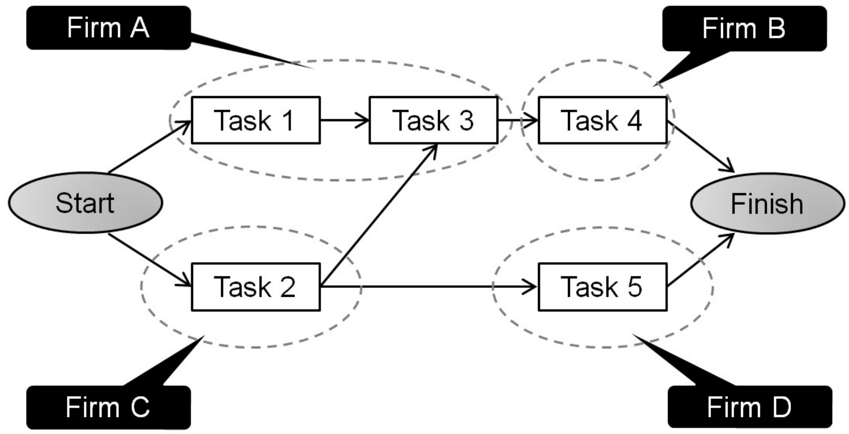


Figure 3.1: Collaboration in a joint project.

the project in time and cost under loss sharing? (2) How do project network and cost structure affect the results? (3) How to design a collaborative partnership that aligns the interests of the firms with that of the project?

To this end, we consider a two-level project network with parallel tasks (e.g., subsystems) in the first level and an integration task (e.g., final assembly) in the second level. Such a project network is quite representative in practice. Each firm faces a time-cost trade-off and must decide its task duration. We study various cost and network structures and characterize the subgame perfect equilibriums either in closed-form or by numerical algorithms. We find that under the loss sharing partnership, there is an inherent mismatch between individual firms' best interests and that of the project. Depending on the cost and network structures, we made a few surprising discoveries, such as (1) the Prisoners' Dilemma: even though keeping the optimal schedule benefits the entire project, it can be in each firm's best interests to delay; (2) the Supplier's Dilemma: if costs are time-dependent, the supplier may have to delay (even at a loss) in order to raise the penalty too high for the manufacturer to delay, to avoid a greater loss; (3) the Coauthors' Dilemma: a firm can expedite its task but cannot expedite the project because if it expedites, other firms will delay. Finally, we present a new "fair sharing" partnership which enhances collaboration the basic form (the loss sharing partnership) by a set of new provisions and prove its capability to align individual firms' financial interests with that of the project.

The chapter is organized as follows. In §3.2, we review the related literature; which is followed by §3.3 where we introduce our models and methodology. In §3.4, we study firms' strategic gaming behaviors under loss sharing. In §3.5, we present the "fair sharing" partnership and prove its effectiveness. We conclude the chapter in §3.6 with a brief summary of our results.

3.2 Literature

This chapter is related to the bodies of literature on project management, economics theory of teamwork, development chain management and project/supply chain interfaces. We shall review related results in each area and point out the difference from our work.

Classic project management literature. The most well known results in this literature include the critical path method (CPM), project evaluation and review techniques (PERT), time-cost analysis (TCA), and resource constrained project scheduling (RCPS). This literature focuses on the scheduling and planning of project(s) within a single firm and thus the main issue is on optimization. We refer the reader to Nahmias (2008) and Jozefowska and Weglarz (2006) for recent surveys. Ours draws the project management details, e.g., cost structure, project network and time-cost trade-off, from this literature but analyzes incentives and gaming behaviors under partnerships in a multi-firm joint project by a game theoretic model.

Classic economics literature of teamwork. The economics literature of teamwork discusses incentives and contracts in general teamwork settings. This literature is vast, we refer the reader to several seminal papers, e.g., Holmstrom (1982), Demski and Sappington (1984), McAfee and McMillan (1986), and Holmstrom and Milgrom (1991), for principal-agent models and moral hazard games; and Bhattacharyya and Lafontaine (1995), Kim and Wang (1998), and Al-Najjar (1997) for the double moral hazard games. Ours enriches and expands this literature by integrating the general economics theory with project management specifics.

Bidding and subcontracting in project management. This body of literature studies project management issues involving multiple firms, such as project bidding and subcontracting. Elmaghraby (1990) studies project bidding under deterministic and probabilistic activity durations from the contractor's perspective, while Gutierrez and Paul (2000) compares fixed price contracts, cost-plus contracts and menu contracts in project bidding from the project owner's perspective. Paul and Gutierrez (2005) studies how to assign tasks to contractors for projects with parallel or serial tasks. Szmerekovsky (2005) studies the impact of payment schedule on contractors' performance. In this model, the owner selects the payment terms in the first place, the contractor then decides the schedule to maximize its net present value. Aydinliyim and Vairaktarakis (2010) considers a set of manufacturers who outsource certain operations to a single third party by booking its capacity, and the third party identifies a schedule that minimizes the total cost for all manufacturers. Ours differs from this literature in two ways: first, we consider collaboration and partnerships which are structurally different from subcontracting as shown in §3.1. Second, all partners considered in this chapter have to contribute to the workload and share the outcome, while in the subcontracting literature, the project owner does not work but only supervises the contractors' work.

Development chain management. This stream of literature studies issues in the development of new products within a single firm and more recently involving multiple firms. For instance, Bhaskaran and Krishnan (2009) studies a development chain with two firms, a focal firm and a partner firm. Their model considers the cost, time, and quality triangle under three partnerships: revenue sharing, investment sharing and innovation sharing. They show that simple revenue sharing does not work well and leads to underinvestment in quality improvements. Alternatively, the investment sharing and innovation sharing, are better than revenue sharing in collaboration. This chapter contributes to this literature by incorporating project management specifics, such as, project network and time-cost trade-off (concepts developed in the classic project management literature) into the analysis.

Project management and supply chain interfaces. This literature studies the management of projects that involve multiple firms from a supply chain perspective and consider project management specifics. It is a fairly new research area but has attracted quite some attentions recently from the operations management community. For instance, Bayiz and Corbett (2005) introduces a principal-multi-agent game to project management by considering projects either with two sequential tasks or with two parallel tasks. They analyze the effectiveness of the fixed-price contracts versus incentive contracts in a subcontracting arrangement. Kwon, Lippman, McCardle, and Tang (2010) analyzes delay payment versus no delay payment in a project management setting where different but parallel tasks are done by different suppliers. They consider a simultaneous game among suppliers while the manufacturer does not contribute to the project but only selects payment regimes. By assuming exponentially distributed task durations, they showed that the delayed payment regime is more preferred by the manufacturer when its revenue is low. In addition, under information symmetry, the delayed payment regime is preferred in the presence of a large number of suppliers. In this chapter, the manufacturer contributes to the workload and so the project network has tasks both in parallel and in sequential. This new network entails a more delicate interaction among the suppliers and the manufacturer, and provides a rich ground for new discoveries and insights.

3.3 The Model and Preliminaries

In this section, we introduce the fundamentals of our model. First, we present the project management specifics such as the project cost structure and project network. Second, we provide more details on the loss sharing and fair sharing partnerships. Finally, we present the game theoretical model and our methodology.

Project Cost Structure. We can classify project costs into two categories: direct cost and indirect cost. Direct cost includes all costs directly contributing to a task, such as the cost of management, labor, material and shipping. Normally, a longer task duration is coupled with a lower direct cost. Indirect cost includes all costs not directly contributing to tasks but depending on the project duration, such as the overhead (e.g., rent, utilities, benefits), interests and financial costs, delay penalty and order cancellation loss. Normally, a longer project duration is coupled with a higher indirect cost. We refer the reader to Nahmias (2008)) for more details.

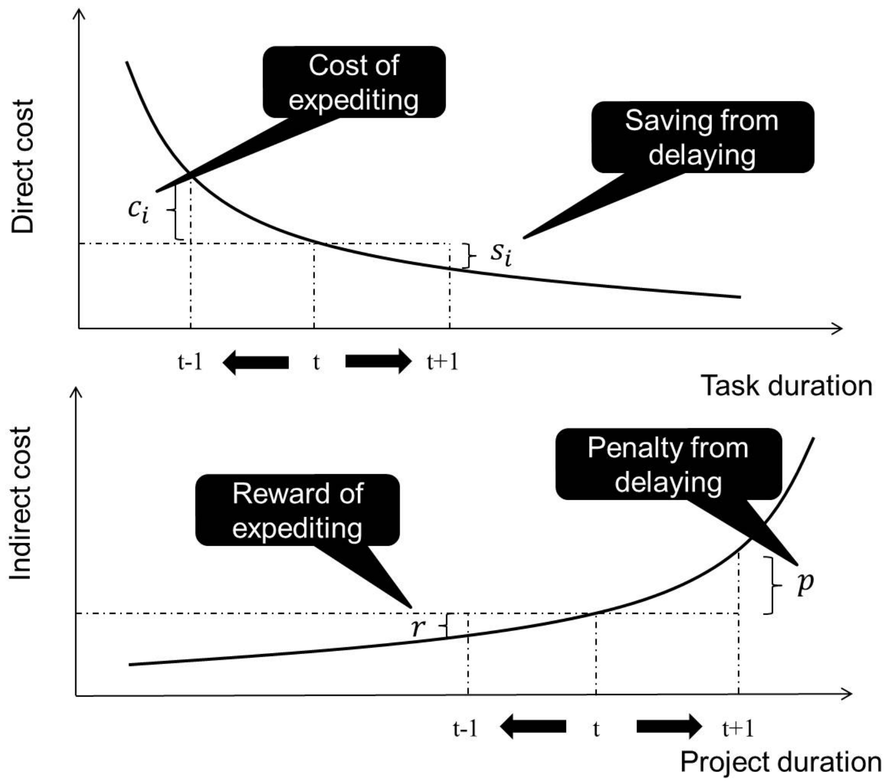


Figure 3.2: Project Cost Structure.

Consistent to a majority of practical situations, we assume that direct cost is convex and decreasing as task duration increases and indirect cost is convex and increasing as project duration increases (Figure 3.2, Nahmias (2008)). If task i is delayed by one period, firm i saves s_i in the direct cost. If the project is delayed by one period, it suffers a penalty p in the indirect cost. Conversely, if task i is expedited by one period, firm i incurs a cost c_i for expediting. If the project is completed one period earlier, it receives a reward r.

Project Network. We consider projects with a network structure shown in Figure 3.3. It has two levels: At level 1, there are several tasks to be completed simultaneously, similar to the design and fabrication of subsystems in the 787 Dreamliner program, the

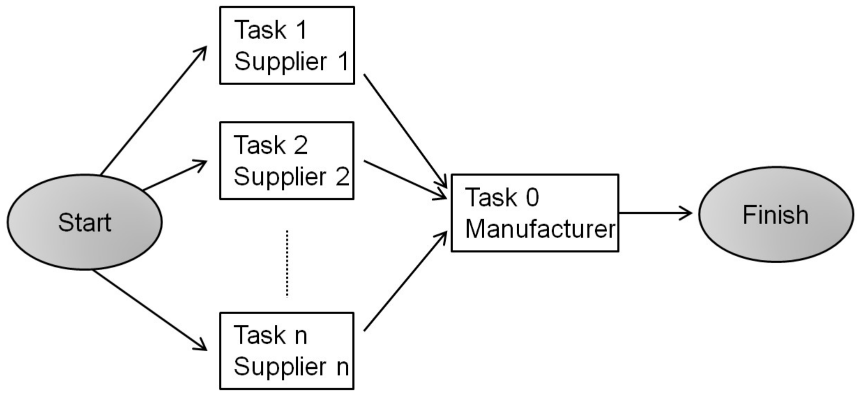


Figure 3.3: Project Network.

writing of individual chapters in a coauthored book, and the development of subsystems and components of the International Space Station (ISS). At level 2, there is only one task that is to integrate and assemble all parts completed in level 1, similar to the system integration task in the 787 Dreamliner program, the integration and proofreading of a coauthored book, and the final assembly and testing task of the ISS. Clearly the task at level 2 cannot start until all tasks at level 1 are completed.

Figure 3.3 shows the general project network, where $n = 1$ denotes the case with only one task at level 1, and thus the project network reduces to two sequential tasks. When $n \geq 2$, there are multiple tasks at level 1, and the project network has an assembly structure. We will discuss these two cases in the chapter.

The Loss Sharing Partnership. In this partnership, each firm pays for the direct and indirect costs of its own task(s), and get paid when the project is done. We observe that under loss sharing, if a firm delays its task, it saves on its direct cost but everyone (including the delayed firm) suffers an increase (a penalty) in indirect cost if the firm's delay results in a project delay. Thus other firms on time are penalized by this firm's delay, and this delayed firm is not fully responsible for the consequences of its action as the penalty is shared among all firms. While this observation presents a "moral hazard" issue well known in the economics literature of teamwork, it is not known exactly how such an issue may affect the time and cost metrics in a project management setting, which is the focus of this chapter.

The Fair Sharing Partnership. This partnership works in the same way as loss sharing except that every firm is fully responsible for the consequence of its action. Intuitively, if one firm causes damage to others, it has to compensate others; if it brings benefit to others, it receives compensation from others; we refer the reader to §3.5 for the exact mechanisms of this partnership.

Game Theoretical Framework. We assume that each task in the 2-level project network is assigned to a different firm. For the ease of exposition, we use "supplier(s)" to name the firm(s) responsible for the tasks at level 1 and "manufacturer" to name the firm responsible for the task at level 2. By the structure of the project network, a two-stage game theoretic model is appropriate for predicting the behaviors of the supplier(s) and the manufacturer in equilibrium. The sequence of events is described as follows (see also Figure 3.4): At the beginning of stage 1, supplier(s) start their tasks and choose task durations. After all suppliers complete their tasks, stage 1 is concluded. At the beginning of stage 2, the manufacturer starts its task and chooses the task duration. When the manufacturer completes its task, stage 2 is drawn to an end and the project is completed. In this game, the suppliers take the lead by taking actions first (anticipating the manufacturer's response) and the manufacturer follows by responding accordingly. We assume information symmetry thus the direct and indirect cost functions of all players are public knowledge. Under either partnership, each firm

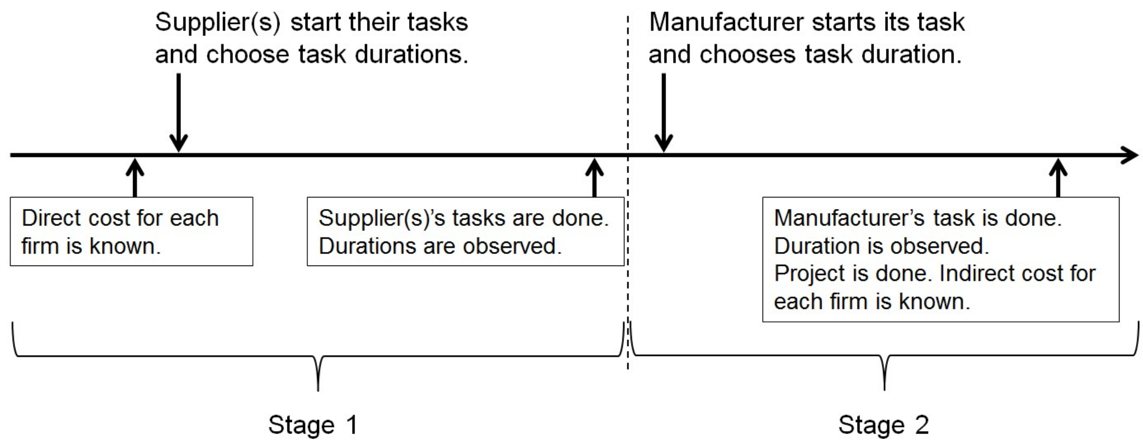


Figure 3.4: Sequence of Events.

aims to maximize its own profit by determining the duration of its own task. We shall derive subgame perfect Nash equilibrium (SPNE) for each case considered below and compare the resulting project performance to the global optimum. If the SPNE is not unique, we shall compare different SPNEs and report the Pareto or strong equilibrium.

Methodology. To understand the firms' strategic behaviors under loss sharing and how they may deviate from the optimal decisions under the "one-firm-does-all" (centralized control) model, we assume that the project starts with an "original schedule" and "original task durations" that are optimal under the centralized control. We first analyze one-period models in which each firm can delay or expedite its original task duration by at most one period. Then we relax this constraint to allow the firms to delay or expedite multiple periods. To study the impact of cost structure and project network on the firms' behaviors, we consider both time-independent and time-dependent costs, and both one supplier and multi-supplier cases.

3.4 The Loss Sharing Partnership

In this section, we study firms' strategic behaviors under the loss sharing partnership. We start with the base model in §3.4.1 which assumes only one supplier and time-independent cost. In this model, each firm can either "keep" the original task duration or "delay" it by one period. In §3.4.2, we relax the time-independent cost assumption in the base model to allow time-dependent costs, for instance, delay penalty per period may increase as the project delay increases. In §3.4.3, we consider the base model but allow each firm an additional option of "expediting" its task by one period. In §3.4.5, we extend the base model to include multiple suppliers, and in the last subsection, §3.4.6, we consider a general model and develop structural results and algorithms for the equilibrium.

3.4.1 The Base Model – The Prisoners' Dilemma

In this section, we consider the base model (defined by Assumption 3.1). Our objective is to understand the impact of collaboration and the loss sharing partnership on the project performance in both time and cost.

Assumption 3.1 *At level 1 of the project network, there is only one task. Each task cannot be expedited but can be delayed by at most one period. If the project is delayed, it is subject to a penalty which is time independent.*

In this model, the supplier and manufacturer only have two options (actions) available: "keep" (keeping the original task duration) or "delay" (delaying it by one period). We use K for "keep" and D for "delay" for simplicity. We assume that firm i is responsible for task i for $i = 0, 1$ where firm 1 (or 0) refers to the supplier (or manufacturer, respectively). The action set, [supplier's action, manufacturer's action], is {[K, K], [D, D], [K, D], [D, K]}. When task i is delayed, firm i receives a saving of s_i in terms of its direct cost. When the project is delayed, a penalty of p per period in terms of the indirect cost is shared by the firms, where firm i pays p_i and $p_0 + p_1 = p$.

Recall that, by assumption, the project starts with an original schedule that is optimal under the centralized control. In other words, the action set [K, K] has a pay-off higher than those under [D, K], [K, D] and [D, D] for the project as a whole. To this end, we need the following necessary condition,

Condition 3.1 *Global Optimum - Base Model:* $s_1 < p$, $s_0 < p$.

We can easily verify Condition 3.1 as follows: at [K, K], there is neither a saving nor a penalty for the project, and thus the pay-off of the project relative to the original schedule is zero. At [D, K], task 1 is delayed by one period but task 0 is kept at its original duration. Thus, we receive a saving of s_1 from task 1 but must pay a penalty of p because the project is delayed by one period. The pay-off of the project is $s_1 - p$ and thus $s_1 < p$ is a necessary condition for [K, K] to outperform [D, K] from the project's perspective. Repeating a similar logic to [K, D] and [D, D] leads to Condition 3.1.

Now we are ready to study the firms' strategic behaviors under the loss sharing partnership and their impact on project performance. Before introducing the general theory, we first present an example (see Figure 3.5) to illustrate the key idea and insight. In this example, task 1 has an original duration of 9 weeks, which can be delayed to 10 weeks with a saving of $s_1 = \$900$. Task 0 has an original duration of 5 weeks which can be delayed to 6 weeks with a saving of $s_0 = \$1200$. The project is due in 14 weeks; each week of delay incurs a penalty of $p = \$1600$ for the project. Clearly, Condition 3.1 is satisfied in the example, and so it is in the project's best interests to keep the original schedule.

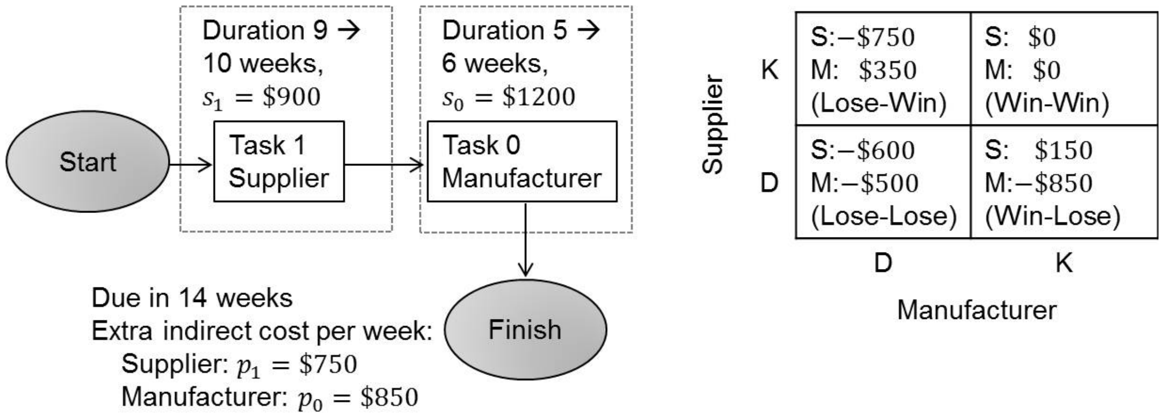


Figure 3.5: An example of the base Model and its pay-off matrix. (K:keep, D:delay)

Under the lost sharing partnership, we assume that upon each week of the project's delay, the supplier's share of the penalty is $p_1 = \$750$ and the manufacturer's share is $p_0 = \$850$. To see what the supplier and the manufacturer would do in their own best interests (i.e., the equilibrium), we consider the following four scenarios:

- **Win-Lose**: firm 1 (the supplier) delays but firm 0 (the manufacturer) keeps its original task duration. In this scenario, firm 1 saves \$900 but must pay \$750 with a net gain of \$150. However, firm 0 must pay \$850 for firm 1's delay. The firms' pay-offs (relative to the original schedule) are $(\pi_1, \pi_0) = (150, -850)$ and the project's pay-off is $-\$700$.

- **Lose-Win**: firm 1 keeps its original task duration but firm 0 delays. In this scenario, firm 0 saves \$1200 but must pay \$850 with a net gain of \$350. However, firm 1 must pay \$750 for the delay caused by firm 0. The firms' pay-offs are $(-750, 350)$ and the project's pay-off is $-\$400$.

- **Lose-Lose**: both firms delay. In this scenario, the project is delayed by two weeks and the firms' pay-offs are $(-600, -500)$. This is the worst scenario for the project as a whole with a total loss of \$1100.

- **Win-Win**: both firms keep their original task duration. This is the best scenario for the project where both the firms and the project lose nothing with a pay-off of zero (relative to the original schedule).

Figure 3.5 summarizes the action sets and the corresponding pay-off matrix. We can see that no matter what the supplier does, the manufacturer's optimal strategy is always to "delay". In other words, "delay" is the dominant strategy for the manufacturer. Similarly, the supplier's best strategy is also to "delay" regardless of the manufacturer's response. Thus, although the "Win-Win" scenario has the best outcome for the project, it is unstable – each firm will find every excuse to delay. The "Lose-Lose" scenario, although having the worst outcome for the project, is the subgame perfect Nash equilibrium (SPNE), as in a typical Prisoners' Dilemma.

We now present the general theory for the base model. Note that in this game, the supplier leads and the manufacturer follows (see §3.3). If the project is finished on time, there is no penalty. For every period of the project delay, the supplier pays a penalty of p_1 and the manufacturer pays the rest which is p_0. The firm whichever delays obtains a saving from the direct cost of its own task. For example, if the supplier delays but the manufacturer keeps the original duration of its task, the supplier saves s_1 from its direct cost which brings its pay-off to be $s_1 - p_1$, and the manufacturer bears a pure penalty of p_0. Figure 3.6 shows the extensive form of the game in the base model.

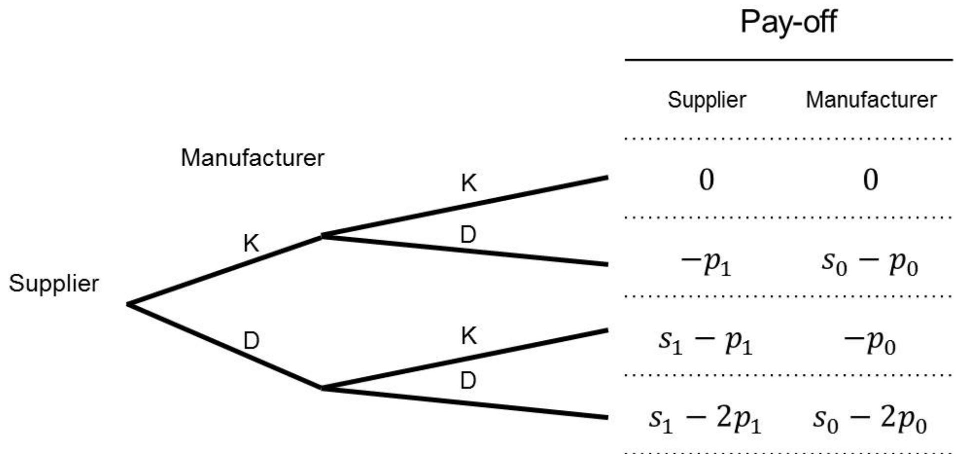


Figure 3.6: The extensive form of the game in the base model.

We derive the following results on the dominant strategies and equilibrium (all proofs of this chapter are presented in the last section unless otherwise mentioned).

Lemma 3.1 (Dominant Strategy): *Under Condition 3.1, when $s_i < p_i$, "keep" is the dominant strategy for firm i, $i = 0, 1$; when $s_i > p_i$, "delay" is the dominant strategy for firm i, $i = 0, 1$.*

For simplicity, we use "S" ("M") to denote the supplier (the manufacturer, respectively).

Theorem 3.1 (Equilibrium): *For the base model, under Condition 3.1, the subgame perfect Nash equilibrium (SPNE) is given by,*

Case	*Condition on S*	*Condition on M*	*Optimal strategy for S*	*M's best response*
1	$s_1 < p_1$	$s_0 < p_0$	K	K
2	$s_1 > p_1$	$s_0 < p_0$	D	K
3	$s_1 < p_1$	$s_0 > p_0$	K	D
4	$s_1 > p_1$	$s_0 > p_0$	D	D

Based on these results, we present the following key insight for the base model under the loss sharing partnership:

The Prisoners' Dilemma: *In the base model, for a schedule to be optimal, we need* $s_1 < p, s_0 < p$ *(Condition 3.1). For the optimal schedule to be the SPNE under loss sharing, a much stronger condition is required, that is,* $s_1 < p_1$ *and* $s_0 < p_0$ *where* $p_1 + p_0 = p$*. Thus, if* $s_1 > p_1$ *and* $s_0 > p_0$ *but* $s_1 < p$ *and* $s_0 < p$*, then it is in each firm's best interests to delay although being on time benefits the entire project.*

3.4.2 The Base Model with Time-dependent Costs – The Supplier's Dilemma

In this section, we relax the "time-independent cost" assumption in the base model to study the impact of time-dependent penalty costs on the results, e.g., the dominant strategies, the Prisoners' Dilemma. We define the model by Assumption 3.2.

Assumption 3.2 *Assumption 3.1 holds here except that project delay penalties are time dependent.*

Let p^1 (or p^2) be the penalty for the 1^{st} (the 2^{nd}, respectively) period of project delay; and let p_i^1 and p_i^2) be the corresponding penalties shared by firm i, where $p_1^1 + p_0^1 = p^1$ and $p_1^2 + p_0^2 = p^2$. The assumption of starting with the optimal schedule and the assumptions of convex and increasing cost functions (see §3.3) mandate,

Condition 3.2 *(1) Global Optimum - Time-Dependent:* $s_1 < p^1$, $s_0 < p^1$. *(2) Monotonicity - Time-Dependent:* $p^1 < p^2$, $p_1^1 < p_1^2$, $p_0^1 < p_0^2$.

To see the impact of time-dependent penalty costs, we slightly modify the example in §3.4.1 (shown in Figure 3.5). In this modified example, everything remains the same except that (1) the saving per week for task 1 is reduced to $s_1 = \$600$ from \$900; (2) the second period delay penalty of the project, p^2, is increased to \$2500 from \$1600, where the supplier bears $p_1^2 = \$1100$ and the manufacturer bears $p_0^2 = \$1400$. Figure 3.7 depicts the modified example. Clearly, Condition 3.2 is satisfied in this example, and it is in the project's best interests to keep the original schedule.

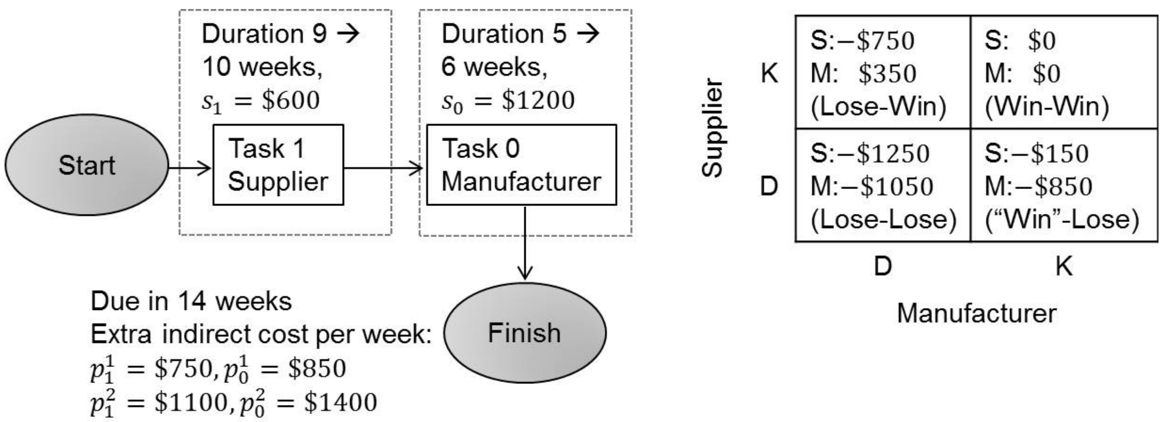


Figure 3.7: An example for the base model with time-dependent costs and its pay-off matrix. (K:keep, D:delay)

We consider the following four scenarios under the loss sharing partnership,

- **"Win"-Lose**: firm 1 (the supplier) delays but firm 0 (the manufacturer) keeps its original task duration. In this scenario, firm 1 saves \$600 but must pay \$750 with a net loss of \$150, while firm 0 must pay \$850. The firms' pay-offs (relative to the original schedule) are $(\pi_1, \pi_0) = (-150, -850)$ and the project's pay-off is $-\$1000$.

- **Lose-Win**: firm 1 keeps its original task duration but firm 0 delays. This scenario is identical to the "Lose-Win" scenario of the example in §3.4.1 with the firms' pay-offs being $(-750, 350)$ and the project's pay-off being $-\$400$.

- **Lose-Lose**: both firms delays. In this scenario, the project is delayed by two weeks and the firms' pay-offs are $(-1250, -1050)$. This is the worst scenario for the project as a whole with a total loss of \$2300.

- **Win-Win**: both firms keep. The firms' pay-offs are $(0, 0)$.

Figure 3.7 summarizes the action set and the pay-off matrix. Clearly, if the supplier (firm 1) keeps its original task duration, the manufacturer's best response is to "delay" because its saving exceeds its penalty of the **1st** period project delay. However, if the supplier delays, the manufacturer's best response is to "keep" its original task duration because now its penalty of the **2nd** period project delay exceeds its saving. Thus the supplier has to delay (even at a loss) to raise the penalty so high that the manufacturer would have to keep, to avoid a greater loss. We call such a phenomenon the "Supplier's Dilemma". It is easy to verify that the SPNE in this example is [D, K].

We now analyze the base model with time-dependent costs in general. We note that the only difference between this model and the base model in §3.4.1 is that when both firms delay, the delay penalty is $p_i^1 + p_i^2$ for firm i. Figure 3.8 shows the extensive form of the game between the supplier and the manufacturer.

We can derive the following results on the dominant strategies and equilibrium.

Lemma 3.2 (Dominant Strategy): *In the base model with time-dependent costs, under Condition 3.2, when $s_0 < p_0^1$, "Keep" is the dominant strategy for the manufacturer; when $s_0 > p_0^2$, "Delay" is the dominant strategy for the manufacturer.*

Theorem 3.2 (Equilibrium): *For the base model with time-dependent costs, under Condition 3.2, the subgame perfect Nash equilibrium is given by:*

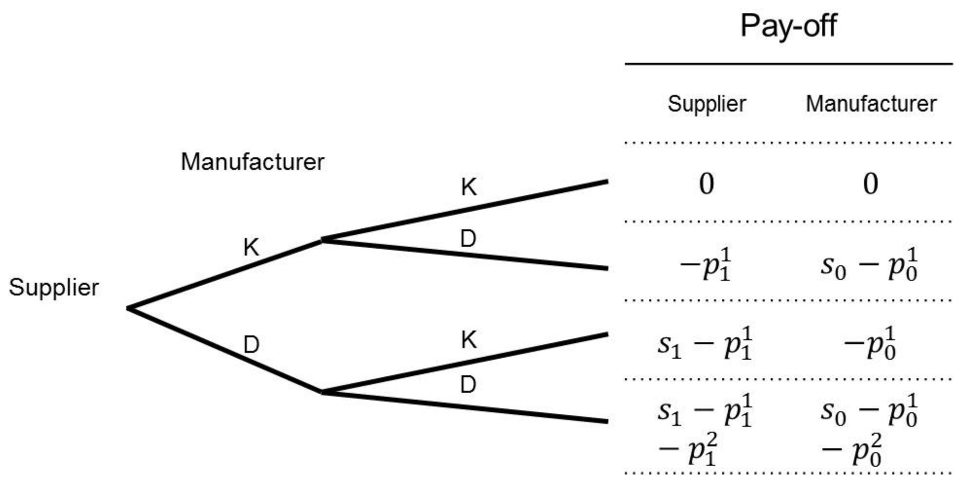


Figure 3.8: The extensive form of the game in the base model with time-dependent costs.

Case	*Condition on S*	*Condition on M*	*Optimal strategy for S*	*M's best response*
1	$s_1 < p_1^1$	$s_0 < p_0^1$	K	K
2	$s_1 > p_1^1$	$s_0 < p_0^1$	D	K
3		$p_0^1 < s_0 < p_0^2$	D	K
4	$s_1 < p_1^2$	$s_0 > p_0^2$	K	D
5	$s_1 > p_1^2$	$s_0 > p_0^2$	D	D

Theorem 3.2 is similar to Theorem 3.1 except for one new case (3rd case in Theorem 3.2): when $p_0^1 < s_0 < p_0^2$ (also illustrated in the example), the manufacturer's best strategy depends on the supplier's action. If the supplier keeps its original task duration, the manufacturer will delay; otherwise, the manufacturer will keep its original task duration. Thus, in this case, the supplier must take the manufacturer's response into account in making its own decision.

Based on these results, we present the following key insight for the base model with time-dependent costs under the loss sharing partnership:

The Supplier's Dilemma: *if $p_0^1 < s_0 < p_0^2$, the supplier has to delay (even at a loss) to raise the penalty too high for the manufacturer to delay, to avoid a greater loss.*

3.4.3 The Base Model with Expediting and Reward – The Coauthors' Dilemma

In this section, we relax the base model by allowing each firm an additional option: expediting by one period (see Assumption 3.3). With the new action of "expediting", the project could be completed earlier than the original schedule. The question is, will this happen in equilibrium under loss sharing?

Assumption 3.3 *Assumption 3.1 holds here except that each task can be expedited by at most one period, and there is a reward per period if the project is expedited.*

We use "E" to denote "expediting". Let c_0 (or c_1) be the cost of expediting (i.e., the additional direct cost) for task 0 (or 1, respectively). Let r be the reward for the project per period expedited, and r_0 and r_1 be rewards received by the firms where $r_1 + r_0 = r$. When a firm expedites, the pay-off functions are different from previous sections where firms cannot expedite. Specifically, if the supplier expedites, the action set [E, K] yields $-c_1 + r_1$ for the supplier and r_0 for the manufacturer, [E, D] yields $-c_1$ for the supplier and s_0 for the manufacturer, and [E, E] yields $-c_1 + 2r_1$ for the supplier and $-c_0 + 2r_0$ for the manufacturer. If the manufacturer expedites, the pay-off functions could be derived in a similar way.

As in all previous sections, we assume that the project starts with an original schedule that is optimal under the centralized control. To this end, Condition 3.3 (Global Optimum) provides a necessary condition. For instance, [E, K] should yield less profit for the entire project than [K, K], which requires $-c_1 + r_1 + r_0 < 0$, and [E, D] should yield less profit for the project than [K, K], which requires $s_0 < c_1$. Condition 3.3 (Monotonicity) comes from the assumption of convex and increasing indirect cost and convex and decreasing direct cost (see §3.3). Condition 3.3 (Loss Sharing) indicates that the monotonicity condition on the project's reward and penalty also applies to each firm's share of the reward and penalty.

Condition 3.3 *(1) Global Optimum - Expediting:* $s_1 < p$, $s_0 < p$; $r < c_1$, $r < c_0$; $s_1 < c_0$, $s_0 < c_1$. *(2) Monotonicity - Expediting:* $r < p$; $s_1 < c_1$, $s_0 < c_0$. *(3) Loss Sharing - Expediting:* $r_1 < p_1$, $r_0 < p_0$.

The extensive form of the game is shown in Figure 3.9. For instance, if the supplier expedites while the manufacturer keeps its original task duration, the supplier gets an award of r_1 but must pay an expediting cost of c_1; the manufacturer gets an award of r_0 without any cost.

We can derive the following results on the dominant strategies and equilibrium.

Lemma 3.3 (Dominant Strategy)*: In the base model with expediting and reward, under Condition 3.3, when* $s_i > p_i$*, "delay" is the dominant strategy for firm* i*,* $i = 1, 0$*; when* $s_i < r_i < p_i < c_i$*, "keep" is the dominant strategy for firm* i*,* $i = 1, 0$*.*

Lemma 3.3 differs from Lemma 3.1 on the conditions for "keep" because we must consider not only "delay" but also "expediting" in this model.

Theorem 3.3 (Equilibrium)*: For the base model with expediting and reward, under Condition 3.3, the subgame perfect Nash equilibrium is given by,*

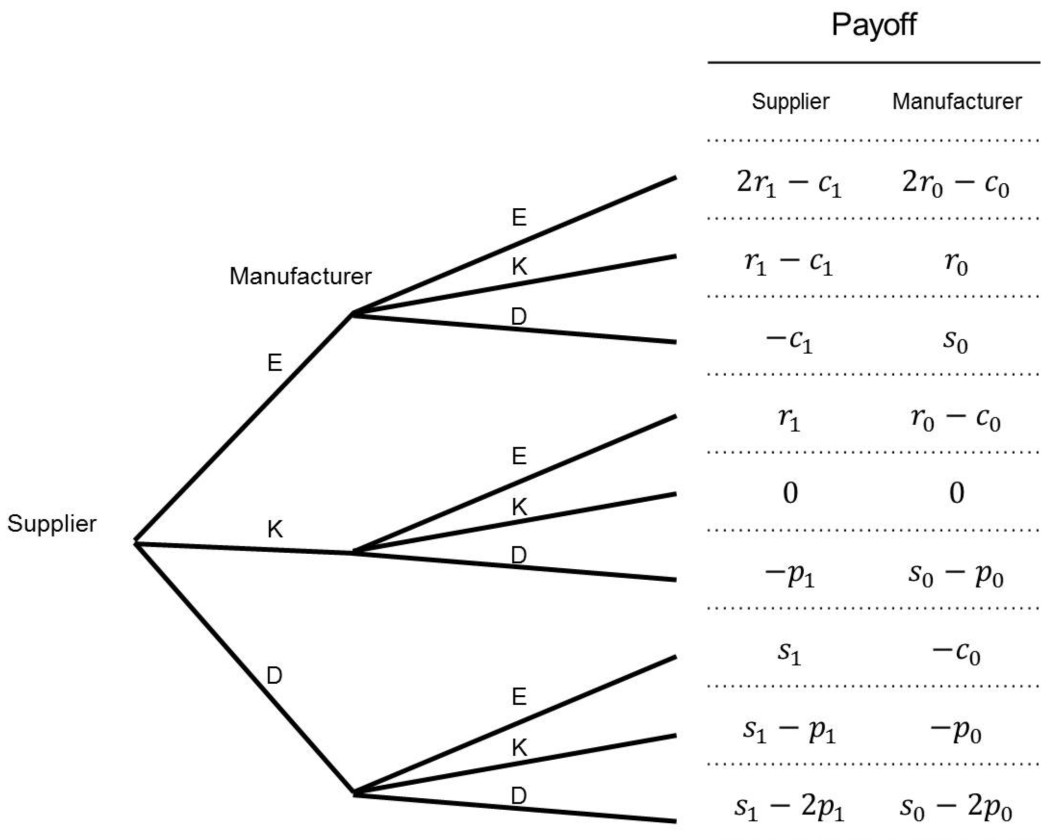


Figure 3.9: The extensive form of the game in the base model with expediting and reward.

Case	*Condition on S*	*Condition on M*	*Optimal strategy for S*	*M's best response*
1		$c_0 < p_0$	*D*	*E*
2	$s_1 < p_1$	$s_0 < p_0 < c_0$	*K*	*K*
3	$s_1 > p_1$	$s_0 < p_0 < c_0$	*D*	*K*
4	$c_1 < p_1$	$s_0 > p_0$	*E*	*D*
5	$s_1 < p_1 < c_1$	$s_0 > p_0$	*K*	*D*
6	$s_1 > p_1$	$s_0 > p_0$	*D*	*D*

Theorem 3.3 is similar to Theorem 3.1 except for the 1st and 4th cases that involve expediting and have equilibriums of [D, E] and [E, D]. We shall first explain the intuition

behind these two new cases and then discuss the other cases.

- **1st case,** $c_0 < p_0$**, [D, E] is the equilibrium**: In this case, the manufacturer faces a delay penalty that is greater than its expediting cost, and so it would do anything to prevent the project from being delayed. Taking advantage of the manufacturer's weakness, the supplier could delay regardless of its own cost structure, and earn a net saving without any penalty. Thus, even if the manufacturer expedites its task, the project will not be expedited because the supplier will delay.

  An example in the book publishing industry: Let's consider a coauthor and a lead author working sequentially on a textbook. The coauthor writes parts of the book and must pass on the manuscripts to the lead author to integrate and complete. The lead author is responsible for the delivery and is very concerned about the deadline. Thus the lead author will do anything possible to finish the book on time. Knowing this, the coauthor will delay as much as what the lead author can catch up without a penalty.

- **4th case,** $c_1 < p_1$ **and** $p_0 < s_0$**, [E, D] is the equilibrium**: In this case, "delay" is the dominant strategy for the manufacturer (by Lemma 3.3). In addition, the supplier faces a delay penalty that is greater than its expediting cost, and so the supplier will have to expedite to prevent the project from being delayed.

  An example in the academic thesis completion: Let's consider a PhD student and his/her advisor. The student shall write the PhD thesis and handle it over to the advisor to read and approve. The student needs to graduate and will do anything possible to complete his/her thesis on time. The advisor, on the other hand, is already established and much less concerned. Knowing the advisor to be bottleneck, the student has to work extra hard in the hope of getting the thesis done on time.

- **2nd case, $s_1 < p_1$ and $s_0 < p_0 < c_0$, [K, K] is the equilibrium**: In this case, the supplier cannot be better off by delaying, so it either keeps or expedites its task. If the supplier keeps, the manufacturer will also keep because either delaying or expediting will make itself worse off. If the supplier expedites, the manufacturer may delay or keep: delaying renders the supplier a pure expediting cost while keeping provides the supplier a reward, r_1, but still insufficient to cover its expediting cost because $r_1 < c_1$ by Condition 3.3 (Global Optimum). So the supplier would choose to keep.

- **3rd case, $s_1 > p_1$ and $s_0 < p_0 < c_0$, [D, K] is the equilibrium**: In this case, “delay” is the dominant strategy for the supplier (by Lemma 3.3). The manufacturer will choose to keep because either delaying or expediting makes itself worse off.

- **5th case, $s_1 < p_1 < c_1$ and $s_0 > p_0$, [K, D] is the equilibrium**: “delay” is the dominant strategy for the manufacturer (by Lemma 3.3). The supplier’s saving from “delay” is less than the delay penalty, which, in turn, is less than its expediting cost. This fact makes “keep” the best strategy for the supplier.

- **6th case, $s_1 > p_1$ and $s_0 > p_0$, [D, D] is the equilibrium**: “delay” is the dominant strategy for both firms.

Theorem 3.3 implies that in the base model with expediting and reward, the project will never be expedited in the equilibrium under the loss sharing partnership as compared to the optimal schedule. We summarize the results in this section by the following dilemma:

The Coauthors' Dilemma: *A firm can expedite its task but cannot expedite the project because if it expedites, the other will delay; if it delays, the other may or may not expedite.*

3.4.4 More Discussion on The Loss Sharing Condition

The intuition behind the loss sharing condition $r_0 < p_0$ and $r_1 < p_1$ (in Condition 3.3) is that a firm's share of the reward should be less than its share of the penalty, consistent to the monotonicity condition of the project, which is $r < p$. This condition is necessary for the Coauthor's Dilemma.

If we relax the loss sharing condition, then the Coauthor's Dilemma may not hold. Here is an example: assuming $r_0 > p_0$, we find that the [E,K] could be the equilibrium under certain conditions, and the project will be expedited and so the Coauthor's Dilemma no longer holds. Specifically, let's consider the case of$p_0 < s_0 < r_0$, where $p_1 > c_1 - r_1$ and $p_1 > (s_1 + c_1 - r_1)/2$ are satisfied, [E,K] turns out to be the equilibrium. We refer the readers to the proof of Theorem 3.3 (in appendix) for technical details.

The intuition behind this case is that if the supplier expedites, the manufacturer would keep because the saving from direct cost is less than the possible reward from expediting and both are less than the expediting cost; if the supplier keeps, the manufacturer would delay because penalty from indirect cost is less than the saving from direct cost; if the supplier delays, the manufacturer will delay to reduce loss. Since the supplier's loss from expediting is less than its delay penalty, it would expedite to avoid more loss.

3.4.5 The Base Model with Multiple Suppliers – The Worst Supplier Dominance

In this section, we extend the base model to include two suppliers at level 1 to study the impact of the project network. The analysis of a N-supplier system is similar. The model is defined in Assumption 3.4 where suppliers play a simultaneous game among themselves anticipating the manufacturer's response to their aggregated actions.

Assumption 3.4 *Assumption 3.1 holds here except that level 1 has two tasks each conducted by a unique supplier, and the manufacturer can only start its task after both suppliers complete their work.*

We denote supplier 1 (2)'s saving in the direct cost from delay to be s_1 (s_2) per period. The project penalty shared by the supplier 1 (or 2) is p_1 (or p_2 respectively) where $p_1 + p_2 + p_0 = p$. A necessary condition for the original schedule to be optimal under the centralized control is provided as follows,

Condition 3.4 *Global Optimum - Two Suppliers:* $s_1 + s_2 < p$, $s_0 < p$.

Without the loss of generality, we assume that the original durations of tasks 1 and 2 are identical (otherwise, the system reduces to the base model as we can ignore the supplier with a shorter duration). The same assumption applies to systems with more than two suppliers which will be discussed later in the chapter.

The extensive form of the game is shown in Figure 3.10.

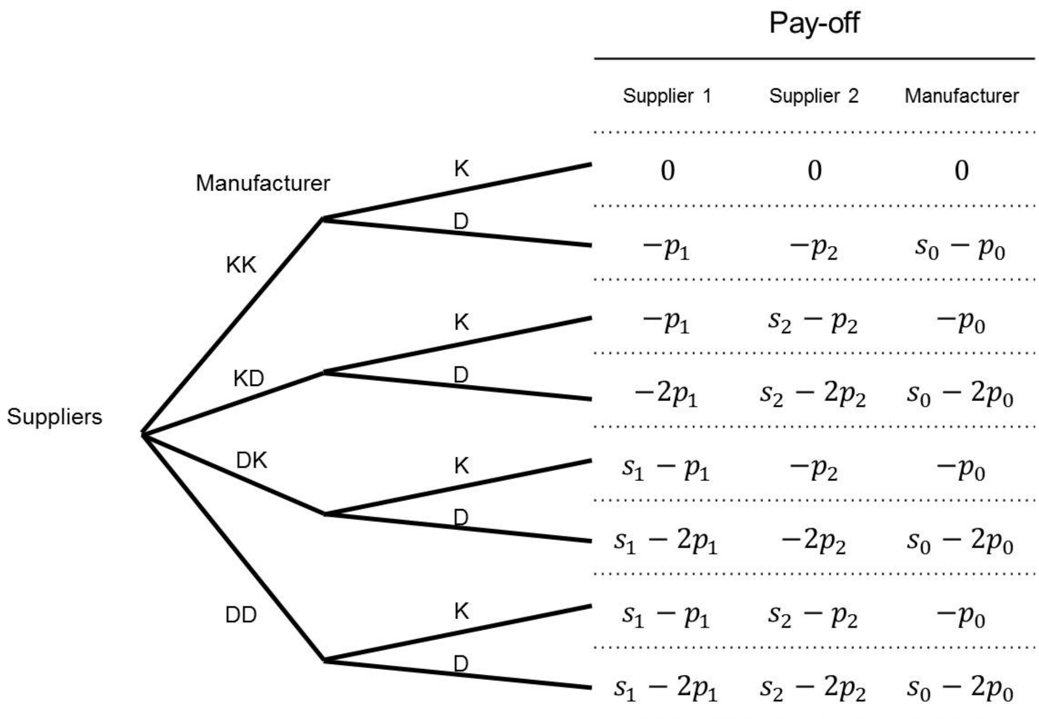


Figure 3.10: The extensive form of the game in the base model with multiple suppliers.

We have the following results on the dominant strategies and equilibrium.

Lemma 3.4 (Dominant Strategy): *In the base model with two suppliers, under Condition 3.4, when $s_0 < p_0$, "keep" is the dominant strategy for the manufacturer; when $s_0 > p_0$, "delay" is the dominant strategy for the manufacturer. When $s_i > p_i$, "delay" is the dominant strategy for supplier i.*

Lemma 3.4 differs from Lemma 3.1 because of the assembly-like structure at level 1 – there is no unilateral condition for a supplier to keep the original duration of its task as the level 1's on time performance depends on both suppliers' actions.

Theorem 3.4 (Equilibrium): *For the base model with two suppliers, under Condition 3.4, the subgame perfect Nash equilibrium is given by,*

Case	*Condition on S*	*Condition on M*	*Optimal strategy for S1, S2*	*M's best response*
1	$s_1 < p_1$ *and* $s_2 < p_2$	$s_0 < p_0$	*K, K*	*K*
2	$s_1 > p_1$ *or* $s_2 > p_2$	$s_0 < p_0$	*D, D*	*K*
3	$s_1 < p_1$ *and* $s_2 < p_2$	$s_0 > p_0$	*K, K*	*D*
4	$s_1 > p_1$ *or* $s_2 > p_2$	$s_0 > p_0$	*D, D*	*D*

Remarks: With two suppliers, the SPNE is no longer unique due to the simultaneous game played among the suppliers in level 1. For instance, when $s_0 < p_0$, the manufacturer keeps its original task duration, and the pay-off matrix for suppliers 1 and 2 is given by:

1\2	K	D
K	$0, 0$	$-p_1, s_2 - p_2$
D	$s_1 - p_1, -p_2$	$s_1 - p_1, s_2 - p_2$

Clearly, if $s_1 < p_1$ and $s_2 < p_2$, both [K, K] and [D, D] are SPNE. We only report [K, K] here because it is Pareto optimal but [D, D] is not.

Theorem 3.4 illustrates the impact of the project network on the equilibrium and project performance, that is, the project is more likely to be delayed with multiple suppliers. For the original schedule to be the SPNE, we require $s_1 < p_1$ and $s_2 < p_2$ (i.e., penalty exceeds saving for both suppliers) and $s_0 < p_0$. If the saving exceeds penalty for any supplier, all suppliers will have to delay in equilibrium. This observation gives rise to the following key insight:

The Worst Supplier Dominance: if one supplier delays, the other supplier(s) have to follow.

Extension to Multiple Suppliers with Expediting and Reward

We now extend the discussion to the scenario with expediting and reward. Based on Assumption 3.4, we define the model in Assumption 3.5.

Assumption 3.5 *Assumption 3.4 holds here except that each task can be expedited by at most one period, and there is a reward per period if the project is expedited.*

A necessary condition for the original schedule to be optimal under the centralized control is provided as follows,

Condition 3.5 *(1) Global Optimum - Two Suppliers and Expediting:* $s_1 + s_2 < p$, $s_0 < p$; $r < c_0$, $r < c_1 + c_2$; $s_1 + s_2 < c_0$, $s_0 < c_1 + c_2$. *(2) Monotonicity - Two Suppliers and Expediting:* $r < p$; $s_1 < c_1$, $s_2 < c_2$, $s_0 < c_0$. *(3) Loss Sharing - Two Suppliers and Expediting:* $r_1 < p_1$, $r_2 < p_2$, $r_0 < p_0$.

Theorem 3.5 (Equilibrium): *For the model based on Assumption 3.5 , under Condition 3.5, the subgame perfect Nash equilibrium is given by,*

Case	*Condition on S*	*Condition on M*	*Optimal strategy for S*	*M's best response*
1		$c_0 < p_0$	*DD*	*E*
2	$s_1 < p_1$ *and* $s_2 < p_2$	$s_0 < p_0 < c_0$	*KK*	*K*
3	$s_1 > p_1$ *or* $s_2 > p_2$	$s_0 < p_0 < c_0$	*DD*	*K*
4	$c_1 < p_1$ *and* $c_2 < p_2$	$s_0 > p_0$	*EE*	*D*
5	($s_1 < p_1$ *and* $s_2 < p_2$) *and* ($c_1 > p_1$ *or* $c_2 > p_2$)	$s_0 > p_0$	*KK*	*D*
6	$s_1 > p_1$ *or* $s_2 > p_2$	$s_0 > p_0$	*DD*	*D*

Theorem 3.5 is similar to Theorem 3.3. We shall explain the intuition behind these cases.

- **1st case, $c_0 < p_0$, [D, D, E] is the equilibrium**: This corresponds to case 1 of Theorem 3.3 for a single supplier. The manufacturer faces a delay penalty that is greater than its expediting cost, and so it would do anything to prevent the project from being delayed. The only difference from case 1 of Theorem 3.3 is that here, both suppliers are taking advantage of the manufacturer's weakness.

- **2nd case, $s_1 < p_1$ and $s_2 < p_2$ and $s_0 < p_0 < c_0$, [K, K, K] is the equilibrium**: In this case, the suppliers cannot be better off by delaying, so each of them either keeps or expedites its own task. If their durations are kept, the manufacturer will also keep because either delaying or expediting will make itself worse off. If their durations are expedited, the manufacturer may delay or keep – delaying renders each supplier a pure expediting cost while keeping provides each supplier a reward, r_1, but still insufficient to cover the expediting cost because $r < c_1 + c_2$ by Condition 3.5 (Global Optimum). So the suppliers would choose to keep.

- **3rd case, ($s_1 > p_1$ or $s_2 > p_2$) and $s_0 < p_0 < c_0$, [D, D, K] is the equilibrium**: In this case, "delay" is the dominant strategy for the two suppliers, because at least one of them has a delay penalty less than the benefit that it gets from "delay". The manufacturer will choose to keep because either delaying or expediting makes itself worse off.

- **4th case, ($c_1 < p_1$ and $c_2 < p_2$ and $s_0 > p_0$, [E, E, D] is the equilibrium**: In this case, "delay" is the dominant strategy for the manufacturer as the delay penalty is less than the benefit that it gets from "delay". In addition, each supplier faces a delay penalty that is greater than its expediting cost, and so each will have to expedite to prevent the project from being delayed.

- **5th case, ($c_1 < p_1$ and $c_2 < p_2$ and ($c_1 > p_1$ or $c_2 > p_2$) and $s_0 > p_0$, [K, K, D] is the equilibrium**: This is the most complicated case. For the manufacturer, "delay" is the dominant strategy. For at least one supplier whose $c_i > p_i$, the expediting cost is higher than the delay penalty, and thus this supplier has no reason to expedite. $s_i < p_i$ implies that this supplier has no reason to delay as well. For the other supplier, we know it will at least keep the duration. The worst supplier will dominate the final decision. Thus, both suppliers will keep.

- **6th case, ($s_1 > p_1$ or $s_2 > p_2$) and $s_0 > p_0$, [D, D, D] is the equilibrium**: "delay" is the dominant strategy for all firms.

Compare 3.5 with Theorem 3.3, we observe that:

- The condition is much more demanding for [E, E, D] in multiple suppliers case than in the single supplier case. We have similar observations between [K,K,K] (or [K,K,D]) and [K,K] ([K,D], respectively).

- The worst case of the two suppliers dominates their actions. In the single supplier case, the supplier only takes care of its own payoff. However, in the multiple suppliers case, each supplier will take the other's action into account.

3.4.6 The General Model

In previous sections, we reveal many managerial insights from the base model and its extensions. In this section, we put all the extensions together into a general model where we also allow each firm to delay or expedite its task by multiple periods (see Assumption 3.6). The question is, do the results obtained from the special cases in previous sections (§3.4.1-3.4.5), especially the Coauthors' Dilemma, still hold in the general model? And how to compute the project schedule in equilibrium?

Assumption 3.6 *The system has multiple suppliers and one manufacturer; each task can be either expedited or delayed by multiple periods; the cost structure, including penalty, reward, saving and expediting costs, are time dependent.*

We first consider the system with a single supplier. For the ease of exposition, we define the strategy pair as (x_1, x_0) where x_1 (or x_0) is an integer and its absolute value represents the number of periods expedited or delayed by the supplier (the manufacturer, respectively) relative to the original schedule. A negative integer means expediting, a positive integer means delaying, and zero means keeping the original task duration.

In this game, the supplier is the first mover and takes an action x_1. Let's define the manufacturer's best response (to the supplier's action) to be $x_0^*(x_1)$. The project duration will therefore be changed by $x_1 + x_0^*(x_1)$. We use superscripts on s_i, c_i, r and p to index the associated periods. For example, if task i is delayed by two periods, then the total saving should be $s_i^1 + s_i^2$ where s_i^1 (s_i^2) is the saving from the 1$^{\text{st}}$ (2$^{\text{nd}}$) period of delay. if task i is expedited by two periods, then c_i^1 (c_i^2) is the cost for the 1$^{\text{st}}$ (2$^{\text{nd}}$) period of expediting. Lastly, we define $\pi_1(x_1, x_0)$ ($\pi_0(x_1, x_0)$) to be the pay-off function for the supplier (the manufacturer, respectively).

For this system, Condition 3.6 (Global Optimum) is necessary for the original schedule to be optimal under the centralized control; Condition 3.6 (Monotonicity) comes from the convex increasing indirect cost and convex decreasing direct cost; finally, Condition 3.6 (Loss Sharing) indicates that the monotonicity condition on project reward and penalty also applies to each player's reward and penalty.

Condition 3.6 *(1) Global Optimum - General:* $\sum_{i=0}^{n} \pi_i(x_1, \cdots, x_n, x_0) \leq 0$ *for any* x_i*,* $i = 0, 1, \ldots, n$*; (2) Monotonicity - General:* $r^k > r^{k+1}$*,* $p^k < p^{k+1}$*,* $s_i^k > s_i^{k+1}$*,* $c_i^k < c_i^{k+1}$ *for any positive integer* k *and any* $i = 0, 1, \ldots, n$*, and* $r^1 < p^1$*,* $s_i^1 < c_i^1$ *for any* $i = 0, 1, \ldots, n$*; (3) Loss Sharing - General:* $r_i^k > r_i^{k+1}$*,* $p_i^k < p_i^{k+1}$ *and* $r_i^1 < p_i^1$ *for* $i = 0, 1, \ldots, n$*.*

We first characterize the pay-off function for the manufacturer for a given action of the supplier.

Lemma 3.5 *Given* $x_1 = a$*,* $\pi_0(a, x_0)$ *is a uni-modal function of* x_0*.*

Lemma 3.5 indicates that the manufacturer has a unique best response to each of the supplier's actions. The following lemma shows some monotonicity properties of the manufacturer's best response function.

Theorem 3.6 (Monotonicity Property)*: As* x_1 *increases,* $x_0^*(x_1)$ *is non-increasing but* $x_1 + x_0^*(x_1)$ *is non-decreasing.*

Lemma 3.6 implies that if the supplier delays more, the manufacturer will delay less, but the project will be delayed for a longer time.

In the case that the task duration is sufficiently long and so x_1 is effectively unbounded from below, the following theorem specifies a limit by which the supplier would expedite its task.

Theorem 3.7 (Expedition Limit): *There exists a* $x_L = \max\{x_1 | x_1 + x_0^*(x_1) = 0\} > -\infty$ *such that if* $x_1 \leq x_L$*, the supplier will be better off if it increases* x_1 *to* x_L*.*

Combining Theorems 3.6-3.7, we arrive at the following key insight,

Corollary 3.1 (The General Coauthor's Dilemma): *No matter by how much each firm expedites its task, the project will never be expedited in equilibrium under the loss sharing partnership.*

For the system with multiple suppliers, we define $x_s = \max\{x_1, \ldots, x_n\}$. We can show that Theorems 3.6-3.7 hold if we replace x_1 by x_s.

To numerically compute the equilibrium (the SPNE), we design an algorithm which enumerates x_1 between x_L and a pre-specified maximum allowable project delay, to find the optimal x_1^* for the supplier. Here is the key idea: we start by setting $x_1 = 0$. First, we search the region of $x_1 < 0$ until x_1 reaches x_L (if $x_L < 0$); second, we search the region of $x_1 > 0$ until we reach the maximum allowable project delay. We keep updating the best π_1 found to date and the corresponding x_1 and x_0, denoted by $(\pi_1^{\max}, x_1^*, x_0^*)$, until the enumeration is completed.

Let U be the maximum allowable project delay, the implementation details of this algorithm are described as follows:

Algorithm

- Step 1 - initialization: set $x_1 \leftarrow 0$. If $s_0^1 < p_0^1$, $x_0^*(0) \leftarrow 0$ otherwise $x_0^*(0)$ equals to i that satisfies $s_0^i > p_0^i$ and $s_0^{i+1} < p_0^{i+1}$. Initialize $\{\pi_1^{\max}, x_1^*, x_0^*\}$ with $\{\pi_1(0, x_0^*(0)), 0, x_0^*(0)\}$. Let $k \leftarrow x_0^*(0)$.

- Step 2 - search the region of $x_1 < 0$: $x_1 \leftarrow x_1 - 1$. Find $x_0^*(x_1)$ by comparing $\pi_0(x_1, k)$ and $\pi_0(x_1, k+1)$: if the former is greater, k remains; otherwise $k \leftarrow k+1$. Compute $\pi_1(x_1, k)$, and update $\{\pi_1^{\max}, x_1^*, x_0^*\}$ with $\{\pi_1(x_1, k), x_1, k\}$ if $\pi_1^{\max} < \pi_1(x_1, k)$. If $x_1 + k > 0$, repeat Step 2, otherwise reset $x_1 \leftarrow 0$, $k \leftarrow x_0^*(x_1)$ and go to Step 3.

- Step 3 - search the region of $x_1 > 0$: if $x_1 + k \leq U$, find $x_0^*(x_1')$ by comparing $\pi_0(x_1', k)$ and $\pi_0(x_1', k-1)$: if the former is greater, k remains the same; otherwise $k \leftarrow k - 1$. Compute $\pi_1(x_1, k)$, and update $\{\pi_1^{\max}, x_1^*, x_0^*\}$ with $\{\pi_1(x_1, k), x_1, k\}$ if $\pi_1^{\max} < \pi_1(x_1, k)$. If $x_1 + k > U$, stop and output the current $\{\pi_1^{\max}, x_1^*, x_0^*\}$.

3.5 The Fair Sharing Partnership

In this section, we present some provisions to enhance collaboration the basic form (i.e., collaboration under the loss sharing partnership); we call the resulting new partnership "fair sharing". The fair sharing partnership is designed to have each partner fully responsible for the consequence of its actions. In principle, if one firm causes damage to other firms, it has to compensate the others. Conversely, if one firm brings benefits to other firms, it shall receive compensations from the others. Our objective of this section is to specify the detailed sharing scheme in the fair sharing partnership for various project networks and cost structures so as to align each partner's best interest with that of the project. We shall first revisit the base model (see §3.4.1) in §3.5.1 to illustrate the key ideas, and then present a complete solution for the general model (see §3.4.6) in §3.5.2.

3.5.1 The Base Model Revisited

In this section, we specify the "fair sharing" partnership for the base model (defined by Assumption 3.1 in §3.4.1) according to the following principle: if firm i delays, it not only suffers its own share of the project delay penalty p_i, but also must reimburse firm j ($j \neq i$) her share of the penalty p_j due to firm i's delay. In this way, each firm is fully responsible for the penalty incurred by its delay. Specifically,

The Fair Sharing Scheme (The Base Model): *if both firms keep their original task duration, no payment is transferred. If only the supplier delays its task, the supplier not only suffers a penalty of p_1, but also pays the manufacturer p_0 to compensate her loss due to the supplier's delay. Similarly, if only the manufacturer delays its task, the manufacturer suffers a penalty of p_0 and must pay the supplier p_1, that is, the supplier's loss due to the manufacturer's delay. If both firms delay, each will compensate the other for the loss caused by its delay, that is, the supplier pays p_0 to the manufacturer and the manufacturer pays back the supplier p_1. In any event, if a firm delays, it will pay the full penalty p.*

The pay-off matrix is shown in Figure 3.11. It is obvious that the action set [K, K] is the SPNE under Condition 3.1 in §3.4.1. Thus fair sharing is capable of aligning individual firms' interests with that of the project in the base model.

Extension to Two Suppliers

The system with multiple suppliers complicates the fair sharing partnership. Let's consider the base model with two suppliers (defined by Assumption 3.4 in §3.4.5) and modify the above sharing scheme as follows.

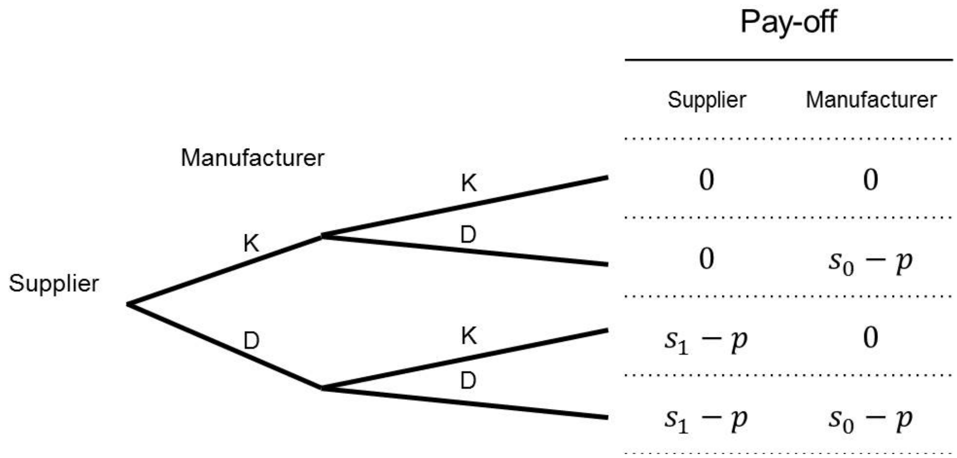


Figure 3.11: Extensive form of the game in base model under fair sharing.

The Fair Sharing Scheme (The Base Model With Two Suppliers): *if the manufacturer delays, it pays p which is the delay penalty of the project. Likewise, if one of the suppliers delays while the other keeps its original task duration, the delayed supplier pays p. If both suppliers delay, they split the penalty according to a rationing rule ($\beta_1 > 0$, $\beta_2 > 0$) where $\beta_1 + \beta_2 = 1$ and supplier 1 (2) pays $\beta_1 p$ ($\beta_2 p$).*

An analysis of the extensive form of the game (shown in Figure 3.12) reveals,

Theorem 3.8 *Consider the base model with two suppliers. Under the fair sharing partnership and Condition 3.4, the SPNE is to keep the original schedule (which is optimal under the centralized control) for any (β_1, β_2) as long as $\beta_1 > 0$, $\beta_2 > 0$ and $\beta_1 + \beta_2 = 1$.*

Note that Theorem 3.8 holds regardless of the value of $\beta_i, i = 1, 0$. Thus, the fair sharing partnership leaves the firms a flexibility in negotiating the contract.

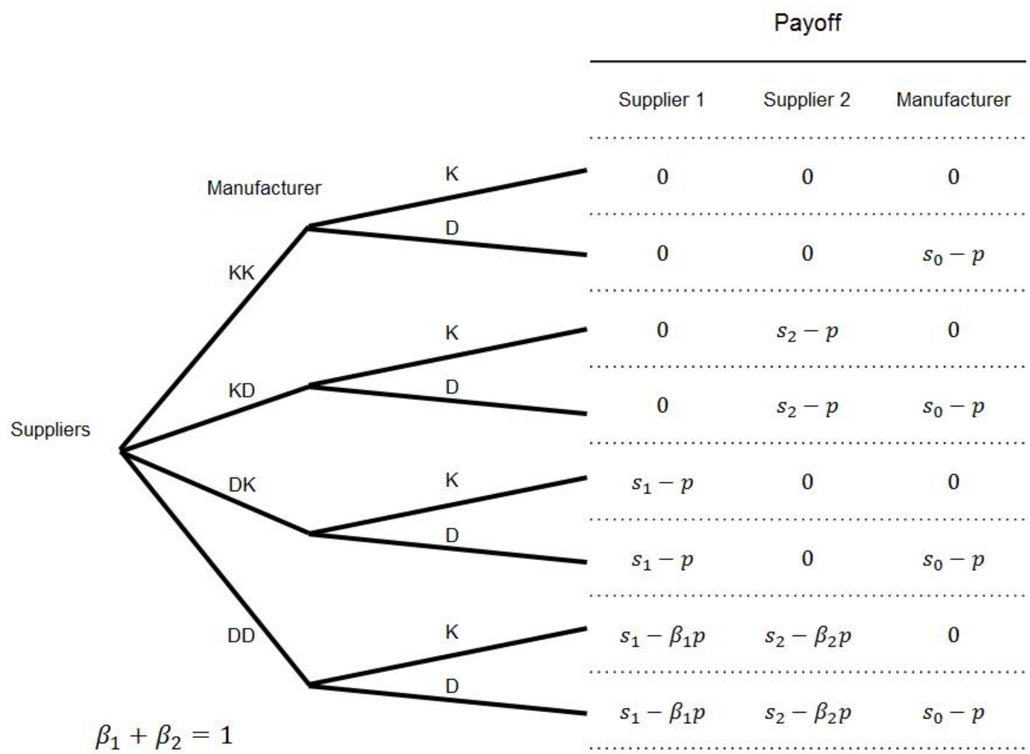


Figure 3.12: Extensive form of the game in base model with two suppliers under fair sharing.

3.5.2 The General Model Revisited

In this section, we present the details of the fair sharing partnership for the general model (defined by Assumption 3.6 in §3.4.6) and prove its effectiveness. Note that fair sharing can be seen as a way to redistribute the incremental indirect cost of the project (either reward or penalty) due to schedule changes among the firms. We denote this incremental indirect cost by B. Under fair sharing, B is distributed to levels 1 and 2 firms. Suppose that level 1 firms (the suppliers) get A_1 and the level 2 firm (the manufacturer) gets A_2, then $A_1 + A_2 = B$. For the ease of exposition, we also define $x_s = \max\{x_1, x_2, \ldots, x_N\}$ where x_s represents the change of level 1 completion date as compared to the original schedule. Using this notation, we specify the fair sharing scheme for the general model in two steps.

The Fair Sharing Scheme (The General Model):

- **Step 1**: *we decide the payment transferred between the two levels by allocating B to levels 1 and 2. If level 1 completion date is expedited by k periods ($x_s = -k$), level 1 firms shall be compensated by the rewards (i.e., savings in the indirect cost) for the project for the first k periods, that is, $A_1 = r^1 + r^2 + \ldots + r^k$. If level 1 completion date is delayed by k periods ($x_s = k$), level 1 firms shall pay the penalty (i.e., the additional indirect cost) for the project for the first k periods, that is, $A_1 = p^1 + p^2 + \ldots + p^k$. After the suppliers' allocation A_1 is determined, the manufacturer's allocation $A_2 = B - A_1$ accordingly.*

- **Step 2**: *we decide the payment transferred within level 1 firms by allocating A_1 among the suppliers. If level 1 completion date is expedited by k periods ($x_s = -k$), then each supplier must have expedited its task by at least k periods. A_1 is the reward and should be shared among all the suppliers. If level 1 completion date is delayed by k periods ($x_s = k$), then each supplier delays its task by at most k periods. A_1 is now the penalty and should be shared on a period-by-period basis among all delayed suppliers. For those suppliers who didn't delay in this case, they neither receive any reward nor share any penalty. More details are provided below.*

To see how Step 1 works, we provide an example:

- **Case 1**: If level 1 is expedited by 5 weeks but level 2 is delayed by 2 weeks, the project is therefore expedited by 3 weeks. Level 1 firms should be rewarded by $r^5, r^4, \ldots, r^1$, among which r^3, r^2, r^1 come from the project's earlier completion, but the rewards r^5 and r^4 are not materialized due to the delay at level 2, and so must be paid by the firm (the manufacturer) at level 2.

- **Case 2**: If level 1 is delayed by 5 weeks but level 2 is expedited by 2 weeks, the project is therefore delayed by 3 weeks. Level 1 firms must pay the penalties $p^1, p^2, \ldots, p^5$. However, p^4 and p^5 are not materialized by the level 2 firm's expedition, and so must be paid to the level 2 firm.

The general pay-off function of the manufacturer (the level 2 firm) is shown in Table 3.1.

Level 1	The manufacturer	Pay-off of the manufacturer
E: $x_s < 0$	E: $x_0 < 0$	$\sum_{i=\lvert x_s\rvert+1}^{\lvert x_s\rvert+\lvert x_0\rvert} r^i - \sum_{i=1}^{\lvert x_0\rvert} c_0^i$
	K: $x_0 = 0$	0
	D: $x_0 > 0$	$-\sum_{i=\lvert x_s+x_0\rvert+1}^{\lvert x_s\rvert} r^i + \sum_{i=1}^{\lvert x_0\rvert} s_0^i$, if $x_s + x_0 \leq 0$
		$-\sum_{i=1}^{\lvert x_s\rvert} r^i - \sum_{i=1}^{\lvert x_s+x_0\rvert} p^i + \sum_{i=1}^{\lvert x_0\rvert} s_0^i$, if $x_s + x_0 > 0$
K: $x_s = 0$	E: $x_0 < 0$	$\sum_{i=1}^{\lvert x_0\rvert} r^i - \sum_{i=1}^{\lvert x_0\rvert} c_0^i$
	K: $x_0 = 0$	0
	D: $x_0 > 0$	$-\sum_{i=1}^{\lvert x_0\rvert} p^i + \sum_{i=1}^{\lvert x_0\rvert} s_0^i$
D: $x_s > 0$	E: $x_0 < 0$	$\sum_{i=1}^{\lvert x_s\rvert} p^i + \sum_{i=1}^{\lvert x_s+x_0\rvert} r^i - \sum_{i=1}^{\lvert x_0\rvert} c_0^i$, if $x_s + x_0 < 0$
		$\sum_{i=\lvert x_s+x_0\rvert+1}^{\lvert x_s\rvert} p^i - \sum_{i=1}^{\lvert x_0\rvert} c_0^i$, if $x_s + x_0 \geq 0$
	K: $x_0 = 0$	0
	D: $x_0 > 0$	$-\sum_{i=\lvert x_s\rvert+1}^{\lvert x_s\rvert+\lvert x_0\rvert} p^i + \sum_{i=1}^{\lvert x_0\rvert} s_0^i$

Table 3.1: The pay-off function of the manufacturer under fair sharing in the general model.

To see how the reward or penalty is shared among the suppliers in Step 2, we show the pay-off functions of the suppliers in Table 3.2. In principle, each supplier is only responsible for the penalty of the periods delayed by itself, and it will not be rewarded if its expedition is not effective – does not lead to an expedition of level 1 completion date.

The three cases of Table 3.2 can be explained as follows:

- **Case 1:** $x_s < 0$. All suppliers share the expediting rewards of $|x_s|$ periods. The

Level 1	Supplier i	Pay-off of supplier i
E: $x_s < 0$	E: $x_i < 0$	$-\sum_{j=1}^{\lvert x_i\rvert} c_i^j + \alpha_i \sum_{j=1}^{\lvert x_s\rvert} r^j$
K: $x_s = 0$	E: $x_i < 0$	$-\sum_{j=1}^{\lvert x_i\rvert} c_i^j$
	K: $x_i = 0$	0
D: $x_s > 0$	E: $x_i < 0$	$-\sum_{j=1}^{\lvert x_i\rvert} c_i^j$
	K: $x_i = 0$	0
	D: $x_i > 0$	$\sum_{j=1}^{\lvert x_i\rvert} s_i^j - \sum_{j=1}^{\lvert x_s\rvert} \beta_i^j p^j$

Table 3.2: The pay-off function of suppliers under fair sharing in the general model. Note: (1) $\alpha_i > 0$ and $\sum_{i=1}^{N} \alpha_i = 1$. (2) $\beta_i^j = 0$ if $j > x_i$, otherwise, $\beta_i^j > 0$. (3) $\sum_{i=1}^{N} \beta_i^j = 1$ for all $j = 1, 2, \ldots, |x_s|$.

pay-off for supplier i is $\pi_i = -\sum_{j=1}^{|x_i|} c_i^j + \alpha_i \sum_{j=1}^{|x_s|} r^j$, where $\alpha_i > 0$ for $i = 1, \ldots, N$ and $\sum_{i=1}^{N} \alpha_i = 1$. Here α_i is supplier i's ration of the reward.

- **Case 2:** $x_s = 0$. If supplier i keeps its original task duration, its pay-off is $\pi_i = 0$; if supplier i expedites, its pay-off is $\pi_i = -\sum_{j=1}^{|x_i|} c_i^j$.

- **Case 3:** $x_s > 0$. If supplier i expedites, its pay-off is $\pi_i = -\sum_{j=1}^{|x_i|} c_i^j$; if it keeps, its pay-off is $\pi_i = 0$; if it delays, its pay-off is $\pi_i = \sum_{j=1}^{|x_i|} s_i^j - \sum_{j=1}^{|x_s|} \beta_i^j p^j$ where β_i^j is supplier i's ration for the penalty of the j^{th} period delayed. If $j > x_i$ (that is, this supplier does not contribute to the j^{th} period of delay), $\beta_i^j = 0$; otherwise $\beta_i^j > 0$ and β_i^j satisfies $\sum_{i=1}^{N} \beta_i^j = 1$ for all $j = 1, 2, \ldots, |x_s|$.

Under this sharing scheme, we have the following result.

Theorem 3.9 *In the general under the fair sharing scheme, "keep" for all firms is the unique SPNE.*

Theorem 3.9 implies that fair sharing is capable of aligning individual firms' interests with that of the project in the general model.

Extension: Starting from A Suboptimal Schedule

So far, we proved the effectiveness of the fair sharing partnership by assuming that the project starts from an original schedule that is optimal under the centralized control. An interesting question is, what happens if we relax this assumption and so the project starts from a suboptimal (or any) schedule?

When starting from an arbitrary schedule, the schedule in equilibrium may differ from the starting schedule under fair sharing. To see this, let's consider an example with a single supplier. Let $p^1 = 220, p^2 = 300, p^3 = 400, \ldots$, $s_i^1 = 250, s_i^2 = 200, \ldots$, and $s_0^1 = 280, s_0^2 = 200, \ldots$. Given such costs, the original schedule is clearly not optimal. In fact, keeping task 1's original duration but delaying task 2's duration by 1 week is the optimal schedule under the centralized control. It is easily to verify that the subgame perfect Nash equilibrium is $x_1 = 1$ and $x_0 = 0$ under fair sharing. Thus, when the project starts from an arbitrary schedule, such a schedule may not be the equilibrium schedule under fair sharing.

Interestingly, the equilibrium schedule will not worsen the project performance under fair sharing relative to the starting schedule. To see this, let's consider the suppliers first. By Table 3.2, a supplier could always choose "keep" in order to get a zero pay-off regardless of the actions of other suppliers and the manufacturer. This is true because fair sharing ensures that each partner is fully responsible for consequences of its actions and so a partner who choose to keep won't be penalized by damages caused by others. By a similar logic, the manufacturer can secure a zero pay-off by choosing "keep" even in such a second mover situation (see Table 3.1) regardless of the suppliers' actions. Suppose in equilibrium, a partner's optimal action is not "keep", then this partner must get a positive pay-off because otherwise, it can always choose "keep" to avoid a negative pay-off.

Proposition 3.1 *In the general model under fair sharing, if the project starts from an arbitrary schedule, all firms can not be worse off in their pay-offs in equilibrium.*

3.6 Conclusions

In this chapter, we consider collaborative partnerships in a two-level project management setting where the workload of the project is spread out to multiple firms (partners). We study the strategic behaviors of the firms under the loss sharing partnership in these joint projects by combing the economics/supply chain gaming models with project management specifics. This chapter highlights the negative impact of collaboration and the loss sharing partnership on the project performance in both time and cost by discovering exactly why and how they can hurt. We find an inherent mismatch between individual firms' best interests and that of the project. Depending on the project network and cost structure, a firm may be motivated to delay even if doing so harms the entire project (the Prisoners' Dilemma); a firm may have to delay (even at a loss) just to prevent the others from delaying, to avoid a much greater loss (the Suppliers' Dilemma); and no matter by how much a firm expedites its task, it cannot expedite the project because other firms will delay (the Coauthors' Dilemma). To resolve the incentive issue, we enhance the loss sharing partnership by a set of provisions with the principle of each firm being fully responsible for the consequences of its action. We present the exact form of the fair sharing partnership and prove its effectiveness in aligning the interests of individual firms with that of the project.

3.7 Appendix: Proofs and Technical Details

Proof of Lemma 3.1

For the supplier with $s_1 < p_1$, if the manufacturer chooses "keep", then $0 > s_1 - p_1$ and so the supplier will choose "keep"; if the manufacturer chooses "delay", then $-p_1 > s_1 - 2p_1$ so that the supplier will choose "keep" as well. Thus, the supplier has a dominant strategy of "keep" when $s_1 < p_1$. Similarly, we can prove that when $s_1 > p_1$, "delay" is the dominant strategy for the supplier.

For the manufacturer with $s_0 < p_0$, if the supplier chooses "keep", then $0 > s_0 - p_0$ and so the manufacturer will choose "keep"; if the supplier chooses "delay", then $-p_0 > s_0 - 2p_0$ so that the manufacturer will choose "keep" as well. Thus, the manufacturer has a dominant strategy of "keep" when $s_0 < p_0$. Similarly, we can prove that when $s_0 > p_0$, "delay" is the dominant strategy for the manufacturer. $\square$

Proof of Theorem 3.1

This theorem is a straightforward result of Lemma 3.1. $\square$

Proof of Lemma 3.2

For the manufacturer with $s_0 < p_0^1$, if the supplier chooses "keep", then $0 > s_0 - p_0^1$ and so the manufacturer will choose "keep"; if the supplier chooses "delay", then $-p_0^1 > s_0 - p_0^1 - p_0^2$ and so the manufacturer will choose "keep" as well. Thus, the manufacturer has a dominant strategy of "keep" when $s_0 < p_0^1$. Similarly, we can prove that when $s_0 > p_0^2$, "delay" is the dominant strategy for the manufacturer. $\square$

Proof of Theorem 3.2

Lemma 3.2 implies,

- when $s_1 > p_1^1$ and $s_0 < p_0^1$, the supplier has a dominant strategy of "delay" and the manufacturer has a dominant strategy of "keep".

- when $s_1 > p_1^2$ and $s_0 > p_0^2$, the supplier has a dominant strategy of "delay" and the manufacturer has a dominant strategy of "delay".

When $s_1 < p_1^1$ and $s_0 < p_0^1$, if the supplier chooses "keep", then the manufacturer will choose "keep" as $0 > s_0 - p_0^1$; if the supplier chooses "delay", then the manufacturer will choose "keep" as $-p_0^1 > s_0 - p_0^1 - p_0^2$. The former strategy gives the supplier a higher pay-off (0) than the latter strategy $(s_1 - p_1^1)$ and thus the supplier will choose "keep" and then the manufacturer will choose "keep".

When $p_0^1 < s_0 < p_0^2$, if the supplier chooses "keep", then the manufacturer will choose "delay" as $s_0 > p_0^1$; if the supplier chooses "delay", then the manufacturer will choose "keep" as $s_0 < p_0^2$. The latter strategy gives the supplier a higher pay-off $(-p_1^1)$ than the former strategy $(s_1 - p_1^1)$ and thus the supplier will choose "delay" and then the manufacturer will choose "keep".

When $s_1 < p_1^2$ and $s_0 > p_0^2$, the manufacturer has the dominant strategy of "delay". Since $-p_1^1 > s_1 - p_1^1 - p_1^2$, the supplier will choose "keep". □

Proof of Lemma 3.3

When $s_0 > p_0$, we know that $s_0 > r_0$ and $r_0 < c_0$ from Condition 3.3. If the supplier chooses "expediting" or "keep", the manufacturer always gets the highest pay-off if it delays. If the supplier chooses "delay", because $p_0 < s_0 < c_0$, "delay" yields the highest pay-off for the manufacturer. Thus, the manufacturer has a dominant strategy of "delay" in this scenario. Similarly, we can prove that the supplier has a dominant strategy of "delay" when $s_1 > p_1$. By a similar analysis, we could prove that when $s_i < r_i < p_i < c_i$, "keep" is the dominant strategy for firm i, $i = 1, 0$. □

Proof of Theorem 3.3

All potential actions are listed below:

<table>
<tr><th>S</th><th>M</th><th>M’s Pay-off</th><th colspan="2">Conditions</th><th>M’s Best Response</th><th>S’s Pay-off</th></tr>
<tr><td rowspan="3">E</td><td>E</td><td>$2r_0 - c_0$</td><td colspan="2"></td><td></td><td></td></tr>
<tr><td>K</td><td>r_0</td><td colspan="2">if $r_0 > s_0$</td><td>K</td><td>$r_1 - c_1$</td></tr>
<tr><td>D</td><td>s_0</td><td colspan="2">if $r_0 < s_0$</td><td>D</td><td>$-c_1$</td></tr>
<tr><td rowspan="3">K</td><td>E</td><td>$r_0 - c_0$</td><td colspan="2"></td><td></td><td></td></tr>
<tr><td>K</td><td>0</td><td colspan="2">if $p_0 > s_0$</td><td>K</td><td>0</td></tr>
<tr><td>D</td><td>$s_0 - p_0$</td><td colspan="2">if $p_0 < s_0$</td><td>D</td><td>$-p_1$</td></tr>
<tr><td rowspan="3">D</td><td>E</td><td>$-c_0$</td><td rowspan="2">if $p_0 > s_0$</td><td>if $p_0 > c_0$</td><td>E</td><td>s_1</td></tr>
<tr><td>K</td><td>$-p_0$</td><td>if $p_0 < c_0$</td><td>K</td><td>$s_1 - p_1$</td></tr>
<tr><td>D</td><td>$s_0 - 2p_0$</td><td colspan="2">if $p_0 < s_0$</td><td>D</td><td>$s_1 - 2p_1$</td></tr>
</table>

- When $p_0 < s_0$, “delay” is the dominant strategy for the manufacturer by Lemma 3.3. The supplier’s pay-off is $-c_1$ with “expediting”, $-p_1$ with “keep”, and $s_1 - 2p_1$ with “delay”. We consider three cases:
  - (a) When $p_1 > c_1$, the supplier’s optimal strategy is “expediting” because $c_1 > s_1$ by Condition 3.3(2) and so $-c_1$ is the largest payoff.
  - (b) When $s_1 < p_1 < c_1$, the supplier’s optimal strategy is “keep”.
  - (c) When $p_1 > c_1$, the supplier’s optimal strategy is “delay”.
- When $s_0 < p_0 < c_0$ and $r_0 > s_0$, “keep” is the dominant strategy for the manufacturer by Lemma 3.3. The supplier’s pay-off is $r_1 - c_1$ with “expediting”, 0 with “keep”, and $s_1 - p_1$ with “delay”. We consider two cases:
  - (a) When $p_1 > s_1$, the supplier’s optimal strategy is “keep” because $r_1 < c_1$ by Condition 3.3(1).
  - (b) When $p_1 < s_1$, the supplier’s optimal strategy is “delay” because $r_1 < c_1$.
- When $s_0 < p_0 < c_0$ and $r_0 < s_0$, there is no dominant strategy for the manufacturer. If the supplier chooses “expediting”, the manufacturer will choose “delay”.

If the supplier chooses “keep” or “delay”, the manufacturer will choose “keep”. Thus, the supplier’s pay-off is $-c_1$ with “expediting”, 0 with “keep”, and $s_1 - p_1$ with “delay”.

  - (a) When $p_1 > s_1$, the supplier’s optimal strategy is “keep”.
  - (b) When $p_1 < s_1$, the supplier’s optimal strategy is “delay”.

- When $p_0 > c_0$ and $r_0 > s_0$, by $c_0 > s_0$ (Condition 3.3(2)) we obtain $p_0 > s_0$. If the supplier chooses “expediting”, the manufacturer will choose “keep”. If the supplier chooses “keep”, the manufacturer will choose “keep”. If the supplier chooses “delay”, the manufacturer will choose “expediting”. (Note: the manufacturer will do whatever it could to prevent project delay.) Given the manufacturer’s optimal response, the supplier’s pay-off is $r_1 - c_1$ with “expediting”, 0 with “keep”, and s_1 with “delay”. Since $r_1 < c_1$ by Condition 3.3(1), the supplier’s optimal strategy is “delay”.

- When $p_0 > c_0$ and $r_0 < s_0$, by $c_0 > s_0$ (Condition 3.3(2)) we obtain $p_0 > s_0$. If the supplier chooses “expediting”, the manufacturer will choose “delay”. If the supplier chooses “keep”, the manufacturer will choose “keep”. If the supplier chooses “delay”, the manufacturer will choose “expediting”. (Note: the manufacturer will do whatever he could to prevent delay.) Given the manufacturer’s optimal response, the supplier’s pay-off is $-c_1$ with “expediting”, 0 with “keep”, and s_1 with “delay”. Clearly, the supplier’s optimal strategy is “delay”.

Summarizing all cases, we have proved the theorem. □

Proof of Lemma 3.4

By Lemma 3.1, the first two results are immediate, that is, when $s_0 < p_0$, "keep" is the dominant strategy for the manufacturer; when $s_0 > p_0$, "delay" is the dominant strategy for the manufacturer.

When $s_1 > p_1$, an enumerating over all options of supplier 2 and the manufacturer finds that supplier 1 archives the highest pay-off when it delays. $\square$

Proof of Theorem 3.4

By Lemma 3.4, as long as one of the suppliers has a dominant strategy of "delay", the other has to delay as well. Otherwise, it suffers a pure penalty. Combining the dominant strategies leads to the theorem. $\square$

Proof of Theorem 3.5

All potential actions are listed below:

<table>
<tr><th>S</th><th>M</th><th>M's Pay-off</th><th colspan="2">Conditions</th><th>M's Best Response</th><th>S's Pay-off</th></tr>
<tr><td rowspan="3">EE</td><td>E</td><td>$2r_0 - c_0$</td><td colspan="2"></td><td></td><td></td></tr>
<tr><td>K</td><td>r_0</td><td colspan="2">if $r_0 > s_0$</td><td>K</td><td>$r_1 - c_1$</td></tr>
<tr><td>D</td><td>s_0</td><td colspan="2">if $r_0 < s_0$</td><td>D</td><td>$-c_1$</td></tr>
<tr><td rowspan="3">KK</td><td>E</td><td>$r_0 - c_0$</td><td colspan="2"></td><td></td><td></td></tr>
<tr><td>K</td><td>0</td><td colspan="2">if $p_0 > s_0$</td><td>K</td><td>0</td></tr>
<tr><td>D</td><td>$s_0 - p_0$</td><td colspan="2">if $p_0 < s_0$</td><td>D</td><td>$-p_1$</td></tr>
<tr><td rowspan="3">DD</td><td>E</td><td>$-c_0$</td><td rowspan="2">if $p_0 > s_0$</td><td>if $p_0 > c_0$</td><td>E</td><td>s_1</td></tr>
<tr><td>K</td><td>$-p_0$</td><td>if $p_0 < c_0$</td><td>K</td><td>$s_1 - p_1$</td></tr>
<tr><td>D</td><td>$s_0 - 2p_0$</td><td colspan="2">if $p_0 < s_0$</td><td>D</td><td>$s_1 - 2p_1$</td></tr>
</table>

Note that due to "worst supplier dominance", we only consider the cases that the two suppliers take the same action.

- When $p_0 < s_0$, also by Assumption 3.5 $r_0 < p_0$ we have $r_0 < s_0$. "delay" is the dominant strategy for the manufacturer. In other words, the manufacturer will choose "delay" no matter what actions the suppliers take. By knowing this, the suppliers' payoff matrix is:

1\2	E	K	D
E	$-c_1$,$-c_2$	$-c_1 - p_1, -p_2$	$-c_1 - 2p_1, s_2 - 2p_2$
K	$-p_1, -c_2 - p_2$	$-p_1, -p_2$	$-2p_1, s_2 - 2p_2$
D	$s_1 - 2p_1, -c_2 - 2p_2$	$s_1 - 2p_1, -2p_2$	$s_1 - 2p_1, s_2 - 2p_2$

- (a) When $p_1 > c_1$ and $p_2 > c_2$, [E,E],[K,K],[D,D] are all Nash equilibrium, however, [E,E] is Pareto optimal Nash equilibrium while the other two are not.
- (b) When $p_1 > s_1$ and $p_2 > s_2$ and ($p_1 < c_1$ or $p_2 < c_2$), [K,K] and [D,D] are both Nash equilibriums. But only [K,K] is Pareto optimal Nash equilibrium.
- (c) When $p_1 < s_1$ or $p_2 < s_2$, only [D,D] is Nash equilibrium.

• When $s_0 < p_0 < c_0$ and $r_0 > s_0$, "keep" is the dominant strategy for the manufacturer. The suppliers' pay-off matrix is:

1\2	E	K	D
E	$r_1 - c_1$,$r_2 - c_2$	$-c_1, 0$	$-c_1 - p_1, s_2 - p_2$
K	$0, -c_2$	$0, 0$	$-p_1, s_2 - p_2$
D	$s_1 - p_1, -c_2 - p_2$	$s_1 - p_1, -p_2$	$s_1 - p_1, s_2 - p_2$

- (a) [E,E] cannot be the equilibrium because it requires $r_1 > c_1$ and $r_2 > c_2$ which violates the assumption.
- (b) When $p_1 > s_1$ and $p_2 > s_2$, [K,K] and [D,D] are both Nash equilibrium. However, only [K,K] is Pareto optimal Nash equilibrium.
- (c) When $p_1 < s_1$ or $p_2 < s_2$, [D,D] is the only Nash equilibrium.

• When $s_0 < p_0 < c_0$ and $r_0 < s_0$, there is no dominant strategy for the manufacturer. If the suppliers' duration is expedited, the manufacturer will delay. If the suppliers' duration is kept or delayed, the manufacturer will keep. The suppliers' pay-off matrix is:

$1\backslash 2$	E	K	D
E	$-c_1$,$-c_2$	$-c_1, 0$	$-c_1 - p_1, s_2 - p_2$
K	$0, -c_2$	$0, 0$	$-p_1, s_2 - p_2$
D	$s_1 - p_1, -c_2 - p_2$	$s_1 - p_1, -p_2$	$s_1 - p_1, s_2 - p_2$

- (a) When $p_1 > s_1$ and $p_2 > s_2$, [K,K] and [D,D] are both Nash equilibrium. However, [K,K] is Pareto optimal Nash equilibrium while [D,D] is not.
- (b) When $p_1 < s_1$ or $p_2 < s_2$, [D,D] is the only Nash equilibrium.

• When $p_0 > c_0$ and $r_0 > s_0$, by Condition 3.5 $c_0 > s_0$ we obtain $p_0 > s_0$. If the suppliers' duration is expedited, the manufacturer will choose "keep". If the suppliers' duration is kept, the manufacturer will choose "keep". If the suppliers choose "delay", the manufacturer will choose "expediting". (Note: the manufacturer will do whatever it could to prevent project delay.) Given the manufacturer's optimal response, the suppliers' pay-off matrix is:

$1\backslash 2$	E	K	D
E	$r_1 - c_1$,$r_2 - c_2$	$-c_1, 0$	$-c_1, s_2$
K	$0, -c_2$	$0, 0$	$0, s_2$
D	$s_1, -c_2$	$s_1, 0$	s_1, s_2

- (a) [E,E] cannot be the equilibrium because $r_1 > c_1$ and $r_2 > c_2$ will violate Condition 3.5.
- (b) [K,K] is not the equilibrium either due to $s_1 > 0$ and $s_2 > 0$.
- (c) [D,D] is the only Nash equilibrium.

• When $p_0 > c_0$ and $r_0 < s_0$, by Condition 3.5 $c_0 > s_0$, we obtain $p_0 > s_0$. If the suppliers' duration is expedited, the manufacturer will choose "delay". If the suppliers' duration is kept, the manufacturer will choose "keep". If the suppliers' duration is delayed, the manufacturer will choose "expediting". (Note:

the manufacturer will do whatever he could to prevent delay.) By knowing this, suppliers' pay-off matrix is:

1\2	E	K	D
E	$-c_1, -c_2$	$-c_1, 0$	$-c_1, s_2$
K	$0, -c_2$	$0, 0$	$0, s_2$
D	$s_1, -c_2$	$s_1, 0$	s_1, s_2

It is clear that [D,D] is the only Nash equilibrium.

Gathering all analysis above, we prove this theorem.

□

Proof of Lemma 3.5

We first prove the lemma for $a < 0$. When $a < 0$, we consider three cases:

(1) If $x_0 \leq 0$, $\pi_0(a, x_0) = r_0^1 + \ldots + r_0^{|a|+|x_0|} - c_0^1 - \ldots - c_0^{|x_0|}$.

(2) If $0 < x_0 \leq |a|$, $\pi_0(a, x_0) = r_0^1 + \ldots + r_0^{|a|-x_0} + s_0^1 + \ldots + s_0^{x_0}$.

(3) If $x_0 > |a|$, $\pi_0(a, x_0) = s_0^1 + \ldots + s_0^{x_0} - p_0^1 - \ldots - p_0^{a+x_0}$.

When $x_0 \in (-\infty, 0)$, $\pi_0(a, x_0)$ is an increasing function in x_0 because $r_0^{|x_0|} < r_0^1 < c_0^1 < c_0^{|x_0|}$ by Condition 3.6. At $x_0 = 0$, $\pi_0(a, 0) > \pi_0(a, -1)$ because $r_0^1 < c_0^1$. Thus, $\pi_0(a, x_0)$ is a monotonically increasing function of x_0 on $x_0 \in (-\infty, 0]$. Note that $\pi_0(a, 0) = r_0^1 + \ldots + r_0^{|a|}$. It is easy to show that when $x_0 \to +\infty$, $\pi_0(a, x_0) \to -\infty$. There always exists $x_0 = \hat{x}_0 \in [0, +\infty)$ that maximizes $\pi_0(a, x_0)$.

We now show that $\pi_0(a, x_0)$ is monotonically increasing in $(-\infty, \hat{x}_0]$ and monotonically decreasing in $[\hat{x}_0, +\infty)$. We discuss three scenarios:

1°, if $\hat{x}_0 = 0$, we have $\pi_0(a, 0) > \pi_0(a, 1)$, indicating that $r_0^{|a|} > s_0^1$. Since both $\{s_0^t\}$ and $\{r_0^t\}$ are decreasing series in t, we have $r_0^{|a|-1} > r_0^{|a|} > s_0^1 > s_0^2 \Rightarrow \pi_0(a, 1) > \pi_0(a, 2)$. By induction, we could prove that $\pi_0(a, x_0)$ is decreasing in $[0, +\infty)$.

2°, if $0 < \hat{x}_0 < |a|$, we have $\pi_0(a, \hat{x}_0) > \pi_0(a, \hat{x}_0 - 1)$ and $\pi_0(a, \hat{x}_0) > \pi_0(a, \hat{x}_0 + 1)$, indicating that $s_0^{\hat{x}_0} > r_0^{|a|-(\hat{x}_0-1)}$ and $r_0^{|a|-\hat{x}_0} > s_0^{\hat{x}_0+1}$. Furthermore, as both $\{s_0^t\}$ and $\{r_0^t\}$ are decreasing series in t, we have $s_0^{\hat{x}_0-1} > s_0^{\hat{x}_0} > r_0^{|a|-(\hat{x}_0-1)} > r_0^{|a|-(\hat{x}_0-2)}$ and $r_0^{|a|-\hat{x}_0} > s_0^{\hat{x}_0+1} > s_0^{\hat{x}_0+2}$, which lead to $\pi_0(a, \hat{x}_0 - 1) > \pi_0(a, \hat{x}_0 - 2)$ and $\pi_0(a, \hat{x}_0 + 1) > \pi_0(a, \hat{x}_0 + 2)$. We first consider the left side of $\hat{x}_0$ and show $\pi_0(a, x_0)$ is monotonically increasing in $[0, \hat{x}_0]$ by induction. The induction assumption is $\pi_0(a, x_0') > \pi_0(a, x_0' - 1)$ where $x_0' \in (0, \hat{x}_0)$. We have $\pi_0(a, x_0') > \pi_0(a, x_0' - 1) \Rightarrow s_0^{x_0'} > r_0^{|a|-x_0'+1} \Rightarrow s_0^{x_0'-1} > s_0^{x_0'} > r_0^{|a|-x_0'+1} > r_0^{|a|-x_0'+2} \Rightarrow \pi_0(a, x_0' - 1) > \pi_0(a, x_0' - 2)$. In addition, when $x_0' = 1$, we could show that $\pi_0(a, 1) > \pi_0(a, 0)$. Thus, $\pi_0(a, x_0)$ is monotonically increasing in $[0, \hat{x}_0]$. Similarly, we could prove that $\pi_0(a, x_0)$ is monotonically decreasing in $[\hat{x}_0, +\infty)$. Recall that $\pi_0(a, x_0)$ is a monotonically increasing function of x_0 on $x_0 \in (-\infty, 0]$, therefore $\pi_0(a, x_0)$ is a concave unimodal function with the peak $x_0 = \hat{x}_0$ when $0 < \hat{x}_0 < |a|$.

3°, if $\hat{x}_0 \geq |a|$, by a similar analysis, it is easy to prove that $\pi_0(a, x_0)$ is a concave unimodal function with the peak $x_0 = \hat{x}_0$.

In summary, $\pi_0(a, x_0)$ is a unimodal function of x_0 when $a < 0$.

Now, we discuss $a \geq 0$. Similarly, we have three cases:

(1) If $x_0 \leq -a$, $\pi_0(a, x_0) = r_0^1 + \ldots + r_0^{-a-x_0} - c_0^1 - \ldots - c_0^{-x_0}$.

(2) If $-a < x_0 \leq 0$, $\pi_0(a, x_0) = -p_0^1 - \ldots - p_0^{a+x_0} - c_0^1 - \ldots - c_0^{-x_0}$.

(3) If $x_0 > 0$, $\pi_0(a, x_0) = s_0^1 + \ldots + s_0^{x_0} - p_0^1 - \ldots - p_0^{a+x_0}$.

When $x_0 > 0$, $\pi_0(a, x_0)$ is a monotonically decreasing function of x_0 because $s_0^{x_0} < s_0^1 < p_0^1 < p_0^{a+x_0}$. Also $\pi_0(a, 0) = -p_0^1 - \ldots - p_0^a$. It is easy to show that when $x_0 \to +\infty$, $\pi_0(a, x_0) \to -\infty$. There always exists $x_0 = \hat{x}_0 \in (-\infty, 0]$ that maximizes $\pi_0(a, x_0)$.

We now show that $\pi_0(a, x_0)$ is monotonically increasing in $(-\infty, \hat{x}_0]$ and monotonically decreasing in $[\hat{x}_0, +\infty)$. We discuss three scenarios:

$1°$, if $\hat{x}_0 = 0$, we have $\pi_0(a, 0) > \pi_0(a, -1)$, indicating that $p_0^a < c_0^1$. Since both $\{p_0^t\}$ and $\{c_0^t\}$ are increasing series in t, we have $p_0^{a-1} < p_0^a < c_0^1 < c_0^2 \Rightarrow \pi_0(a, -1) > \pi_0(a, -2)$. By induction, we could prove that $\pi_0(a, x_0)$ is decreasing in $[0, +\infty)$.

$2°$, if $-a < \hat{x}_0 < 0$, we have $\pi_0(a, \hat{x}_0) > \pi_0(a, \hat{x}_0 - 1)$ and $\pi_0(a, \hat{x}_0) > \pi_0(a, \hat{x}_0 + 1)$, indicating that $p_0^{a+\hat{x}_0} < c_0^{-\hat{x}_0+1}$ and $p_0^{a+\hat{x}_0+1} > c_0^{-\hat{x}_0}$. Furthermore, as both $\{c_0^t\}$ and $\{p_0^t\}$ are increasing series in t, we have $c_0^{-\hat{x}_0+2} > c_0^{-\hat{x}_0+1} > p_0^{a+\hat{x}_0} > p_0^{a+\hat{x}_0-1}$ and $p_0^{a+\hat{x}_0+2} > p_0^{a+\hat{x}_0+1} > c_0^{-\hat{x}_0} > c_0^{-\hat{x}_0-1}$, which lead to $\pi_0(a, \hat{x}_0 - 1) > \pi_0(a, \hat{x}_0 - 2)$ and $\pi_0(a, \hat{x}_0 + 1) > \pi_0(a, \hat{x}_0 + 2)$. By induction, we could simply prove that $\pi_0(a, x_0)$ is monotonically increasing in $[-a, \hat{x}_0]$ and monotonically decreasing in $[\hat{x}_0, 0]$. Recall that $\pi_0(a, x_0)$ is a monotonically decreasing function of x_0 on $x_0 \in (0, +\infty]$, therefore $\pi_0(a, x_0)$ is a concave unimodal function with the peak $x_0 = \hat{x}_0$ when $-a < \hat{x}_0 < 0$.

$3°$, if $\hat{x}_0 \leq -a$, by a similar analysis, it is easy to prove that $\pi_0(a, x_0)$ is a concave unimodal function with the peak $x_0 = \hat{x}_0$.

In summary, $\pi_0(a, x_0)$ is a unimodal function of x_0 when $a \leq 0$.

In conclusion, for both cases of $a < 0$ and $a \geq 0$, we have proved that given $x_1 = a$, $\pi_0(a, x_0)$ is a uni-modal function of x_0. $\square$

Proof of Theorem 3.6

We first show that when $x_1 \to -\infty$, $x_0^*(x_1) > 0$ and $x_1 + x_0^*(x_1) < 0$.

- When $x_1 < 0$, the supplier expedites; the manufacturer will never expedite because a negative x_0 yields $r_0^{|x_1+x_0|} < r_0^1 < c_0^1$. Consider the manufacturer's response in three scenarios: (1) $x_0 < |x_1|$, (2) $x_0 = |x_1|$, $\pi_0(x_1, x_0) = s_0^1 + \ldots + s_0^{|x_1|}$.

(3) $x_0 > |x_1|$, $\pi_0(x_1, x_0) = s_0^1 + \ldots + s_0^{|x_0|} - p_0^1 - \ldots - p_0^{x_1+x_0}$.

- Scenario (1) yields the highest pay-off for the manufacturer when $x_1 \to -\infty$. Explanation: In scenario (3), when $x_1 \to -\infty$, $x_0 \to +\infty$ and thus $s_0^{x_0} \to 0$ and $p_0^{x_1+x_0} \to +\infty$. It is clear that scenario (2) yields a higher pay-off than scenario (3). Next, let $x_0 = |x_1| - 1$. $\pi_0(x_1, |x_1| - 1) = s_0^1 + \ldots + s_0^{|x_1|-1} + r_0^1$. When $x_1 \to -\infty$, $s_0^{|x_1|} < r_0^1$ and thus $\pi_0(x_1, |x_1| - 1) > \pi_0(x_1, |x_1|)$. Hence, when $x_1 \to -\infty$, $x_0^*(x_1) < |x_1|$. In other words, when $x_1 \to -\infty$, $x_0^*(x_1) > 0$ and $x_1 + x_0^*(x_1) < 0$.

Now we start from $x_1 \to -\infty$ and increase x_1 by one unit each time to see how $x_0^*(x_1)$ and $x_1 + x_0^*(x_1)$ will change.

When $x_1 \to -\infty$, $x_0^*(x_1) > 0$, $x_1 + x_0^*(x_1) < 0$, so that $\pi_0(x_1, x_0^*(x_1)) = s_0^1 + \ldots + s_0^{x_0^*(x_1)} + r_0^1 + \ldots + r_0^{|x_1+x_0^*(x_1)|}$. $x_0^*(x_1)$ being the best response requires conditions $\pi_0(x_1, x_0^*(x_1)) > \pi_0(x_1, x_0^*(x_1) - 1)$ and $\pi_0(x_1, x_0^*(x_1)) > \pi_0(x_1, x_0^*(x_1) + 1)$ which are equivalent to $s_0^{x_0^*(x_1)} > r_0^{|x_1+x_0^*(x_1)-1|}$ and $r_0^{|x_1+x_0^*(x_1)|} > s_0^{x_0^*(x_1)+1}$. Let $x_1' = x_1 + 1$, to find the manufacturer's best response, we compare the following pay-offs as(assuming $x_0^*(x_1) - 2 \geq 0$ and $|x_1 + x_0^*(x_1) + 1| > 0$):

(1) $\pi_0(x_1', x_0^*(x_1) - 2) = r_0^1 + \ldots + r_0^{|x_1+1+x_0^*(x_1)-2|} + s_0^1 + \ldots + s_0^{x_0^*(x_1)-2}$.

(2) $\pi_0(x_1', x_0^*(x_1) - 1) = r_0^1 + \ldots + r_0^{|x_1+1+x_0^*(x_1)-1|} + s_0^1 + \ldots + s_0^{x_0^*(x_1)-1}$.

(3) $\pi_0(x_1', x_0^*(x_1)) = r_0^1 + \ldots + r_0^{|x_1+1+x_0^*(x_1)|} + s_0^1 + \ldots + s_0^{x_0^*(x_1)}$.

(4) $\pi_0(x_1', x_0^*(x_1) + 1) = r_0^1 + \ldots + r_0^{|x_1+1+x_0^*(x_1)+1|} + s_0^1 + \ldots + s_0^{x_0^*(x_1)+1}$.

Because $r_0^{|x_1+1+x_0^*(x_1)-2|} < s_0^{x_0^*(x_1)} < s_0^{x_0^*(x_1)-1}$ and $r_0^{|x_1+1+x_0^*(x_1)|} > r_0^{|x_1+x_0^*(x_1)|} > s_0^{x_0^*(x_1)+1}$, we have $\pi_0(x_1', x_0^*(x_1)-2) < \pi_0(x_1', x_0^*(x_1)-1)$ and $\pi_0(x_1', x_0^*(x_1)) > \pi_0(x_1', x_0^*(x_1)+1)$. We can easily verify that when $x_0^*(x_1) - 2 = -1$ and $|x_1 + x_0^*(x_1) + 1| = 0$, these inequalities still hold. By the unimodality property of Lemma 3.5, $x_0^*(x_1) - 1 \leq x_0^*(x_1+1) \leq x_0^*(x_1)$. In other words, when x_1 increases one unit, the manufacturer's best response is to either reduce the corresponding $x_0^*(x_1)$ by one unit or keep it the same until $x_1 + x_0^*(x_1)$ reaches 0.

When x_1 gradually increases, we will encounter the following three scenarios: (1) $x_1 < 0$, $x_0^*(x_1) \geq 0$, $x_1 + x_0^*(x_1) \geq 0$; (2) $x_1 \geq 0$, $x_0^*(x_1) \geq 0$, $x_1 + x_0^*(x_1) \geq 0$; (3) $x_1 \geq 0$, and $x_0^*(x_1) < 0$, $x_1 + x_0^*(x_1) > 0$. In case (1), for $x_0^*(x_1)$ to be the best response, we require conditions $\pi_0(x_1, x_0^*(x_1)) > \pi_0(x_1, x_0^*(x_1) - 1)$ and $\pi_0(x_1, x_0^*(x_1)) > \pi_0(x_1, x_0^*(x_1) + 1)$ which are equivalent to $s_0^{x_0^*(x_1)} < p_0^{x_1 + x_0^*(x_1)}$ and $s_0^{x_0^*(x_1)+1} < p_0^{x_1 + x_0^*(x_1)+1}$. Similarly, for case (2), we have the same conditions as case (1): $s_0^{x_0^*(x_1)} < p_0^{x_1 + x_0^*(x_1)}$ and $s_0^{x_0^*(x_1)+1} < p_0^{x_1 + x_0^*(x_1)+1}$; for case (3), we have $c_0^{-x_0^*(x_1)} < p_0^{x_1 + x_0^*(x_1)+1}$ and $c_0^{-x_0^*(x_1)+1} > p_0^{x_1 + x_0^*(x_1)}$. In each case, we compare $\pi_0(x_1', x_0^*(x_1) - 2)$, $\pi_0(x_1', x_0^*(x_1) - 1)$, $\pi_0(x_1', x_0^*(x_1))$ and $\pi_0(x_1', x_0^*(x_1) + 1)$ in a way similar to the analysis in the previous paragraph, and reach the following statement for all cases: when x_1 increases one unit, the manufacturer's best response is to either reduce the corresponding $x_0^*(x_1)$ by one unit or keep it the same.

In summary, for $x_1 \in (-\infty, +\infty)$, when x_1 increases by one unit, $x_0^*(x_1)$ will either decrease by one unit or remain the same and therefore $x_1 + x_0^*(x_1)$ will not decrease. $\square$

Proof of Theorem 3.7

In the proof of Theorem 3.6, we have shown that when $x_1 \to -\infty$, $x_1 + x_0^*(x_1) < 0$. On the other hand, when $x_1 = 0$, $x_0^*(0)$ should be greater than or equal to 0 so as to guarantee original schedule to be the global optimum; $x_1 + x_0^*(x_1)$ is therefore greater than or equal to 0. From Theorem 3.6, we know that $x_1 + x_0^*(x_1)$ will increase by one unit or hold still each time when x_1 increases by one unit. There must exist x_L such that $x_L = \max\{x_1 | x_1 + x_0^*(x_1) = 0, x_1 \leq 0\}$.

For any $x_1 < x_L$ and $x_1 + x_0^*(x_1) \leq -1$, the supplier's pay-off is $\pi_1(x_1, x_0^*(x_1)) = r_1^1 + \ldots + r_1^{|x_1+1+x_0^*(x_1)|} + r_1^{|x_1+x_0^*(x_1)|} - c_1^1 - \ldots - c_1^{|x_1+1|} - c_1^{|x_1|}$. When the supplier expedites one period less, x_1+1, the manufacturer's best response is either $x_0^*(x_1)$ or $x_0^*(x_1)-1$ by Theorem 3.6. (1) If the manufacturer's best response is $x_0^*(x_1)$, then the supplier's pay-off is $r_1^1 + \ldots + r_1^{|x_1+1+x_0^*(x_1)|} - c_1^1 - \ldots - c_1^{|x_1+1|}$. Note that $r_1^{|x_1+x_0^*(x_1)|} \leq r_1^1 < c_1^1 < c_1^{|x_1|}$ by Condition 3.6(1), the supplier actually improves its pay-off by increasing x_1 to $x_1 + 1$. (2) If the manufacturer's best response is $x_0^*(x_1) - 1$, the supplier's pay-off is $r_1^1 + \ldots + r_1^{|x_1+1+x_0^*(x_1)-1|} - c_1^1 - \ldots - c_1^{|x_1+1|}$. The supplier also improves its pay-off. Therefore the supplier could continuously improve its pay-off by increasing x_1 until $x_1 + x_0^*(x_1) = -1$.

When $x_1 < x_L$ and $x_1 + x_0^*(x_1) = -1$, the supplier's pay-off is $r_0^1 - c_1^1 - \ldots - c_1^{|x_1+1|} - c_1^{|x_1|}$. If it expedites one period less, $x_1 + 1$, the manufacturer's best response is either $x_0^*(x_1)$ or $x_0^*(x_1) - 1$. The former one yields the supplier a pay-off of $-c_1^1 - \ldots - c_1^{|x_1+1|}$. Note that $r_1^1 < c_1^1 \leq c_1^{|x_1|}$. The supplier has a higher pay-off at $x_1 + 1$ than that at x_1. The latter one is the same as the case discussed in the previous paragraph which is shown that the supplier could improve its pay-off from x_1 to $x_1 + 1$. In other words, at $x_1 + x_0^*(x_1) = -1$, the supplier could also improve its pay-off until $x_1 + x_0^*(x_1) = 0$.

When $x_1 < x_L$ and $x_1 + x_0^*(x_1) = 0$, the supplier's pay-off is $-c_1^1 - \ldots - c_1^{|x_1|}$. If the supplier expedites one period less, as long as $x_1 + x_0^*(x_1)$ is still equal to 0, the supplier always gets its pay-off improved.

In summary, x_L always exists and for any $x_1 < x_L$, we have $\pi_1(x_1, x_0^*(x_1)) < \pi_1(x_L, x_0^*(x_L))$. $\square$

Proof of Theorem 3.8

It is obvious that the manufacturer has a dominant strategy of "keep". Using a backward induction, the pay-off matrix between supplier 1 and supplier 2 is:

$1\backslash 2$	K	D
K	$0, 0$	$0, s_2 - p$
D	$s_1 - p, 0$	$s_1 - \beta_1 p, s_2 - \beta_2 p$

By Condition 3.4, [D, K] or [K, D] cannot be the equilibrium because $s_1 < p$ and $s_2 < p$. [D, D] cannot be the equilibrium either because $s_1 - \beta_1 p$ and $s_2 - \beta_2 p$ cannot be larger than 0 at the same time, otherwise $s_1 + s_2 < p$ from Condition 3.4 is violated. We could verify that [K, K] is the only equilibrium. Note that we do not have to specify β_1 and β_2 completely. $\square$

Proof of Theorem 3.9

Not every supplier would like to expedite because $-\sum_{j=1}^{|x_i|} c_i^j + \alpha_i \sum_{j=1}^{|x_s|} r^j$ is not positive for every i, otherwise we violate the assumption that the original schedule is the optimal schedule. So $x_s \geq 0$ and thus no supplier would like to expedite. On the other hand, no supplier would like to delay because those suppliers who delayed have to share the penalty. By Condition 3.6, at least one of them is losing money. Because this fact applies to any group of suppliers who delay, no supplier would like to delay and so "Keep" is the dominant strategy for every supplier.

Knowing that suppliers will always keep, the manufacturer's pay-off is: (1) $\sum_{i=1}^{|x_0|} r^i - \sum_{i=1}^{|x_0|} c_0^i$ if it expedites $|x_0|$; (2) 0 if it keeps; (3) $-\sum_{i=1}^{|x_0|} p^i + \sum_{i=1}^{|x_0|} s_0^i$ if it delays. By Condition 3.6(1), the pay-offs in (1) and (3) are all less than 0. Thus, the best strategy for the manufacturer is "keep". $\square$

Chapter 4

Continuous Time and Stochastic Duration Models

In Chapter 3, we discussed the firms' strategic behaviors under loss sharing and fair sharing partnerships in a deterministic and discrete-time setting. In this chapter, we consider its extension in two directions: (1) a continuous time model, see Section 4.1; (2) a stochastic duration model, see Section 4.2. In Section 4.1, the key question is, will the Coauthor's Dilemma still hold when time is continuous and the cost functions are continuous in time? In Section 4.2, the key question is, what is the impact of stochastic task durations on the Prisoner's Dilemma?

4.1 Continuous Time Model on Loss Sharing Partnership

In this section, we assume that time is continuous and the cost functions are continuous in time. We no longer distinguish the notation between the expediting reward (the expediting cost) and the delay penalty (delay saving) for indirect cost (direct cost, respectively). We focus on loss sharing partnership as it is widely used in practice. Specifically, we define the following notation.

- T_i: the duration of each task i, $i = 0, 1, 2, ..., n$; $T_i \in [a_i, b_i]$ with a_i and b_i being finite positive numbers.

- $C_i(T_i)$: cost of task i, it represents the direct cost of task i, $i = 0, 1, 2, ..., n$.

- T: total project duration.
- $R(T)$: revenue of the project, it represents the indirect cost of the project.
- α_i: the share of revenue for task i ($i = 0, 1, 2, ..., n$).

Assumption 4.1 *We make the following assumption on the cost functions:*

1. *The manufacturer starts its work only after all the suppliers complete their tasks.*
2. $C_i(T_i)$ *is decreasing and convex in* T_i, *for* $i = 0, 1, 2, ..., n$.

Assumption 4.1 is consistent to our assumptions on cost structure and project network in §**??**. Part 2 of Assumption 4.1 is consistent with direct cost assumption commonly made in project management literature. Intuitively, the more we crash the task duration, the higher the cost per unit of time crashed. The cost functions under this assumption are very general, including constant and linear cost functions as special cases. By Assumption 4.1, the total project duration, T, is equal to $\max\{T_1, T_2, ..., T_n\} + T_0$.

Assumption 4.2 $R(T)$ *is decreasing and concave in* T; $\alpha_i \in (0, 1)$ *for* $i = 0, 1, ..., n$, *and* $\sum_{i=0,1,...,n} \alpha_i = 1$.

Assumption 4.2 is consistent to the indirect cost where it is typically assumed that the earlier the project is completed, the higher the reward or the less the penalty. The concavity of the revenue function comes from higher penalty cost per unit time as delay increases, as we often observe in practice. We should point out that this revenue function is very general, including fixed-price and linear incentive contract as special cases.

4.1.1 Global Optimal Solution – Centralized Case

In the centralized case where all tasks are done by one firm, the total pay-off of the project is given as follows:

$$\pi \ = \ R(T) - C_0(T_0) - \sum_{i=1}^{n} C_i(T_i). \tag{4.1}$$

To determine the optimal durations for tasks under centralized control, we first observe that the function $T = f(T_0, T_1, ..., T_n) = \max_{i=1,2,...,n}\{T_i\} + T_0$ is jointly convex in $(T_0, T_1, ..., T_n)$ (the proof is straightforward, we omit it). For simplicity, we define vector $\bar{T} = (T_0, T_1, ..., T_n)$; we can also rewrite $R(T) = R(f(\bar{T}))$.

Lemma 4.1 *$R(f(\bar{T}))$ is jointly concave in $\bar{T} = (T_0, T_1, ..., T_n)$.*

Proof. Consider $\bar{T} \neq \bar{T}'$ and $\beta \in [0, 1]$. By the joint convexity of $f(\bar{T})$ in $\bar{T}$, we must have

$$f(\beta\bar{T} + (1-\beta)\bar{T}') \leq \beta f(\bar{T}) + (1-\beta) f(\bar{T}').$$

In addition, $R(T)$ is decreasing in T which implies

$$R(f(\beta\bar{T} + (1-\beta)\bar{T}')) \geq R(\beta f(\bar{T}) + (1-\beta) f(\bar{T}')).$$

Finally, the concavity of $R(T)$ in T indicates

$$R(\beta f(\bar{T}) + (1-\beta) f(\bar{T}')) \geq \beta R(f(\bar{T})) + (1-\beta) R(f(\bar{T}')).$$

The joint concavity of $R(f(\bar{T}))$ in $\bar{T}$ follows by the last two inequalities. $\square$

Because $-C_0(T_0) - \sum_{i=1}^{n} C_i(T_i)$ is jointly concave in $\bar{T}$ and $R(f(\bar{T}))$ is also jointly concave in $\bar{T}$, π is a jointly concave function in $\bar{T}$. Thus, the optimal task durations $\bar{T}^* = (T_0^*, T_1^*, \cdots, T_n^*)$ must exist and be finite. Assuming that the feasible region of

T_i, $[a_i, b_i]$, is sufficiently large for all i, then there must exist a solution $\bar{T}^*$ such that $T_1^* = T_2^* = \cdots = T_n^*$. This is true because if for instance $T_1^* < T_2^*$, then increasing T_1^* to be equal to T_2^* would lower the cost of task 1 while keeping the everything else unchanged.

4.1.2 Subgame Perfect Nash Equilibrium – Decentralized Case

In the decentralized case, each firm makes its decision to maximize its pay-off. α_0 and α_i $(i = 1, 2, ..., n)$ are the pre-determined sharing ratio among the manufacturer and suppliers; $\alpha_0 + \sum_{i=1}^{n} \alpha_i = 1$. The manufacturer's pay-off is given by:

$$\pi_0 = \alpha_0 R(T) - C_0(T_0).$$

The supplier i's pay-off is given by:

$$\pi_i = \alpha_i R(T) - C_i(T_i).$$

By the structure of the project network, the suppliers move first. The manufacturer observes the duration from stage 1, the suppliers' stage, and decides its own action. As noted in Chapter 3, the suppliers play a simultaneous game among themselves and a sequential game together with the manufacturer.

Lemma 4.2 *For a given T_0, the suppliers play a simultaneous game. The subgame perfect Nash equilibrium exists. And in all possible equilibriums, suppliers must have identical task duration, i.e., $\widetilde{T}_1 = \widetilde{T}_2 = \cdots = \widetilde{T}_n = T_s$.*

Proof. Supplier i's pay-off function is $\pi_i = \alpha_i R(T) - C_i(T_i)$. By Lemma 4.1, π_i is concave in $\bar{T}$. Because the strategy space is compact and convex, there exists at least one subgame perfect Nash equilibrium.

Consider a subgame perfect Nash equilibrium $(\widetilde{T}_1, \widetilde{T}_2, \cdots, \widetilde{T}_n)$. Suppose $\widetilde{T}_1 > \widetilde{T}_2$, we can simply increase $\widetilde{T}_2$ to $\max_{i=1,\ldots,n}\{\widetilde{T}_i\}$ which will increase (or at least maintain) supplier 2's pay-off while keeping all other suppliers' pay-offs unchanged. Thus any Nash equilibrium must satisfy $\widetilde{T}_1 = \widetilde{T}_2 = \cdots = \widetilde{T}_n = T_s$. □

From this lemma, all suppliers have identical duration for their tasks in the equilibrium. Let $T_s = \widetilde{T}_1 = \widetilde{T}_2 = \cdots = \widetilde{T}_n$. We now consider the sequential game played between the suppliers as a whole and the manufacturer. Due to the sequential nature, the manufacturer decides its own duration T_0 after observing the suppliers' duration, T_s. Anticipating the manufacturer's response, the suppliers determine their best duration T_s upfront.

For the suppliers as a whole, the total pay-off π_s is given as follows:

$$\pi_s \;=\; (1-\alpha_0)R(T_s+T_0) - \sum_{i=1}^{n} C_i(T_s). \tag{4.2}$$

Each supplier has its pay-off function as follows:

$$\pi_i \;=\; \alpha_i R(T_s+T_0) - C_i(T_i). \tag{4.3}$$

According to Lemma 4.2, there should be at least one dominating supplier that holds the longest duration at stage 1. For all the other suppliers, they have no choice but accepting this longest duration because the worst supplier dominates the duration of stage 1.

For the manufacturer, its pay-off function is given by:

$$\pi_0 \;=\; \alpha_0 R(T_s+T_0) - C_0(T_0). \tag{4.4}$$

Given T_s, we define the manufacturer's response as,

$$g(T_s) = \arg\max_{T_0}\{\alpha_0 R(T_s+T_0) - C_0(T_0)\}. \tag{4.5}$$

The first order condition $\frac{\partial \pi_0}{\partial T_0} = 0$ results in

$$\alpha_0 R^{'}(T_s + T_0) - C_0^{'}(T_0) = 0,$$

and consequently $g(T_s)$ must satisfy

$$\alpha_0 R^{'}(T_s + g(T_s)) - C_0^{'}(g(T_s)) = 0 \tag{4.6}$$

Suppose supplier i dominates other suppliers, $T_s = T_i$. The first order condition for supplier i when $\frac{d\pi_i(T_i, g(T_i))}{dT_i} = 0$ results in

$$\alpha_i R^{'}(T_s + g(T_s))(1 + g^{'}(T_s)) - C_i^{'}(T_s) = 0 \tag{4.7}$$

For the manufacturer, its pay-off is $\pi_0(T_s, T_0) = \alpha_0 R(T_s + T_0) - C_0(T_0)$. For a given T_s, $\pi_0(T_s, T_0)$ is a concave function in T_0, thus there exists a unique $g(T_s)$ which maximizes the manufacturer's pay-off. For the dominating supplier i, $\pi_i(T_s) = \alpha_i R(T_s + g(T_s)) - C_i(T_s)$. Since T_s is defined in a finite region, there must exist a finite $\widetilde{T}_s$ that maximizes supplier i's pay-off $\pi_i(T_s)$. The pair, $(\widetilde{T}_s, \widetilde{T}_0 = g(\widetilde{T}_s))$, is the subgame perfect Nash equilibrium.

The following lemma characterizes $g(T_s)$.

Lemma 4.3 $-1 < g^{'}(T_s) < 0$, *the manufacturer's task duration is a decreasing function of* T_s.

Proof. Taking derivative of Eq. (4.6) with respect to T_s yields

$$\alpha_0 R^{''}(T_s + g(T_s))(1 + g^{'}(T_s)) = C_0^{''}(g(T_s))g^{'}(T_s).$$

Thus,

$$g^{'}(T_s) = \frac{\alpha_0 R^{''}(T_s + g(T_s))}{-\alpha_0 R^{''}(T_s + g(T_s)) + C_0^{''}(g(T_s))}. \tag{4.8}$$

Because $R^{''} < 0$ and $C_0^{''} > 0$, then $g^{'}(T_s) < 0$. Rewrite

$$g^{'}(T_s) = -1 + \frac{C_0^{''}(g(T_s))}{-\alpha_0 R^{''}(T_s + g(T_s)) + C_0^{''}(g(T_s))},$$

Thus $g^{'}(T_s) > -1$. □

Note that this lemma indicates: (1) when the suppliers' duration increases, the manufacturer's best duration will decrease; (2) for each unit of time that the suppliers' duration increases, the manufacturer's best duration will decrease less than one unit. This result is consistent with Theorem 3.6 in Chapter 3.

We now compare the optimal project duration in the centralized case, $T_s^* + T_0^*$, with that in the decentralized case, $\widetilde{T}_s + \widetilde{T}_0$.

Theorem 4.1 (The Continuous-time Coauthor's Dilemma) $T_s^* + T_0^* \leq \widetilde{T}_s + \widetilde{T}_0$, *the total duration in the decentralized case is always greater than or equal to the optimal duration in the centralized case.*

Proof. By Lemma 4.1, T_s^* and T_0^* are given by the first order condition of Eq. (4.1). Thus,

$$R^{'}(T_s^* + T_0^*) = \sum_{i=1}^{n} C_i^{'}(T_s^*) \tag{4.9}$$

$$R^{'}(T_s^* + T_0^*) = C_0^{'}(T_0^*) \tag{4.10}$$

Under decentralized control, $\widetilde{T}_s$ and $\widetilde{T}_0$ should satisfy,

$$\alpha_i R^{'}(\widetilde{T}_s + \widetilde{T}_0)(1 + g^{'}(\widetilde{T}_s)) = C_i^{'}(\widetilde{T}_s), \tag{4.11}$$

$$\alpha_0 R^{'}(\widetilde{T}_s + \widetilde{T}_0) = C_0^{'}(\widetilde{T}_0), \tag{4.12}$$

where $g^{'}(\widetilde{T}_s) = \dfrac{\alpha_0 R^{''}(\widetilde{T}_s + \widetilde{T}_0)}{-\alpha_0 R^{''}(\widetilde{T}_s + \widetilde{T}_0) + C_0^{''}(\widetilde{T}_0)}$. By Eq. (4.11),

$$\frac{R^{'}(\widetilde{T}_s + \widetilde{T}_0)}{C_i^{'}(\widetilde{T}_s)} = \frac{1}{\alpha_i(1 + g^{'}(\widetilde{T}_s))}.$$

By Lemma 4.3, we know $-1 < g^{'}(\widetilde{T}_s) < 0$. Thus, $\dfrac{1}{\alpha_i(1+g^{'}(\widetilde{T}_s))} > 1$, and $\dfrac{R^{'}(\widetilde{T}_s+\widetilde{T}_0)}{C_i^{'}(\widetilde{T}_s)} > 1$. By Eq. (4.9), we know that $\dfrac{R^{'}(T_s^*+T_0^*)}{\sum_{i=1}^{n} C_i^{'}(T_s^*)} = 1$. Thus,

$$\frac{R^{'}(\widetilde{T}_s+\widetilde{T}_0)}{C_i^{'}(\widetilde{T}_s)} > \frac{R^{'}(T_s^*+T_0^*)}{\sum_{i=1}^{n} C_i^{'}(T_s^*)}. \tag{4.13}$$

For simplicity, we define $G(T_s, T_0) = \dfrac{R^{'}(T_s+T_0)}{C_i^{'}(T_s)}$. By Assumptions 4.1-4.2,

$$R^{'} < 0, R^{''} < 0; \quad C_i^{'} < 0, C_i^{''} > 0 \quad \text{for all} \quad i = 0, 1, ..., n. \tag{4.14}$$

We easily arrive at,

$$\frac{\partial G}{\partial T_s} > 0 \quad and \quad \frac{\partial G}{\partial T_0} > 0 \tag{4.15}$$

Now, we are ready to discuss the relationship between $\widetilde{T}_s$ and T_s^*, and between $\widetilde{T}_0$ and T_0^*. There could be four cases:

(a). $\widetilde{T}_s \geq T_s^*$ and $\widetilde{T}_0 \geq T_0^*$.

(b). $\widetilde{T}_s \leq T_s^*$ and $\widetilde{T}_0 \leq T_0^*$.

(c). $\widetilde{T}_s \geq T_s^*$ and $\widetilde{T}_0 \leq T_0^*$.

(d). $\widetilde{T}_s \leq T_s^*$ and $\widetilde{T}_0 \geq T_0^*$.

By Eq. (4.13) and (4.15), we can see that case (b) would never happen. Consider case (a), it is obvious that $T_s^* + T_0^* \leq \widetilde{T}_s + \widetilde{T}_0$.

Consider case (c), because $\widetilde{T}_0 \leq T_0^*$ and $C_0^{'}(.)$ is increasing, we must have $C_0^{'}(\widetilde{T}_0) \leq C_0^{'}(T_0^*)$. By Eq. (4.10) and (4.12), we arrive at $\alpha_0 R^{'}(\widetilde{T}_s+\widetilde{T}_0) \leq R^{'}(T_s^*+T_0^*)$. Because $\alpha_0 R^{'}(\widetilde{T}_s+\widetilde{T}_0) \geq R^{'}(\widetilde{T}_s+\widetilde{T}_0)$, we obtain $R^{'}(\widetilde{T}_s+\widetilde{T}_0) \leq R^{'}(T_s^*+T_0^*)$. Because $R^{'}(\cdot)$ is decreasing, we must have $T_s^* + T_0^* \leq \widetilde{T}_s + \widetilde{T}_0$.

The proof of case (d) follows that of case (c). Specifically, because $\widetilde{T}_s \leq T_s^*$ and $\sum_{i=1}^n C_i^{'}(\cdot)$ is increasing, $C_i^{'}(\widetilde{T}_s) < \sum_{i=1}^n C_i^{'}(\widetilde{T}_s) \leq \sum_{i=1}^n C_i^{'}(T_s^*)$. By Eq. (4.9) and (4.11), we get $(1+g^{'}(\widetilde{T}_s))\alpha_i R^{'}(\widetilde{T}_s+\widetilde{T}_0) \leq R^{'}(T_s^*+T_0^*)$. Because $0 < (1+g^{'}(\widetilde{T}_s))\alpha_i < 1$, $(1+g^{'}(\widetilde{T}_s))\alpha_i R^{'}(\widetilde{T}_s+\widetilde{T}_0) \geq R^{'}(\widetilde{T}_s+\widetilde{T}_0)$. Thus, $R^{'}(\widetilde{T}_s+\widetilde{T}_m) \leq R^{'}(T_s^*+T_0^*)$. Because $R^{'}(\cdot)$ is decreasing, we must have $T_s^*+T_0^* \leq \widetilde{T}_s+\widetilde{T}_0$.

Summarizing all cases, $T_s^*+T_0^* \leq \widetilde{T}_s+\widetilde{T}_0$ always holds. □

This theorem indicates that the decentralized case always results in a longer duration for the project than the centralized case. This is consistent to the Coauthor's Dilemma in Chapter 3.

We now determine the share of revenue, α_0, α_i, so as to incorporate the individual rationality constraints, which mandate that the manufacturer and the suppliers must make a positive pay-off in order to participate in the project. For the manufacturer, we must have

$$\pi_0 = \alpha_0 R(T_s+T_0) - C_0(T_0) \geq 0. \tag{4.16}$$

For the suppliers, we consider a weak condition, which is

$$\pi_s = (1-\alpha_0)R(T_s+T_0) - \sum_{i=1}^n C_i(T_s) \geq 0. \tag{4.17}$$

Combining Eqs. (4.16)-(4.17), we have the following lower and upper bounds for α_0,

$$\alpha_0 \geq \frac{C_0(T_0)}{R(T_s+T_0)}, \tag{4.18}$$

$$\alpha_0 \leq 1 - \frac{\sum_{i=1}^n C_i(T_s)}{R(T_s+T_0)}. \tag{4.19}$$

Or equivalently,

$$\frac{C_0(T_0)}{R(T_s+T_0)} \leq \alpha_0 \leq 1 - \frac{\sum_{i=1}^n C_i(T_s)}{R(T_s+T_0)}. \tag{4.20}$$

Thus, α_0 must satisfy the constraints in Eq. (4.20) for the game to be individually rational.

4.1.3 Special Cases

In this section, we consider two special cases relevant to practice and develop stronger results. We shall first consider linear revenue function and then constant assembly cost at stage 2.

Linear Revenue Function

Linear revenue function corresponds to the cases where a linear incentive contract is used for the entire project. Let $R(T) = -aT + K$, then

$$g(T_s) = \arg\max_{T_0} \alpha_0 R(T_s + T_0) - C_0(T_0) = \alpha_0 K - a\alpha_0 T_s + \arg\max_{T_0}[-a\alpha_0 T_0 - C_0(T_0)].$$

Thus, $g(T_s)$ is a constant and independent of T_s, therefore $g^{'}(T_s) = 0$.

With linear revenue function, Eq. (4.7) can be simplified into $\alpha_i(-a) - C_i^{'}(\widetilde{T}_s) = 0$. Thus $-a = \dfrac{C_i^{'}(\widetilde{T}_s)}{\alpha_i}$. This implies that if the manufacturer keeps a larger share of the total pay-off, i.e., a larger α_0, the suppliers will defer their tasks more.

We can also simplify Eq. (4.6) into $\alpha_0(-a) - C_0^{'}(\widetilde{T}_0) = 0$, and thus $-a = \dfrac{C_0^{'}(\widetilde{T}_0)}{\alpha_0}$. This implies that if the manufacturer keeps a larger share of the total pay-off, i.e., a larger α_0, the manufacturer will work faster. In summary,

$$-a = \frac{C_i^{'}(\widetilde{T}_s)}{\alpha_i}, -a = \frac{C_0^{'}(\widetilde{T}_0)}{\alpha_0}. \tag{4.21}$$

We now identify the relationship between the equilibrium $(\widetilde{T}_s, \widetilde{T}_0)$ and the global optimal solution (T_s^*, T_0^*). The linear revenue function allows us to obtain a stronger result on the relationship.

For the centralized case, we first note that

$$\pi = R(T_0 + T_s) - C_0(T_0) - \sum_{i=1}^{n} C_i(T_s) = K - a(T_0 + T_s) - C_0(T_0) - \sum_{i=1}^{n} C_i(T_s).$$

The first order conditions for T_s,T_0 are $-a = C_0^{'}(T_0)$,$-a = \sum_{i=1}^{n} C_n^{'}(T_s)$. So, the global optimal solution (T_s^*, T_0^*) satisfies,

$$-a = C_0^{'}(T_0^*), -a = \sum_{i=1}^{n} C_n^{'}(T_s^*). \tag{4.22}$$

Comparing Eq. (4.21) and (4.22), we can conclude,

$$T_s^* \leq \widetilde{T}_s, \quad T_0^* \leq \widetilde{T}_0.$$

This property is stronger than Lemma 4.1 where $T_s^* + T_0^* \leq \widetilde{T}_s + \widetilde{T}_0$.

Constant Assembly Cost

Suppose $C_0(T_0) = K_0$. By Eq. (4.8), $g^{'}(T_s) = -1$. Eq. (4.7) becomes

$$C_i^{'}(T_s) = 0. \tag{4.23}$$

This equation indicates that the dominating supplier only needs to minimize its total cost in the equilibrium without considering the revenue function.

4.2 Stochastic Task Durations

In this section, we consider stochastic task durations for the base model defined in Assumption 3.1. Specifically,

Assumption 4.3 *At level 1 of the project network, there is only one task. Each task cannot be expedited but can be delayed by at most one period. The supplier and the manufacturer will no longer decide the task duration directly. Instead, they choose to* *exert either high effort or low effort. Depending on the effort, the task has different probability to delay. Both the manufacturer and the supplier do not know the other's choice but observe the other's task duration. If the project is delayed, it is subject to a penalty which is time independent.*

In this model, the supplier will choose its effort level between e_1^H and e_1^L where e_1^H represents the high effort level and e_1^L represents the low effort level; the manufacturer will choose effort level between e_0^H and e_0^L where e_0^H represents the high effort level and e_0^L represents the low effort level. If a firm i exerts high effort e_i^H, its task (task i) will have a low probability q_i^L to get delayed. On the other hand, if the firm exerts low effort e_i^L, task i will have a high probability q_i^H to be delayed. Note that $q_i^L < q_i^H$ and $q_i^L + q_i^H$ is not necessarily equal to 1.

For the cost structure, it is natural to assume that when firm i exerts low effort, there would be a saving s_i as compared to exerting high effort.

If the task is delayed, there would be an extra direct cost associated with the effort for a longer duration. That is, a supplier exerts an effort but the task is unfortunately delayed by one period, it has to pay the associated direct cost occurred in this period. We define f_i^H to be the extra direct cost from high effort and f_i^L to be the extra direct cost from low effort respectively. It is natural to assume that $f_i^H > f_i^L$.

Since the project starts with the an original schedule that is optimal under centralized control. The condition is the same as Condition 3.1. We rewrite it here.

Condition 4.1 *Global Optimum:* $s_1 < p$, $s_0 < p$.

With this condition, we study the strategic behavior of firms under the loss sharing and fair sharing partnerships.

4.2.1 The Loss Sharing Partnership

To analyze the subgame perfect Nash equilibrium under the loss sharing partnership, we start from the last stage and go backward. The manufacturer will observe whether the supplier's task is delayed or kept.

- If the supplier's task is delayed, the manufacturer's expected pay-offs are $E(\pi_0(e_0^H)) = -p_0(1-q_0^L) - (2p_0 + f_0^H)q_0^L$ and $E(\pi_0(e_0^L)) = s_0 - p_0(1-q_0^H) - (2p_0 + f_0^L)q_0^H$.
- If the supplier's task is on time, they are $E(\pi_0(e_0^H)) = -(p_0 + f_0^H)q_0^L$ and $E(\pi_0(e_0^L)) = s_0 - (p_0 + f_0^L)q_0^H$.

For the ease of illustration, we define $\delta_1 = f_1^H - f_1^L$ and $\delta_0 = f_0^H - f_0^L$. δ_i represents the difference in extra direct cost between the high effort and the low effort when the task is delayed. The manufacturer has dominant strategies at stage 2.

Lemma 4.4 *when condition $s_0 + \delta_0 q_0^L < (q_0^H - q_0^L)(p_0 + f_0^L)$ holds, e_0^H is the dominant strategy for the manufacturer; when condition $s_0 + \delta_0 q_0^L > (q_0^H - q_0^L)(p_0 + f_0^L)$ holds, e_0^L is the dominant strategy for the manufacturer.*

Proof. If the supplier's task is delayed, $E(\pi_0(e_0^H)) - E(\pi_0(e_0^L)) = p_0(q_0^H - q_0^L) - s_0 - f_0^H q_0^H$. Since $\delta_0 = f_0^H - f_0^L$, the equation becomes $p_0(q_0^H - q_0^L) - s_0 - (f_0^L + \delta_0)q_0^L + f_0^L q_0^H = (p_0 + f_0^L)(q_0^H - q_0^L) - s_0 - \delta_0 q_0^L$. When $s_0 + \delta_0 q_0^L < (q_0^H - q_0^L)(p_0 + f_0^L)$, the manufacturer's best strategy is to choose e_0^H, otherwise its best strategy is to choose e_0^L.

If the supplier is on time, $E(\pi_0(e_0^H)) - E(\pi_0(e_0^L)) = p_0(q_0^H - q_0^L) - s_0 - f_0^H q_0^L + f_0^L q_0^H = (p_0 + f_0^L)(q_0^H - q_0^L) - s_0 - \delta_0 q_0^L$. When $s_0 + \delta_0 q_0^L < (q_0^H - q_0^L)(p_0 + f_0^L)$, the manufacturer's best strategy is to choose e_0^H, otherwise its best strategy is to choose e_0^L. $\square$

Going back to stage 1, we derive the supplier's dominant strategies as below.

Lemma 4.5 *When $s_1 + \delta_1 q_1^L < (q_1^H - q_1^L)(p_1 + f_1^L)$, e_1^H is the dominant strategy for the supplier; when $s_1 + \delta_1 q_1^L > (q_1^H - q_1^L)(p_1 + f_1^L)$, e_1^L is the dominant strategy for the supplier.*

The proof is similar to Lemma 4.4 and thus omitted.

To see the meaning of the conditions in Lemmas 4.4-4.5, we consider a special case where $f_i^H = f_i^L = 0$ for $i = 0, 1$ which indicates that there is no extra direct cost. Then the left-hand-side of the conditions is the saving while the right-hand-side of the conditions is the probability adjusted penalty because the delay is stochastic. More specifically,

- when $s_0 < (q_0^H - q_0^L)p_0$, the saving by low effect is smaller than the discounted delay penalty for the manufacturer. The manufacturer will choose e_0^H.
- When $s_0 > (q_0^H - q_0^L)p_0$, the saving by low effect is greater than the discounted delay penalty for the manufacturer. The manufacturer will choose e_0^L.
- When $s_1 < (q_1^H - q_1^L)p_1$, the saving by low effect is smaller than the discounted delay penalty for the supplier. The supplier will choose e_1^H;
- When $s_1 > (q_1^H - q_1^L)p_1$, the saving by low effect is greater than the discounted delay penalty for the supplier. The supplier will choose e_1^L.

The conditions here are an extension of the conditions in Lemma 3.1 from deterministic durations to stochastic duration.

To see the impact of stochastic task durations, we now compare Lemmas 4.4-4.5 to Lemma 3.1 of the deterministic case in Chapter 3 Section 3.4.1. Lemma 3.1 shows that when $s_0 < p_0$, the manufacturer will choose "keep"; when $s_0 > p_0$, the manufacturer will choose "delay". Similarly, when $s_1 < p_1$, the supplier will choose "keep"; when $s_1 > p_1$, the supplier will choose "delay".

Because $0 < q_0^H - q_0^L < 1$ and $0 < q_1^H - q_1^L < 1$, they can be regarded as discount factors. In the first case, we note that $s_0 < (q_0^H - q_0^L)p_0 < p_0$, which implies that fixing p_0, s_0 has to be smaller in the stochastic case than the deterministic case to ensure that the manufacturer chooses "keep", or conversely, fixing s_0, p_0 has to be higher in the stochastic case than the deterministic case to ensure that the manufacturer chooses "keep". Thus, the condition for a high effort is more difficult to satisfy for the manufacturer in the stochastic case than the condition for "keep" in the deterministic case. The same insight applies to the supplier.

4.2.2 The Fair Sharing Partnership

The scheme for fair sharing is introduced in Section 3.5. If a firm's task is delayed by one period, the firm will pay the full penalty cost for the delay including its own penalty cost and a compensation to the other firm.

For the supplier, if it exerts a high effort, e_1^H, its task will have $1 - q_1^L$ chance to be finished on time and q_1^L chance to be delayed. If the task is on time, its pay-off is zero; if delayed, the pay-off is $-p - f_1^H$. Thus, the expected pay-off for a high effort supplier is $-(p + f_1^H)q_1^L$. If the supplier exerts a low effort, e_1^L, its task will have $1 - q_1^H$ chance to be finished on time and q_1^H chance to be delayed. If the task is on time, the pay-off is s_1; if delayed, the pay-off is $s_1 - p - f_1^L$. The expected pay-off for a low effort supplier is $s_1 - (p + f_1^L)q_1^H$. We have $E(\pi_1(e_1^H)) - E(\pi_1(e_1^L)) = p(q_1^H - q_1^L) - f_1^H q_1^L - f_1^L q_1^H - s_1$. Under the fair-sharing partnership, we have the following dominate strategies for the supplier.

Lemma 4.6 *When $s_1 < p(q_1^H - q_1^L) - f_1^H q_1^L - f_1^L q_1^H$, e_1^H is the dominant strategy for the supplier; when $s_1 > p(q_1^H - q_1^L) - f_1^H q_1^L - f_1^L q_1^H$, e_1^L is the dominant strategy for the supplier.*

Similarly, we have the dominant strategies for the manufacturer.

Lemma 4.7 *When $s_0 < p(q_0^H - q_0^L) - f_0^H q_0^L - f_0^L q_0^H$, e_0^H is the dominant strategy for the manufacturer; when $s_0 > p(q_0^H - q_0^L) - f_0^H q_0^L - f_0^L q_0^H$, e_0^L is the dominant strategy for the manufacturer.*

Considering the special case of $f_1^L = f_1^H = 0$, we have $E(\pi_1(e_1^H)) - E(\pi_1(e_1^L)) = p(q_1^H - q_1^L) - s_1$. The choice of "high effort" not only depends on the penalty cost and the saving in direct cost, but also on delay probabilities. A similar observation can be made for the manufacturer, where we have $E(\pi_0(e_0^H)) - E(\pi_0(e_0^L)) = p(q_0^H - q_0^L) - s_0$. Comparing the results here to those in Section 3.5.1 in Chapter 3, the fair sharing partnership in case of uncertain task durations requires more restrictive conditions to ensure that both firms exert a high effort than in the case of deterministic durations. Comparing the results here to those in Section 4.2.1, we can see that both firms are more likely to exert a high effort in the fair-sharing partnership than in the loss sharing partnership.

4.3 Conclusions

In this chapter, we extend Chapter 3 in two directions. First, we consider a continuous-time model and show that the general Coauthor's Dilemma holds. The continuous-time model provides a more general functional form between time and cost as compared to the discrete model in Chapter 3. Second, we relax the deterministic task durations and introduce uncertainties where a task can be delayed even if a high effort is exerted. Our analysis shows that the uncertainty in task durations has a significant impact on the firms' optimal strategies. Specifically, under the loss sharing partnership, both firms are less likely to exert a high effort with uncertainty than what they would do without uncertainty; the firms' behaviors are also changed in the fair sharing partnership which does not guarantee a high effort exerted by both firms although it does provide stronger motivation than the loss sharing partnership for them to do so.

Going forward beyond the scope of this chapter, research on supply chain and project management interfaces promises to be fruitful to both practitioners and academicians because of the high impact on practice, and the potential of exciting theoretical discoveries and insights by integrating two rich bodies of literature. The potential in coordinating the project-driven supply chains (or joint projects) has recently been recognized in both academia and industry. While there is ample work to be done, we suggest the following future research directions:

1. *Empirical Studies*: The recent slips of the 787 Dreamliner and Airbus 380 have drawn the attention of both practitioners and academicians on how to ensure successful innovation by collaboration. While theoretical models can be built to aid the development of the next mega project, empirical studies should also be done to discover what really happened in these programs.

2. *Uncertainties in Projects*: We have done some pilot study in Section 4.2 on uncertainties. There are still many other ways to introduce uncertainties to our model. While deterministic model greatly simplifies the analysis and thus allows us to establish clean results on incentives and gaming behaviors in joint projects, it is of

a great interest to allow randomness in project durations, task failure rates, and much more to potentially integrate the economics/supply chain incentive theory with project evaluation and review technique (PERT).

Bibliography

Al-Najjar, Nabil I. 1997. Incentive Contracts in Two-Sided Moral Hazards with Multiple Agents. *J. Econ. Theory* **74**(1) 174–195.

Aquilano, Nicholas J., Dwight E. Smith. 1980. A formal set of algorithms for project scheduling with critical path scheduling/material requirements planning. *J. Oper. Manag.* **1**(2) 57–67.

Axsater, Sven. 1990. Simple Solution Procedures for a Class of Two-Echelon Inventory Problems. *Oper. Res.* **38**(1) 64–69.

Axsater, Sven. 2007. *Inventory control*, vol. 90. Springer.

Aydinliyim, Tolga, GL Vairaktarakis. 2010. Coordination of Outsourced Operations to Minimize Weighted Flow Time and Capacity Booking Costs. *Manuf. Serv. Oper. Manag.* **12**(2) 236–255.

Bayiz, Murat, Charles J. Corbett. 2005. Coordination and Incentive Contracts in Project Management under Asymmetric Information. *SSRN eLibrary* .

Bhaskaran, Sreekumar R., V. Krishnan. 2009. Effort, Revenue, and Cost Sharing Mechanisms for Collaborative New Product Development. *Manage. Sci.* **55**(7) 1152–1169.

Bhattacharyya, Sugato, Francine Lafontaine. 1995. Double-sided Moral Hazard and the Nature of Share Contracts. *Rand J. Econ.* **26**(4) 761–781.

Brown, Karen A, Thomas G Schmitt, Richard J Schonberger, Stephen Dennis. 2004. Quadrant Homes Applies Lean Concepts in a Project Environment. *Interfaces (Providence).* **34**(6) 442–450.

Demski, Joel S., David Sappington. 1984. Optimal Incentive Contracts With Multiple Agents. *J. Econ. Theory* **33**(1) 152–171.

Dodin, B, A A Elimam. 2001. Integrated project scheduling and material planning with variable activity duration and rewards. *IIE Trans.* **33**(11) 1005–1018.

Elfving, J A, G Ballard, U Talvitie. 2010. Standardizing logistics at the corporate level towards lean logistics in construction. *Proc. 18th Annu. Conf. Int. Gr. Lean Constr.*. 222–231.

Elmaghraby, Salah E. 1990. Project Bidding under Deterministic and Probabilistic Activity Durations. *Eur. J. Oper. Res.* **49**(1) 14–34.

Exostar, LLC. 2007. *Boeing 787: global supply chain management takes flight*. Case Study.

Graves, Stephen C, Sean P Willems. 2000. Optimizing strategic safety stock placement in supply chains. *Manuf. Serv. Oper. Manag.* **2**(1) 68–83.

Graves, Stephen C, Sean P Willems. 2003. Supply chain design: safety stock placement and supply chain configuration. *Handbooks Oper. Res. Manag. Sci.* **11** 95–132.

Graves, Stephen C, Sean P Willems. 2005. Optimizing the Supply Chain Configuration for New Products. *Manage. Sci.* **51**(8) 1165–1180.

Gutierrez, Genaro, Anand Paul. 2000. Analysis of the Effects of Uncertainty, Risk-pooling, and Subcontracting Mechanisms on Project Performance. *Oper. Res.* **48**(6) 927–938.

Hadley, George, Thomson M Whitin. 1963. *Analysis of inventory systems*. Prentice Hall.

Hariharan, Rema, Paul Zipkin. 1995. Customer-Order Information, Leadtimes, and Inventories. *Manage. Sci.* **41**(10) 1599–1607.

Hausman, Warren H., Hau L. Lee, Alex X. Zhang. 1998. Joint demand fulfillment probability in a multi-item inventory system with independent order-up-to policies. *Eur. J. Oper. Res.* **109**(3) 646–659.

Holmstrom, Bengt. 1982. Moral Hazard in Teams. *Bell J. Econ.* **13**(2) 324–340.

Holmstrom, Bengt, Paul Milgrom. 1991. Multitask principal-agent analyses: Incentive contracts, asset ownership, and job design. *JL Econ. Org.* **7** 24.

Horng, T.C., K. Bozdogan. 2007. *Comparative analysis of supply chain management practices by Boeing and Airbus: long-term strategic implications*. Presentation at the MIT Lean Aerospace Initiative, April 18, 2007.

Józefowska, Joanna, Jan Weglarz. 2006. *Perspectives in modern project scheduling*, vol. 92. Springer.

Jozefowska, Joanna, Jan Weglarz. 2006. *Perspectives in Modern Project Scheduling*. 1st ed. Springer, New York.

Kerwin, K. 2005. BW 50: A new blueprint at Pulte Homes. *BusinessWeek, Oct. 3rd* .

Kim, Son Ku, Susheng Wang. 1998. Linear Contracts and the Double Moral-Hazard. *J. Econ. Theory* **82** 342–378.

Kwon, H. Dharma, Steven A. Lippman, Kevin F. McCardle, Christopher S. Tang. 2010. Project Management Contracts with Delayed Payments. *Manuf. Serv. Oper. Manag.* **12**(4) 692–707.

McAfee, R. Preston, John McMillan. 1986. Bidding for contracts: a principal-agent analysis. *Rand J. Econ.* **17**(3) 326–338.

Nahmias, Steven. 2008. *Production and Operations Analysis (McGraw-Hill/Irwin Series Operations and Decision Sciences)*. McGraw-Hill/Irwin.

OBrien, William J, Kerry London, Ruben Vrijhoef. 2002. Construction supply chain modeling: a research review and interdisciplinary research agenda. *Proc. IGLC*, vol. 10. 1–19.

Ozdamar, L., G. Ulusoy. 1995. A survey on the resource-constrained project scheduling problem. *IIE Trans.* **27**(5) 574–586.

Paul, Anand, Genaro Gutierrez. 2005. Simple Probability Models for Project Contracting. *Eur. J. Oper. Res.* **165**(2) 329–338.

Pinedo, Michael L. 2005. *Planning and Scheduling in Manufacturing and Services*. Springer.

Porteus, Evan L. 2002. *Foundations of stochastic inventory theory*. Stanford University Press.

Rubinstein, Reuven Y, Alexander Shapiro. 1993. *Discrete event systems: Sensitivity analysis and stochastic optimization by the score function method*, vol. 346. Wiley New York.

Schmitt, Thomas, Bruce Faaland. 2004. Scheduling recurrent construction. *Nav. Res. Logist.* **51**(8) 1102–1128.

Shah, M, Y Zhao. 2009. Construction Resource ManagementICM Inc. *Rutgers Bus. Sch. Case Study. Newark, NJ* .

Simchi-Levi, David, Yao Zhao. 2011. Performance evaluation of stochastic multi-echelon inventory systems: a survey. *Adv. Oper. Res.* **2012**.

Smith-Daniels, Dwight E., Nicholas J. Aquilano. 1984. Constrained resource project scheduling subject to material constraints. *J. Oper. Manag.* **4**(4) 369–387.

Smith-Daniels, Dwight E, Vicki L Smith-Daniels. 1987. Optimal project scheduling with materials ordering. *IIE Trans.* **19**(2) 122–129.

Somerville, C Tsuriel. 1999. Residential construction costs and the supply of new housing: endogeneity and bias in construction cost indexes. *J. Real Estate Financ. Econ.* **18**(1) 43–62.

Szmerekovsky, Joseph G. 2005. The Impact of Contractor Behavior on the Client's Payment-Scheduling Problem. *Manage. Sci.* **51**(4) 629–640.

Tommelein, Iris D, Glen Ballard, P Kaminsky. 2009. Supply chain management for lean project delivery. *Constr. supply Chain Manag. Handb.* 6.

Tommelein, Iris D, Kenneth D Walsh, James C Hershauer. 2003. *Improving capital projects supply chain performance*. Construction Industry Institute, University of Texas at Austin.

Vaidyanathan, Kalyan, Gregory Howell. 2007. Construction supply chain maturity modelconceptual framework. *Proc. IGLC*, vol. 15. 170–180.

Vrijhoef, Ruben, Lauri Koskela. 2000. The four roles of supply chain management in construction. *Eur. J. Purch. supply Manag.* **6**(3) 169–178.

Walsh, Kenneth D, James C Hershauer, Iris D Tommelein, Tobi A Walsh. 2004. Strategic positioning of inventory to match demand in a capital projects supply chain. *J. Constr. Eng. Manag.* **130**(6) 818–826.

Xu, Xin, Yao Zhao. 2010. *Some economic facts of the prefabricated housing*. Rutgers Business School Industry Report, Newark, NJ.

Xu, Xin, Yao Zhao. 2011. *Build-to-Performance – Boeing 787 Dreamliner*. Rutgers Business School Case Study, Newark, NJ.

Yeo, K.T, J.H Ning. 2002. Integrating supply chain and critical chain concepts in engineer-procure-construct (EPC) projects. *Int. J. Proj. Manag.* **20**(4) 253–262.

Zhao, Yao, David Simchi-Levi. 2006. Performance Analysis and Evaluation of Assemble-to-Order Systems with Stochastic Sequential Lead Times. *Oper. Res.* **54**(4) 706–724.

Zipkin, Paul. 1991. Evaluation of base-stock policies in multi-echelon inventory systems with compound-poisson demands. *Nav. Res. Logist.* **38**(3) 397–412.

Zipkin, Paul Herbert. 2000. *Foundations of inventory management*, vol. 2. McGraw-Hill New York.

Vita

Xin Xu

2009-2014 Ph. D. in Supply Chain Management, Rutgers University

2006-2009 M. Eng. in Automation from Tsinghua University

2002-2006 B. Eng. in Automation from Tsinghua University

2009-2013 Graduate assistant, Department of Supply Chain Management, Rutgers University