Doctoral Dissertation

Synthesis of Time-varying Positive Linear Systems via Geometric Programming

Chengyan Zhao

Program of Intelligent Cyber-Physical Systems
Graduate School of Science and Technology
Nara Institute of Science and Technology

Supervisor: Kenji Sugimoto
Intelligent System Control Lab. (Division of Information Science)

Submitted on March 11, 2021

A Doctoral Dissertation
submitted to Graduate School of Science and Technology,
Nara Institute of Science and Technology
in partial fulfillment of the requirements for the degree of
Doctor of ENGINEERING

Chengyan Zhao

Thesis Committee:

Supervisor	Kenji Sugimoto (Professor, Division of Information Science)
	Shoji Kasahara (Professor, Division of Information Science)
	Yukiharu Uraoka (Professor, Division of Material Science)
	Takamitsu Matsubara (Associate Professor, Division of Information Science)
	Masaki Ogura (Associate Professor, Graduate School of Information Science and Technology, Osaka University)

Synthesis of Time-varying Positive Linear Systems via Geometric Programming*

Chengyan Zhao

Abstract

In the field of control system, there lies an important class of linear systems, called positive linear systems. For simple introduction, the time-invariant, finite-dimensional single input, single output systems, described by state equations of the form

$$\dot{x}(t) = Ax(t) + bu(t)$$

where the state matrix A is Metzler (the off-diagonal entries are nonnegative). Positive linear systems, as another linear systems, satisfy the properties of general linear systems and also the peculiar ones. Thus, positive linear systems could be studied by other mathematical theories, like Linear Programming, Nonnegative Matrices and Geometric Programming.

In the dynamic system, the abrupt changes of working environment, failures of sensors or actuators, and the working points of nonlinear systems can be regarded as the phenomena of time-varying or stochastic switching, which can be widely found in energy, chemical process, communication, social network, and other practical systems. Due to these reasons, the models adopted for the design of control systems are more and more complicated and specific. Recent years, switched linear systems which are regarded as a special class of hybrid systems attract a lot of attention in modeling the time-varying feature of dynamic systems. Due to the wide existence of nonnegative variables in the world, examples

*Doctoral Dissertation, Graduate School of Science and Technology Nara Institute of Science and Technology, March 11, 2021.

like, power, traffic, bio-chemistry, positive linear systems become the hot spot in the control research field recently. By using the property of nonnegative matrix theory, the study of positive systems derives many new results than the general linear system theory. In this situation, this dissertation aims at building new frameworks for the positive time-varying linear systems. This dissertation will make deep discussion on two issues of the positive linear systems with time-varying state. One is the stability optimization of stochastic switched state. The other is the general time-varying finite-time control. The main contributions are as follows:

The stability optimization problem for the positive semi-Markov jump linear systems is investigated. Firstly, for extending the stochastic switching rule in modeling the positive switched linear systems to a more general case, the Markov process adopted in the previous works is extended to semi-Markov process because Markov process has the limitation in modeling the switching phenomena in real problems. By utilizing the spectral radius based stability analysis result of positive semi-Markov jump linear system, the framework for solving the stability optimization problem is proposed via geometric programming. Specifically, the problems of tuning the coefficients of the system matrices for maximizing the exponential decay rate of the system under a budget-constraint and minimizing the parameter tuning cost under the decay rate constraint is investigated. By using a result from the matrix theory on the log-log convexity of the spectral radius of nonnegative matrices, the stability optimization problems are reduced to convex optimization problems under certain regularity conditions on the system matrices and the cost function. Finally, the validity and effectiveness of the proposed results are illustrated by using an example from the population biology.

The finite-time control problem for discrete-time positive linear system with time-varying state is studied by adopting geometric programming. Although several interesting control problems appearing in population biology, economics, and network epidemiology can be described as the class of finite-time control problems, an efficient solution to the control problem has not been yet found in the literature. In this dissertation, an optimization framework for solving the class of finite-time control problems via convex optimization is proposed. Finally, the effectiveness of the proposed method is illustrated by a numerical simulation in the context of dynamical product development processes.

Based on the theoretical results of Chapter 4, we solve the resource allocation of Product Development (PD) process. In this thesis, we formulate the dynamic resource allocation of the PD process as a convex optimization problem. Specially, we build and solve two variants of this issue: the budget-constrained problem and the performance-constrained problem. By using convex optimization, we propose a framework to optimally solve large problem instances at a relatively small computational cost. The solutions to both problems exhibit similar trends regarding resource allocation decisions and performance evolution. Furthermore, we show that the product architecture affects resource allocation, which in turn affects the performance of the PD process. By introducing centrality metrics for measuring the location of the modules and design rules within the product architecture network, we find that resource allocation decisions correlate to their metrics. These results provide simple, but powerful, managerial guidelines for efficiently designing and managing the PD process.

Keywords:

Positive systems, switched linear systems, semi-Markov process, finite-time constrol, stability optimization, geometric programming, convex optimization, biology population control, product management, centrality.

Contents

List of Figures

1 Introduction

1.1 Positive linear systems

Linear time-invariant systems, where the state, in-put, and out-put matrices with respect to nonnegative orthant, are said to be positive linear systems [32]. Numerous examples of dynamic systems with nonnegative variables are found in biology [7], epidemics [37], chemistry [13], air-transportation [14], and project management [15]. Compared with general linear systems, positive linear systems hold the character that all its signals are confined to one nonnegative region in multidimensional space. By utilizing the mathematical theory of positive system built on the theory of nonnegative matrices [16] and the unified theoretical method for positive systems [17], a large number of results have been reported about the analysis and synthesis of positive systems such as positive realization [18], reachability and controllability [19] [20], and stabilization [72].

For the synthesis problems of positive linear systems [32], the commonly recognized approach is to building co-positive linear Lyaponov function. By utilizing the character of positive systems, this way has the advantage compared with the quadratic Lyaponov function that used for general linear system, because the co-positive linear Lyaponov function based method can be solved by linear programming which has the less variables and higher efficiency for computation. However, the linear programming approach has the constraint that the controller design problem does allow the parameter tunning feature to be nonlinear, which restricts the feasibility of applying to practical problems. The key idea of this thesis is to find the relationship between positive linear system and optimization method for building the more specific computation framework. To simply state the idea, we know that linear programming deal with the linear function with real variables. In this situation, the parameter tunning function in the controller design problem must be liner or constant. Our motivation is if there exists a optimization approach dealing the positive nonlinear function program? Fortunately, geometric programming (see Section 2) is an ideal method for addressing a class of nonnegative nonlinear function called posynomials.

1.2 Positive stochastic jump linear systems

Recently, positive linear systems with time-varying or stochastic switching features [10] are deeply studied because the real physical systems always suffer from the sudden change in the mode of interconnection between subsystems, failure of the components, and human intervention during their operation. To describe this phenomenon, stochastic process model is adopted for mathematically illustrating the jump feature among these subsystems. Currently, the most investigated model is the Markov jump linear systems [1–6, 9, 33] which is regarded as an important class of stochastic switched dynamical systems and have applications in mobile robots [36], epidemic processes [37], and networked control systems [38]. Several important issues on Markov jump linear systems [34] have been addressed in the literature including controllability and stabilizability [11, 23, 27, 31, 39], robust optimal control [21, 22, 24–26, 40], sampled-data control [28–30, 41], and game theory [42]. Furthermore, the class of systems includes the basic class of stochastic dynamical systems with an independent and identically distributed parameters [43, 44]. Despite the aforementioned advances, modelling by a Markov jump linear system suffers from the limitation that the sojourn time of the systems must follow an exponential distribution. This restriction is not necessarily satisfied in practice; a typical example arises in modeling the occurrence of component failures in the context of fault tolerant control systems, where the probability density functions of failure rates are well-explained by Weibull distributions [45].

One natural way to overcome this limitation is to allow the sojourn times to follow non-exponential distributions, which results in a broader class of stochastic dynamical systems called *semi-Markov jump linear systems* [46]. In this context, we can find in the literature several results toward the analysis and control of the class of systems. The authors in [47] have presented sufficient conditions for the moment stability of semi-Markov jump linear systems. Huang and Shi [48] derived linear matrix inequalities (LMIs) for the robust state-feedback control of semi-Markov jump linear systems. We can also find several LMI-based approaches for further advanced types of synthesis methodologies [49–52]. However, the design methodologies in the aforementioned references can be conservative because their derivation relies on approximations for avoiding the difficulty in dealing with non-exponential distributions.

We show that the problem of tuning the parameters of parametrized *positive* semi-Markov jump linear systems can be efficiently and exactly solved without introducing any conservatism. We specifically show that, under the assumption that the parametrization is described by posynomial functions [53], the problems of finding the parameters maximizing the decay rate of the parametrized system and minimizing the parameter tuning cost can be transformed to convex optimization problems. The reduction to convex optimization problems is exact because we do not rely on any approximations that are employed in the aforementioned references. Instead of employing an approximation, in this thesis we utilize the stability characterization of positive semi-Markov jump linear systems [46] as well as the log-log convexity result on the spectral radius of nonnegative matrices [54]. The theoretical result in this thesis is illustrated by an example in the context of the population biology [55].

1.3 Positive time-varying linear systems

The concept of finite-time stability [64], which is concerned with the stability property of dynamical systems over a finite time window, is of practical importance due to its effectiveness in solving realistic control problems appearing in several fields including robotics [65], spacecraft control [66], and multi-agent systems [67]. We find in the literature several advances in the field; for example, Bhat and Bernstein [68] proposed a finite-time stability criteria for continuous-time autonomous systems. Amato et al. [69] proposed a sufficient condition for finite-time stability and control for time-invariant linear systems through the Lyapunov function approach. Hong [70] considered the finite-time control and stabilizability for a class of controllable systems. The authors in [71, 72] studied a finite-time synthesis problem for the nonlinear systems.

Recently, finite-time control problems have been actively investigated in the context of positive systems [32], which are dynamical systems whose state variables are confined to be within the positive orthant and naturally arise in various application areas including pharmacology [74], epidemiology [37,75], and communication networks [76]. For example, the authors in [77] derived a necessary and sufficient condition for the finite-time stability of switched positive linear systems by using the co-positive Lyapunov approach. Colaneri et al. [56] established the convexity of the norm of the state variable of a class of positive time-varying linear systems with respect to the diagonal entries of the state matrix. However, the practical applicability of this convexity result is not necessarily enough to cover some applications of positive linear systems because the convexity property is limited to the diagonals of the state matrix of the system, as shall be discussed later in this thesis. Furthermore, there is a lack of frameworks for considering the cost associated with control input such as the one for chemical [7] and medical [73] interventions.

Extending the framework in [62] for linear time-invariant positive systems, in this thesis we propose an optimization framework to solve a class of finite-time control problems for discrete-time *time-varying* positive linear systems. We formulate the finite-time control problem as an optimization problem, in which the parameter cost as well as the performance evaluation functions are described by posynomial functions [53]. We then show that the finite-time control problem can be transformed into a geometric program, which can be efficiently solved via convex optimization. In the derivation of these results, we do not restrict the tunable entries of the system matrix to its diagonals; therefore, the contribution of this thesis lies in showing a form of convexity of the problem with respect to *any* of the entries of the state matrix.

1.4 Application research: Optimal resource allocation for product development

Successful Product Development (PD) requires careful allocation of development resources. Allocating resources to various subsystems and modules within the product system requires a deep understanding of many complex interactions.

These interactions arise from various sources; namely, due to the physical interdependencies between the different subsystems in the product itself (i.e., the product architecture), the arrangements of organizations that will carry out the development process (i.e., the social network behind the organization), and the structure of the development process (i.e., predecessor relationships between development activities) [94]. In particular, this paper is focused on obtaining an understanding of the product architecture and its role in resource allocation decisions during PD [1].

Product architecture is usually described by a continuum between an integral product architecture to a modular one. In integral architectures, the product functions are shared by product modules (i.e., physical elements), and in modular architectures, each function is delivered by a separate element or module. Thus, integrality creates interdependence between product elements or modules. This interdependency, in turn, results in complexity. That is, some of the interdependencies may not be known in advance, or their influence on product and PD process performance may also be unknown. Within this complex PD environment, several studies have argued that the product architecture may evolve from integral to modular [8, 95].

In this paper, we investigate how the product architecture may influence the resource allocation decision to various modules using an optimization framework. Using this framework, we can investigate the tendency for product architectures to evolve form integral to modular architectures. The aim of this paper is to check whether the location of a module within the product architecture can offer PD managers insights into optimal resource allocation decisions.

Several authors have formulated and analyzed the PD problem by analogy to dynamic linear systems (e.g., [35, 96]). In their analysis, they assumed that all tasks in the design structure matrix (DSM) proceed in parallel, where the DSM is a matrix representation of the development network. At any iteration stage, one

[1] Product architecture is not the only driver for resource allocation decisions. Other drivers, such as existing product lineup, competitive products, product demand and price, technological advancements, consumer taste changes, balancing the development portfolio, etc., may play an influential role (Terwiesch and Ulrich, 2008). However, we focus on product architecture since it is the central issue in our proposed model."Terwiesch, C., Ulrich, K. (2008). Managing the opportunity portfolio. Research-Technology Management, 51(5), 27-38.

unit of work on one task results in a fraction of rework for the other dependent tasks during the next iteration stage. The dependency between tasks is captured by the numerical values in the DSM. As such, the work completed in a current design iteration is a linear function of the work completed in the previous design iteration, with the linear weights being the numerical values in the DSM.

Other authors have used complexity theory to describe and analyze the PD process. They showed how the underlying network topologies and statistical structural properties provide direct information about the functionality, dynamics, robustness, and fragility of these PD projects. Also, the authors in [97] argued that modules could be optimized independently if interface standards between modules are left unchanged. Similarly, Luo [98] used the NK framework to show how different product architectural patterns can influence product evolvability.

Network analysis has also been used for analyzing PD project network [99,100]. For example, the analysis of the network structure (i.e., statistical properties) for various software and hardware development projects revealed that these networks have both small world and scale free network patterns. Additionally, they demonstrated that complex design networks are highly robust to the failure of randomly selected design components, but weak for failures targeting specific components (such as hub components). Similarly, Sosa et al. [101] found that the analysis of the network structure of complex product designs (particularly, the existence of hubs in the design network) impacts the quality of the product being developed.

More recently, the authors in [8] have formulated the PD resource allocation problem as a nonlinear optimization problem. Furthermore, the authors proposed a dynamic model in which there are several investment runs (or rounds) during the PD process. This formulation allowed the investigation of several interesting hypotheses, including the impact of architecture on performance evolution from integral to modular systems.

The aim of this paper is to offer a more efficient optimization approach based on convex optimization techniques, which would allow us to find the globally optimal allocation of development resources. In this direction, we first adopt a discrete-time linear system to represent the work transformation feature in the PD process. Then, we propose an optimization framework where the resource allocation problem of the PD process can be transformed into a convex optimization problem. We then apply our framework to symmetric and synthetic product architectures to reveal the trends of the evolution of optimal investment. Finally, analyzing case studies with asymmetric PD architectures, we gain into the resource allocation problem and provide a guide for designing and managing the PD process.

This thesis is organized as follows. After giving mathematical preliminaries in Section 3.1, we formulate the stabilization problem of positive semi-Markov jump linear systems and state the main result. The derivation of the main result is presented in Section 3.2. In Section 3.3, we illustrate the validity and effectiveness of the result with numerical simulations. In Section 4.1, we formulate the finite-time control problem studied in this thesis. In Section 4.2, we introduce our assumptions on the system and cost functions and, then, state our main result. Finally, in Chapter 5, we illustrate the effectiveness of our results by solving the optimal resource allocation problem that arises in the context of managing product development processes.

2 Mathematical Preliminaries

The following notations are used in this dissertation. Let $\mathbb{R}$, $\mathbb{R}_+$, $\mathbb{R}_{++}$ denote the set of real, nonnegative, and positive numbers, respectively. Let $\mathbb{N}$ denote the set of positive integers. Let $\mathbb{R}^{n\times n}$ denote the set of $n \times n$ real matrices. The identity matrix of order n is denoted by I_n. We let $x \geq 0$ be a nonnegative vector, if the entries of x are all nonnegative. We say that a square matrix is Metzler if the off-diagonal entries of the matrix A are nonnegative. We let the entrywise logarithm operation $\log[\cdot]\colon \mathbb{R}^n_{++} \to \mathbb{R}^n$ be defined by $(\log[v])_i = \log v_i$ for all $i \in \{1, \dots, n\}$. Likewise, we define the entrywise exponentiation $\exp[\cdot]\colon \mathbb{R}^n \to \mathbb{R}^n_{++}$ in the same manner. We say that a matrix is nonnegative if all the entries of the matrix are nonnegative. Let (Ω, M, P) be a probability space. The expected value of a random variable X on Ω is denoted by $E[X]$. We denote the spectral radius of A by $\rho(A)$. We define the entrywise logarithm of a vector $v \in \mathbb{R}^n_{++}$ by $\log[v] = [\log v_1, \dots, \log v_n]^\top$. The entrywise exponential operation $\exp[\cdot]$ is defined in the same manner. A_{ij} stands for (i, j) entry of the matrix A. L^*

Definition 1. *[53] Let v_1, …, v_n denote n real positive variables.*

1. *We say that a real function $g(v)$ is a monomial if there exist $c > 0$ and $a_1, \dots, a_n \in \mathbb{R}$ such that $g(v) = cv_1^{a_1} \dots v_n^{a_n}$.*

2. *We say that a real function $f(v)$ is a posynomial if f is a sum of monomials of v.*

3. *We also say that a real function is a* generalized posynomial *if it can be formed from posynomials using the operations of addition, multiplication, positive (fractional) power, and maximum.*

The following lemma shows the log-convexity of posynomials.

Lemma 1. *[53] Let $f\colon \mathbb{R}^n_+ \to \mathbb{R}_+\colon x \mapsto f(x)$ be a posynomial function. Then, the function*

$$F\colon \mathbb{R}^n \to \mathbb{R}\colon w \mapsto \log f(\exp[w])$$

is convex.

The log-convexity of posynomials allows us to solve a class of optimization problems called geometric programs efficiently, as summarized in the following proposition.

Proposition 1. *If g_i are monomials and f_j are posynomials. We say the following optimization problem is the geometric programming problem*

$$\begin{aligned} \underset{\theta\in\Theta}{\text{minimize}} \quad & f_0(\theta) \\ \text{subject to} \quad & f_j(\theta) \leq 1,\ j = 1, \dots, p, \\ & g_i(\theta) = 1,\ i = 1, \dots, q, \end{aligned}$$

can be transformed into a convex optimization problem through the logarithmic variable transformation

$$\theta = \exp[z],\ z \in \mathit{\Gamma} \subset \mathbb{R}^m.$$

Then, we obtain the convex optimization problem with the following form:

$$\begin{aligned} \underset{z\in\mathit{\Gamma}}{\text{minimize}} \quad & \log f_0(\exp[z]) \\ \text{subject to} \quad & \log f_j(\exp[z]) \leq 0,\ j = 1, \dots, p, \\ & \log g_i(\exp[z]) = 0,\ i = 1, \dots, q. \end{aligned}$$

3 Stability optimization of positive semi-Markov jump linear systems via convex optimization

In this chapter, we study the problem of optimizing the stability of positive semi-Markov jump linear systems. We specifically consider the problems of tuning the coefficients of the system matrices for maximizing the exponential decay rate of the system under a budget-constraint and minimizing the parameter tuning cost under the decay rate constraint. By using a result from the matrix theory on the log-log convexity of the spectral radius of nonnegative matrices, we show that the stability optimization problems are reduced to convex optimization problems under certain regularity conditions on the system matrices and the cost function. We illustrate the validity and effectiveness of the proposed results by using an example from the population biology.

3.1 Problem formulation and main results

Let us consider a parameterized family of switched linear systems of the form

$$\Sigma_\theta : \frac{dx}{dt} = A_{\sigma(t)}(\theta)x(t), \quad x(0) = x_0 \in \mathbb{R}^n, \tag{3.1}$$

where $x(t) \in \mathbb{R}^n$ is the state vector, σ is a piecewise-constant function taking values in the set $\{1, \dots, N\}$, and $A_1(\theta), \dots, A_N(\theta) \in \mathbb{R}^{n\times n}$ are matrices parametrized by the parameter θ belonging to a set $\Theta \subset \mathbb{R}^\ell$.

In this chapter, we assume that each subsystem has a positivity (see, e.g., [32, 57]). We also assume that σ is a semi-Markov process [58]; i.e., we assume that the evolution of σ is governed by the following probabilities:

$$\Pr\{\sigma(t+h)=j \mid \sigma(t)=i\} = \begin{cases} \lambda_{ij}(h)h + o(h), & \text{if } j \neq i, \\ 1+\lambda_{ii}(h)h+o(h), & \text{if } j = i, \end{cases}$$

where $\lambda_{ij}(h)$ represents a time-varying transition rate from mode i to mode j, $\lambda_{ii}(h) = -\sum_{j=1,i\neq j}^{N} \lambda_{ij}(h)$, and $o(h)$ is little-o notation defined by $\lim_{h\to 0} o(h)/h = 0$. The above assumptions are summarized into the following definition.

Definition 2 ([46]). *Let $\theta \in \Theta$. We say that the system Σ_θ is a* positive semi-Markov jump linear system *if the initial state x_0 is nonnegative, the matrices $A_1(\theta)$, …, $A_N(\theta)$ are Metzler, and σ is a semi-Markov process taking values in $\{1,\dots,N\}$.*

For $t \geq 0$ and $x_0 \in \mathbb{R}_+^n$, we let $x(t;x_0)$ denote the trajectory of the system Σ_θ at time t and with the initial condition $x(0) = x_0$. This chapter is concerned with the stability property of the system Σ_θ given as follows:

Definition 3 ([46, 59]). *We say that Σ_θ is* exponentially mean stable *if there exist $\alpha > 0$ and $\beta > 0$ such that, for every x_0 and $\sigma(0)$,*

$$E[\|x(t;x_0)\|] \leq \alpha e^{-\beta t}\|x(0)\|.$$

If Σ_θ is mean stable, then the exponential decay rate *of the system Σ_θ is defined by*

$$\gamma_\theta = -\sup_{x_0\in\mathbb{R}_+^n} \limsup_{t\to\infty} \frac{\log E[\|x(t;x_0)\|]}{t}.$$

In this chapter, we consider a budget-constrained stability optimization problem described as follows. Consider the situation where a limited amount of resource available is given for tuning the parameter θ to improve the stability of the system Σ_θ. We let a real function C denote the cost for achieving a specific parameter θ. In this context, we formulate our stability optimization problem as follows:

Problem 1 (Budget-constrained stabilization)**.** *Let a real number $\bar{C}$ be given. Find the parameter $\theta \in \Theta$ such that the exponential decay rate γ_θ is maximized, while the budget constraint*

$$C(\theta) \leq \bar{C}$$

is satisfied.

In the budget-constrained optimization problem, we need to distribute the constrained parameter cost to $A_i(\theta)$ to obtain the maximized decay rate. However, there is another situation where the optimization object is minimizing the parameter tuning cost $C(\theta)$ while satisfying the performance constraint (decay rate). From this perspective, we formulate an alternative optimization problem as follows:

Problem 2 (Performance-constrained stabilization)**.** *Let a positive number $\bar{\gamma}$ be given. Find the parameter $\theta \in \Theta$ such that the parameter tuning cost $C(\theta)$ is minimized, while the performance constraint*

$$\gamma_\theta \geq \bar{\gamma}$$

is satisfied.

In our main results, we show that Problems 1 and 2 reduce to convex optimization problems. In order to state the main results, we place certain regularity assumptions on the system matrices $A_i(\theta)$ and the cost function $C(\theta)$. For this purpose, we introduce the class of functions called monomials and posynomials [53]. We say that a function $F\colon \mathbb{R}^n_{++} \to \mathbb{R}_{++}$ is a *monomial* if there exist $c > 0$ and real numbers $a_1, \ldots, a_n$ such that

$$F(v) = cv_1^{a_1} v_2^{a_2} \cdots v_n^{a_n}$$

Then, we say that a function F is a *posynomial* if F is a sum of finite number of monomials.

The following mild and reasonable assumption is necessary for ensuring Problems 1 and 2 reduce to convex optimization problems.

Assumption 1. *The following conditions hold true:*

1. *For each* $k = 1, \ldots, N$*, there exists an* $n \times n$ *Metzler matrix* M_k *such that each entry of the matrix*

$$\bar{A}_k(\theta) = A_k(\theta) - M_k$$

is either a posynomial in θ *or zero.*

2. $C(\theta)$ *is a posynomial in* θ*.*

3. *There exist posynomials* $\phi_1(\theta), \ldots, \phi_m(\theta)$ *and positive constants* $\bar{\phi}_1, \ldots, \bar{\phi}_m$ *such that*

$$\Theta = \{\theta \in \mathbb{R}^\ell \colon \phi_1(\theta) \leq \bar{\phi}_1, \ldots, \phi_m(\theta) \leq \bar{\phi}_m\}.$$

4. *The sojourn times of the semi-Markov process* σ *are uniformly bounded, i.e., there exists* $T > 0$ *such that the sojourn times are less than or equal to* T *with probability one.*

Remark 1. Major examples of positive linear time-invariant systems satisfying Assumption 1 include networked epidemic processes [60], population dynamics [55], and dynamical buffer networks [61] (for further discussions, see [62]). For example, in the containment problem for networked epidemic processes [60], the parameter θ corresponds to the infection and recovery rates of the nodes, while the cost function $C(\theta)$ would indicate the cost for medical resources to tune the rates.

Also, we introduce the following notations to state the main results of this chapter. Let σ_d be the embedded Markov chain of σ (see, e.g., [58]). For $i, j \in \{1, \ldots, N\}$, let p_{ij} denote the transition probability of σ_d, i.e., let p_{ij} denote the probability that the discrete-time Markov chain σ_d transitions into state j from state i in one time step. Also, let h_{ij} denote the random variable representing the sojourn time of σ at the mode j after jumping from the mode i, and $f_{ij}(h_{ij})$ denote the corresponding probability density function.

Theorem 1. *Let* $\bar{C} > 0$ *be given. For each* $\theta \in \Theta$ *and* $g > 0$*, define the matrix* $\mathscr{A}(\theta, g) \in \mathbb{R}^{(nN)\times(nN)}$ *as the block matrix whose* (i, j)*-block is defined by*

$$[\mathscr{A}(\theta, g)]_{ij} = p_{ji} \int_0^T e^{(A_j(\theta)+gI)\tau} f_{ji}(\tau)\, d\tau \in \mathbb{R}^{n\times n}. \tag{3.2}$$

Define the set

$$\log[\mathit{\Theta}] = \{\log[\theta] \colon \theta \in \mathit{\Theta}\} \subset \mathbb{R}^\ell.$$

Assume that $u = u^\star$ and $v = v^\star$ solve the following optimization problem:

$$\begin{aligned} \underset{u\in\log[\mathit{\Theta}],\, v\in\mathbb{R}}{\text{minimize}} \quad & -v \\ \text{subject to} \quad & \log\rho(\mathscr{A}(\exp[u], e^v)) \leq 0, \\ & \log C(\exp[u]) \leq \log \bar{C}, \\ & \log \phi_i(\exp[u]) \leq \log \bar{\phi}_i, \quad i = 1, \dots, \ell. \end{aligned} \tag{3.3}$$

Then,

$$\theta = \exp[u^\star]$$

solves Problem 1 and attains the exponential decay rate $e^{v^\star}$. Furthermore, the optimization problem (3.3) *is convex.*

We can also show that Problem 2 can be solved by the following convex optimization problem.

Theorem 2. *Assume that $u = u^\star$ solves the following optimization problem:*

$$\begin{aligned} \underset{u\in\log[\mathit{\Theta}],\, v\in\mathbb{R}}{\text{minimize}} \quad & \log C(\exp[u]) \\ \text{subject to} \quad & \log\rho(\mathscr{A}(\exp[u], e^v)) \leq -\log\bar{\gamma}, \\ & \log \phi_i(\exp[u]) \leq \log \bar{\phi}_i, \\ & \qquad\qquad i = 1, \dots, \ell. \end{aligned}$$

Then,

$$\theta = \exp[u^\star]$$

solves Problem 2 and attains the minimized cost $C(\exp[u^\star])$. Furthermore, the optimization problem in Theorem 2 is convex.

3.2 Proof

In this section, we give the proof of the main results. Because the proof of Theorem 2 is similar to that of Theorem 1, we only present the proof of Theorem 1.

We first prepare a few lemmas for the proof. The first lemma gives a characterization of the exponential decay rate of the system Σ_θ in terms of the spectral radius of the matrix $\mathscr{A}(\theta, g)$ defined in the theorem.

Lemma 2. *Let $\theta \in \Theta$ and $g > 0$ be arbitrary. The following statements are equivalent:*

- *The exponential decay rate of Σ_θ satisfies $\gamma_\theta > g$.*
- $\rho(\mathscr{A}(\theta, g)) < 1$.

Proof. Assume $\gamma_\theta > g$. Then, the positive semi-Markov jump linear system

$$\frac{dx}{dt} = (A_{\sigma(t)}(\theta) + gI)x(t)$$

is exponentially mean stable. Therefore, by Theorem 2.5 in [46], the matrix $\mathscr{A}(\theta, g)$ has a spectral radius less than one, as desired. The proof of the opposite direction can be proved in the same manner and, therefore, is omitted.

We then recall the following celebrated result by [54]. We say that an $\mathbb{R}_{++}$-valued function $f(x)$ is superconvex if $\log f(x)$ is convex.

Lemma 3 ([54])**.** *Let $A\colon \mathbb{R}^\ell \to \mathbb{R}_{++}^{n\times n}$ be a function. Assume that each entry of A is either a superconvex function or the zero function. Then, the mapping $\mathbb{R}^\ell \to \mathbb{R}_{++}\colon x \mapsto \rho(A(x))$ is superconvex.*

We finally state the following lemma concerning the superconvexity of posynomials.

Lemma 4 ([53])**.** *Let $f\colon \mathbb{R}_{++}^n \to \mathbb{R}_{++}$ be a posynomial. Then, the mapping $\mathbb{R}^n \to \mathbb{R}_{++} : u \mapsto f(\exp[u])$ is superconvex.*

Let us now prove Theorem 1.

Lemma 2 shows that the solution of Problem 1 is given by the following optimization problem:

$$\begin{aligned} \underset{\theta\in\Theta,\, g>0}{\text{minimize}} \quad & -g \\ \text{subject to} \quad & \rho(\mathscr{A}(\theta, g)) \leq 1, \\ & C(\theta) \leq \bar{C}, \\ & \phi_i(\theta) \leq \bar{\phi}_i,\ i = 1, \dots, \ell. \end{aligned} \tag{3.4}$$

Performing the variable transformations

$$u = \log[\theta], \quad v = \log g$$

as well as taking logarithms in the objective functions and constraints, we can equivalently reduce (3.4) into the optimization problem in Theorem 1. Therefore, to complete the proof of theorem, we need to show the convexity of the optimization problem in Theorem 1. The convexity of the constraints in Theorem 1 is a direct consequence of the superconvexity of posynomials stated in Lemma 4. In the remaining of this section, we shall show the convexity of the mapping

$$\log[\Theta] \times \mathbb{R} \to \mathbb{R} \colon (u, v) \mapsto \log \rho(\mathscr{A}(\exp[u], e^v)).$$

For each $k = 1, \dots, N$, we define the matrix function

$$\tilde{A}_k(\theta) = A_k(\theta) - M_k + gI.$$

Then, equation (3.2) shows that

$$[\mathscr{A}(\theta, g)]_{ij} = p_{ji} \int_0^T e^{(\tilde{A}_j(\theta)+M_j)\tau} f_{ji}(\tau)\, d\tau,$$

where f_{ji} denotes the probability density function of the sojourn time h_{ji}. This equation and the Lie-product formula

$$e^{A+B} = \lim_{K\to\infty} (e^{A/K} e^{B/K})^K$$

for square matrices A and B (see, e.g., [63]) yield that

$$\begin{aligned}[\mathscr{A}(\theta, g)]_{ij} &= p_{ji} \int_0^T \lim_{K\to\infty} \left(e^{\frac{\tau \tilde{A}_j(\theta,g)}{K}} e^{\frac{\tau M_j}{K}} \right)^K f_{ji}(\tau)\, d\tau \\ &= \lim_{K,L\to\infty} \mathit{\Gamma}_{ij}^{(K,L)}(\theta, g)\end{aligned}$$

where, for positive integers K and L, the $n \times n$ matrix $\mathit{\Gamma}_{ij}^{(K,L)}(\theta, g)$ is defined by

$$\mathit{\Gamma}_{ij}^{(K,L)}(\theta, g) = p_{ji} \sum_{\ell=1}^{L} \frac{T}{L} \left(e^{\frac{\ell T \tilde{A}_j(\theta,g)}{KL}} e^{\frac{\ell T M_j}{KL}} \right)^K f_{ji}(\ell T/L).$$

Therefore, if we define

$$\mathit{\Gamma}_{ij}^{(K,L,M)}(\theta, g) = \\ p_{ji} \sum_{\ell=1}^{L} \frac{T}{L} \left(\sum_{m=0}^{M} \frac{1}{m!} \left(\frac{\ell T \tilde{A}_j(\theta, g)}{KL} \right)^m e^{\frac{\ell T M_j}{KL}} \right)^K f_{ji}\left(\frac{\ell T}{L}\right)$$

then, by the definition of matrix exponentials, we obtain the following expression:

$$[\mathscr{A}(\theta, g)]_{ij} = \lim_{K,L,M\to\infty} \mathit{\Gamma}_{ij}^{(K,L,M)}(\theta, g). \tag{3.5}$$

Let us show that each entry of the matrix $\mathit{\Gamma}_{ij}^{(K,L,M)}$ is either a posynomial in θ and g or zero. Since the matrix M_j is assumed to be Metzler (Assumption 1.1), the matrix $e^{\ell T M_j/KL}$ is nonnegative for all K and L. Also, each entry of the matrix $\tilde{A}(\theta, g)$ is either a posynomial or zero by Assumption 1.1. Since the set of posynomials is closed under additions and multiplications, each entry of the matrix power $(\ell T \tilde{A}_j(\theta, g)/KL)^m$ is either a posynomial of θ and g or zero as well. From the above observation, we conclude that each entry of the matrix $\mathit{\Gamma}^{(K,L,M)}(\theta, g)$ is a posynomial with the variables θ and g or zero.

We are now ready to complete the proof of the theorem. Define the $(nN)\times(nN)$ matrix $\mathscr{A}^{(K,L,M)}(\theta, g)$ as the block matrix whose (i, j)-block equals $\mathit{\Gamma}_{ij}^{(K,L,M)}(\theta, g)$ for all $i, j \in \{1, \dots, N\}$. Then, by Lemmas 3 and 4, the mapping

$$(u, v) \mapsto \rho(\mathscr{A}^{(K,L,M)}(\exp[u], e^v))$$

is superconvex. Since (3.5) shows that the mapping $\mathscr{A}$ is a point-wise limit of the mapping $\mathscr{A}^{(K,L,M)}$, taking a limit preserves superconvexity, and the spectral radius operator $\rho(\cdot)$ is continuous, we obtain the convexity of the mapping (3.2). This completes the proof of convexity of the optimization problem (3.3), as desired. □

Remark 2. From the proof of Theorem 1, we see that Problem 1 can be formulated as the problem (3.4) even without Assumption 1. Assumption 1 then allows us to reduce the optimization problem (6) into a convex optimization problem.

Corollary 1. *We can also show that a performance-constrained counterpart of Problem 1 can be solved via convex optimization. Let us consider the following optimization problem: For a given $\bar{\gamma} > 0$, find $\theta \in \Theta$ such that the cost function $C(\theta)$ is minimized, while the requirement*

$$\gamma_\theta \geq \bar{\gamma}$$

In the same way as the proof of Theorem 1, we can show that the solution of the optimization problem is given by

$$\theta = \exp[u^\star],$$

where $u = u^\star$ solves the following convex optimization problem:

$$\begin{aligned} \underset{u \in \log[\Theta],\, v \in \mathbb{R}}{\text{minimize}} \quad & \log C(\exp[u]) \\ \text{subject to} \quad & \log \rho(\mathscr{A}(\exp[u], e^v)) \le -\log \bar{\gamma}, \\ & \log \phi_i(\exp[u]) \le \log \bar{\phi}_i, \\ & \qquad\qquad i = 1, \dots, \ell. \end{aligned}$$

The proof of this corollary is the same manner as Theorem 1.

3.3 Example: biological population control

In this section, we illustrate the validity and effectiveness of the main results with an example in the context of population biology [55]. Many biological populations are exposed to environmental fluctuations, from daily regular cycles of light and temperature to irregular fluctuations of nutrients and pH levels. To survive through the fluctuating environment, many biological populations employ a protection mechanism called *bet-hedging* to increase robustness against the fluctuations of environment. In brief, bet-hedging means that the biological populations exhibit several phenotypes that have different growth rates among the possible environments.

For the bet-hedging population model, we consider a biological community with n phenotypes living in a randomly fluctuating environment with N possible environmental types. Let g_k^i denote the growth rate of phenotype i under environment k. In the bet-hedging population, individuals may switch their phenotype at any time but stochastically. We let ω_k^{ji} denote the instantaneous rate at which an individual having phenotype j switches its phenotype to i under environment k. Let $x_i(t)$ denote the number of individuals having phenotype i at time t and $\sigma(t)$ denote the environment type at time t. Then, the dynamics of population in phenotype i can be expressed by the following differential equation [55]

$$\Sigma : \frac{dx_i}{dt} = g^i_{\sigma(t)} x_i(t) + \sum_{j \neq i}^{n} \omega^{ji}_{\sigma(t)} x_j(t),$$

where $\omega_k^{ii} = -\sum_{j=1, j\neq i}^{n} \omega_k^{ij}$. The fluctuation is governed by a semi-Markov process $\sigma(t) \in \{1, \dots, N\}$ as mentioned in Definition 2. Fig. 3.1 shows a schematic picture of this model for $n = 2$ and $N = 2$, i.e., individuals present two types of phenotypes in two environments.

Let us consider the problem of driving the entire population into extinction through biological intervention. Assume that L different types of antibiotics are available for suppressing growth rates. Let $c_\ell(\alpha_\ell)$ ($\ell \in \{1, \dots, L\}$) denote the cost for dosing α_ℓ unit of the ℓth antibiotics, which is assumed to reduce the growth rate of the ith phenotype population by $\Delta_\ell g^i(\alpha_\ell)$ independent of the current environment types. In this situation, we can reduce the growth rate of the ith phenotype population to $g^i_{\sigma(t)} - \sum_{\ell=1}^L \Delta_\ell g^i(\alpha_\ell)$ with the associated total cost $C(\alpha) = \sum_{\ell=1}^L c_\ell(\alpha_\ell)$. The resulting population dynamics admits the representation

$$\Sigma' : \frac{dx_i}{dt} = \left(g^i_{\sigma(t)} - \sum_{\ell=1}^{L} \Delta_\ell g^i(\alpha_\ell) \right) x_i(t) + \sum_{j \neq i}^{n} \omega^{ji}_{\sigma(t)} x_j(t).$$

Let us allow the following box constraint

$$0 \leq \alpha_\ell \leq \bar{\alpha}_\ell \tag{3.6}$$

on the amount of doses. Under this scenario, we consider the following optimal intervention problem:

Problem 3 (Optimal intervention problem)**.** *Let $\bar{C}$ be a positive constant. Assume that σ is a semi-Markov process satisfying Assumption 1.4. Find the set of dose amounts $\alpha = (\alpha_1, \dots, \alpha_L)$ to maximize the exponential decay rate of the system Σ' while satisfying*

$$C(\alpha) \leq \bar{C}. \tag{3.7}$$

In this numerical example, we assume that the cost for antibiotics is linear with their dose amount, i.e., we let

$$c_\ell(\alpha_\ell) = r_\ell \alpha_\ell$$

for a constant $r_\ell > 0$ for all ℓ. As for the suppression $\Delta_\ell g^i(\alpha_\ell)$ of the growth rates, we adopt the increasing function that presents diminishing marginal benefit on the dosage

$$\Delta_\ell g^i(\alpha_\ell) = s^i_\ell \frac{\underline{\theta}_\ell^{-q_\ell} - (\alpha_\ell + \underline{\theta}_\ell)^{-q_\ell}}{\underline{\theta}_\ell^{-q_\ell} - (\bar{\alpha}_\ell + \underline{\theta}_\ell)^{-q_\ell}}$$

where $\underline{\theta}_\ell > 0$, $s_\ell^i \geq 0$, and $q_\ell > 0$ are parameters. These parameters allow us to realize various shapes of the suppression functions, including the dose-proportional suppression illustrated in Fig. 3.2. We notice that the zero dose of the ℓth antibiotic does not change the growth rate, i.e., $\Delta_\ell g^i(\alpha_\ell)(0) = 0$, while the maximum dose achieves the full performance $\Delta_\ell g^i(\bar{\alpha}_\ell) = s_\ell^i$.

Let us show that the optimal intervention problem reduces to Problem 1. We introduce an auxiliary variable

$$\theta_\ell = \underline{\theta}_\ell + \alpha_\ell$$

that is to be optimized. If we define $\bar{\theta}_\ell = \underline{\theta}_\ell + \bar{\alpha}_\ell$, then the constraint (3.6) is rewritten as the block constraint

$$\underline{\theta}_\ell \leq \theta_\ell \leq \bar{\theta}_\ell,$$

which can be expressed using posyonmial functions [60]. Therefore, Assumption 1.3 is satisfied. Let us define the variable $\theta = (\theta_1, \dots, \theta_L)$. Then, we can rewrite the system Σ' into the form (3.1), where the matrices $A_1(\theta)$, ..., $A_N(\theta)$ are defined by

$$[A_k(\theta)]_{ii} = \tilde{g}_k^i + \sum_{\ell=1}^{L} s_\ell^i (\underline{\theta}_\ell^{-q_\ell} - \bar{\theta}_\ell^{-q_\ell})^{-1} \theta_\ell^{-q_\ell}$$

with $\tilde{g}_k^i = g_k^i - \omega_k^{ii} - \sum_{\ell=1}^{L} s_\ell^i \underline{\theta}_\ell^{-q_\ell} / (\underline{\theta}_\ell^{-q_\ell} - \bar{\theta}_\ell^{-q_\ell})$, and

$$[A_k(\theta)]_{ij} = \omega_k^{ij}$$

for $i \neq j$. Therefore, if we define the diagonal matrix $M_k = \operatorname{diag}(\tilde{g}_k^1, \dots, \tilde{g}_k^n)$, then each entry of the matrix $A_k(\theta) - M_k$ is a posynomial in the variables θ or zero. Hence, Assumption 1.1 is satisfied. Furthermore, the cost constraint (3.7) can be rewritten as

$$\sum_{\ell=1}^{L} r_\ell \theta_\ell \leq \bar{C} + \sum_{\ell=1}^{L} r_\ell \underline{\theta}_\ell$$

in terms of posynomials of the variable θ. Since all the conditions in Assumption 1 are satisfied, the optimal intervention problem can be efficiently solved by convex optimization as shown in Theorem 1.

For simplicity of presentation, we focus on the case of $n = N = 2$ in this numerical example. Throughout the simulation, we fix a part of the parameters as follows: $\omega_1^{21} = \omega_1^{12} = 0.1$, $\omega_2^{21} = \omega_2^{12} = 0.5$, $g_1^1 = 1$, $g_1^2 = -0.5$, $g_2^1 = -1$, $g_2^2 = 0.5$, $\bar{\alpha}_1 = \bar{\alpha}_2 = 1$, $\underline{\theta}_1 = \underline{\theta}_2 = 10$, $q_1 = q_2 = 0.01$, $s_1^1 = s_1^2 = s_2^1 = s_2^2 = 1$, $c_1 = c_2 = 1$, and $\bar{C} = 2$. Also, we assume that the environment keeps switching from one to another, and that the sojourn time of each environment follows the Weibull distribution having the probability density function

$$f_{ij}(h_{ij}) = \begin{cases} \gamma_{ij} \frac{k_{ij}}{\lambda_{ij}} \left(\frac{h_{ij}}{\lambda_{ij}}\right)^{k_{ij}-1} e^{-(h_{ij}/\lambda_{ij})^{k_{ij}}}, \\ \qquad\qquad\qquad\qquad \text{if } 0 < h_{ij} \leq T, \\ 0, \quad \text{otherwise,} \end{cases} \tag{3.8}$$

where λ_{ij} is the range parameter for adjusting the expectation of the sojourn time h_{ij}, and k_{ij} is the shape parameter. We truncate this density function at a finite time T to satisfy Assumption 1.4, and γ_{ij} is the constant for normalizing the integral of the truncated density function. We remark that setting $k_{ij} = 1$ recovers the case of sojourn times following exponential distributions (i.e., the Markovian case).

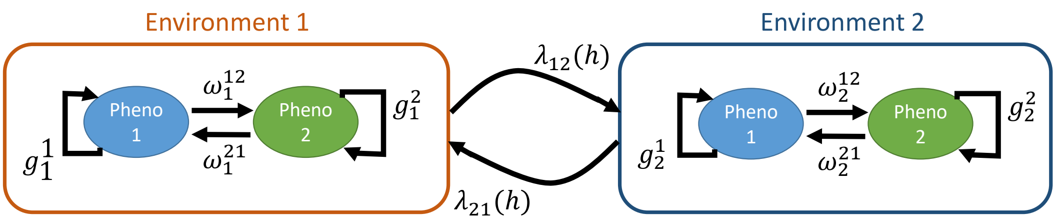


Figure 3.1: Bet-hedging population

In our simulation, we investigate the trade-off between the parameter λ_{ij} and k_{ij} to see the impact of the extension from Markov process to semi-Markov process. For problem solving, we adopt the commonly used off-the-shelve software for convex optimization problem: **fmincon** routine in MATLAB. For this purpose, we consider the situation in which the parameter λ_{ij} and k_{ij} of the probability density function f_{ij} in (3.8) can be tuned under the constraint of equally fixed expectation $E[h_{12}] = E[h_{21}] = 6$. For various values of λ_{12} and k_{21}, we present the values of the optimized exponential decay rate in Fig. 3.3. We can observe that the optimized decay rate nontrivially depends on k_{21} (i.e., the shape of the density f_{21}). Specifically, when $k_{21} > 1$, modeling the bet-hedging population via

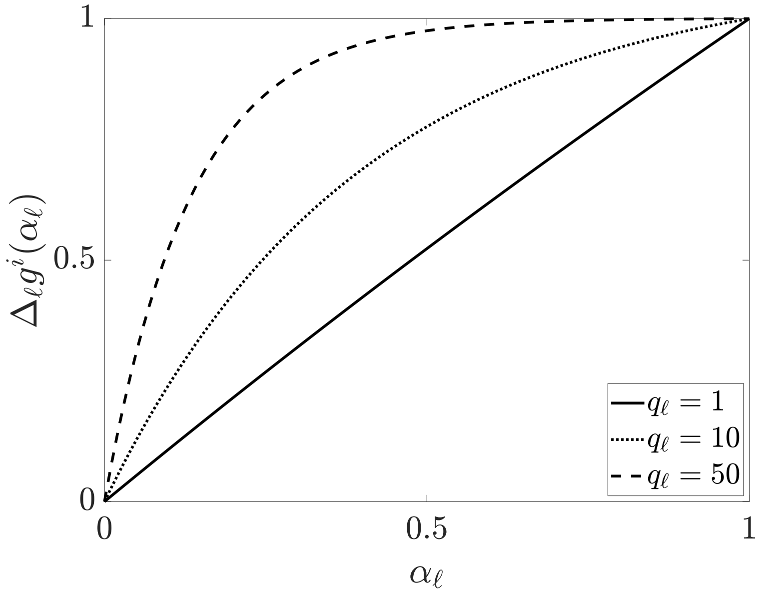


Figure 3.2: Three realizations of the doasge-performance function with the parameter.

semi-Markov process can increase the stability of system. On the other hand, in the case of $k_{21} < 1$, we can see that the stability deteriorates. This observation shows that the trade-off between the parameter of semi-Markov process is not trivial on the stability of system. Also, it is clearly seen that our proposed framework extended the possible optimized solutions compared with the result in Markov jump linear systems, in which the optimized solutions are only represented by the dashed line in Fig. 3.3. The biological population of phenotype 1 before and after medical intervention is illustrated in Fig. 3.4.

Let us also investigate the dependency of the optimized decay rate on the shape parameter q_ℓ of the dosage-performance function (3.3). In this simulation, we change the values of q_1 and q_2 from 1 to 100 under the constraint $q_1 = q_2$. We also change the value of the budget $\bar{C}$ in the interval $[1, 2]$. We solve the optimal intervention problem for various pairs of $q_1 = q_2$ and $\bar{C}$ and obtain the optimized decay rates, as shown in Fig. 3.5. We can see that for the fixed budget on total cost $\bar{C}$, the increase on q_ℓ results the smaller decay rate, i.e., the stronger the diminishing property of the antibiotic, the higher decay rate can be obtained.

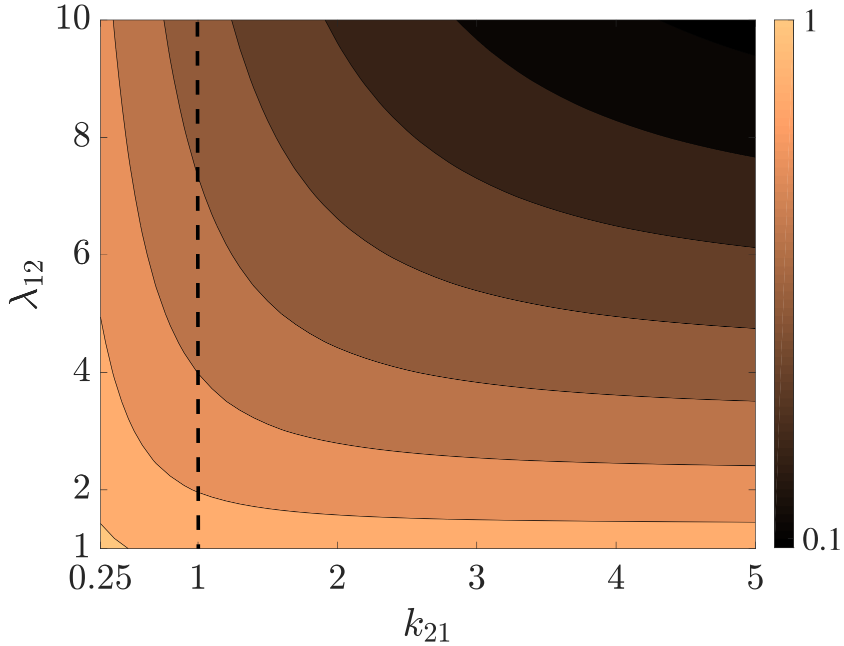


Figure 3.3: The optimized exponential decay rate for various values of λ_{12} and k_{21}. Dashed line indicates the optimized decay rate under the Markov process ($k_{12} = k_{21} = 1$).

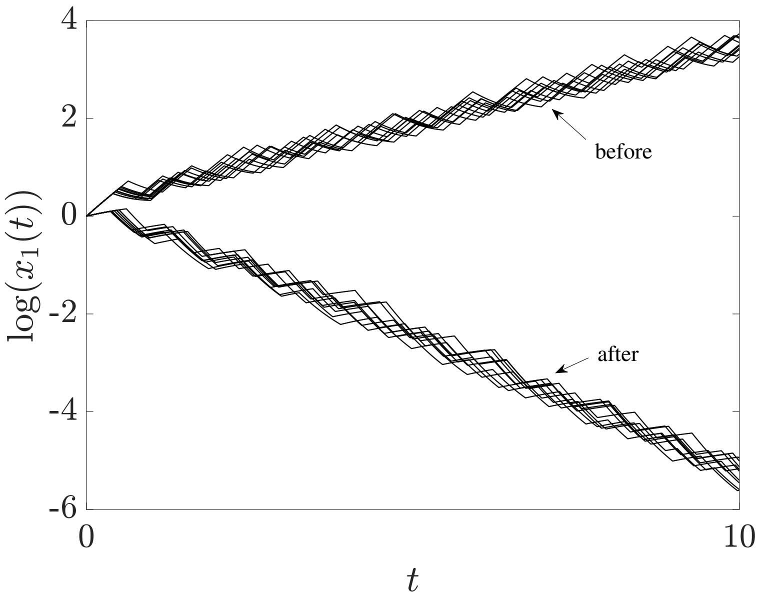


Figure 3.4: 10 realizations of the biological populations of phenotype 1 in the log form when $E[h_{12}] = E[h_{21}] = 6$ and $\lambda_{12} = \lambda_{21} = 6.67$ and $k_{12} = k_{21} = 5$. In this situation, $\sigma(t)$ follows the semi-Markov process.

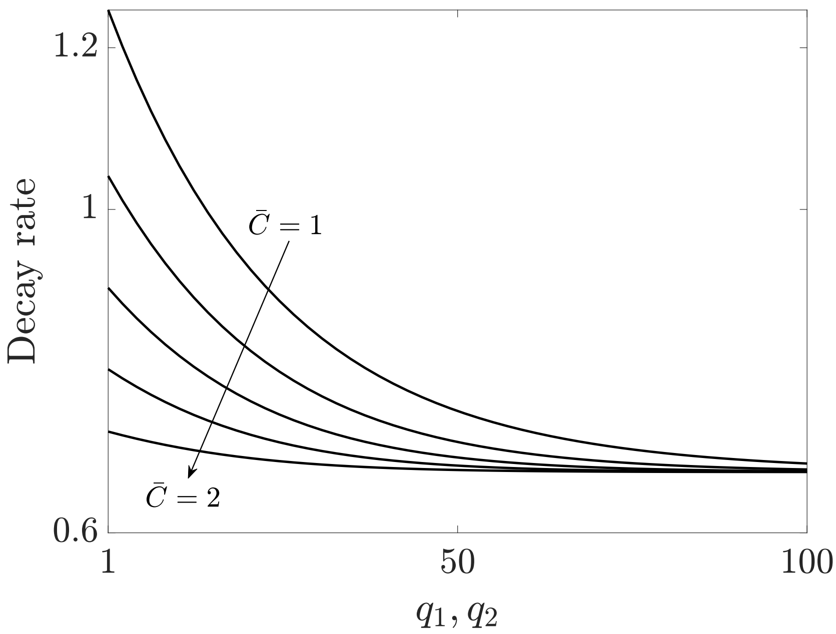


Figure 3.5: The optimized exponential decay rate for $\bar{C}$ = 1, 1.2, 1.4, 1.6, 1.8, and 2 with the variation of parameters of the dosage-performance functions $q_1 = q_2$.

4 Finite-time control for discrete-time positive linear systems with time-varying state via convex optimization

In this chapter, we study a class of finite-time control problems for discrete-time positive linear systems with time-varying state parameters, where the state variables are functions of time. From the property of posynomials, it is easily derived that the discrete time series can be illustrate by posynomials. Hence, there exists the possibility that both the parameter tuning function and the stability constraints in the discrete-time positive linear systems could be expressed by posynomials. Therefore, the motivation lies that the Finite-time control problem could be solved by geometric programming. Although several interesting control problems appearing in population biology, economics, and network epidemiology can be described as the class of finite-time control problems, an efficient solution to the control problem has not been yet found in the literature. In this chapter, we propose an optimization framework for solving the class of finite-time control problems via convex optimization. We illustrate the effectiveness of the proposed method by a numerical simulation in the context of dynamical product development processes.

4.1 Finite-time control problem

In this section, we describe the finite-time control problem studied in this chapter. In Section 4.1.1, we describe the system studied in this chapter and state the necessary assumptions. In Section 4.1.2, we formulate the finite-time control problem as an optimization problem.

4.1.1 System Model

In this chapter, we consider the following parametrized time-varying linear system defined on a finite time interval:

$$\Sigma_\theta : x(k+1) = (A(k) + K(k; \theta(k)))x(k),\ k = 0, \dots, T,$$

where $x(k) \in \mathbb{R}^n$ is the state variable, $A(k) \in \mathscr{A} \subset \mathbb{R}^{n\times n}$ $(k = 0, \dots, T)$ is a time-varying state matrix, and

$$K(k; \theta(k)) \in \mathscr{K} \subset \mathbb{R}^{n\times n},\ k = 0, \dots, T,$$

is the control matrix parametrized by the vector $\theta(k)$ belonging to a set $\Theta \subset \mathbb{R}^{n_\theta}$. We assume that the set $\mathscr{K}$ is bounded. Our objective in this chapter is to present an optimization framework for tuning the parameter $\theta(k)$ in such a way that the finite-time stability of the system is guaranteed, under the positivity assumption on the system. The positivity of discrete-time time-varying linear systems is formally defined as follows.

Definition 4. *[32] We say that the time-varying linear system*

$$\Sigma : x(k+1) = M(k)x(k)$$

*is (*internally*)* positive *if for any initial condition $x(0)$ with nonnegative entries, the corresponding state trajectory $x(k)$ is nonnegative for all $k \geq 0$.*

For positive time-varying linear systems, we define the notion of finite-time stability [12] as follows.

Definition 5. *Let T be a positive integer. Suppose that a positive number ϵ as well as positive vectors v and $\ell(k)$ $(k \in \{1, \dots, T\})$ are given. We say that Σ is* finite-time stable *if the trajectory of the system satisfies*

$$x^\top(k)\ell(k) < \epsilon,\ k = 1, \dots, T,$$

for all initial states $x(0)$ satisfying $x^\top(0)v \leq 1$.

In this chapter, we place the following assumption on the parameterized system Σ_θ for ensuring its positivity.

Assumption 2. *The matrix $A + K$ is nonnegative for all $A \in \mathscr{A}$ and $K \in \mathscr{K}$.*

We then introduce cost and performance functions as follows. For each control action $K(k; \theta(k))$, the control parameter $\theta(k)$ at time k comes with an associated cost. In this chapter, we suppose that a cost function for tuning the parameter $\theta(k)$ is given by the following functional:

$$L \colon \mathbb{R}^{n_\theta} \to \mathbb{R} \colon \theta(k) \mapsto L(\theta(k)).$$

Let $x(\cdot; \theta(k))$ denote the solution of the system Σ_θ. In order to measure the stability property of the system, we use the functional

$$J(\theta(k)) = \|x(\cdot; \theta(k))\|_p$$

where $p > 0$ is a constant and $\|\cdot\|_p$ denotes the ℓ^p-norm of a sequence of real vectors.

4.1.2 Problem Formulation

In this section, we present two types of optimization problems for the finite-time control of the parametrized system Σ_θ. We first present the budget-constrained optimization problem to minimize J while satisfying the constraint on the cost function L as well as the finite-time stability. Formally, the budget-constrained optimization problem is stated as follows:

Problem 4. *Let a constant $\bar{L}$ be given. Find a sequence of variables $\theta = \{\theta(k)\}_{k=0}^{T}$ such that*

$$L(\theta) \leq \bar{L}$$

and the system Σ_θ is finite-time stable in the sense of Definition 5, while minimizing the cost function $J(\theta)$.

Mathematically, the budget-constrained finite-time control problem can be stated as

$$\begin{aligned} \underset{\theta \in \Theta^{T+1}}{\text{minimize}} \quad & J(\theta) && (4.1a) \\ \text{subject to} \quad & x^\top(k;\theta(k))\ell(k) < \epsilon, && (4.1b) \\ & L(\theta) \le \bar{L}. && (4.1c) \end{aligned}$$

Likewise, by exchanging the roles of the objective function and constraints in the budget-constrained optimization problem, we obtain the performance-constrained optimization problem

Problem 5. *Let a constant $\bar{J}$ be given. Find a sequence of variables $\theta = \{\theta(k)\}_{k=0}^{T}$ such that*

$$J(\theta) \le \bar{J}$$

and the system Σ_θ is finite-time stable in the sense of Definition 5, while minimizing the cost function $L(\theta)$.

As in (4.1), we can mathematically formulate the performance-constrained finite-time control problem as the following:

$$\begin{aligned} \underset{\theta \in \Theta^{T+1}}{\text{minimize}} \quad & L(\theta) \\ \text{subject to} \quad & x^\top(k;\theta(k))\ell(k) < \epsilon, \\ & J(\theta) \le \bar{J}. \end{aligned}$$

4.2 Main result

In this section, we present our optimization framework for solving the budget-constrained and performance-constrained finite-time control problems. Under proper assumptions, we show that the problems can be transformed into convex optimization problems.

In Problems 4 and 5, the functional $J(\theta)$ is typically a nonlinear function. Furthermore, the cost functional $L(\theta)$ is often nonlinear in applications due to the their physical characteristics such as the dosage-effect relation in the therapy control processes. For these reasons, Problems 4 and 5 are not trivial to solve directly. However, in this chapter, we show that a mild set of assumptions allow us to reduce the problems to geometric programming, which can be efficiently solved via convex optimization [53].

Let us first give a brief overview of geometric programming. We start from stating the following definition.

Definition 6. *[53] Let $v_1, \ldots, v_n$ denote n real positive variables. We say that a real function $g(v)$ is a monomial if there exist $c > 0$ and $a_1, \ldots, a_n \in \mathbb{R}$ such that $g(v) = cv_1^{a_1} \cdots v_n^{a_n}$. We say that a real function $f(v)$ is a posynomial if f is a sum of monomials of v.*

The following lemma shows the log-convexity of posynomials.

Lemma 5. *[53] Let $f\colon \mathbb{R}_+^n \to \mathbb{R}_+\colon x \mapsto f(x)$ be a posynomial function. Then, the function*

$$F\colon \mathbb{R}^n \to \mathbb{R}\colon w \mapsto \log f(\exp[w])$$

is convex.

The log-convexity of posynomials allows us to solve a class of optimization problems called geometric programs efficiently, as summarized in the following proposition [53].

Proposition 2. *Let $g_1(\theta), \ldots, g_q(\theta)$ be monomials and $f_0(\theta), \ldots, f_p(\theta)$ be posynomials. Assume that variables $\theta \in \Theta$ satisfy Definition 6. We say that the following optimization problem*

$$\begin{aligned} \underset{\theta \in \Theta}{\text{minimize}} \quad & f_0(\theta) \\ \text{subject to} \quad & f_i(\theta) \leq 1,\ i = 1, \ldots, p, \\ & g_j(\theta) = 1,\ j = 1, \ldots, q, \end{aligned}$$

can be transformed into a convex optimization problem through the logarithmic variable transformation

$$\theta = \exp[z],\ z \in \Gamma \subset \mathbb{R}^m.$$

Then, we obtain the convex optimization problem with the following form:

$$\begin{aligned} \underset{z \in \Gamma}{\text{minimize}} \quad & \log f_0(\exp[z]) \\ \text{subject to} \quad & \log f_i(\exp[z]) \leq 0,\ i = 1, \dots, p, \\ & \log g_j(\exp[z]) = 0,\ j = 1, \dots, q. \end{aligned}$$

To exploit the log-convexity of posynomials, we first place the following assumption on the structure of the parametrized time-varying linear system Σ_θ:

Assumption 3. *Define the matrix* $K_{\min} \in \mathbb{R}^{n \times n}$ *by*

$$[K_{\min}]_{ij} = \inf\{K_{ij} \colon K \in \mathscr{K}\}.$$

Then, the matrix

$$\tilde{A}(k) = A(k) + K_{\min} \tag{4.4}$$

is nonnegative for all k*.*

We remark that, in Assumption 3, the existence of the matrix $K_{\min}$ is guaranteed by the boundedness of the set K. Furthermore, this assumption is not very restrictive and is satisfied in the examples that we discuss in this section.

Using the matrix $\tilde{A}$ in (4.4), we rewrite the parametrized system Σ_θ as

$$\Sigma_\theta : x(k+1) = (\tilde{A}(k) + \tilde{K}(k; \theta(k)))x(k),\ k \in \{0, \dots, T\},$$

where

$$\tilde{K}(k; \theta(k)) = K(k; \theta(k)) - K_{\min}$$

is a nonnegative matrix. For this nonnegative matrix as well as the parameter space Θ, we place the following assumption.

Assumption 4. *The following conditions hold true:*

1. *There exist a sequence of the posynomials* $f_1(\theta)$, …, $f_N(\theta)$ *such that*

$$\Theta = \{\theta \in \mathbb{R}^m_{++} \colon f_1(\theta) \leq 1, \dots, f_N(\theta) \leq 1\}.$$

2. There exist posynomials $\kappa_{ij}(k;\theta(k))$ ($i,j \in \{1,\ldots,n\}$ and $k \in \{0,\ldots,T\}$) such that

$$\tilde{K}(k;\theta(k)) = \{[\kappa_{ij}(k;\theta(k))]_{i,j} \colon \theta \in \Theta\}$$

3. $L(\theta)$ is a posynomial.

We can now present the first main result of this chapter; namely, we can show that Problem 4 can be solved via convex optimization.

Theorem 4.2.1. *The solution of the following convex optimization problem is given by $z = \{z(k)\}_{k=0}^{T}$, where $z(k)$ belongs to the set $\Gamma \subset \mathbb{R}^m$.*

$$\begin{aligned}
&\underset{z\in\Gamma^{T+1}}{\text{minimize}} && \log J(\exp[z]) && \text{(4.5a)}\\
&\text{subject to} && \log x^\top(k;\exp[z(k)])\ell(k) < \log\epsilon, && \text{(4.5b)}\\
& && \log L(\exp[z]) \leq \log \bar{L}. && \text{(4.5c)}
\end{aligned}$$

Then, the solution of Problem 4 is given by

$$\theta(k) = \exp[z(k)]. \tag{4.6}$$

Proof. Under the log transformation $z(k) = \log[\theta(k)]$, the constraints (4.1a), (4.1b) and (4.1c) are equivalent to the constraints (4.5a), (4.5b) and (4.5c), respectively. Therefore, the solution of Problem 4 given by (4.6) becomes the solutions of optimization problem (4.5). From Lemma 5, we can get that (4.5c) is convex if the cost function $L(\theta)$ follows the posynomials. Also, for the convexity of (4.5b), $x^\top(k;\theta(k))\ell(k)$ is a linear function which is definitely a posynomial function. For the convexity of (4.5a), we can derive that the entry of state vector is posynomials from the expansion of $x(k;\theta(k))$:

$$\begin{aligned}
x(k;\theta(k)) = (\tilde{A}(k-1) + \tilde{K}(k-1;\theta(k-1)))\cdots\\
(\tilde{A}(0) + \tilde{K}(0;\theta(0)))x(0).
\end{aligned}$$

From the previous assumptions, we can see that if the performance measurement function follows posynomials, $J(\theta)$ is convex under the log transformation. From Proposition 2, we can see that Theorem 4.2.1 is a convex optimization problem. □

Likewise, the performance-constrained form of finite-time control problem can also be solved through the following optimization problem:

Corollary 2. *The solution of the following convex optimization problem is given by* $z = \{z(k)\}_{k=0}^{T}$*, where* $z(k)$ *belongs to the set* $\Gamma \subset \mathbb{R}^m$*.*

$$\begin{aligned} \underset{z \in \Gamma^{T+1}}{\text{minimize}} \quad & \log L(\exp[z]) \\ \text{subject to} \quad & \log x^\top(k; \exp[z(k)])\ell(k) < \log \epsilon, \\ & \log J(\exp[z]) \le \log \bar{J}. \end{aligned}$$

Then, the solution of Problem 5 is given by

$$\theta(k) = \exp[z(k)].$$

4.3 Example: Product Development Management

In this section, we illustrate the effectiveness of our proposed framework by solving the dynamic optimal resource allocation problem for the automotive appearance design process in the car manufacturing industry.

In this chapter, we adopt the automotive appearance design example presented in [35], which contains the following tasks: 1) carpet, 2) center console, 3) door trim panel, 4) garnish trim, 5) overhead system, 6) instrument panel, 7) luggage trim, 8) package tray, 9) seats and 10) steering wheel. Suppose there are T development rounds during the process, the dynamic process of the remaining work on each task can be represented by the discrete-time positive linear system $x(k+1) = Ax(k),\ k \in \{0, \ldots, T\}$, where $x(k)$ is the remaining work vector, A is the work transition matrix which is nonnegative. In this chapter, we adopt the dynamic model in [8]

$$A_k(\phi_k, \gamma_k) = \begin{bmatrix} \phi_{1,k} + \Delta_1 & \cdots & \prod_{\ell=1}^{k} \gamma_{1n,\ell} \\ \prod_{\ell=1}^{k} \gamma_{21,\ell} & \cdots & \prod_{\ell=1}^{k} \gamma_{2n,\ell} \\ \vdots & \ddots & \vdots \\ \prod_{\ell=1}^{k} \gamma_{n1,\ell} & \cdots & \phi_{n,k} + \Delta_n \end{bmatrix},$$

where the value of the off-diagonal entries of the work transition matrix is updated with the accumulated effect in the previous investment rounds $(k-1, k-2, \ldots, 0)$. $\phi_k = \{\phi_{1,k}, \ldots, \phi_{n,k}\}$ represents the adjustable work efficiency of the task, while $\gamma_k = \{\gamma_{ij,k}\}, (i, j = 1, \ldots, n, i \neq j)$ are the off-diagonal entires of $A_k(\phi_k, \gamma_k)$ which represent the ratio of the extra work transferred among the tasks with progress. Furthermore, we let $\Delta_{:,k}$ to represent the abrupt change on ϕ_k (e.g., equipment fault, conflict on schedule or the absence of engineer). During the intermittence of the development process, the managers allocate a fixed amount of resource to prompt the development process (i.e., tuning the parameter of work transition matrix). We assume that the parameters can be tuned within the following intervals:

$$0 < \phi_{i,k}^{\min} \leq \phi_{i,k} \leq \phi_{i,k}^{\max}, \quad 0 < \gamma_{ij,k}^{\min} \leq \gamma_{ij,k} \leq \gamma_{ij,k}^{\max}.$$

Specifically, the resource can be allocated on the tasks (i.e., diagonal entries of $A_k(\phi_k, \gamma_k)$) to promote the efficiency, or on the off-diagonals to reduce the ratio of the generated work among the related tasks. Furthermore, suppose that the initial value of $A_k(\phi_k, \gamma_k)$ is given by $\phi_{i,k}^{\max}, \gamma_{ij,k}^{\max}$, we have to pay $f_i(\phi_{i,k})$ unit of cost for tuning the work efficiency of module i from $\phi_{i,k}^{\max}$ to $\phi_{i,k}$. Likewise, we let the cost for tuning $\gamma_{ij,k}$ equal to $g_{ij}(\gamma_{ij,k})$. The total cost for the kth investment round is calculated by taking the sum of the cost in all the entries of $A_k(\phi_k, \gamma_k)$:

$$B_k(\phi_k, \gamma_k) = \sum_{i,j=1}^{n} (f_i(\phi_{i,k}) + g_{ij}(\gamma_{ij,k})). \tag{4.7}$$

Usually, a dynamic product development process contains dozens or hundreds of tasks and several investment rounds. Moreover, from the discussion in Section 4.2, the dynamic resource allocation problem for product development process is also a nonlinear optimization problem. Thus, finding the optimal strategy is a difficult problem which can not easily be solved by the experience based method. For checking the satisfaction for Assumption 3, we can see that $A_k(\phi_k, \gamma_k),\ k \in \{0, \ldots, T\}$ is a sequence of nonnegative matrices with the directly tuning parameters ϕ_k, γ_k belonging to positive numbers. Then, by utilizing the knowledge in [53], the cost function (4.7) can be modeled with posynomials. Thus, the problem satisfies the assumptions and definitions in our theorem. By using Theorem 4.2.1, we can transform the optimal resource allocation problem

Table 4.1: Work transition matrix of automotive appearance design

	$A_{0,1}$	$A_{0,2}$	$A_{0,3}$	$A_{0,4}$	$A_{0,5}$	$A_{0,6}$	$A_{0,7}$	$A_{0,8}$	$A_{0,9}$	$A_{0,10}$
$A_{0,1}$	0.85	0.12	0.02	0.06	0.06				0.06	
$A_{0,2}$	0.1	0.53	0.04			0.3	0.02		0.24	0.02
$A_{0,3}$	0.02	0.04	0.47	0.08		0.24	0.02		0.18	0.02
$A_{0,4}$	0.06		0.18	0.68		0.14	0.1	0.02	0.08	
$A_{0,5}$	0.04				0.83					
$A_{0,6}$		0.3	0.26	0.16		0.28	0.06		0.02	0.2
$A_{0,7}$		0.02	0.02	0.1		0.06	0.76	0.06	0.04	
$A_{0,8}$				0.1			0.06	0.83	0.16	
$A_{0,9}$	0.08	0.24	0.18	0.08		0.04	0.04	0.16	0.63	0.2
$A_{0,10}$		0.02	0.02			0.26			0.2	0.7

of automotive appearance design process into the finite-time control problem for positive linear system.

In our case study, we select the performance-constrained problem, which aims at minimizing the total investments while satisfying the constraint on the total remaining work. For the problem initialization, we unify the initial value of remaining work with $x(0)_i = 1$, $(i = 1, \dots, 10)$ (i.e., all the tasks at the beginning of development process have 100% work remained). We set the investment rounds $T = 5$ and take the sum of the remaining work after the final investment round $\sum_{i=1}^{n} x_i(T)$ as performance evaluation. Furthermore, we set the constraint value of the total remaining work with $0.001 \times \sum_{i=1}^{n} x_i(0)$ (i.e., the remaining work is 0.1% of the beginning) for judging the accomplish of process. The initial value of $A_k(\phi_k, \gamma_k)$ is given in Table 4.1. Let the entries of $A_k(\phi_k, \gamma_k)$ be tuned within the interval $[0.1, 1]$ (i.e., the component can be accelerated between $[0\% - 90\%]$). Finally, we let the variance on the efficiency of each task $\varDelta_i$ varies between $[-0.2, 0.2]$. For the finite-time stability constraint in (4.1b), we set $\ell(k) = \eta(k)x(0)$, where $\eta(k) = e^{-k}$, $(k = 1, \dots, 5)$, and $\epsilon = 1$.

For the cost function, we adopt the following posynomial function:

$$f_{ij}(\gamma_{ij}) = c_{ij} \left(\frac{1}{(\gamma_{ij})^p} - \frac{1}{(\varOmega_{ij})^p} \right),$$

where $p > 0$ is the parameter for tuning the shape of cost function, and $c_{ij}, \varOmega_{ij}$,

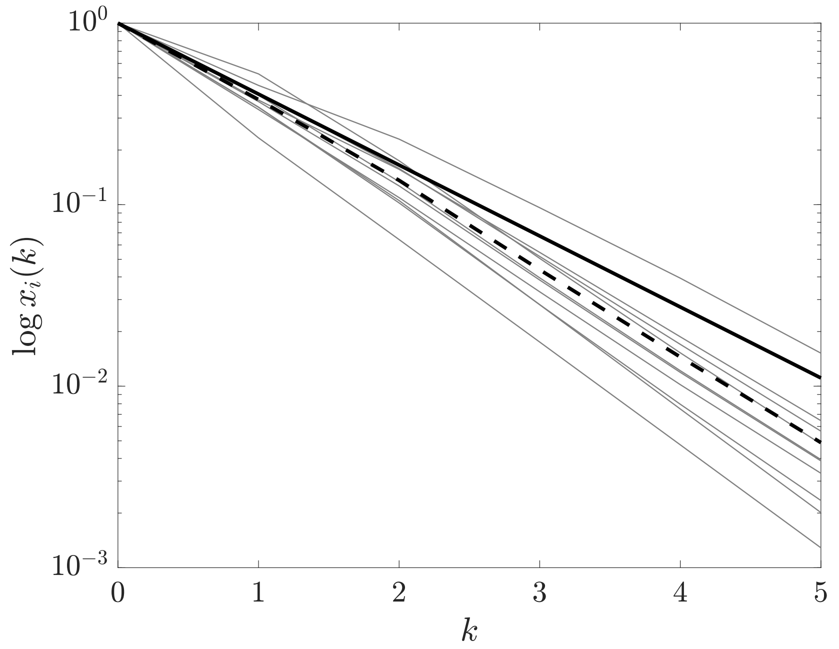


Figure 4.1: Gray line: $\log x_i(k)$; Solid line: finite-time stability constraint; Dashed line: average value of $\log x_i(k)$.

$i, j = \{1, \ldots, 10\}$ are positive numbers for fitting the data. For simplicity, we unify the parameters of all cost functions with $c_{ij} = 1$, $p = 1$, $\Omega_{ij} = 1$. In this case, for example, if $\gamma_{ij} = 1$ (i.e., the corresponding entry in $A_k(\phi_k, \gamma_k)$ is not tuned), then $f_{ij}(\gamma_{ij}) = 0$ which means the cost is 0.

Fig. 4.1 shows that despite satisfying the object function, the dashed line does not exceed the prescribed boundedness (i.e., the designed strategy meets the constraint of finite-time stability). Through solving the convex optimization problem, we are sure to get the optimal decision variables. However, from Fig. 4.2 and Fig. 4.3, we can get the trends of the decision variables γ and ϕ, which means that the manager can foresee the trends before the process is put into effect. The information from Fig. 4.2 and Fig. 4.3 is especially useful for the stage of product development system design, where the manager can modify the structure of the work transition matrix based on the technology of management engineering to improve the performance of the product development system via the earlier design approach.

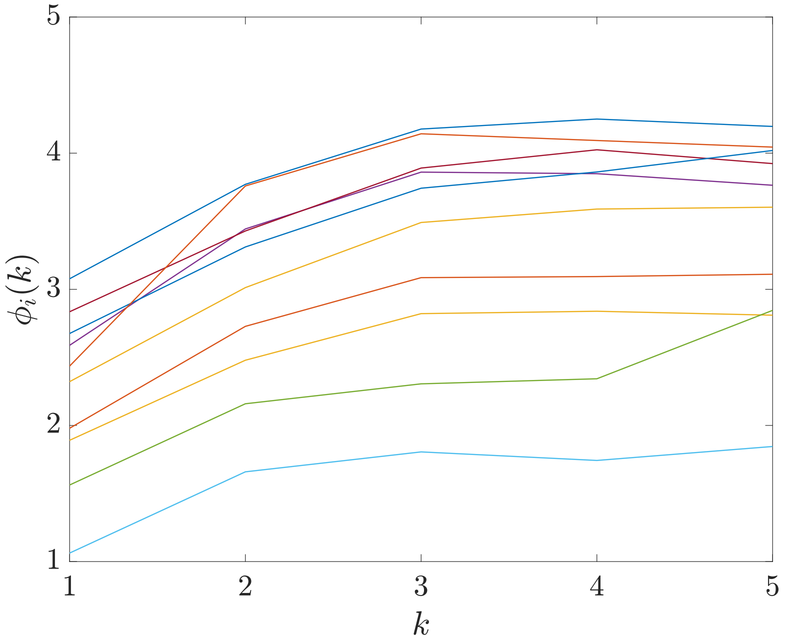


Figure 4.2: The investments in ϕ_i versus investment round k.

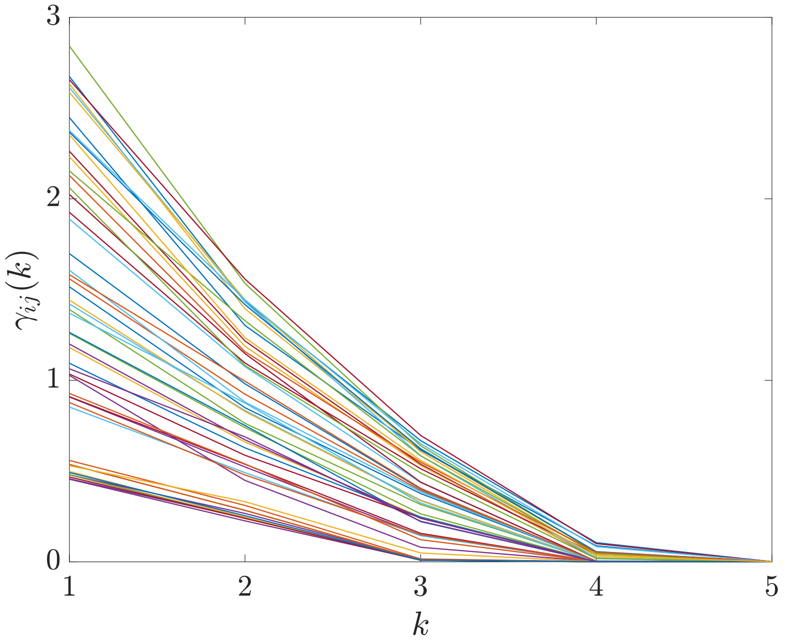


Figure 4.3: The investments in γ_{ij} versus investment round k.

5 Optimal resource allocation of dynamic product development via convex optimization

In this chapter, we first review the dynamic model of the product development (PD) process proposed in [8]. Then, from the perspective of system and control, we show that the work transition feature in the PD process can be expressed by a discrete-time linear system. Finally, we formulate the optimal resource allocation problem of the PD process as the budget-constrained optimization and the performance-constrained problem separately.

5.1 Work transformation matrix

In PD, the product architecture is built not only by the constituent parts that define the product system (i.e., modules or components), but also by the interaction relationships between these parts (i.e., dependency structure) [78]. In this thesis, we assume that the product architecture has been determined in the early design stage. That is, the modules and their dependency structure (i.e., design rules) have been established. In this situation, we focus on improving the performance of PD system through allocating the development resources to the various modules and design rules over all investment rounds (i.e., design iterations).

We start the problem formulation by reviewing the dynamic PD model presented in [8]. Suppose that there are n modules and T investment rounds during the PD process, we let $P_i(k)$ represent the amount of the remaining work in the ith module after finishing the kth investment round. The remaining work of all modules is defined by the vector

$$P(k) = \begin{bmatrix} P_1(k) \\ \vdots \\ P_n(k) \end{bmatrix}.$$

The performance of the PD process is evaluated by the sum of the remaining work in each module [79], which implies that the less total remaining work, the higher performance the product system has. Thus, the sum of the remaining work from all modules can be adopted as a measurement of the performance of the PD system at each round, which is expressed by the following equation:

$$\sum_{i=1}^{n} P_i(k).$$

At each iteration stage, the module finishes a certain amount of remaining work, and sends/receives the produced work (i.e., a fraction of rework) to/from its dependent modules. To describe this work transformation process, we use a discrete-time linear system expressed by the following equation:

$$P(k+1) = A_k(\phi_k, \gamma_k)P(k), \ k = 0, \ldots, T, \tag{5.1}$$

where $A_k(\phi_k, \gamma_k)$ is the work transformation matrix (WTM),

$$\phi_k = \{\phi_{1,k}, \ldots, \phi_{n,k}\}$$

is the vector of the work completion rate of modules, and

$$\gamma_k = \{\gamma_{ij,k}\} \quad (i, j = 1, \ldots, n, \ i \neq j)$$

are the updated value of design rules. The value of the inter-module variable $\gamma_{ij,k}$ represents the work flow strength from module i to j at kth investment round, i.e., at round $k+1$, the accumulated produced work to module j is the sum of the multiplication of the remaining work $P_i(k)$ on module i and $\gamma_{ij,k}$. For an established product architecture, the performance of the product system can be further improved by investing in both modules (i.e., determining the work completion rate in the certain iteration stage) and design rules (i.e., reducing the dependency strength between two modules). We assume that ϕ_k, γ_k can be tuned within the following intervals:

$$0 < \underline{\phi}_{i,k} \le \phi_{i,k} \le \bar{\phi}_{i,k}, \quad 0 < \underline{\gamma}_{ij,k} \le \gamma_{ij,k} \le \bar{\gamma}_{ij,k},$$

where $\bar{\phi}_{i,k}, \bar{\gamma}_{ij,k}$ are the initialized parameter values in (5.1), and $\underline{\phi}_{i,k}, \underline{\gamma}_{ij,k}$ are the limitations of parameters. For the PD process with multiple investment rounds, the authors in [8] showed that unlike the memoryless feature in the investment for the modules, the investment in the design rules for reducing the work flow strength has a cumulative effect. That is to say, $\gamma_{ij,k}$ in the kth round is updated to include the values of the design rules that resulted from the investment in the $(k-1)$th round. Therefore, the updated value of the design rules at the kth iteration $\gamma_{ij,k}$ is the multiplication of the updated value of the design rules from the 0th to the kth round, which is expressed as $\prod_{\ell=1}^{k} \gamma_{ij,\ell}$. The specific form of the $A_k(\phi_k, \gamma_k)$ is given by

$$A_k(\phi_k, \gamma_k) = \begin{bmatrix} \phi_{1,k} & \prod_{\ell=1}^{k} \gamma_{12,\ell} & \cdots & \prod_{\ell=1}^{k} \gamma_{1n,\ell} \\ \prod_{\ell=1}^{k} \gamma_{21,\ell} & \phi_{2,k} & \cdots & \prod_{\ell=1}^{k} \gamma_{2n,\ell} \\ \vdots & \vdots & \ddots & \vdots \\ \prod_{\ell=1}^{k} \gamma_{n1,\ell} & \prod_{\ell=1}^{k} \gamma_{n2,\ell} & \cdots & \phi_{n,k} \end{bmatrix}.$$

Suppose that we can use the development resources to update the value of ϕ_k, γ_k. That is, we can use the resources to tune the work completion rate ϕ_k and the dependency strength γ_k. Moreover, we assume that there is an associated cost $f_i(\phi_{i,k})$ for tuning the value from $\bar{\phi}_{i,k}$ to $\phi_{i,k}$. Likewise, $g_{ij}(\gamma_{ij,k})$ is the cost for tuning the value $\bar{\gamma}_{ij,k}$ to $\gamma_{ij,k}$. Then, the total cost in the kth investment round equals

$$B_k(\phi_k, \gamma_k) = \sum_{i=1}^{n} f_i(\phi_{i,k}) + \sum_{i=1}^{n} \sum_{i \neq j} g_{ij}(\gamma_{ij,k}). \tag{5.2}$$

Form the perspective of project manager, optimally allocating the development resources to the maximum extent to obtain the maximized profit is imperative, especially when a huge project is carried out. However, making the optimal resources allocation strategy for thousands of decision variables just by experience and intuition seems not very effective. Thus, a mathematical programming formulation for finding the optimal investment strategy is essential.

5.2 Optimization problem

As mentioned in Section 5.1, at each iteration, PD managers can use a certain amount of development resources to improve the performance of the product system. Particularly, the resources can be allocated on a module for tuning its work completion rate or on the design rule for reducing the dependency strength between two certain modules. Assuming that given a set of budgets for each investment round during the whole development process, how should we make the allocation strategy to minimize the total remaining work of the PD process? It is clear that the minimized remaining work stands for obtaining the maximized work efficiency of the PD system. Based on this question, we formulate the budget-constrained problem as follows:

Problem 6 (Budget-constrained optimization)**.** *Assume that, given $P(0)$, there are T investment rounds with the corresponding budgets $\bar{B}_k > 0$ ($k = 1, \ldots, T$) for resource allocation during the PD process, as well as the cost functions $f_i(\phi_{i,k})$ and $g_{ij}(\gamma_{ij,k})$. Find a sequence of decision variables for allocating the investment resources in modules $\phi = \{\phi_{i,k}\}_{k=1}^{T}$ ($i = 1, \ldots, n$) and design rules $\gamma = \{\gamma_{ij,k}\}_{k=1}^{T}(i, j = 1, \ldots, n,\ \ i \neq j)$ to minimize the total remaining work at the end of Tth round while satisfying the budget constraints on the investment resources in each round.*

Mathematically, we formulate the budget-constrained problem as:

$$\underset{\phi,\gamma}{\text{minimize}} \sum_{i=1}^{n} P_i(T) \tag{5.3a}$$

$$\text{subject to} B_k(\phi_k, \gamma_k) \leq \bar{B}_k, \tag{5.3b}$$

$$0 < \underline{\phi}_{i,k} \leq \phi_{i,k} \leq \bar{\phi}_{i,k},$$

$$0 < \underline{\gamma}_{ij,k} \leq \gamma_{ij,k} \leq \bar{\gamma}_{ij,k},\ \ k = 1, \ldots, T. \tag{5.3c}$$

For the budget-constrained problem, our goal is to make the optimal resource allocation strategy to accelerate the PD process to the greatest extent. However, PD managers also face with another case when the prescribed target on the remaining work (i.e., as a proxy for judging the completion of the PD process) at

T is set, how to make the resources allocation strategy to minimize the cost? In this case, the performance-constrained problem can be formulated as follows:

Problem 7 (Performance-constrained optimization)**.** *Assume that, given $P(0)$, there are T investment rounds and the prescribed remaining work constraint $\bar{P}_T > 0$, as well as the cost functions $f_i(\phi_{i,k})$ and $g_{ij}(\gamma_{ij,k})$. Find a sequence of decision variables for allocating the development resources in modules $\phi = \{\phi_{i,k}\}_{k=1}^{T}$ $(i = 1, \dots, n)$ and design rules $\gamma = \{\gamma_{ij,k}\}_{k=1}^{T}$ $(i, j = 1, \dots, n,\ i \neq j)$ to minimize the total investment resource while satisfying the constraint on remaining work.*

As in (5.3), we can mathematically build the performance-constrained problem as the following:

$$\underset{\phi,\gamma}{\text{minimize}} \sum_{k=1}^{T} B_k(\phi_k, \gamma_k) \tag{5.4a}$$

$$\text{subject to} \sum_{i=1}^{n} P_i(T) \leq \bar{P}_T, \tag{5.4b}$$

$$0 < \underline{\phi}_{i,k} \leq \phi_{i,k} \leq \bar{\phi}_{i,k},$$

$$0 < \underline{\gamma}_{ij,k} \leq \gamma_{ij,k} \leq \bar{\gamma}_{ij,k}.\ k = 1, \dots, T.$$

The difficulty of solving the budget-constrained problem and the performance-constrained problem mainly stems from the nonlinearity of the functions (5.3a), (5.4a) and constraints (5.3b), (5.4b). That is, the budget-constrained problem and the performance-constrained problem become nonlinear optimization problems. Although there are some numerical solutions for this case based on heuristic methods [80, 81], such techniques can cause the solution to be trapped in a local optimal point. Moreover, the computation cost of the heuristic solver grows rapidly with the increase in problem size (i.e., the number of modules, design rules and the investment rounds). Thus, there exists a need for developing a computation framework that can deliver the optimal solution for a relatively large size of the resources allocation problem.

5.3 Solution using convex optimization

In this section, we present an optimization framework for efficiently solving the budget-constrained problem and the performance-constrained problem. Under

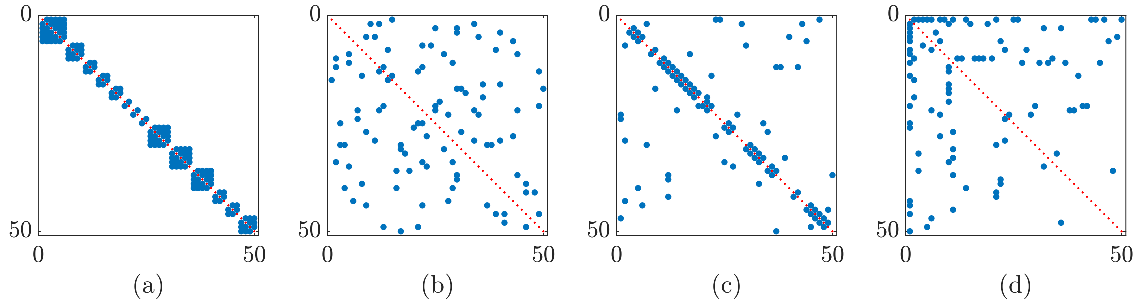


Figure 5.1: Four DSM architectures. All with 50 modules and 100 design rules: (a) Block diagonal network, (b) Erdős-Rényi network (Random), (c) Watz-Strogatz network (Small world), (d) Barabási-Albert network (Scale free). The diagonals in the DSM represent the location of modules, and the off-diagonals show the dependencies between modules.

the relatively mild assumption on the cost function, we can show that problems can be transformed into convex optimization problems. Let us begin with reviewing the definition of posynomials with the following:

Definition 7 ([53]). *Let $v = \{v_1, \ldots, v_n\}$ denote n real positive variables.*

1. *We say that a real function $g(v)$ is a monomial if there exist $c > 0$ and a set of real numbers $a_1, \ldots, a_n$ such that $g(v) = cv_1^{a_1} \cdots v_n^{a_n}$.*

2. *We say that a real function $f(v)$ is a posynomial if f is a sum of monomials of v.*

3. *We also say that a real function is a* generalized posynomial *if it can be formed from posynomials using the operations of addition, multiplication, positive (fractional) power, and maximum.*

To precisely model the cost features, the nonlinearity from the practical problem can not be ignored. From Definition 7, we can see that the posynomials are nonlinear functions which can be used for fitting real data from practical PD problems. For specific techniques to fit posynomials to real data, we refer readers to [53]. Also, the following lemma shows the convexity property of posynomials, which is essential in transforming the budget-constrained problem and the performance-constrained problem into convex optimization problems.

Lemma 6 ([53])**.** If f is a posynomial, then, the function $x \mapsto \log f(\exp[x])$ is convex.

As mentioned earlier, the nonlinearity of the real data can be fitted by posynomials. From Definition 7, we can see that the range of the posynomials is in the nonnegative number field. However, in practice, real data may run out of the nonnegative area. Thus, normalizing the range of the cost function to the nonnegative field is necessary (i.e., adjust the minimum value of the cost function larger than 0). For this reason, we make the following assumption for ensuring the nonnegativity of the cost function. We assume that the cost function has the following form:

$$f_i(\phi_{i,k}) = f_i^+(\phi_{i,k}) - f_i^+(\bar{\phi}_{i,k}),$$
$$g_{ij}(\gamma_{ij,k}) = g_{ij}^+(\gamma_{ij,k}) - g_{ij}^+(\bar{\gamma}_{ij,k}).$$

The essential part of the cost function is the first term $f_i^+(\phi_{i,k})$, while the second term $(-f_i^+(\bar{\phi}_{i,k}))$ is for normalizing the cost function as $f_i(\bar{\phi}_{i,k}) = 0$, similarly for $g_{ij}(\gamma_{ij,k})$, which means that the zero investment yields no cost.

The resulting optimization problems (5.3) and (5.4) are not trivial to solve directly because the nonlinearity in the work transformation process function (5.3a), (5.4b) and the resource cost functions (5.3b), (5.4a). Although there exist heuristic optimization methods that can solve this problem, the solution is local optimal due to the constraint of the algorithm. In a real product development project, especially for the complex case that contains hundreds of modules and design rules, finding the global optimal resource allocation strategy can bring great benefit for the company and share holders. Therefore, it is necessary to establish an efficient computation framework for obtaining the global optimal solution to the problems (5.3) and (5.4).

The following theorem allows us to overcome the difficulty and solve the budget-constrained problem and the performance-constrained problem via convex optimization and is the main theoretical result of this thesis.

Theorem 3. *Problems 6 and 7 reduce to convex optimization problems. In this appendix, we illustrate how we can reduce Problems 6 and 7 to convex optimization* *problems. For the total cost function in* (5.2)*, we define*

$$B_k^+(\phi_k,\gamma_k) = \sum_{i=1}^n f_i^+(\phi_{i,k}) + \sum_{i=1}^n \sum_{i\neq j} g_{ij}^+(\gamma_{ij,k}),$$
$$B_k^-(\phi_k,\gamma_k) = \sum_{i=1}^n f_i^+(\bar{\phi}_{i,k}) + \sum_{i=1}^n \sum_{i\neq j} g_{ij}^+(\bar{\gamma}_{ij,k}).$$

Under this notation, we can show that the solution of the budget-constrained problem is given by

$$\phi = \exp[x],\ \ \gamma = \exp[y], \tag{5.5}$$

where $\exp[\cdot]$ *is the entrywise exponential function of the variables, and* $x = \{x_k\}_{k=1}^T$ *and* $y = \{y_k\}_{k=1}^T$ *solve the following* convex *optimization problem:*

$$\begin{aligned} &\underset{x,y,\mathit{\Gamma}}{\text{minimize}}\,\mathit{\Gamma} \\ &\text{subject to} \log B_k^+(x_k,y_k) \leq \log(\bar{B}_k + B_k^-), && (5.6a)\\ &\qquad \log \sum_{i=1}^n P_i(T) \leq \mathit{\Gamma}, && (5.6b)\\ &\qquad \log \underline{\phi}_{i,k} \leq x_{i,k} \leq \log \bar{\phi}_{i,k}, && (5.6c)\\ &\qquad \log \underline{\gamma}_{ij,k} \leq y_{ij,k} \leq \log \bar{\gamma}_{ij,k}. && (5.6d)\end{aligned}$$

Let us give a brief proof of this statement. Under Lemma 6, it can easily be seen that (5.3a)*,* (5.3b)*, and* (5.3c) *in the budget-constrained problem are equivalent to* (5.6a)*,* (5.6b)*,* (5.6c)*, and* (5.6d)*. Therefore, the solution of the optimization problem* (5.6) *given by* (5.5) *is the solution of the budget-constrained problem. Under this equivalence, we show the convexity of the optimization problem* (5.6)*. It is sufficient to show that constraints* (5.6a) *and* (5.6b) *are convex if the performance functions* (5.3a)*,* (5.3b) *and the cost function* (5.2) *follow Definition 7.*

Similarly, the next theorem shows that the performance-constrained problem can also be solved with the same optimization framework. Specifically, Then, the solution of the performance-constrained problem is given by (5.5)*, where* $x =$

$\{x_k\}_{k=1}^T$ and $y = \{y_k\}_{k=1}^T$ solve the following convex optimization problem

$$
\begin{aligned}
&\underset{x,y,\Psi}{\text{minimize}} \Psi \\
&\text{subject to} \log \sum_{i=1}^{n} P_i(T) \leq \log \bar{P}_T, \\
&\qquad \log \sum_{k=1}^{T} B_k^+(x_k, y_k) \leq \log(\Psi + \sum_{k=1}^{T} B_k^-), \\
&\qquad \log \underline{\phi}_{i,k} \leq x_{i,k} \leq \log \bar{\phi}_{i,k}, \\
&\qquad \log \underline{\gamma}_{ij,k} \leq y_{ij,k} \leq \log \bar{\gamma}_{ij,k}.
\end{aligned}
$$

We omit the proof of this statement because it is similar to the one for the budget-constrained problem.

5.4 Experimental setup, analysis and discussion of results

In this section, we show the effectiveness of the proposed framework by solving relatively large-size PD problems with different product architectures. Furthermore, by investigating the solution, we reveal the trends, structure, and relationship of the decision variables. In Section 5.4.1, we introduce four typical DSM architectures embedded in our simulation experiments. In Section 5.4.2, we give the specific form of the cost function. Then, in Section 5.4.3, we present the optimal solution of the budget-constrained problem, perform its analysis, and discuss the results. Likewise, in Section 5.4.4, the optimal solution of the performance-constrained problem is demonstrated and discussed. In Section 5.4.5, we statistically investigate the impact of product architecture on the optimal resources allocation.

5.4.1 DSM architecture

As mentioned earlier, the design structure matrix (DSM) is a matrix representation of the development network which can have a particular architecture [82].

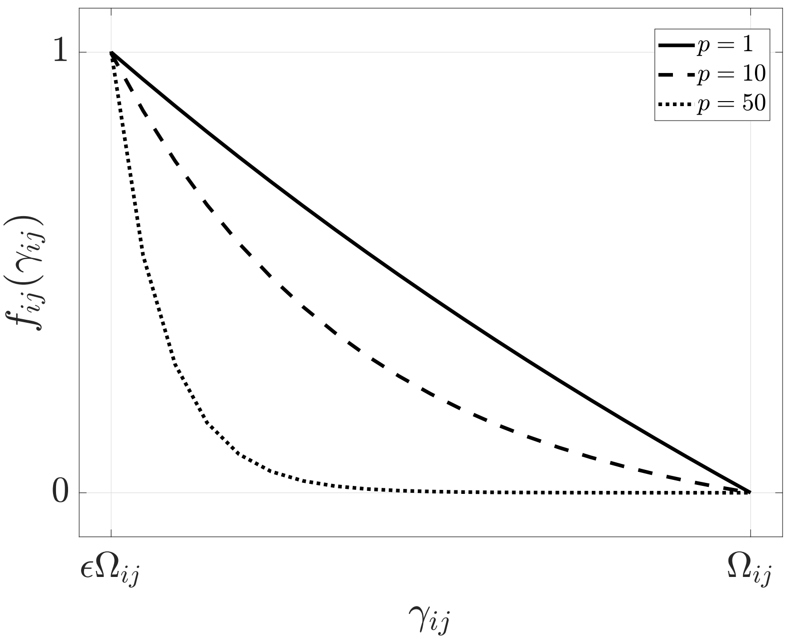


Figure 5.2: Three cost functions with $p = 1$, 10, and 50. $f_{ij}(\Omega_{ij}) = 0$ represents that no resource is allocated, where Ω_{ij} denotes the initial value of the certain entry in WTM. $f_{ij}(\epsilon\Omega_{ij}) = 1$ indicates the upper bound of the allocated resources where we can to obtain the fully improved value.

For this reason, the DSM architecture in our experiment is determined by the following network models: the Block diagonal [82], the Erdős-Rényi (random) [83], the Watz-Strogatz (small world) [84] and the Barabási-Albert (scale free) [85] graphs.

Fig. 5.1 shows the four DSM architectures used in this thesis. On one end, the Block diagonal network represents a typical modular architecture, where the dependencies between modules are divided into dependent groups (no interactions between the groups), and the modules in each group are fully dependent (see Fig. 5.1 (a) [86]). Alternatively, the Watz-Strogatz network and the Barabási-Albert network represent the other extreme, called integral architecture. The Watz-Strogatz network in Fig. 5.1 (c) shows the small world property, where most modules dependencies are local, but few dependencies exist between the distant modules [8, 84]. The Barabási-Albert network in Fig. 5.1 (d) illustrates the preferential attachment feature of the PD project. The project starts with few modules and as the design process unravels the new modules are linked to the old modules [8, 85]. We adopt the Erdős-Rényi network in Fig. 5.1 (b) for randomly setting the dependency structure in the DSM, which serves as a benchmark to other patterned DSM architectures. For other DSM architectures, we refer the readers to [8, 87, 88].

5.4.2 Cost function

As mentioned in Section 5.1, the resources allocated on the modules and design rules result in a reduction of the parameters in (5.1). Based on this, we claim that the cost function should be a decreasing function, and satisfy Definition 7. Thus, we use the following cost function:

$$f_{ij}(\gamma_{ij}) = c_{ij}\left(\frac{1}{(\gamma_{ij})^p} - \frac{1}{(\Omega_{ij})^p}\right),$$

where $\gamma_{ij,k}$ $(i, j = 1, \ldots, n,\ i \neq j)$ is the updated value of the parameter in the WTM, p is a positive integer for tuning the shape of the concerned cost function, and c_{ij}, Ω_{ij} are positive numbers for fitting the value of the cost function to satisfy Definition 7. Then, we make the following assumption to show the diminishing return property, which ensures the convexity of the cost function as well. Suppose that there is a fixed increment $\epsilon_{ij} > 0$ on γ_{ij}, and let $\Delta f_{ij}(\gamma_{ij}) = f_{ij}(\gamma_{ij} - \epsilon_{ij}) - f_{ij}(\gamma_{ij})$ represent the cost for tuning γ_{ij} to $\gamma_{ij} - \epsilon_{ij}$. The diminishing return property means that the parameter tuning cost $\Delta f_{ij}(\gamma_{ij})$ increases with γ_{ij}, and also implies the convexity of f_{ij}. In practice, the parameters of the cost function are carefully assigned by the managers and the work teams (e.g., see [?, 8]). Fig. 5.2 shows three realizations of the cost function under different values of p.

5.4.3 Analysis and discussion of the budget-constrained problem

In this subsection, we optimally solve the budget-constrained problem through our proposed framework. Then, we investigate the evolution of decision variables during the budget-constrained PD process. Finally, we introduce the centrality metrics for measuring the importance of modules and design rules, and study whether the allocated resources or the remaining work in each module or design rule correlates with its centrality.

In this simulation experiment, for testing the effectiveness of solving a relatively large scale PD problem [89], we produce the DSMs of size 50 and hold the total number of dependencies to 100 for each DSM architecture. We set the number of investment rounds $T = 5$, and the budget $\bar{B}_k = 300$ for each investment round. For initializing the parameters of the WTM, we unify $\bar{\phi}_{i,k} = 0.5$ $(k = 1, \ldots, 5, i = 1, \ldots, 50)$ and $\bar{\gamma}_{ij,k} = 0.05$ $(i, j = 1, \ldots, 50, i \neq j)$ for all the experiments. For all the cost functions, we unify the parameters with $c_{ij} = 1$, $p = 1$, $\Omega_{ij} = 1$, and $\epsilon = 0.1$, which indicates that the ϕ and γ can be updated between $[0 - 0.9]$ of the initial value. From the parameter initialization, we can see that the values of $\phi_{i,k}$ and $\gamma_{ij,k}$ can be tuned within the intervals $[0.05, 0.5]$ and $[0.005, 0.05]$, respectively. We conduct the experiments with the selected DSM architectures in Fig 5.1, and observe the following response variables: the remaining work in modules, the investment in the modules and the design rules, and the dependency strength between modules.

For problem solving, we adopt the commonly used off-the-shelve software for convex optimization problem: fmincon routine in MATLAB. From the experiment setup, we can see that the total number of the decision variables is $(50+100)\times 5 = 750$, which reaches the standard size of large PD process. Through running the experiment on the desktop with common configuration (i.e., Intel Core i7-7700 and 8GB memory), the average time for solving the optimization problem is 210 minutes, which illustrates that our framework is capable for solving the much larger-scale problems.

Fig. 5.3 shows the evolution of the resources allocation variables $(\phi_{i,k},\ \gamma_{ji,k})$, the remaining work $P_i(k)$ and the dependency strength of design rules in solving the budget-constrained problem. Particularly, Fig. 5.3 (a) shows the decreasing trends of the remaining work in each module, which indicates that the PD process is in progress. However, this experiment contains more than 1 investment round, so just by observing the remaining work from Fig. 5.3 (a) is not easily for us to distinguish the effect of the investment in each investment round. Thus, we define the completion rate $\xi(k)$ for each investment round by

$$\xi(k) = \frac{\sum_{i=1}^{n} P_i(k) - \sum_{i=1}^{n} P_i(k+1)}{\sum_{i=1}^{n} P_i(k)}$$

as an index for measuring the corresponding performance in each round. From Fig. 5.4, we can observe the evolution of the performance during the process, where the performance of product system is monotonically improved with successive investment. Compared with the experiment with no investment (dashed line), we can derive that the PD process is accelerated with the allocated resources. As seen in Fig. 5.4, with the investment in progress, the performance reaches its saturation. We also notice that the limits of performance are different from the DSM architectures, which implies that the DSM architecture must be taken into consideration for further studying the performance. We note that the high performance of the block diagonal network is consistent with the finding in [87] that real design networks have a nested hierarchical network structure.

Figs. 5.3 (b) and 5.3 (c) show the evolution of investment in modules and design rules, where we can see that the investment in modules increases, while the investment in design rules decreases. This phenomenon is in line with the result in [8] that there is a shift in resource allocation from design rules to modules as the development process progresses. Moreover, it is worth noting that the modular architecture consumed more resources on the modules compared with design rules, while integral architecture consumed more resources on the design rules. Thus, the evolution of the decision variables also confirms that the product architecture evolves from integral to modular as the product matures. In Fig. 5.3 (d), we can see that the dependency values of the design rules tends to 0, which indicates that the dependency strength between modules is reduced or nearly eliminated by successive investment in design rules.

Next, we carry out a further investigation on whether there is a relationship between the optimal cumulative investment in a specific module. We define the cumulative allocated resources on the module during the process by

$$\mu_i = \sum_{k=1}^{T} f_i(\phi_{i,k})$$

Table 5.1: Pearson correlation analysis for the budget-constrained problem (Figs. 5.9-5.12)

DSM		Eigenvector	PageRank	Closeness
Block diagonal	Remaining work	0.993	-	0.999
	Investment in module	0.992	-	0.989
	Investment in DRs	0.993	-	0.979
Random	Remaining work	0.561	0.947	0.867
	Investment in module	0.518	0.944	0.886
	Investment in DRs	0.443	0.573	0.595
Small world	Remaining work	0.501	0.968	0.531
	Investment in module	0.483	0.972	0.526
	Investment in DRs	0.377	0.614	0.364
Scale free	Remaining work	0.493	0.994	0.948
	Investment in module	0.427	0.987	0.968
	Investment in DRs	0.666	0.751	0.734

and its related design rules

$$\rho_i = \sum_{k=1}^{T} \left(\sum_{j=1}^{n} g_{ij}(\gamma_{ij,k}) + \sum_{j=1}^{n} g_{ji}(\gamma_{ji,k}) \right).$$

We also define the cumulative allocated resources on design rules by

$$\rho_{ij} = \sum_{k=1}^{T} \left(g_{ij}(\gamma_{ij,k}) + g_{ji}(\gamma_{ji,k}) \right).$$

Fig. 5.5 shows a positive correlation between the investment in the modules and its related design rules. This observation can be used as a managerial guideline for resource allocation: if a module is assigned with a certain amount of resources, then corresponding amount of resource must be allocated to its related design rules.

After discussing the results in Fig. 5.3, we introduce the centrality metrics (i.e., importance measures) for the modules and design rules to investigate whether there is a relationship between the investment in module/design rule and its centrality. For describing the importance of the modules and design rules, we adopt three centrality metrics [90]: the Eigenvector, the PageRank, and the Closeness centrality. For simplicity, we normalize each centrality metric to 1. Throughout the thesis, we let the centrality metric of the ith module be denoted by r_i, and the centrality metric of a design rule between the ith and the jth module be denoted by r_{ij}, respectively. Then, Figs. 5.9-5.12 show the the dependence on the centrality measures of remaining work $P_i(T)$, cumulative resource allocation μ_i on the modules, and cumulative resource allocation ρ_{ij} on the design rules. From Figs. 5.9-5.12, we observe that the extent of correlation varies with different centrality metrics. To decide which centrality metric performs the best in describing the correlation, we adopt Pearson correlation [91] to help us select the proper centrality metric. A perfect Pearson correlation 1 occurs when each of the variables is a perfect monotone function of the other. On the contrary, 0 means that there is completely no correlation between the two set of numbers. Table 5.1 shows the result of Pearson correlation for Figs. 5.9-5.12. From Table 5.1, we can see that PageRank performs the best except for the block diagonal case because the definition of PageRank is infeasible for measuring the block diagonal network. Fortunately, both the Eigenvector centrality and the Closeness centrality perform a strong linear correlation, which can be used for the block diagonal case.

From Figs. 5.9-5.12, we find that there exists a relatively strong correlation between the investment/remaining work and its PageRank centrality under the four DSM architectures. So, as a quick heuristic, we can assign the budget as a function of centrality instead of solving a complex optimization problem, as previously found in [92]. Moreover, PD managers can use PageRank centrality as a proxy to allocate development resources for the modules.

For the correlation between the investment in the design rule and its centrality in Figs. 5.9 (c)-5.12 (c), we can see that the extent of correlation in Figs. 5.9 (c)-5.11 (c) is not strong enough for drawing a conclusion. However, for the block diagonal case in Fig. 5.12 (c), we observe a different phenomenon that hundred of variables are overlapped to a few points, and are strongly correlated with the Eigenvector and the Closeness centrality. Specifically, we notice that design rules belonging to different sub-blocks but with the same size receive the same amount of investment. In other words, the investment in design rules is independent of the block to which it belongs. This independence is caused by the special structure of the block diagonal network, in which all the sub-blocks are independent of each other.

5.4.4 Analysis and discussion of the performance-constrained problem

In this subsection, we solve the performance-constrained problem via convex optimization. Although we have revealed the trends of the decision variables and the internal relations for the budget-constrained problem, we cannot conclude that the same situation also exists in the performance-constrained problem.

As in Section 5.4.3, we perform the simulation experiment on a controlled set of product architectures. For initializing the performance-constrained problem, we adopt the same parameters setting as the budget-constrained problem. Based on the formulation of the performance-constrained problem, we set the constraint for the total remaining work of the final investment round to $\bar{P}_T = 0.01$, which can be regarded as a threshold for judging the accomplishment of the PD process. For example, suppose that the total remaining work at the beginning is normalized to 1, if we set $\bar{P}_T = 0.01$, it means that when the total remaining work is 1% its initial value, we can say that the project is finished. We conduct the experiments with the selected DSM architectures in Fig 5.1, and observe the following response variables: the remaining work in modules, the investment in the modules and the design rules, the dependency strength between modules, and the total investment in each round.

Fig. 5.6 shows the evolution of decision variables and the remaining work of the performance-constrained problem. From Figs. 5.6 (a)-(c), we can see that the solution of the performance-constrained problem exhibits similar trends to the budget-constrained problem. Particularly, the solution shows that the product architecture evolves from an integral to a modular as successive investment are made on the modules and the design rules. In Fig. 5.6 (d), it is worth noting that there is a decreasing tendency on the total investment during the PD process, which contradicts our intuition that the resources should be equally allocated for

Remaining work	Random	Small world	Scale free	Block diagonal
Random	—	0.265	0.085	4.95×10^{-23}
Small world	—	—	0.013	1.48×10^{-21}
Scale free	—	—	—	7.04×10^{-25}
Block diagonal	—	—	—	—

Table 5.2: p-values from ANOVA test of the total remaining work in Fig. 5.7 (a).

Invest in modules	Random	Small world	Scale free	Block diagonal
Random	—	0.579	3.03×10^{-5}	1.06×10^{-23}
Small world	—	—	1.63×10^{-7}	3.04×10^{-28}
Scale free	—	—	—	7.63×10^{-19}
Block diagonal	—	—	—	—

Table 5.3: p-values from ANOVA test of the total investment in modules in Fig. 5.7 (c).

each investment round. In the performance-constrained problem, we also find the positive correlation between the investment in the module and its related design rules as the budget-constrained problem. Further analysis similar to the ones for the performance-constrained problem allows us to draw the same set of conclusions as the one we obtained for the budget-constrained problem. The details are omitted.

5.4.5 Analysis of different DSM architectures

In this subsection, we carry out an analysis of variance on the product architecture to investigate whether the product architecture affects the resource allocation and the performance of the designed PD system. It is important to remark that the

Investment in DRs	Random	Small world	Scale free	Block diagonal
Random	—	0.389	3.63×10^{-6}	1.91×10^{-27}
Small world	—	—	2.54×10^{-10}	1.38×10^{-30}
Scale free	—	—	—	1.56×10^{-18}
Block diagonal	—	—	—	—

Table 5.4: p-values from ANOVA test of the total investment in design rules in Fig. 5.7 (e)

Total investment	Random	Small world	Scale free	Block diagonal
Random	—	0.325	2.95×10^{-8}	4.95×10^{-5}
Small world	—	—	1.45×10^{-13}	3.18×10^{-4}
Scale free	—	—	—	1.34×10^{-18}
Block diagonal	—	—	—	—

Table 5.5: p-values from ANOVA test of the total investment in Fig. 5.8 (a).

Invest in modules	Random	Small world	Scale free	Block diagonal
Random	—	0.532	0.023	2.39×10^{-6}
Small world	—	—	0.009	5.01×10^{-7}
Scale free	—	—	—	1.65×10^{-5}
Block diagonal	—	—	—	—

Table 5.6: p-values from ANOVA test of the total investment in modules in Fig. 5.8 (c).

Investment in DRs	Random	Small world	Scale free	Block diagonal
Random	—	0.028	2.59×10^{-5}	1.61×10^{-5}
Small world	—	—	3.61×10^{-4}	1.66×10^{-4}
Scale free	—	—	—	2.31×10^{-14}
Block diagonal	—	—	—	—

Table 5.7: p-values from ANOVA test of the total investment in design rules in Fig. 5.8 (e).

set of synthetic networks we use for our analysis is not intended to replicate all the aspects of real design networks, specifically the significant difference between the distribution of in- and out-degrees in the product architecture [87, 88].

In this experiment, we used the four DSM architectures introduced in Fig. 5.1 and selected the three response variables: total remaining work, total investment in modules, and total investment in design rules. To detect any statistical difference, we randomly generate 50 sample networks for each type of product architecture. In all the problems, we unify the parameters of the WTM and the cost functions as in the previous sections. We solve these problems with the proposed framework in this thesis. The results of the budget-constrained problem and the performance-constrained problem sorted by product architecture are shown in Figs. 5.7 and 5.8, respectively. We use Boxplots to illustrate the maximum, th e minimum, the variance, and the mean value of the investment and the remaining work. We observe that the architecture affects the resource allocation, which in turn affects the remaining work, investment in modules and design rules of the PD process. To statistically investigate the dependence, we also carry out one-way ANOVA tests [93] between pairs of data. We adopt the p-value from the ANOVA test as an index to illustrate the difference between each two networks. From Figs. 5.7 and 5.8, we observe that the difference among the networks except the random network is statistically significant, while the difference between the random and small world network is often not significant. This tendency could be partly attributed to the similarity in the construction rules for these two networks.

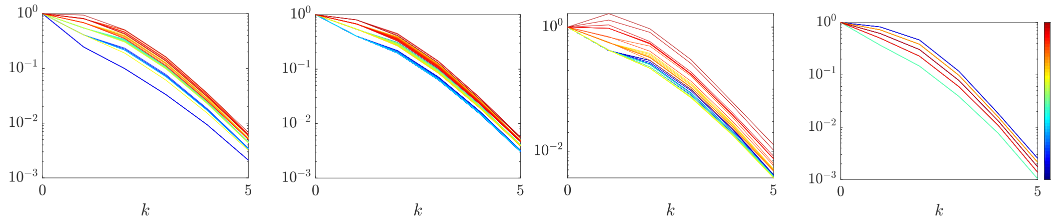
For the remaining work of the budget-constrained problem (Fig. 5.7 (a)), we can see that the Block diagonal architecture has the minimum remaining work (the best performance) compared with the other three architectures, which indicates that the modular architecture performs better than integral architecture (i.e., small world and scale free). Besides, we also notice that although the Block diagonal has the best performance, the variability is larger than the Small world case. This result implies that the stability of product architecture can not be neglected in designing the DSM structure. From Fig. 5.7 (b) and Fig. 5.7 (c), we confirm that there exists a variance on the investment in modules and design rules with different product architectures. Also, from Fig. 5.8 for the performance-constrained problem, we can see that for meeting the same target of the remaining work, the Block diagonal architecture costs the minimum resources among the four architectures, which also indicates that the modular architecture performs better than integral architecture.

5.4.6 Limitation and robustness

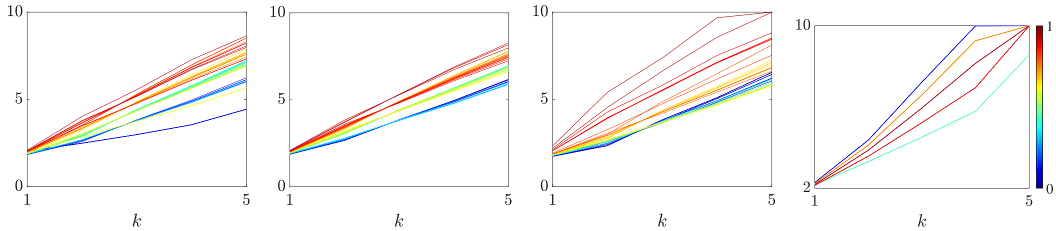
We discuss the limitation and robustness of our analysis in this section. First, as we remarked in the introduction, the set of synthetic networks we use for our analysis does not necessarily cover all the types of realistic design networks. As revealed in the seminal works in [87,88], empirical product development architectures often exhibit distinctive asymmetry between the distributions of incoming and outgoing links. For this reason, we investigate real product development processes having asymmetric architectures in the next section.

Second, our findings above are dependent on the the current choice of relevant parameters. In order to examine the robustness of our analysis to the choice of parameters, we let the shape parameter p of the cost function (4.3) vary between $[1, 10]$ and perform simulations. We then found that the solution (i.e., the resulting pattern of resource allocation) exhibits the same trend, which can suggest the robustness of our analysis. We also vary the other shape parameter c_{ij} to find that, the larger c_{ij}, the lower the effect of the investment on the development process due to the higher price of the resources.

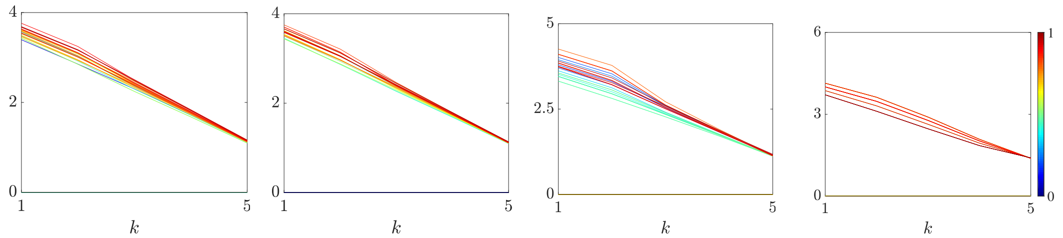
Random	Small world	Scale free	Block diagonal



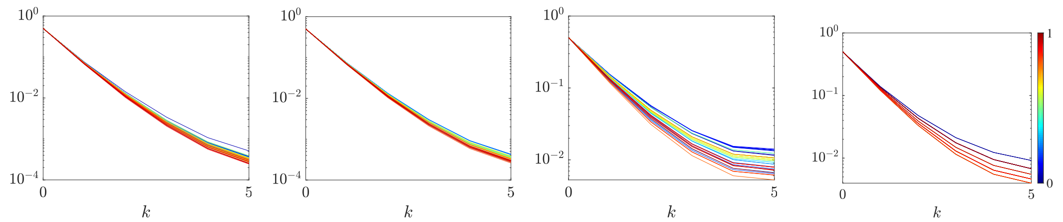
(a) Remaining work in modules versus investment round.



(b) Investment in modules versus investment round.



(c) Investment in design rules versus investment round.



(d) The dependency strength of design rules versus investment round.

Figure 5.3: The optimal solution of the budget-constrained problem. Color lines distinguish the importance of modules/design rules via the PageRank. Colorbar indicate the importance (PageRank) of the Y-axis values.

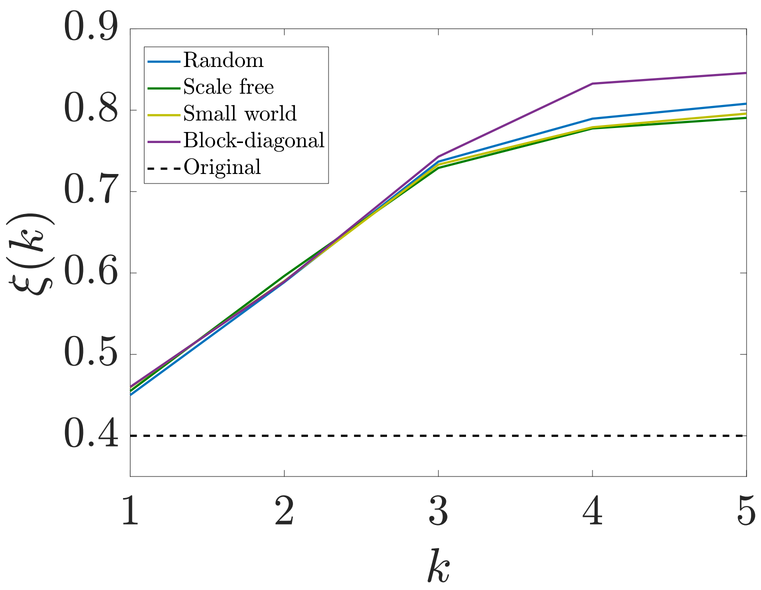
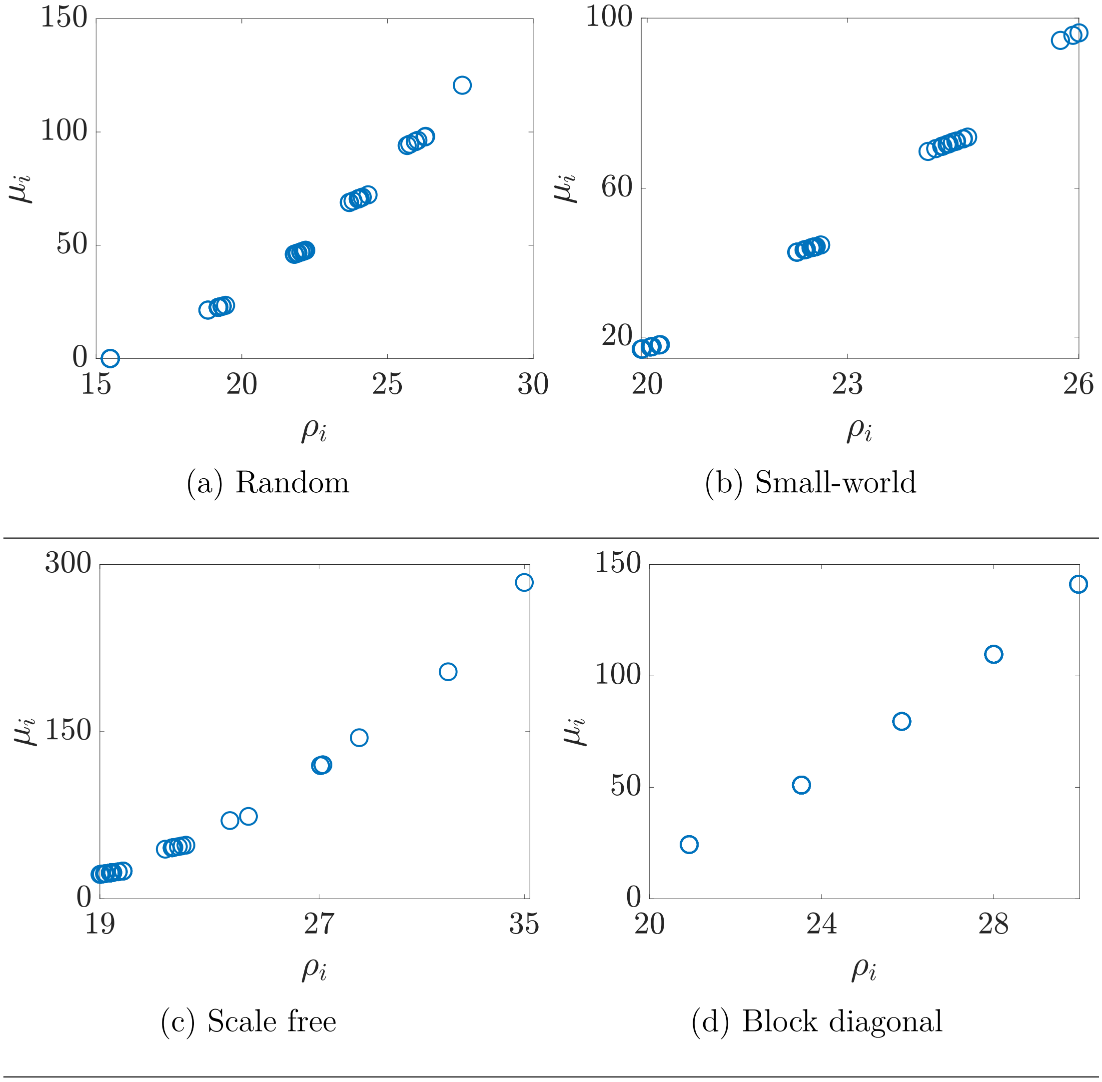


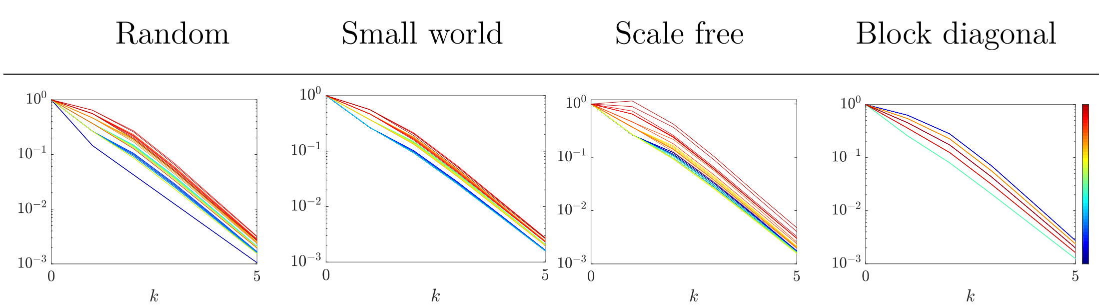
Figure 5.4: Performance evolution of the budget-constrained problem.



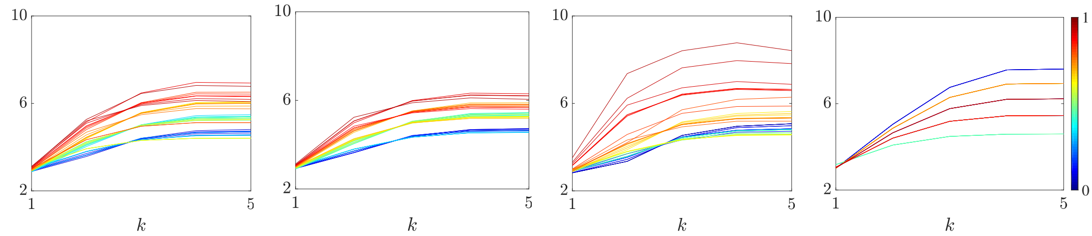
(a) Random (b) Small-world

(c) Scale free (d) Block diagonal

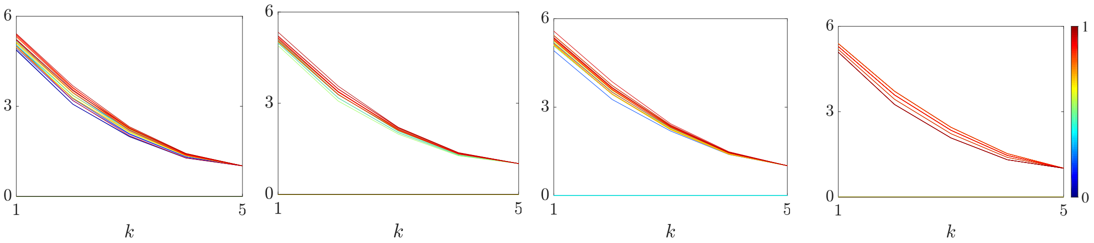
Figure 5.5: The correlation between the investment in module and the total investment in its dependent design rules in the optimal solution of the budget-constrained problem.



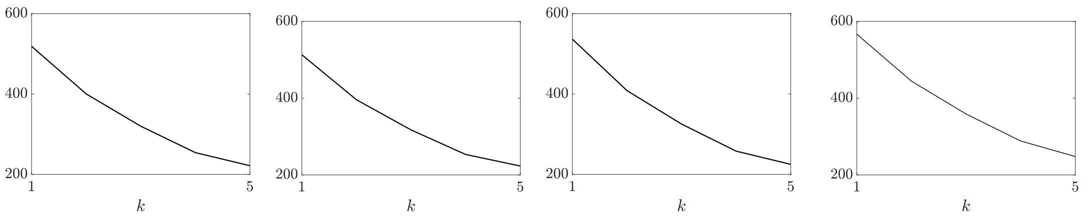
(a) Remaining work in modules versus investment round.



(b) Investment in modules versus investment round.

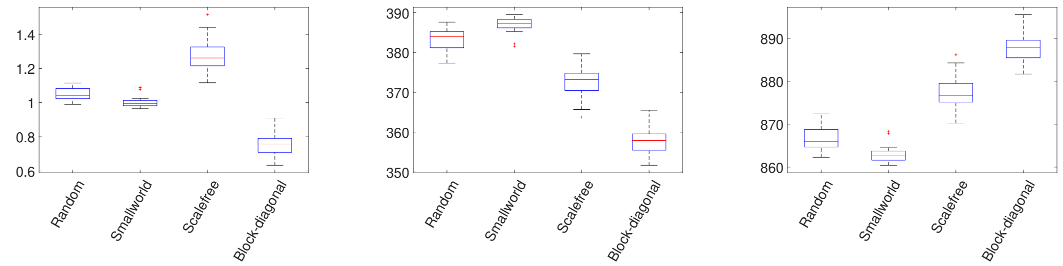


(c) Investment in design rules versus investment round.



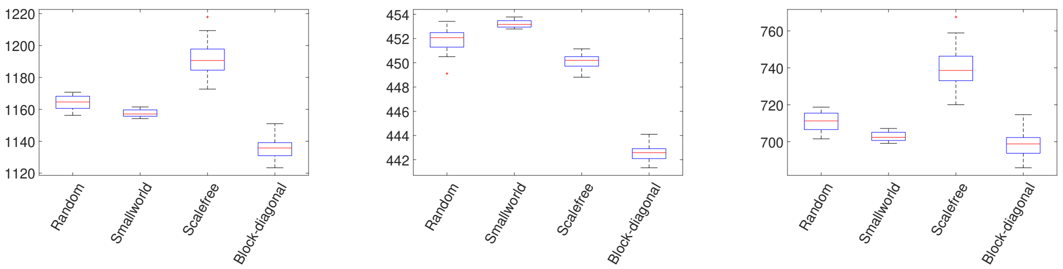
(d) The total resource cost versus investment round.

Figure 5.6: The optimal solution of the performance-constrained problem. Color lines distinguish the importance of module/design rules via the PageRank.



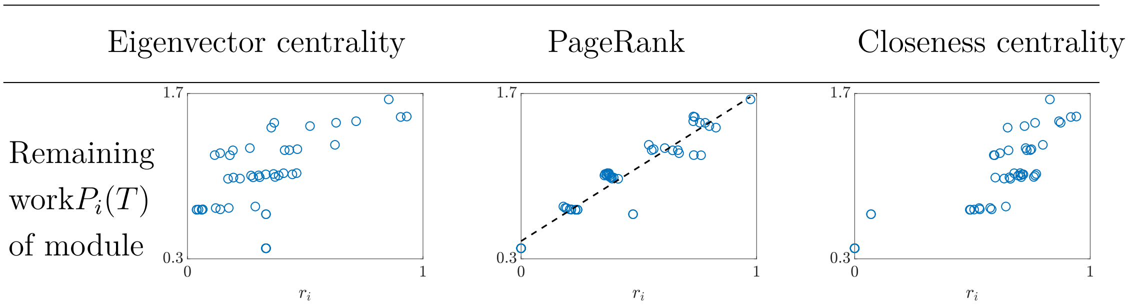
(a) Remaining work (b) Total invest in modules (c) Total invest in design rules

Figure 5.7: The total investment and performance of the budget-constrained problem versus different DSM architectures.

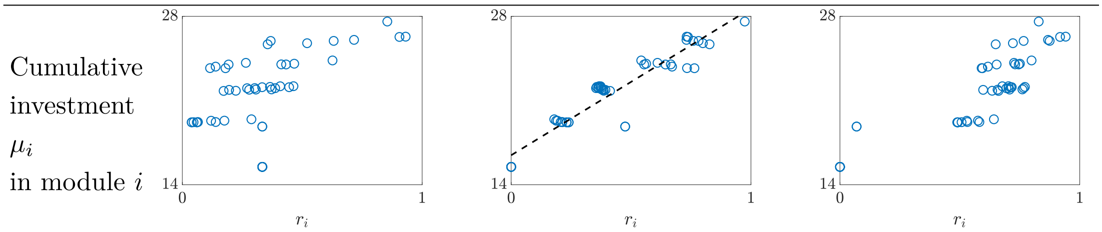


(a) Remaining work (b) Total invest in modules (c) Total invest in design rules .

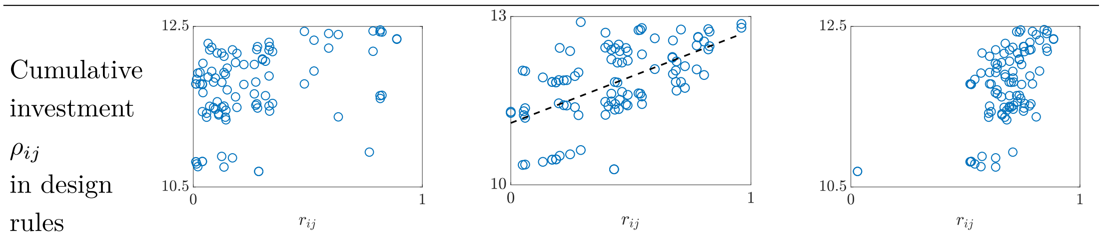
Figure 5.8: The total investment and performance of performance-constrained problem versus different DSM architectures.



(a) Remaining work of module versus its centrality measures in the network.

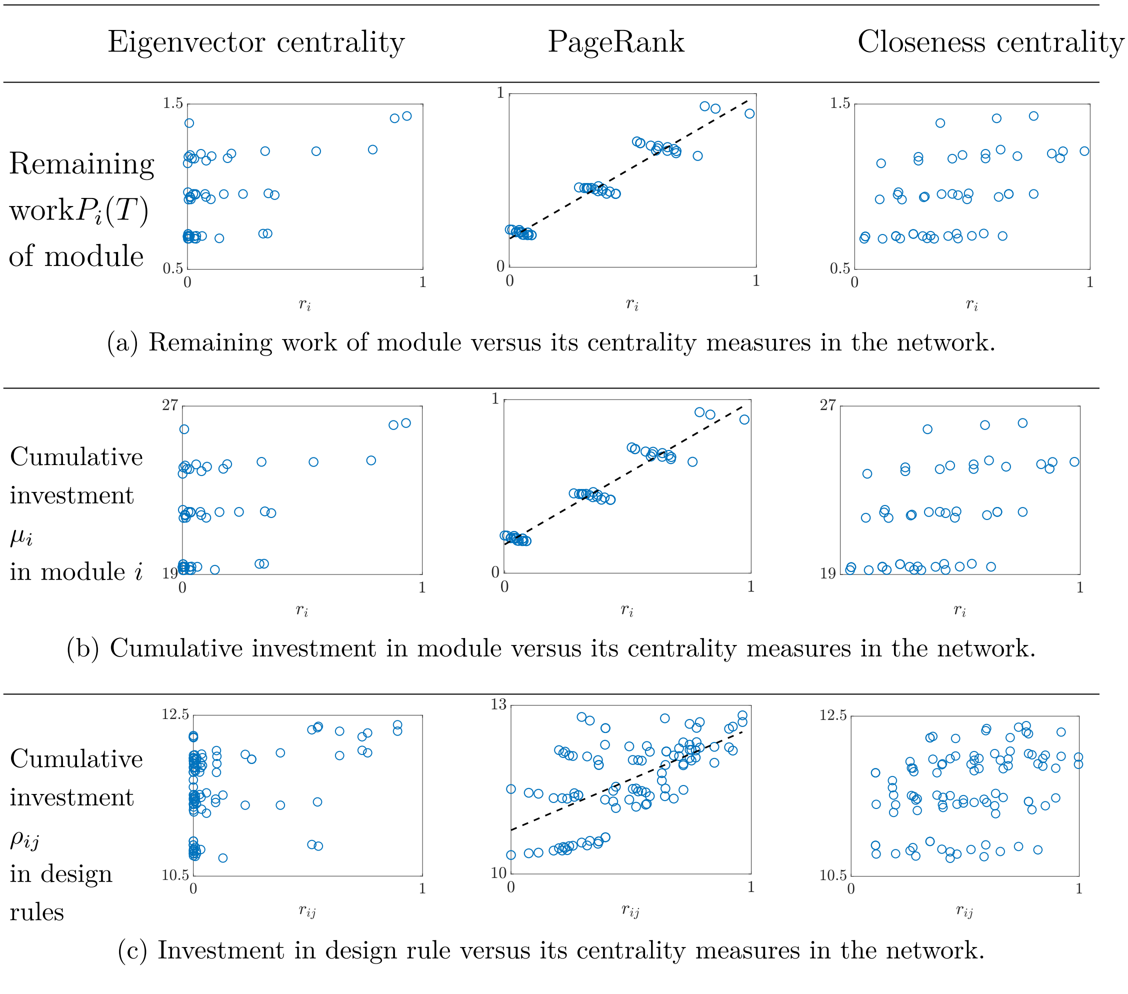


(b) Cumulative investment in module versus its centrality measures in the network.



(c) Investment in design rule versus its centrality measures in the network.

Figure 5.9: The remaining work, investment in modules and design rules of Problem 6 versus their centrality measures in the Erdős-Rényi (random) network. Dash line: Linear regression line.



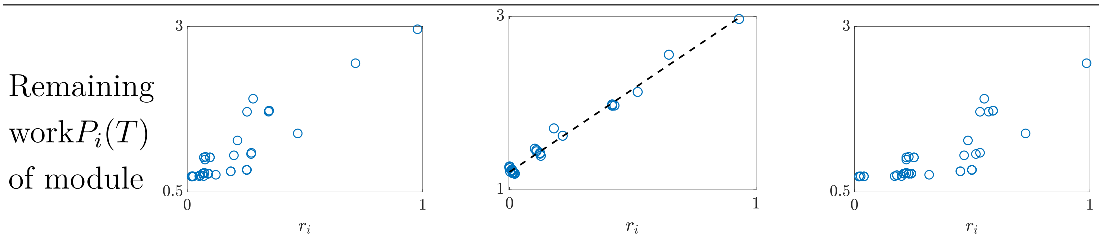
(a) Remaining work of module versus its centrality measures in the network.

(b) Cumulative investment in module versus its centrality measures in the network.

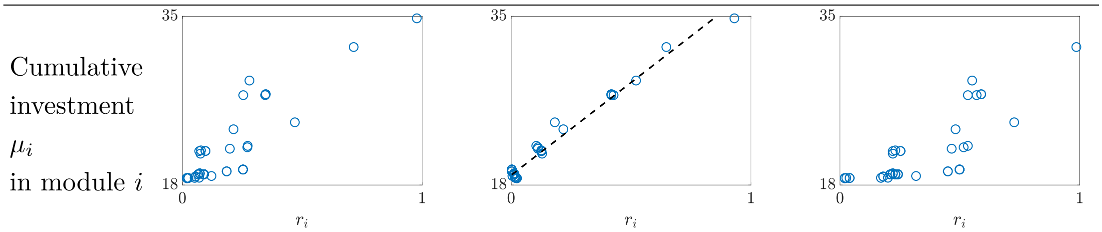
(c) Investment in design rule versus its centrality measures in the network.

Figure 5.10: The remaining work, investment in modules and design rules of Problem 6 versus their centrality measures in the Watz-Strogatz (small world) network. Dash line: Linear regression line.

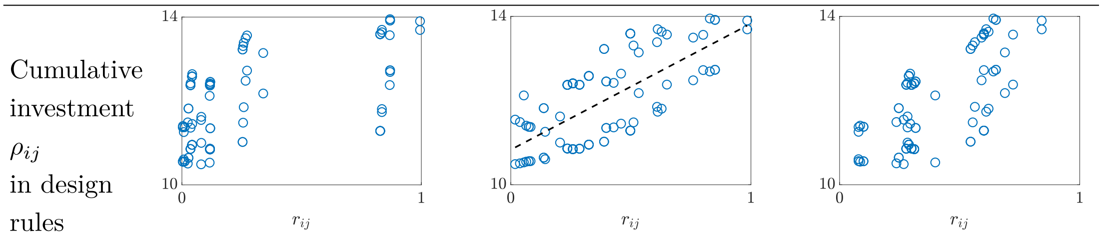
	Eigenvector centrality	PageRank	Closeness centrality



(a) Remaining work of module versus its centrality measures in the network.

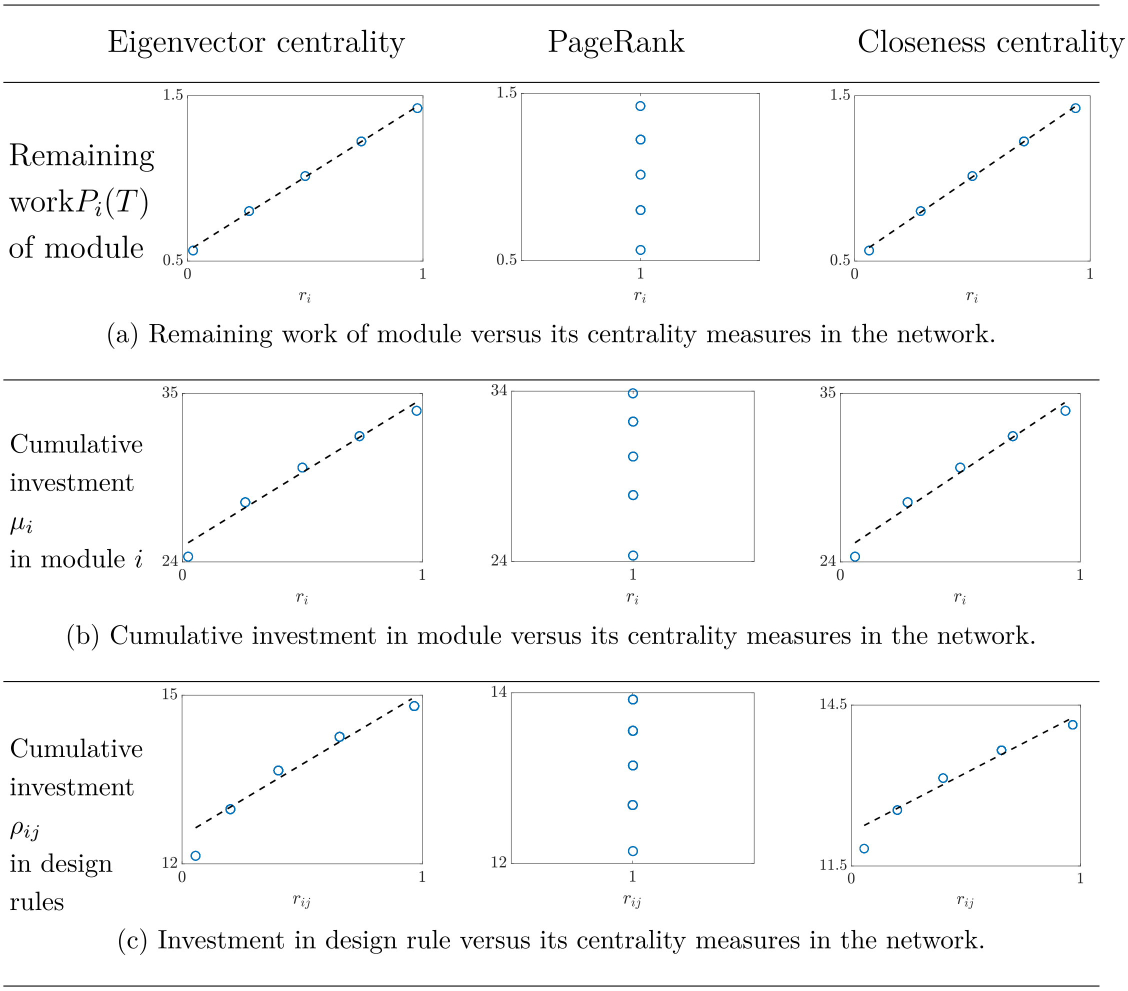


(b) Cumulative investment in module versus its centrality measures in the network.



(c) Investment in design rule versus its centrality measures in the network.

Figure 5.11: The remaining work, investment in modules and design rules of Problem 6 versus theirr centrality measures in the Barabási-Albert (scale-free) network. Dash line: Linear regression line.



(a) Remaining work of module versus its centrality measures in the network.

(b) Cumulative investment in module versus its centrality measures in the network.

(c) Investment in design rule versus its centrality measures in the network.

Figure 5.12: The remaining work, investment in modules and design rules of Problem 6 versus their centrality measures in the Block-diagonal network. Dash line: Linear regression line.

6 Discussion

In this thesis, we have studied the two issues of positive time-varying linear systems: the feature of time-varying governed by the semi-Markov switching process and the discrete-time switched case. Moreover, as a novel computation framework, the optimal resource allocation of dynamic product development problem is solved by the proposed results.

For the stabilization of positive semi-Markov jump linear systems, in this thesis, we only address the fundamental stabilization problem. For other issues in control problems, there also exist many stability measurements can be adopted for further study, like, the L_1 control, sign-stability, time-delayed systems, and robust control. Besides that, it is also worth to explore the stochastic process under the more general situation. For example, if the semi-Markov jump process shows a partial information of the transition matrix (i.e., the switching information between some subsystems are unknown), the stabilization problem would be a challenge one and worth studied. For the application research, examples like, the fault tolerant control, the epidemic spreading control, and the wireless power transmission control can also be generalized to the semi-Markov jump situation based on the results in this thesis.

For the finite-time control for positive time-varying linear systems, we assume that the entries in the parameterized state matrices follow the posynomials. However, if the entries include the negative expressions of posynomials, the optimization problem becomes non-convex, which can be regarded as a more general case compared to the problem studied in this thesis. For this situation, the non-convex optimization method would be utilized to study the computation framework. For solving the real problems, we derived the result under the assumption that the switching signal is known before the controller design. However, in solving practical problems, the accuracy of the modeled switching signal impacts a lot on the performance of the designed controller. Therefore, it is necessary for us to adopt the statistics approach or the machine learning methods for fitting the real switching signal. For example, the optimal energy operation in the data center relies on the requests from the data users. However, it is known that the data request is not deterministic, but it follows the tendency on statistics. In this situation, it is worth for us to explore the learning method based approached for solving the application problems.

In Chapter 5, we tested the feasibility of the theoretical result proposed in Chapter 4 for solving the optimal resources allocation of product development process. Moreover, we combined the network analysis method and our results to build a synthesis framework for solving the optimization of complex networks. Based on this results, it is also possible to solve other complex networks based model like, wireless networks, epidemic spreading networks, and multi-agent systems. One limitation of the framework proposed in this paper is that it does not consider the time-delay effect; so, dynamic investment problem with time-delay need to be considered in future work; especially, if the parameters of the PD system are updated after a certain period. Then, the investment decision making problem becomes applicable to a more general situation.

7 Conclusions

In this thesis, we studied two issues of positive time-varying linear systems via geometric programming and proposed the novel frameworks for addressing the related control problems. we first studied the stabilization problem of positive semi-Markov jump linear systems. By utilizing the spectral property of nonnegative matrices, we proposed a novel computation framework that the optimal performance of the system can be formulated to a convex optimization problem which is solved by optimizing the spectral radius of the matrix under the budget-constrained of the system parameter. Then, we checked the validity through a simulation example of the biological propagation which illustrates the relations among these parameters.

Then, for the discrete-time positive linear system with time-varying state, we studied a class of finite-time control problems for the discrete-time time-varying positive linear systems constrained by the parameter tuning cost. By utilizing the convexity property of posynomial functions, we have shown that the finite-time control problem can be transformed into a convex optimization problem. Finally, we have illustrated the effectiveness of our framework by a numerical simulation on product development processes. In the future work, one of the possible extension of our work is to consider the time-delay effect; especially, if the parameters of the PD system are updated after a certain period. Then, the investment decision making problem becomes a more general situation.

In the application research (Chapter 5), our results provide PD managers with an efficient tool to allocate development resources optimally for the budgetconstrained problem and performance-constrained problem, where the resources can be allocated on both modules and design rules. Although we carried out the experiments with two types of problems, and with different product architectures for each problem, the evolution of the investment and remaining work exhibit similar trends, which shows that the evolution property of the PD process is independent of the problem formulation and product architecture. Moreover, the investment and performance in modules also illustrate that certain correlations exist despite the problem formulation and product architecture, which also confirms that these trends and correlations are the intrinsic properties of the PD process. In the analysis of different PD architectures, we show that the architecture of the product affects resource allocation which in turn affects the performance of the PD process. Design and managerial guidelines can result from the direct analysis of the PD architecture. Specifically, for development engineers, our result can be used for selecting the product architecture which leads to maximum performance. On the other hand, when the PD architecture is fixed, our proposed framework helps PD managers in deciding on the optimal budget proportions to be allocated to modules and to design rules.

Acknowledgements

Pursuing the doctoral degree in Japan was always my dream when I entered university in China. Although I went to the industrial field after finishing the master course and work for three years, I still not give up my dream on pursuing my doctoral degree. So, firstly, I would give my heart gratitude to Professor Kenji Sugimoto for giving me the chance and kindly help during the three years study. Without your support, I could not achieve this progress and finish my doctoral course. Moreover, Professor Sugimoto's rigorous scholarship leaves me a deep impression, which set up an example on academic researcher. I sincerely wish Professor Sugimoto good health and happiness.

Especially, I want to express my special appreciation and thanks to Professor Masaki Ogura. Before I join the research group of Professor Ogura, I have much misunderstanding on doing research. Through the three years study, Professor Ogura teaches me the right way of the selection of a subject, building the idea, writing the academic report, making presentation and help me cultivate a good habit on academic research. I really enjoy the research work with Professor Ogura. Moreover, Professor Ogura also magnanimously tolerates my carelessness and mistakes, and always give me confidence.

I would also like to thank other committee members, Professor Yukiharu Uraoka, Professor Shoji Kasahara and Professor Takamitsu Matsubara for serving as my committee members even at hardship. Thank you for giving me so much comments during my study period. I also want to thank you for letting my defense be an enjoyable moment, and for your brilliant comments and suggestions, thanks to you.

Furthermore, I would like to thank my parents for their wise counsel and sympathetic ear. You are always there for me. Finally, there are my friends, who were of great support in deliberating over our problems and findings, as well as providing pleasant distractions to rest my mind outside of my research.

References

[1] D. De Farias, J. Geromel, J. Do Val, and O. Costa, "Output feedback control of Markov jump linear systems in continuous-time," *IEEE Transactions on Automatic Control*, vol. 45, no. 5, pp. 944–949, 2000.

[2] H. Lin and P. J. Antsaklis, "Stability and stabilizability of switched linear systems: A survey of recent results," *IEEE Transactions on Automatic Control*, vol. 54, no. 2, pp. 308–322, 2009.

[3] O. L. Costa, M. D. Fragoso, and M. G. Todorov, *Continuous-time Markov Jump Linear Systems.* Berlin, Heidelberg, Germany: Springer-Verlag, 2013.

[4] P. Shi and F. Li, "A survey on Markovian jump systems: Modeling and design," *International Journal of Control, Automation and Systems*, vol. 13, no. 1, pp. 1–16, 2015.

[5] Peng Shi, E.-K. Boukas, and R. Agarwal, "Control of Markovian jump discrete-time systems with norm bounded uncertainty and unknown delay," *IEEE Transactions on Automatic Control*, vol. 44, no. 11, pp. 2139–2144, 1999.

[6] M. D. Fragoso and O. L. V. Costa, "A unified approach for stochastic and mean square stability of continuous-time linear systems with Markovian jumping parameters and additive disturbances," *SIAM Journal on Control and Optimization*, vol. 44, no. 4, pp. 1165–1191, 2005.

[7] M. Ogura, M. Wakaiki, H. Rubin, and V. M. Preciado, "Delayed bet-hedging resilience strategies under environmental fluctuations," *Physical Review E*, vol. 95, no. 5, p. 052404, 2017.

[8] A. Yassine and J. Naoum-Sawaya, "Architecture, performance, and investment in product development networks," *Journal of Mechanical Design*, vol. 139, no. 1, 011101, 2016.

[9] X. Feng, K. Loparo, Y. Ji, and H. Chizeck, "Stochastic stability properties of jump linear systems," *IEEE Transactions on Automatic Control*, vol. 37, no. 1, pp. 38–53, 1992.

[10] Yuguang Fang and K. Loparo, "Stochastic stability of jump linear systems," *IEEE Transactions on Automatic Control*, vol. 47, no. 7, pp. 1204–1208, 2002.

[11] P. Bolzern, P. Colaneri, and G. De Nicolao, "Markov jump linear systems with switching transition rates: Mean square stability with dwell-time," *Automatica*, vol. 46, no. 6, pp. 1081–1088, 2010.

[12] F. Amato, R. Ambrosino, C. Cosentino, and G. D. Tommasi, "Finite-time stabilization of impulsive dynamical linear systems," *Nonlinear Analysis: Hybrid Systems*, vol. 5, no. 1, pp. 89–101, 2011.

[13] De. Leenheer. P and Angeli. D, "Monotone chemical reaction networks," *Journal of Mathematical Chemistry.*

[14] K. Gopalakrishnan, H. Balakrishnan, and R. Jordan, "Stability of networked systems with switching topologies," *In IEEE conference on decision and control (CDC)*, 2016.

[15] C. Zhao, M. Ogura, M. Kishida, A. Yassine, "Optimal resource allocation for dynamic product development process via convex optimization," *Research in Engineering Design*, 2020.

[16] A. Berman and R. J. Plemmons, "Nonnegative matrices in the mathematical sciences," *SIAM*, 1994.

[17] D. Luenberger, "Introduction to dynamic systems: theory, models, and applications," *New York: Wiley*, 1979.

[18] L. Benvenuti and L. Farina, "A tutorial on the positive realization problem," *IEEE Transactions on Automatic Control*, vol. 49, no. 5, pp. 651–664, 2003.

[19] L. Caccetta and V. G. Rumchev, “A survey of reachability and controllability for positive linear systems,” *Annals of Operations Research*, vol. 98, no. 1-4, pp. 101–122, 2000.

[20] Y. Ebihara, D. Peaucelle, and D. Arzelier, “L_1 gain analysis of linear positive systems and its application,” *IEEE Conference on Decision and Control and European Control* , pp. 4029–4034, 2011.

[21] M. S. Mahmoud and P. Shi, “Robust stability, stabilization and H_∞ control of time-delay systems with Markovian jump parameters,” *International Journal of Robust Nonlinear Control*, vol. 13, no. 8, pp. 755–784, 2003.

[22] S. Xu, J. Lam, and X. Mao, “Delay-dependent H_∞ control and filtering for uncertain Markovian jump systems with time-varying delays,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 54, no. 9, pp. 2070–2077, 2007.

[23] X. Zhao and Q. Zeng, “New robust delay-dependent stability and H_∞ analysis for uncertain Markovian jump systems with time-varying delays,” *Journal of the Franklin Institute*, vol. 347, no. 5, pp. 863–874, 2010.

[24] Y. Cao and J. Lam, “Stochastic stabilizability and H_∞ control for discrete-time jump linear systems with time delay,” *Journal of the Franklin Institute*, vol. 336, no. 8, pp. 1263–1281, 1999.

[25] Y. Cao and J. Lam, “Robust H_∞ control of uncertain Markovian jump systems with time-delay,” *IEEE Transactions on Automatic Control*, vol. 45, no. 1, pp. 77–83, 2000.

[26] W. Chen, Z. Guan, and P. Yu, “Delay-dependent stability and H_∞ control of uncertain discrete-time Markovian jump systems with mode-dependent time delays,” *Systems & Control Letters*, vol. 52, no. 5, pp. 361–376, 2004.

[27] J. Xiong and J. Lam, “Stabilization of discrete-time Markovian jump linear systems via time-delayed controllers,” *Automatica*, vol. 42, no. 5, pp. 747–753, 2006.

[28] A. Cetinkaya and T. Hayakawa, “Discrete-time switched stochastic control systems with randomly observed operation mode,” in *52nd IEEE Conference on Decision and Control*, 2013, pp. 85–90.

[29] A. Cetinkaya and T. Hayakawa, “Stabilizing discrete-time switched linear stochastic systems using periodically available imprecise mode information,” in *2013 American Control Conference*, 2013, pp. 3266–3271.

[30] A. Cetinkaya and T. Hayakawa, “Sampled-mode-dependent time-varying control strategy for stabilizing discrete-time switched stochastic systems,” in *2014 American Control Conference*, 2014, pp. 3966–3971.

[31] P. Bolzern, P. Colaneri, and G. De Nicolao, “Stochastic stability of positive Markov jump linear systems,” *Automatica*, vol. 50, no. 4, pp. 1181–1187, 2014.

[32] L. Farina and S. Rinaldi, *Positive Linear Systems.* Hoboken, NJ, USA: John Wiley & Sons, Inc., 2000.

[33] S. Zhu, Q.-L. Han, and C. Zhang, “L_1-stochastic stability and L_1-gain performance of positive Markov jump linear systems With time-delays: Necessary and sufficient conditions,” *IEEE Transactions on Automatic Control*, vol. 62, no. 7, pp. 3634–3639, 2017.

[34] O. L. V. do Costa, R. P. Marques, and M. D. Fragoso, *Discrete-Time Markov Jump Linear Systems.* London, U.K.: Springer-Verlag, 2005.

[35] A. Yassine, N. Joglekar, D. Braha, S. Eppinger, and D. Whitney, “Information hiding in product development: the design churn effect,” *Research in Engineering Design*, vol. 14, no. 3, pp. 145–161, 2003.

[36] A. P. Bowling, “ Dynamic performance, mobility, and agility of multilegged robots,” *Journal of Dynamic Systems, Measurement, and Control*, vol. 128, no. 4, pp. 765–777, 2006.

[37] M. Ogura and V. M. Preciado, “Stability of spreading processes over time-varying large-scale networks,” *IEEE Transactions on Network Science and Engineering*, vol. 3, no. 1, pp. 44–57, 2016.

[38] J. P. Hespanha, P. Naghshtabrizi, and Y. Xu, “A survey of recent results in networked control systems,” *Proceedings of the IEEE*, , vol. 95, no. 1, pp. 138–162, 2007.

[39] Y. Ji and H. J. Chizeck, “Controllability, stabilizability, and continuous-time Markovian jump linear quadratic control,” *IEEE Transactions on Automatic Control*, vol. 35 no. 7, pp. 777–788, 1990.

[40] O.L.V. Costa and R.P. Marques, “Mixed H_2/H_∞-control of discrete-time Markovian jump linear systems,” *IEEE Transactions on Automatic Control*, vol. 43, no. 1, pp. 95–100, 1998.

[41] J. C. Geromel and G. W. Gabriel, “Optimal $\mathscr{H}_2$ state feedback sampled-data control design of Markov Jump Linear Systems,” *Automatica*, vol. 54, pp. 182–188, 2015.

[42] H. Mukaidani, H. Xu, T. Shima, and V. Dragan, “ A stochastic multiple-leader–follower incentive stackelberg strategy for markov jump linear systems,” *IEEE Control Systems Letters*, vol. 1, no. 2, pp. 250–255, 2017.

[43] M. Ogura and C. F. Martin, “Generalized joint spectral radius and stability of switching systems,” *Linear Algebra and its Applications*, vol. 439, no. 8, pp. 2222–2239, 2013.

[44] Y. Hosoe and T. Hagiwara, “Equivalent stability notions, Lyapunov inequality, and its application in discrete-time linear systems with stochastic dynamics determined by an i.i.d. process,” *IEEE Transactions on Automatic Control*, vol. 64, no. 11, pp. 4764–4771, 2019.

[45] B. W. Johnson, “Design and Analysis of Fault-Tolerant Digital Systems,” *Addison-Wesley*, 1989.

[46] M. Ogura and C. F. Martin, “Stability analysis of positive semi-Markovian jump linear systems with state resets,” *SIAM Journal on Control and Optimization*, vol. 52, pp. 1809–1831, 2014.

[47] H. Schioler, M. Simonsen, and J. Leth, “Stochastic stability of systems with semi-Markovian switching,” *Automatica*, vol. 50, no. 11, pp. 2961–2964, 2014.

[48] J. Huang and Y. Shi, “Stochastic stability and robust stabilization of semi-Markov jump linear systems,” *International Journal of Robust and Nonlinear Control*, vol. 23, no. 18, pp. 2028–2043, 2012.

[49] L. Chen, X. Huang, and S. Fu, “Observer-based sensor fault-tolerant control for semi-Markovian jump systems,” *Nonlinear Analysis: Hybrid Systems*, vol. 22, pp. 161–177, 2016.

[50] B. Jiang, Y. Kao, H. R. Karimi, and C. Gao, “Stability and stabilization for singular switching semi-Markovian jump systems with generally uncertain transition rates,” *IEEE Transactions on Automatic Control*, vol. 63, no. 11, pp. 3919–3926, 2018.

[51] H. Shen, F. Li, S. Xu, and V. Sreeram, “Slow state variables feedback stabilization for semi-Markov jump systems with singular perturbations,” *IEEE Transactions on Automatic Control*, vol. 63, no. 8, pp. 2709–2714, 2018.

[52] Y. Wei, J. H. Park, J. Qiu, L. Wu, and H. Y. Jung, “Sliding mode control for semi-Markovian jump systems via output feedback,” *Automatica*, vol. 81, pp. 133–141, 2017.

[53] S. Boyd, S. J. Kim, L. Vandenberghe, and A. Hassibi, “A tutorial on geometric programming,” *Optimization and Engineering*, vol. 8, no. 1, pp. 67–127, 2007.

[54] J. F. C. Kingman, “A convexity property of positive matrices,” *Quarterly Journal of Mathematics*, vol. 12, no. 1, pp. 283–284, 1961.

[55] E. Kussell and S. Leibler, “Phenotypic diversity, population growth, and information in fluctuating environments,” *Science*, vol. 309, pp. 2075–2078, 2005.

[56] P. Colaneri, R. Middleton, Z. Chen, D. Caporale, and F. Blanchini, "Convexity of the cost functional in an optimal control problem for a class of positive switched systems," *Automatica*, , vol. 50, no. 4, pp. 1227–1234, 2014.

[57] Y. Ebihara, "H_2 analysis of LTI systems via conversion to externally positive systems," *IEEE Transactions on Automatic Control*, vol. 63, no. 8, pp. 2566–2572, 2018.

[58] J. Janssen and R. Manca, "Applied Semi-Markov Processes," *Springer-Verlag*, 2006.

[59] X. Feng, K. A. Loparo, Y. Ji, and H. J. Chizeck, "Stochastic stability properties of jump linear systems," *IEEE Transactions on Automatic Control*, vol. 37, pp. 38–53, 1992.

[60] V. M. Preciado, M. Zargham, C. Enyioha, A. Jadbabaie, and G. J. Pappas, "Optimal resource allocation for network protection against spreading processes," *IEEE Transactions on Control of Network Systems*, vol. 1, no. 1, pp. 99–108, 2014.

[61] A. Rantzer and M. E. Valcher, "A tutorial on positive systems and large scale control," *57th IEEE Conference on Decision and Control*, no. 978, pp. 3686–3697, 2018.

[62] M. Ogura, M. Kishida, and J. Lam, "Geometric programming for optimal positive linear systems," *IEEE Transactions on Automatic Control*, vol. 65, no. 11, pp. 4648–4663, 2020.

[63] J. E. Cohen, "Convexity of the dominant eigenvalue of an essentially nonnegative matrix," *Proceedings of the American Mathematical Society*, vol. 81, no. 4, pp. 657–658, 1981.

[64] P. Dorato, "Short time stability in linear time-varying systems," *In Proceedings of the IRE international convention record part 4*, pp. 83–87, 1961.

[65] S. Yu, X. Yu, B. Shirinzadeh, and Z. Man, "Continuous finite-time control for robotic manipulators with terminal sliding mode," *Automatica*, vol. 41, no. 11, pp. 1957–1964, 2005.

[66] H. Du, S. Li, and C. Qian, "Finite-time attitude tracking control of spacecraft with application to attitude synchronization," *IEEE Transactions on Automatic Control*, vol. 56, no. 11, pp. 2711–2717, 2011.

[67] M. Srinivasan, S. Coogan, and M. Egerstedt, "Control of multi-agent systems with finite time control barrier certificates and temporal logic," *In 2018 IEEE Conference on Decision and Control*, pp. 1991–1996, 2018.

[68] S. P. Bhat and D. S. Bernstein, "Finite-time stability of continuous autonomous systems," *SIAM Journal on Control and Optimization*, vol. 38, no. 3, pp. 751–766, 2000.

[69] F. Amato, M. Ariola, and P. Dorato, "Finite-time control of linear systems subject to parametric uncertainities and disturbances," *Automatica*, vol. 37, no. 9, pp. 1459–1463, 2001.

[70] Y. Hong, "Finite-time stabilization and stabilizability of a class of controllable systems," *Systems & Control Letters*, vol. 46, no. 4, pp. 231–236, 2002.

[71] X. Zhang, G. Feng, and Y. Sun, "Finite-time stabilization by state feedback control for a class of time-vary nonlinear systems," *Automatica*, vol. 48, no. 3, pp. 499–504, 2012.

[72] W. Haddad, S. Nersesov, and L. Du, "Finite-time stability for time-varying nonlinear dynamical systems," *In American Control Conference*, pp. 4135–4139, 2008.

[73] J. Köhler, C. Enyioha, and F. Allgöwer, "Dynamic resource allocation to control epidemic outbreaks a model predictive control approach," *In American Control Conference*, pp. 1546–1551, 2018.

[74] E. A. Hernandez-Vargas, P. Colaneri, and R. H. Middleton, “Optimal therapy scheduling for a simplified HIV infection model,” *Automatica*, vol. 49, no. 9, pp. 2874–2880, 2013.

[75] C. Nowzari, V. M. Preciado, and G. J. Pappas, “Analysis and control of epidemics: A survey of spreading processes on complex networks,” *IEEE Control Systems*, vol. 36, no. 1, pp. 26–46, 2016.

[76] R. Shorten, F. Wirth, and D. Leith, “A positive systems model of TCP-like congestion control: asymptotic results,” *IEEE/ACM Transactions on Networking*, vol. 14, no. 3, pp. 616–629, 2006.

[77] G. Chen and Y. Yang, “Finite-time stability of switched positive linear systems,” *International Journal of Robust and Nonlinear Control*, vol. 24, pp. 179–190, 2012.

[78] K. Ulrich, “The role of product architecture in the manufacturing firm,” *Research Policy*, vol. 24, no. 3, pp. 419–440, 1995.

[79] N.R. Joglekar, A. Yassine, S.D. Eppinger, and D.E. Whitney, “Performance of coupled product development activities with a deadline,” *Management Science*, vol. 47, no. 12, pp. 1605–1620, 2001.

[80] J. Alcaraz, C. Maroto, and R. Ruiz, “Solving the multi-mode resource-constrained project scheduling problem with genetic algorithms,” *Journal of the Operational Research Society*, vol. 54, no. 6, pp. 614–626, 2003.

[81] F.F. Boctor, “A new and efficient heuristic for scheduling projects with resource restrictions and multiple execution modes,” *European Journal of Operational Research*, vol. 90, no. 2, pp. 349–361, 1996.

[82] J. W. Rivkin and M. Siggelkow, “Patterned interactions in complex systems: implications for exploration,” *Management Science*, vol. 53, no. 7, pp. 1068–1085, 2007.

[83] P. Erdös and A. Rényi, “On random graphs I,” *Publicationes Mathematicae Debrecen 6*, pp. 290–297, 1959.

[84] K.T. Ulrich and S.D. Eppinger, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, no. 6684, pp. 440–442, 2008.

[85] A.L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.

[86] T.L. Yu, A. Yassine, and D.E. Goldberg, "An information theoretic method for developing modular architectures using genetic algorithms," *Research in Engineering Design*, vol. 18, no. 2, pp. 91–109, 2007.

[87] D. Braha and Y. Bar-Yam, "Information flow structure in large-scale product development organizational networks," *Journal of Information Technology*, vol. 19, no. 4, pp. 244–253, 2004.

[88] D. Braha and Y. Bar-Yam, "Topology of large-scale engineering problem-solving networks," *Physical Review E*, vol. 69, no. 1, 016113, 2004.

[89] C.Y. Baldwin and K.B. Clark, "Design Rules," *The MIT Press*, 2000.

[90] M. Newman, "Networks," *Oxford University Press*, 2010.

[91] J. Benesty, J. Chen, Y. Huang and I. Cohen, "Pearson Correlation Coefficient," *Springer*, Berlin, 2009.

[92] J. F. Maier, D. C. Wynn, W. Biedermann, U. Lindemann and P. J. Clarkson, "Simulating progressive iteration, rework and change propagation to prioritise design tasks," *Research in Engineering Design*, vol. 25, no. 4, pp. 283–307, 2014.

[93] B.G. Tabachnick and L.S. Fidell, "Experimental Designs Using ANOVA," *Duxbury*, 2007.

[94] A. Yassine, "Managing the development of complex product systems: an integrative literature review," *IEEE Transactions on Engineering Management*, pp. 1–18, 2019.

[95] C.M. Christensen, “Disruption, disintegration and the dissipation of differentiability,” *Industrial and Corporate Change*, vol. 11, no. 5, pp. 955–993, 2002.

[96] M. Ogura, J. Harada, M. Kishida and A. Yassine, “Resource optimization of product development projects with time-varying dependency structure,” *Research in Engineering Design*, vol. 30, no. 3, pp. 435–452, 2019.

[97] K. Frenken and S. Mendritzki, “Optimal modularity: a demonstration of the evolutionary advantage of modular architectures,” *Journal of Evolutionary Economics*, vol. 22, no. 5, pp. 935–956, 2012.

[98] J. Luo, “A simulation-based method to evaluate the impact of product architecture on product evolvability,” *Research in Engineering Design*, vol. 26, no. 4, pp. 355–371, 2015.

[99] D. Batallas and A. Yassine, “Information leaders in product development organizational networks: social network analysis of the design structure matrix,” *IEEE Transaction on Engineering Management*, vol. 43, no. 4, pp. 570–582, 2006.

[100] S. Collins, A. Yassine, and S. Borgatti, “Evaluating product development systems using network analysis,” *Systems Engineering Journal*, vol. 12 no. 1, pp. 55–68, 2009.

[101] M. Sosa, J. Mihm, and T. Browning , “Degree distribution and quality in complex engineered systems,” *Journal of mechanical design*, vol. 133, no. 10, 101008, 2011.

Publication List

Journal Articles

[1] W. Mei, C. Zhao, M. Ogura, and K. Sugimoto, "Mixed H_2/H_∞ control of delayed Markov jump linear systems," *IET Control Theory & Applications*, vol. 14, no. 15, pp. 2076–2083, 2020.

[2] C. Zhao, M. Ogura and K. Sugimoto, "Stability optimization of positive semi-Markov jump linear systems via convex optimization," *SICE Journal of Control, Measurement, and System Integration*, vol. 13, no. 5, pp. 233–239, 2020.

[3] C. Zhao, M. Ogura, M. Kishida and A. Yassine, "Optimal resource allocation for dynamic product development process via convex optimization," *Research in Engineering Design*, 2020.

Conference Proceedings

[1] C. Zhao, M. Ogura and K. Sugimoto, "Finite-time control of discrete-time positive linear systems via convex optimization," in *the 59th SICE Annual Conference*, pp. 1230–1235, 2020.

[2] M. Ogura and C. Zhao, "DC programming for optimization of dynamic buffer networks," in *the 8th Multi-symposium on Control Systems*, pp. 1D1-25, 2021.