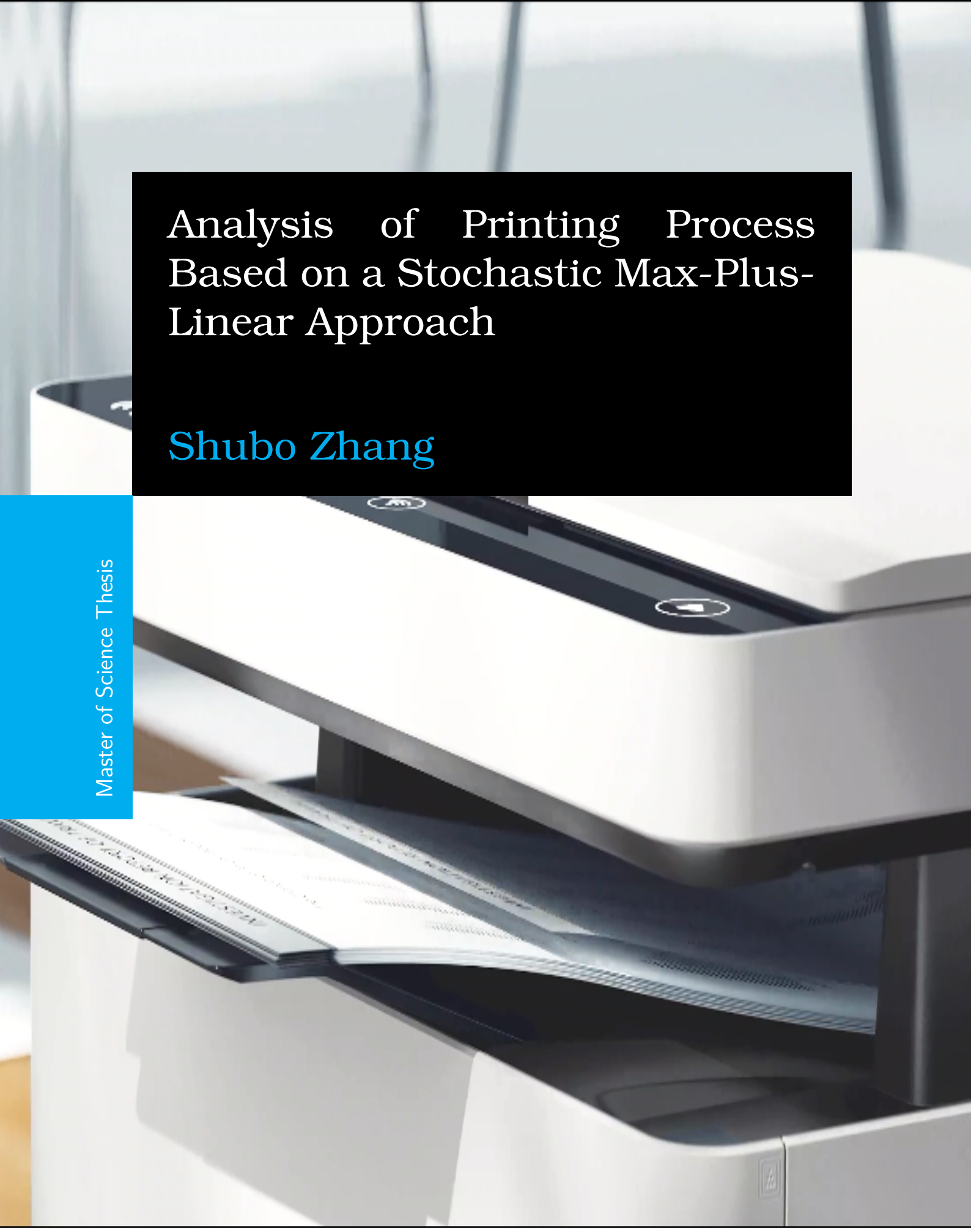


Analysis of Printing Process Based on a Stochastic Max-Plus-Linear Approach

Shubo Zhang

Master of Science Thesis

TU Delft Delft University of Technology

Delft Center for Systems and Control

Analysis of Printing Process Based on a Stochastic Max-Plus-Linear Approach

Master of Science Thesis

For the degree of Master of Science in Systems and Control at Delft University of Technology

Shubo Zhang

October 28, 2021

Faculty of Mechanical, Maritime and Materials Engineering (3mE) · Delft University of Technology

Copyright © Delft Center for Systems and Control (DCSC)
All rights reserved.

Abstract

The scheduling algorithm of the printer is an important factor that affects printing efficiency. For current printers, paper scheduling often follows the first-in-first-out principle, so it is often not optimal. The printer system is a type of semi-cyclic discrete-event system with synchronization but no concurrency. The system contains a set of operations that vary over a limited sequence of operations, which can be further modelled as a Switching Max-Plus-Linear (SMPL) system through max-plus-linear algebra, and can be switched between different modes of operation. Since max-plus algebra has a significant analogy with conventional algebra, some properties in conventional algebra can be applied to such kind of system, making the system easier to achieve optimal control.

In this report, a printing scheduler modelled by the max-plus algebra is presented. We review and summarize the basic knowledge of the Max-Plus-Linear (MPL) system from the existing literature in the first half of the report. We first introduce the basic properties of max-plus algebra and SMPL systems. Then we turn to the stochastic case, and consider the two types of stochastic uncertainty that may be included in the SMPL system, namely stochastic parametric uncertainty and stochastic mode switching uncertainty. Next, we review a Model Predictive Control (MPC) approach that can achieve optimal scheduling of systems containing two random uncertainties.

In the second half of the report, based on the content summarized in the first half, we make a further derivation of the printer scheduling system modelled by SMPL approach. We first present the modelling framework of the printer. Three working modes, namely duplex mode, idle mode and simplex mode, are modelled separately and merged into a compact form. Then a scheduler that takes the feeding and processing time of each sheet of paper as design variables is introduced. It can find the global optimal schedule of different types of paper by solving the Mixed-Integer Linear Programming (MILP) problem. After that, we discuss cases involving switching, and consider switching between two sizes of paper and two working modes. Three possible intermediate switching modes are designed. Finally, we take noise/interference into consideration, discuss the impact of noise on scheduling, and the changes that the previously proposed methods may require to achieve optimal scheduling in the presence of noise.

Table of Contents

Acknowledgements

I would like to express my sincerest appreciation to all those who once gave me support to complete my master thesis project. Special gratitude is given to my supervisor dr.ir.A.J.J. van den Boom, whose academic expertise and encouraging supervision helped me better work on my project. I would also like to extend this thanks to ir. A. Gupta, who kindly provided me constructive suggestions and technical facilitation during the project.

Furthermore, I would also like to express my thanks especially to prof.dr.ir. B. De Schutter, whose inspiring ideas in the literature aid me better in navigating project work throughout the stages. Another heartfelt appreciation goes to dr.ir. J. Xu and dr.ir. S. Farahani, whose approximate methods proposed in the literature are also of great help to this project.

Last but not least, I would like to additionally acknowledge my family and friends for their consistent spiritual support during the tough COVID-19 period. Their encouragement allowed me to maintain a positive attitude and complete the project effectively.

Delft, University of Technology — Shubo Zhang
October 28, 2021

"Yesterday is history, tomorrow is a mystery, today is a gift of God, which is why we call it the present."

— *Bill Keane*

Chapter 1

Introduction

1-1 Printer Scheduling Problem

In recent years, with the rapid development of various industries, the scale of enterprises has continued to expand and their performance has continued to grow. Business activities triggered by this have become more frequent, and people's daily reading volume has also increased significantly. Although the use of electronic documents can reduce the demand for some paper documents, in many industries such as construction, machining, surveying, etc., the importance of paper documents is still very high and cannot be replaced. Due to increased competition and limited budgets, companies have higher requirements in choosing printing solutions. They not only hope to have better purchase cost and use cost but also hope that the printer is faster and the printer supports more types of jobs.

The printers currently used in the market are mostly small printers. This type of printer can only print on a single sheet of paper, and the next sheet of paper can only be started after the previous sheet of paper is completed. Although the price of this type of printer is low, it has the following disadvantages: the high cost of consumables, for users with greater printing needs, the overall cost required is high; errors are prone to occur, this type of printer not only has a high probability of paper overlap but also a high rate of component damage; the printing logic of the printer is simple, which also leads to relatively low printing efficiency. Therefore, this type of printer cannot meet the ever-increasing customer demand in the market.

In order to have a higher market share, printer manufacturers continue to update their technology. The development of new machines with high modularity and high speed has been summarized as the common goal of major manufacturers [1]. This type of printer is highly automated and can perform continuous large-scale printing jobs with a low error rate. It can process multiple of different types of sheets at the same time, usually consisting of a paper source, a system for transporting paper, and a module for transporting images to paper. The crucial elements include Paper Input Module (PIM), Image Transfer Station (ITS), Invert Module (IM), Re-entry Module (RM) and Paper Output Module (POM) (as shown in Figure 1-1) [2].

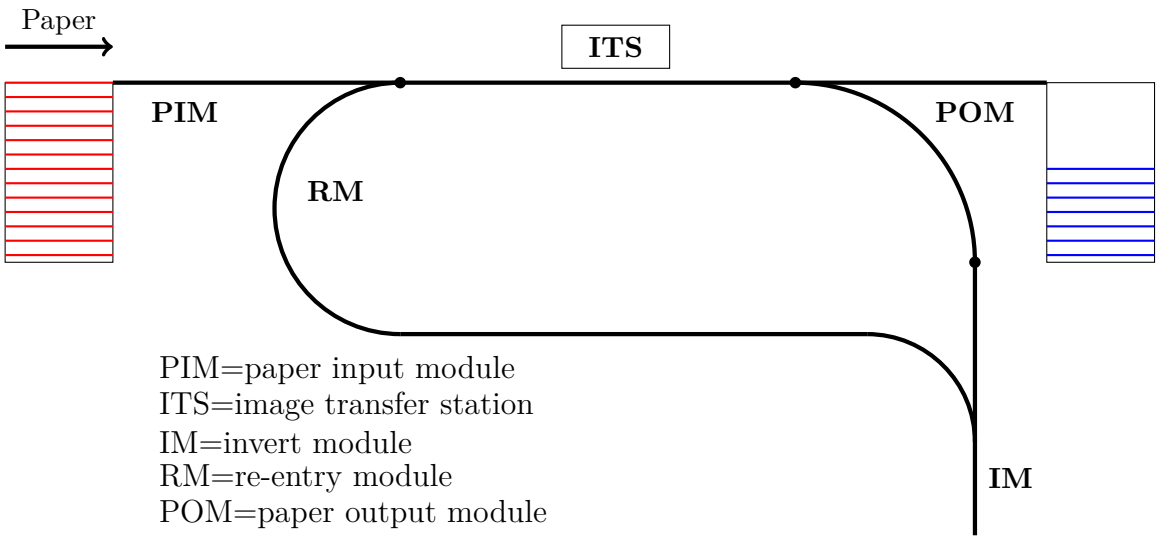


Figure 1-1: A Schematic of the Printer Modules

Since some of the events in the printer are taken place simultaneously, a supervisory controller which schedules the modules in order to optimize certain criteria is required to ensure that the image can be printed accurately at the required position [3] [4]. However, during the duplex printing, some of the resources are involved twice per sheet, which means the system has to be operated cyclically. This kind of system can be modelled as Discrete Event Systems (DESs), which in general will lead to an NP-hard Problem in conventional algebra [5]. Since conventional algorithms are difficult to find the optimal solution for such problems, the current printer scheduling is usually not optimal.

Fortunately, there exists a subclass of DESs in which only synchronization and no concurrency or choice occur. This type of DESs can be modelled by max-plus algebra and can be converted into a kind of linear system under max-plus operation, which is named as Max-Plus-Linear (MPL) systems [6]. In addition, the MPL system has been extended to a Switching Max-Plus-Linear (SMPL) system, in which different operating modes can be modelled [7].

Due to the excellent modelling ability of max-plus algebra on this kind of DESs, SMPL system has already been used to model the printer system [1] [2]. The supervisory controller based on max-plus algebra can not only generate global optimal scheduling for multiple types of sheets, but also deal with the system with noise/disturbance. The optimal scheduling problem is finally transformed into a Mixed-Integer Linear Programming (MILP) problem, which can be solved by a variety of algorithms. In this report, how to implement the SMPL system to the printer system and how to obtain optimal scheduling with SMPL system in different cases will be discussed.

1-2 Research Questions

The issues to be studied in the report are as follows:

Q1. How to convert the printer system to SMPL system?

Q2. How to achieve optimal scheduling of printers with max-plus algebra?

Q3. How to order the system when mode/paper size switching happens?

Q4. What happens to the optimal scheduling problem when the system is disturbed?

Q5. In which aspects can the method be further improved?

1-3 Outline of the Report

In this report, Chapters 2-4 are mainly the summary of the literature study content, and Chapters 5-8 are mainly the author's further derivation based on the literature study. The specific organizational structure of this report is as follows:

- Chapter 2: the foundations of max-plus algebra, MPL systems, and SMPL systems are explained. In addition, this chapter also discusses the dynamic graph and stability criterion of MPL systems and SMPL systems.
- Chapter 3: parameter uncertainties and switching uncertainties of SMPL systems are introduced respectively and a model containing both types of stochastic is presented.
- Chapter 4: a general Model Predictive Control (MPC) method for stochastic SMPL systems is introduced.
- Chapter 5: the printer scheduling model and related graphs are introduced. And state equations of three modes of the printer are derived. Finally, a compact form of the state equations is given.
- Chapter 5: solve the optimal control problem for one mode printers in the deterministic case by transforming it into a MILP problem.
- Chapter 7: introduce switching in the printer, and design intermediate modes for different situations.
- Chapter 8: on the basis of the previous chapters, discuss the switching with stochasticity.
- Chapter 9: a brief summary and conclusion of the report are given, and possible future work is discussed.

Chapter 2

Max-Plus Algebra and Max-Plus-Linear Systems

In this chapter, the basic operations of the max-plus algebra and the max-plus-algebraic matrix will be given first. Then the properties of max-plus-linear systems and switching max-plus-linear systems will be discussed respectively.

2-1 Basic Operations of Max-Plus Algebra

According to [8] and [9], define $\varepsilon = -\infty$ and $\mathbb{R}_\varepsilon = \mathbb{R} \cup \{\varepsilon\}$, then the max-plus-algebraic addition ($\oplus$) and multiplication ($\otimes$) can be defined as:

$$x \oplus y = \max(x, y) \tag{2-1}$$

$$x \otimes y = x + y \tag{2-2}$$

for any x,y $\in \mathbb{R}_\varepsilon$.

Let $r \in \mathbb{R}$, the r-th max-plus-algebraic power of $x \in \mathbb{R}$ is denoted by $x^{\otimes^r}$ and corresponds to rx in conventional algebra.

There exist remarkable analogies between the max-plus operation ($\oplus$ and $\otimes$) and conventional operation (addition and multiplication): many theorems of conventional linear algebra can be translated to the max-plus algebra. One major difference to note is that there exists no inverse element in the max-plus algebra with respect to $\oplus$ in $\mathbb{R}_\varepsilon$ [6]. We can regard ε as the 'zero' from linear algebra ($x \oplus \varepsilon = x = \varepsilon \oplus x$ and $x \otimes \varepsilon = \varepsilon = \varepsilon \otimes x$), and 0 as the 'one' from linear algebra ($x \otimes 0 = x = 0 \otimes x$).

The rules of the evaluation order of the max-plus-algebraic operator correspond to the rules of conventional algebra, which means that max-plus-algebraic power has the highest priority, and max-plus-algebraic multiplication has a higher priority than max-plus-algebraic addition [6].

Based on these operation rules, a special type of functions, namely max-affine functions (or max-plus-scaling functions, donated S_{mpns}) is introduced, i.e., functions f of the form

$$f(z) = \max_i \left(\alpha_{i,1} z_1 + \cdots + \alpha_{i,n} z_n + \beta_i\right) \tag{2-3}$$

with variable $z \in \mathbb{R}_\varepsilon^{n_z}$, constants $\alpha_{i,j} \in \mathbb{R}^+$ and $\beta_i \in \mathbb{R}$, where n_z denotes the size of the vector z and $\mathbb{R}^+$ denotes the set of non-negative real numbers. According to [10], S_{mpns} is closed under the operations maximization, addition, and multiplication by a scalar.

2-2 Max-Plus-Algebraic Matrix Operations

According to [8] and [9], define matrices $A, B \in \mathbb{R}_\varepsilon^{m\times n}$ and $C \in \mathbb{R}_\varepsilon^{n\times p}$, and then apply the basic max-plus-algebraic operations to the matrices, the following max-plus-algebraic matrix operations can be obtained:

$$(A \oplus B)_{ij} = a_{ij} \oplus b_{ij} = \max\left(a_{ij}, b_{ij}\right) \tag{2-4}$$

$$(A \otimes C)_{ij} = \bigoplus_{k=1}^{n} a_{ik} \otimes c_{kj} = \max_{k=1,\cdots,n} (a_{ik} + c_{kj}) \tag{2-5}$$

Another important operations on matrices is max-plus Hadamard product (or named as max-plus element-wise multiplication) "$\odot$", which requires both matrices have the same size. Multiplying two matrices $A^{n\times m} \in \mathbb{R}_\varepsilon$ and $B^{n\times m} \in \mathbb{R}_\varepsilon$ results in:

$$\begin{aligned} A \odot B &= \begin{bmatrix} a_{11} & \cdots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nm} \end{bmatrix} \odot \begin{bmatrix} b_{11} & \cdots & b_{1m} \\ \vdots & \ddots & \vdots \\ b_{n1} & \cdots & b_{nm} \end{bmatrix} \\ &= \begin{bmatrix} a_{11} \otimes b_{11} & \cdots & a_{1m} \otimes b_{1m} \\ \vdots & \ddots & \vdots \\ a_{n1} \otimes b_{n1} & \cdots & a_{nm} \otimes b_{nm} \end{bmatrix} \\ &= \begin{bmatrix} a_{11} + b_{11} & \cdots & a_{1m} + b_{1m} \\ \vdots & \ddots & \vdots \\ a_{n1} + b_{n1} & \cdots & a_{nm} + b_{nm} \end{bmatrix}. \end{aligned}$$

Similar to the conventional algebra, the matrix $\varepsilon_{m\times n}$ with all the elements of ε is the max-plus-algebraic zero matrix, satisfying $(\varepsilon_{m\times n})_{ij} = \varepsilon,\ \forall i, j$; the max-plus-algebraic identity matrix E is defined as a diagonal matrix, satisfying $E_{ii} = 0,\ \forall i$ and $E_{ij} = \varepsilon,\ \forall i, j$ *with* $i \neq j$.

For a matrix $A \in \mathbb{R}_\varepsilon^{n\times n}$, the max-plus-algebraic matrix power is defined as $A^{\otimes^0} = E$ and $A^{\otimes^k} = A \otimes A^{\otimes^{(k-1)}}$ for $k \in \mathbb{N} \setminus \{0\}$. If for a max-plus diagonal matrix $S = \mathrm{diag}_\oplus(s_1, \cdots, s_n)$ all values s_i are finite, then the inverse of S is equal to $S^{\otimes^{-1}} = \mathrm{diag}_\oplus(-s_1, \cdots, -s_n)$. There holds $S \otimes S^{-1} = S^{\otimes^{-1}} \otimes S = E$.

2-3 Max-Plus-Linear Systems

In [9], [8], [11] and [7], it has been proposed that a type of Discrete Event Systems (DESs) with synchronization and no concurrency or choice occur can be modelled using max-plus operation. This type of systems are named as Max-Plus-Linear (MPL) DESs, which can be described by the following form

$$x(k) = A(k) \oplus x(k-1) \otimes B(k) \otimes u(k) \tag{2-6}$$
$$y(k) = C(k) \otimes x(k) \tag{2-7}$$

with $A \in \mathbb{R}_{\varepsilon}^{n\times n}$, $B \in \mathbb{R}_{\varepsilon}^{n\times m}$, and $C \in \mathbb{R}_{\varepsilon}^{l\times n}$, where n is the number of states, m is the number of inputs, l is the number of outputs, and k is the event counter. The matrices A, B, and C usually consist of sums or maximizations of internal process times, transportation times, etc. [8]. One thing that needs to be noted is that the states are event times, which means the state $x(k)$, the input $u(k)$ and the output $y(k)$ contains the time instants at which the internal events, input events and output events occur for the k-th time respectively.

2-4 Switching Max-Plus-Linear Systems

According to [7], Switching Max-Plus-Linear (SMPL) systems are a type of discrete event systems that can switch between different modes of operation. In each mode, the system is described by a max-plus-linear state equation and a max-plus-linear output equation. For different modes, it may have different system matrices. If the systems are in mode $\ell(k) \in \{1, \cdots, n_L\}$ for every step k, then SMPL systems can be described as

$$x(k) = A^{(\ell(k))} \otimes x(k-1) \oplus B^{(\ell(k))} \otimes u(k) \tag{2-8}$$
$$y(k) = C^{(\ell(k))} \otimes x(k) \tag{2-9}$$

where the matrices $A^{(\ell(k))}$, $B^{(\ell(k))}$ and $C^{(\ell(k))}$ are the system matrices for mode $\ell(k)$.

In general, the moments of switching are determined by a switching mechanism [7]. The mode $\ell(k)$ is mainly depend on the previous state $x(k-1)$, the previous mode $\ell(k-1)$, the input variable $u(k)$ and an auxiliary control variable $v(k)$:

$$\ell(k) = \phi_s\left(x(k-1), \ell(k-1), u(k), v(k)\right) \tag{2-10}$$

where ϕ_s is the switching function.

For a given integer N, the set of all possible consecutive mode switching vectors is denoted as

$$\mathcal{L}_N = \left\{ [\ell_1, \cdots, \ell_N]^T \mid \ell_m \in \{1, \cdots, n_L\}, m = 1, \cdots, N \right\} \tag{2-11}$$

Denote the probability of switching to mode $\ell(k)$ given $\ell(k-1), x(k-1), u(k)$ and $v(k)$ by

$$P\left[L = \ell(k) \mid \ell(k-1), x(k-1), u(k), v(k)\right] \tag{2-12}$$

For deterministic switching (systems without uncertainties), the probability functions are piecewise constant with values either 0 or 1.

The model given in Equation (2-8) is often referred to as an explicit SMPL model. Another type of model, named as implicit SMPL model, is given by

$$x(k) = A_0^{(\ell(k))} \otimes x(k) \oplus A_1^{(\ell(k))} \otimes x(k-1) \oplus B_0^{(\ell(k))} \otimes u(k) \tag{2-13}$$

The implicit SMPL model can be transformed into the explicit form (2-8), with

$$A^{(\ell(k))} = \left[A_0^{(\ell(k))}\right]^* \otimes A_1^{(\ell(k))}$$
$$B^{(\ell(k))} = \left[A_0^{(\ell(k))}\right]^* \otimes B_0^{(\ell(k))}$$

where $\left[A_0^{(\ell(k))}\right]^*$ is max-plus Kleene star of $A_0^{(\ell(k))}$, which is given by:

$$\left[A_0^{(\ell(k))}\right]^* = E \oplus A_0^{(\ell(k))} \oplus \left[A_0^{(\ell(k))}\right]^{\otimes^2} \oplus \cdots \oplus \left[A_0^{(\ell(k))}\right]^{\otimes^n}.$$

However, the system matrices obtained by rewriting the implicit SMPL model into an explicit SMPL are usually quite complex, which makes it hard to introduce additional constraints in the final Mixed-Integer Linear Programming (MILP) problem, and the optimization with the explicit model is thus much slower than the optimization with the implicit model [2]. Therefore, sometimes using implicit models in model predictive scheduling may be more efficient.

2-5 Dynamic Graph

There exists a close relation between max-plus algebra and graphs. Important properties such as irreducibility and structural controllability can be determined from the precedence graph of a max-plus system [8] [6]. However, if there exists frequent mode switching in the max-plus system, the precedence graph cannot be used anymore since the mode $\ell(k)$ and the A-matrix of the system may change for every cycle k. For a better understanding of the switching behaviour, the dynamic graph concept was introduced [12]. The definition is given as follows:

Definition 1. Consider an implicit SMPL system for a given mode sequence $\ell \in \mathcal{L}_N$. The dynamic graph $G = (G_0^1, G_1^1, G_0^2, G_1^2, \cdots, G_0^m, G_1^m, , \cdots, H^m)$ is a sequence of graphs, where $G_0^k = \left(X^k, E_0^k\right)$ is a directed graph with only non-positive circuit weights, $G_1^k = \left(X^k, X^{k-1}, E_1^k\right)$, is a directed bipartite graph with E_1^k being the set of edges from X^{k-1} to X^k, and $H^k(X^k, U^k, E_{(u)}^k)$ is a directed bipartite graph with $E_{(u)}^k$ being the set of edges from U^k to X^k. The nodes X_k represent the state of a system at cycle k and the nodes U_k correspond to the input of the system at cycle k. The weight of the edge of G_0^k from node $\left[X^k\right]_j$ to $\left[X^k\right]_i$ is equal to $[A_0(\ell(k))]_{i,j}$, the weight of the edge of G_1^k from node $\left[X^{k-1}\right]_j$ to $\left[X^k\right]_i$ is equal to $[A_1(\ell(k))]_{i,j}$, and the weight of the edge of H^k from node $\left[U^k\right]_j$ to $\left[X^k\right]_i$ is equal to $[B(\ell(k))]_{i,j}$.

The main advantage of the dynamic graph is that it can handle the switching characteristics of the SMPL system [2].

2-6 Stability Condition Analysis

In this section, the stability criteria of max-plus-linear systems and switching max-plus-linear systems will be discussed based on the research of [13] and [2].

2-6-1 Stability Conditions for Max-Plus-Linear Systems

In this report, we adopt the notion of stability for discrete-event systems from [14], in which a discrete-event system is considered to be stable if all its buffer levels remain bounded. Let $r(k)$ be the reference signal (or named as the due date), which is a reference for the time that the output event should occur. If the output event occurs after the due date, a delay will happen in the system. For a proper operation of the system, the delays should remain bounded.

Definition 2. An asymptotically increasing reference signal $r(k)$ is defined as

$$r(k) = \rho k + d(k), \text{ where } |d_i(k)| \leq d_{max}, \forall i \tag{2-14}$$

where r and d are vectors and ρ is a scalar, satisfying $\rho > 0$.

All the buffer levels in a discrete-event system are bounded if the dwelling times of the parts or batches in the system remain bounded [13]. So for an MPL system with an asymptotically increasing reference signal $r(k)$, stability is achieved if and only if there exist finite constants k_0, M_{yr}, M_{yx} and M_{xu} such that

$$|y_i(k) - r_i(k)| \leq M_{yr}, \quad \forall i \tag{2-15}$$

$$|y_i(k) - x_j(k)| \leq M_{yx}, \quad \forall i, j \tag{2-16}$$

$$|x_j(k) - u_m(k)| \leq M_{xu}, \quad \forall j, m \tag{2-17}$$

for all $k > k_0$. Condition (2-15) means that the delay between the actual output date $y(k)$ and the due date $r(k)$ remains bounded (for $y - r < \infty$), and on the other hand, that the stock time will remain bounded (for $r - y < \infty$). Conditions (2-16) and (2-17) mean that the throughput time (i.e. the time between the starting date $u(k)$ and the output date $y(k)$) is bounded.

In summary, for a max-plus-linear system with a reference signal $r(k)$, condition (2-14)-(2-17) imply finite buffer levels.

2-6-2 Stability Conditions for Switching Max-Plus-Linear Systems

Similar to max-plus-linear systems, stability is not an intrinsic feature of the SMPL system, but it also depends on the reference signal. We use the concept of maximum autonomous growth rate:

Definition 3. Consider an SMPL system of the form (2-8) and (2-9) with system matrices $A^{(\ell)}, \ell = 1, \cdots, n_L$. Define the matrices $A_\alpha^{(\ell)}$ with $\left[A_\alpha^{(\ell)}\right]_{i,j} = \left[A^{(\ell)}\right]_{i,j} - \alpha$. Define the set $S_{\text{fin},n}$ of all $n \times n$ max-plus diagonal matrices with finite diagonal entries, that is

$$S_{\text{fin},n} = \left\{S \mid S = \text{diag}_\oplus\left(s_1, \cdots .s_n\right), s_i \text{ is finite}\right\}$$

The maximum autonomous growth rate λ of the SMPL system is defined by

$$\lambda = \min\left\{\alpha \mid \exists S \in S_{\text{fin},n} \text{ such that } \left[S \otimes A_\alpha^{(\ell)} \otimes S^{\otimes^{-1}}\right]_{i,j} \leq 0, \quad i,j,\ell\right\} \tag{2-18}$$

The maximum growth rate λ can be easily computed by solving a linear programming problem.

Definition 4. Let $\alpha \in \mathbb{R}$ be given. Define the matrices $A_\alpha^{(\ell)}$ with $\left[A_\alpha^{(\ell)}\right]_{i,j} = \left[A^{(\ell)}\right]_{i,j} - \alpha$. An SMPL system is structurally controllable if there exists a finite positive integer N such that for all $\widetilde{\ell} = [\ell_1, \cdots, \ell_N]^T \in \mathcal{L}_N$, the matrices

$$\Gamma_\alpha^N(\widetilde{\ell}) = \begin{bmatrix} A_\alpha^{(\ell_N)} \otimes \cdots \otimes A_\alpha^{(\ell_2)} \otimes B^{(\ell_1)} & \cdots & A_\alpha^{(\ell_N)} \otimes A_\alpha^{(\ell_{N-1})} \otimes B^{(\ell_{N-2})} & A_\alpha^{(\ell_N)} \otimes B^{(\ell_{N-1})} & B^{(\ell_N)} \end{bmatrix}$$

are row-finite, i.e. in each row there is at least one entry different from ε.

Definition 5. Let $\alpha \in \mathbb{R}$ be given. Define the matrices $A_\alpha^{(\ell)}$ with $\left[A_\alpha^{(\ell)}\right]_{i,j} = \left[A^{(\ell)}\right]_{i,j} - \alpha$. An SMPL system is structurally observable if there exists a finite positive integer M such that for all $\widetilde{\ell} = [\ell_1, \cdots, \ell_M]^T \in \mathcal{L}_M$, the matrices

$$O_\alpha^M(\widetilde{\ell}) = \begin{bmatrix} C_\alpha^{(\ell_N)} \otimes A_\alpha^{(\ell_N)} \otimes \cdots \otimes A_\alpha^{(\ell_2)} \\ \vdots \\ C_\alpha^{(\ell_N)} \otimes A_\alpha^{(\ell_N)} \otimes A_\alpha^{(\ell_{N-1})} \\ C_\alpha^{(\ell_N)} \otimes A_\alpha^{(\ell_N)} \\ C_\alpha^{(\ell_N)} \end{bmatrix}$$

are column-finite, i.e. in each column there is at least one entry different from ε.

Remark: Note that the structural controllability and structural observability are structural properties and do not depend on the actual value of α. If a SMPL system is structurally controllable (observable) for one finite value of α it is structurally controllable (observable) for any finite value of α [13].

Theorem 1. Consider an SMPL system with maximum autonomous growth rate λ and consider a reference signal (2-14) with grow rate ρ. Define the matrices $A_\rho^{(\ell)}$ with $\left[A_\rho^{(\ell)}\right]_{i,j} = \left[A^{(\ell)}\right]_{i,j} - \rho$. Now if

1. $\rho > \lambda$,
2. the system is structurally controllable, and

3. the system is structurally observable,

then any input signal

$$u(k) = \rho k + \mu(k), \text{ where } |\mu_i(k)| \leq \mu_{max}, \quad \forall i, \forall k \tag{2-19}$$

for a finite value μ_{max}, will stabilize the SMPL system.

The proof of the theorem is given by [13].

Chapter 3

Uncertainties in Stochastic Switching Max-Plus-Linear Systems

In the previous chapter, the deterministic Switching Max-Plus-Linear (SMPL) systems have been introduced. However, in reality, due to modelling errors and perturbations/noise, or due to the random sequence of synchronization of the event steps, uncertainties are introduced into the system. The uncertainties of SMPL system can in general be classified as parametric uncertainty [10] and mode switching uncertainty [13]. In this chapter, both kinds of uncertainties are assumed to be stochastic quantities. An introduction of each kind of stochastic SMPL will be given first, and then a summary of these two kinds of uncertainties will be made. Finally, one approach of how to combine both types of uncertainties into one SMPL system will be presented based on the result from [15].

3-1 Switching Max-Plus-Linear Systems with Uncertain Parameter

3-1-1 Introduction

In contrast to conventional linear systems, where uncertainties such as noise and disturbances are usually modelled by including an extra term in the system equations and considered to be additive terms, the influence of noise and disturbances in SMPL systems is not max-plus-additive, but max-plus-multiplicative [8] [10]. Therefore, uncertainties in SMPL systems are included in the system matrices. Ignoring the uncertainties will result in poor tracking behaviour or even an unstable closed loop.

Consider the following stochastic SMPL system:

$$x(k) = A^{(\ell(k))} \otimes x(k-1) \oplus B^{(\ell(k))} \otimes u(k) \tag{3-1}$$

$$y(k) = C^{(\ell(k))} \otimes x(k) \tag{3-2}$$

where $A^{(\ell(k))}$, $B^{(\ell(k))}$ and $C^{(\ell(k))}$ represent uncertain system matrices due to modelling errors or disturbances for k-th event.

If all such uncertainties caused by the disturbances and errors in the estimation of physical variables are gathered into an uncertainty vector $e(k)$, and assuming that $e(k)$ is a stochastic variable that captures the complete time-varying aspect of the system, then $A^{(\ell(k))}$, $B^{(\ell(k))}$ and $C^{(\ell(k))}$ will belong to the set of max-affine functions of $e(k)$ [10], i.e.

$$A^{(\ell(k))} \in S^{n\times n}_{mpns}\left(e(k)\right), \qquad B^{(\ell(k))} \in S^{n\times n}_{mpns}\left(e(k)\right), \qquad C^{(\ell(k))} \in S^{n\times n}_{mpns}\left(e(k)\right)$$

Since $S_{mpns}\left(e(k)\right)$ is closed under the operations maximization, addition, and multiplication by a scalar, when applying the control algorithm such as MPC to the system, the cost function J and some of the constraints consist of the system matrices will also be stochastic and belong to the set of $S_{mpns}\left(e(k)\right)$.

Let

$$f\left(e(k)\right) = \max_i \left(\alpha_{i,1}e(k)_1 + \cdots + \alpha_{i,n}e(k)_n + \beta_i\right) = \max_i \left(\alpha_i^T e(k) + \beta_i\right)$$

be the function in the set of $S_{mpns}\left(e(k)\right)$, where $\alpha_{i,n}$ and β_i are constant coefficients.

To solve such kind of stochastic constrained optimisation problems, two stochastic quantities are important, namely the expectation of $f(e(k))$:

$$\mathbb{E}\left[f(e(k))\right] = \mathbb{E}\left[\max_i \left(\alpha_i^T e(k) + \beta_i\right)\right] \tag{3-3}$$

where $\mathbb{E}\left[.\right]$ denotes the expectation,

and the chance constraint that $f(e(k))$ is less than B:

$$\mathbb{P}\left[f(e(k)) \leq B\right] = \mathbb{P}\left[\max_i \left(\alpha_i^T e(k) + \beta_i\right) \leq B\right] \tag{3-4}$$

where $\mathbb{P}\left[.\right]$ denotes the probability and $B \in \mathbb{R}$ is a constant.

The computation difficulty of these two quantities varies with the type of stochastic. In the following two sub-sections, the computation or approximation methods of these two stochastic quantities will be discussed respectively.

3-1-2 Computation or Approximation Method of Expectation

The most common method for calculating the expectation of the max-affine expressions is numerical integration, which is in general exact but complex and time-consuming. The computation time is severely affected by the types of stochastic. This method is usually used for stochastic systems with uncomplicated probability density functions, such as uniform distribution, and sometimes normal distribution, but the computation time is still quite high.

The second method is analytic integration based on piecewise affine probability density functions [10] [16]. The method can be used for stochastic systems with uniform distribution or normal distribution probability density function. However, even though the method results in an analytic solution, its complexity still increases drastically as the number of stochastic variables or the order of the system increase.

The third method is variability expansion based on Taylor series [17] [18]. Since this method is an analytical method and does not resort to simulation, in principle, the higher-order moments of the performance characteristics of stochastic systems can be calculated. However, the level of the complexity of the main problem remains too high.

Another approximated method is Monte Carlo (MC) simulation [19] [20]. The method consists of approximating the expectation of the max-affine function on the basis of a large number of random samples. The method has a relatively good approximation effect but the computation time is still a little bit long.

Finally, another effective approximation approach is proposed by [18]. The approximate value of the expectation is based on the raw moments of a random variable, which results in a much lower computational complexity and a much lower computation time, while still ensuring a good performance. The details of this approach are as follows:

Consider the following expression for the expectation of the max-affine function:

$$\mathbb{E}\left[\max_{j=1,\cdots,n}\left(\beta_j+\gamma_j^T e\right)\right] \tag{3-5}$$

where $\beta_j \in \mathbb{R}$ and $\gamma_j \in \mathbb{R}^{n_e}$. For simplicity, define $y_j = \beta_j + \gamma_j^T e$.

According to the norm properties, the following proposition [18] is given:

Proposition 1.

$$\begin{aligned}
\mathbb{E}\left[\max\left(y_1,\cdots,y_n\right)\right] &\leq \mathbb{E}\left[\max\left(|y_1|,\cdots,|y_n|\right)\right] \\
\mathbb{E}\left[\max\left(|y_1|,\cdots,|y_n|\right)\right] &\leq \mathbb{E}\left[\left(|y_1|^p,\cdots,|y_n|^p\right)^{1/p}\right] \\
\mathbb{E}\left[\left(|y_1|^p,\cdots,|y_n|^p\right)^{1/p}\right] &\leq \left(\sum_{j=1}^{n}\mathbb{E}\left[y_j\right]^p\right)^{1/p}
\end{aligned} \tag{3-6}$$

In order to reduce the error, define an offset L such that $x_j = y_j - L$ is almost positive, and then the expectation of the max-affine function given in Equation (3-5) can be converted to:

$$\begin{aligned}
\mathbb{E}\left[\max_{j=1,\cdots,n}\left(\beta_j+\gamma_j^T e\right)\right] &= \mathbb{E}\left[\max\left(y_1,\cdots,y_n\right)\right] \\
&= \mathbb{E}\left[\max\left(y_1-L,\cdots,y_n-L\right)\right]+L \\
&= \mathbb{E}\left[\max\left(x_1,\cdots,x_n\right)\right]+L \\
&\leq \mathbb{E}\left[\max\left(|x_1|,\cdots,|x_n|\right)\right]+L \\
&\leq \mathbb{E}\left[\left(|x_1|^p,\cdots,|x_n|^p\right)^{1/p}\right]+L \\
&\leq \left(\sum_{j=1}^{n}\mathbb{E}\left[|x_j|^p\right]\right)^{1/p}+L
\end{aligned} \tag{3-7}$$

Assuming p is an even integer greater than or equal to 2, then $\mathbb{E}\left[x_p\right] = \mathbb{E}\left[|x|^p\right]$.

Therefore, based on Equation (3-7) and Jensen's Inequality, the upper bound of $\mathbb{E}\Big[\max_{j=1,\cdots,n}\Big(\beta_j+\gamma_j^T e\Big)\Big]$ can be approximated as follows:

$$\mathfrak{U}\left(\mathbb{E}\left[\max_{j=1,\cdots,n}\left(\beta_j+\gamma_j^T e\right)\right]\right)\leq\left(\sum_{j=1}^{n}\mathbb{E}\left[\left(\beta_j+\gamma_j^T e-L\right)^p\right]\right)^{1/p}+L \tag{3-8}$$

for p a positive even integer and for independent random variables y_j, $j=1,\cdots,n$.

Remark. 1 The offset L is used to reduce the error in the first step of the Equation (3-6). If the variables are all positive, the first inequality changes to equal. If y_j is drawn from a distribution with a finite domain (such as the uniform distribution), then L can be defined such that $L\leq y_j$ for $j=1,\cdots,n$ and the first inequality hence becomes equality. However, if y_j is drawn from a distribution with an infinite or a left semi-infinite domain such as the normal distribution, the inequality will never be reduced to equal, and the error will always exist. The error can only be decreased by defining L such that it is less than or equal to "almost" all y_j for $j=1,\cdots,n$.

Remark. 2 From Remark. 1, the offset L should be selected as large as possible to reduce the error in the first inequality of Equation (3-7). However, in the second inequity, where the infinite norm is replaced by the p-norm, the error is reduced by choosing a small value of L. Therefore, we should weigh the trade-offs when choosing L. Similarly, in the second inequity, p should be chosen as large as possible to reduce the error; while in the last step, a smaller value of p will result in less error. Thus, the choice of L and p values should depend on the properties of a given system.

3-1-3 Computation or Approximation Method of Chance Constraints

Similar to the computation or the approximation of the expectation of max-affine function, the following approaches are proposed to compute or approximate the chance constraints of max-affine functions.

The first method is exact computation (numerical integration). As discussed in the previous sub-section, the method has the advantage of high accuracy but the computational burden is extremely high. The computation time is affected by the types of stochastic.

The second method is analytic expression [10] [16]. Even though the method results in an analytic solution, as the number of stochastic variables or the order of the system increases, the amount of offline calculation is still high. However, online computations using analytic solutions (such as using the Model Predictive Control (MPC) controller built with the analytic solution) are quite fast.

The third method is MC simulation, and its principle is the same as that mentioned in the previous sub-section.

Another method is pointwise approximation [21], the chance constraints are approximated and substituted with a finite number of pointwise constraints at independently generated scenarios of the uncertainties.

Finally, two other approximated approaches are proposed by [20] and [22], in which the chance constraints are converted into reduced forms based on some probabilistic inequalities. The details of these two methods are given as follows:

Consider the following constraints:

$$G\,\widetilde{u}(k) + H\,\widetilde{y}(k) \leq h(k) \tag{3-9}$$

with $\widetilde{u}(k) = \begin{bmatrix} u^T(k) & \cdots & u^T(k+N_p-1) \end{bmatrix}^T$ and $\widetilde{y}(k) = \begin{bmatrix} y^T(k) & \cdots & y^T(k+N_p-1) \end{bmatrix}^T$, G and H being constant matrix, N_p being a constant representing the prediction horizon of MPC process (given in Section 4-1).

The components of $\widetilde{y}(k)$ are assumed to be max-affine functions of $\widetilde{w}(k)$ and $\widetilde{u}(k)$, where $\widetilde{w}(k) = \begin{bmatrix} w^T(k) & \cdots & w^T(k+N_p-1) \end{bmatrix}^T$ is uncertain and only its distribution is supposed to be known, which means that the constraints given in Equation (3-9) are also random.

To reformulate the random constraints (3-9), they are required to satisfy for sufficiently many realizations of $\widetilde{w}(k)$, that is the following chance constraints:

$$\Pr\left\{G\,\widetilde{u}(k) + H\,\widetilde{y}(k) \leq h(k)\right\} \geq 1-\epsilon \tag{3-10}$$

where $\epsilon \in (0,1)$ is the probability of possible violation of Equation (3-9).

Further transform it, we have:

$$\begin{aligned} &\Pr\left\{G\,\widetilde{u}(k) + H\,\widetilde{y}(k) \leq h(k)\right\} \\ =\ &\Pr\left\{G\,\widetilde{u}(k) + H\,\widetilde{y}(k) - h(k) \leq 0\right\} \\ =\ &\Pr\left\{\max_{i=1,\cdots,c}\left(G\,\widetilde{u}(k) + H\,\widetilde{y}(k) - h(k)\right)_i \leq 0\right\} \end{aligned}$$

Since the vector $G\,\widetilde{u}(k) + H\,\widetilde{y}(k) - h(k)$ only contains affine operations on the components of $\widetilde{u}(k)$ and $\widetilde{y}(k)$, and the components of $\widetilde{y}(k)$ are max-affine functions of $\widetilde{w}(k)$ and $\widetilde{u}(k)$, assuming that H has nonnegative entries, then each component of $G\,\widetilde{u}(k) + H\,\widetilde{y}(k) - h(k)$ is also a max-affine function of $\widetilde{w}(k)$ and $\widetilde{u}(k)$. Let $m = \sum_{i=1}^{c} n_i$ where n_i is the number of affine expressions appearing in the maximization for the ith component of $G\,\widetilde{u}(k) + H\,\widetilde{y}(k) - h(k)$, we have:

$$\begin{aligned} &\Pr\left\{\max_{i=1,\cdots,c}\left(G\,\widetilde{u}(k) + H\,\widetilde{y}(k) - h(k)\right)_i \leq 0\right\} \\ =\ &\Pr\left\{\max_{i=1,\cdots,m}\left(z_i(k)\right) \leq 0\right\} \end{aligned} \tag{3-11}$$

with $z(k) = \Lambda \begin{bmatrix} w(k-1) & \widetilde{w}(k) \end{bmatrix}^T + \Gamma\widetilde{u}(k) + \Xi(k)$ for some appropriately sized matrices and vectors Λ,Γ, and Ξ.

Thus, the chance constraint (3-10) is equivalent to

$$\Pr\left\{\max_{i=1,\cdots,m}\left(z_i(k)\right) \leq 0\right\} \geq 1-\epsilon \tag{3-12}$$

Although the probability in (3-12) can be computed by numerical integration method, the amount of calculation remains quite huge, thus converting it into the following two forms that can be effectively evaluated:

a) **Based on Boole's Inequality**

Proposition 2. If $\sum_{i=1}^{m} \Pr\{z_i(k) > 0\} \leq \epsilon$, then $\Pr\left\{\max_{i=1,\cdots,m}(z_i(k)) \leq 0\right\} \geq 1-\epsilon$

b) **Based on Chebyshev's Inequality**

Assuming the components of $\widetilde{w}(k)$ are independent and identically distributed random variables. Let μ_w and Σ_w be the mean vector and covariance matrix of $\widetilde{w}(k)$. Define

$$\mu_z(k) = \Lambda\mu_w + \Gamma\widetilde{u}(k) + \Xi(k) \tag{3-13}$$

$$\Sigma_z = \Lambda\Sigma_w\Lambda^T \tag{3-14}$$

According to the properties of the mean and covariance matrix of random vectors, $\mu_z(k)$ and Σ_z are the mean vector and covariance matrix of $z(k)$.

Proposition 3. If Σ_z is a positive definite matrix, let $\lambda_{\min}\left(\Sigma_z^{-1}\right) > 0$ be the smallest eigenvalue of the matrix Σ_z^{-1}. Let $\bar{\mu}_z(k) = \max_{i=1,\cdots,m}\mu_{z,i}(k)$. If $\bar{\mu}_z(k) < 0$ and

$$\frac{m}{\bar{\mu}_z^2(k)\lambda_{\min}\left(\Sigma_z^{-1}\right)} \leq \epsilon$$

then

$$\Pr\left\{\max_{i=1,\cdots,m}(z_i(k)) \leq 0\right\} \geq 1-\epsilon$$

The proof of two propositions can be found from [20] and [22].

3-2 Switching Max-Plus-Linear Systems with Uncertain Switching

3-2-1 Introduction

Switching max-plus-linear systems with switching uncertainties (also named as Randomly Switching Max-Plus-Linear (RSMPL) systems) are a type of SMPL systems in which the order of synchronization of the event steps may vary randomly, or cannot be determined a priori [13]. In other words, compared with the deterministic switching max-plus-linear systems discussed in the previous section, given the current mode $\ell(k)$, it may still be impossible to determine the next mode $\ell(k+1)$ of RSMPL systems. The mode switching of RSMPL depends on a stochastic sequence, i.e. the mode switching variable $\ell(k)$ is a stochastic process. Based on the types of the stochastic, RSMPL systems can be classified into two categories:

Consider the following SMPL system:

$$x(k) = A^{(\ell(k))} \otimes x(k-1) \oplus B^{(\ell(k))} \otimes u(k) \tag{3-15}$$

$$y(k) = C^{(\ell(k))} \otimes x(k) \tag{3-16}$$

If the probability of switching is unknown and cannot be determined, which means the switching is totally random and its probability function of stochastic switching

$$P_s(\ell(k-1), \ell(k)) = P\left[L = \ell(k) \mid \ell(k-1), x(k-1), u(k), v(k)\right]$$

is not regular (i.e uniform or piecewise affine on polyhedral partition), then what can be derived is that the switching probability is bounded satisfying:

$$0 \leq P\left[L = \ell(k) \mid \ell(k-1), x(k-1), u(k), v(k)\right] \leq 1 \tag{3-17}$$

$$\sum_{\ell(k)=1}^{n_L} P\left[L = \ell(k) \mid \ell(k-1), x(k-1), u(k), v(k)\right] = 1 \tag{3-18}$$

If the probability of switching is deterministic, which means the probability function of stochastic switching

$$P_s(\ell(k-1), \ell(k)) = P\left[L = \ell(k) \mid \ell(k-1), x(k-1), u(k), v(k)\right]$$

is uniform or piecewise affine on polyhedral partition of the space of the variables $\ell(k-1), x(k-1), u(k), v(k)$ [13], then some approximation methods can be applied to reduce the computational burden, the details will be given in the next sub-section.

3-2-2 Computation or Approximation Method of Switching Probability

In this sub-section, we will focus on the RSMPL systems with uniform or piecewise affine probability functions and discuss how to compute or approximate the switching probability.

The most common method is the direct method, which calculates the probability for each switching sequence respectively and treats each switching sequence as a deterministic SMPL system, that is:

$$P\left(\widetilde{\ell}(k) \mid \ell(k-1)\right) = P_s(\ell(k-1), \ell(k)) \cdot P_s(\ell(k), \ell(k+1)) \cdots P_s(\ell(k+N_p-2), \ell(k+N_p-1)) \tag{3-19}$$

where P_s denotes the switching probability and N_p represents prediction horizon for the MPC process.

The result of this algorithm is accurate. However, it has the drawback that the number of linear constraints and the number of optimization variables increase rapidly with the increase of the prediction horizon and the number of modes in the system, which is not suitable for systems with a large number of modes or large prediction horizons.

Another effective method to solve the problem is the scenario-based algorithm [23], that is, instead of computing all possible realizations of $\widetilde{\ell}(k)$, only the most probable mode switching vectors will be considered. This approach is not quite effective when a uniform switching probability is given. The algorithm mainly consists of the following two steps:

- **Step 1:** Select n_{max} paths of length $N_p - 1$ in the search tree $\mathcal{T}$ using a random selection or a greedy approach. This results in a candidate set of realizations $\mathcal{L}_{N_p}^{\text{red}} = \left\{\bar{\ell}^1, \cdots, \bar{\ell}^{n_{max}}\right\}$. Define

$$\pi^{red} = \min_{\bar{\ell} \in \mathcal{L}_{N_p}^{\text{red}}} P\left(\bar{\ell} \mid \ell(k)\right) \tag{3-20}$$

- **Step 2:** Apply a breadth-first search in $\mathcal{T}$, cutting a subtree originating in a node n if $P(n) \leq \pi^{\text{red}}$, and updating $\mathcal{L}_{N_p}^{\text{red}}$ whenever a leaf node n_{leaf} is encountered such that $P(n_{leaf}) > \pi^{\text{red}}$; in the latter case, the node $\bar{\ell}$ in $\mathcal{L}_{N_p}^{\text{red}}$ with the lowest probability is removed and replaced by $\bar{\ell}^{real}(n_{leaf})$ and π^{red} is updated accordingly.

Remark. In Step 2, the idea of cutting off the subtree originating in the node n if $P(n) \leq \pi^{\text{red}}$ is that any leaf node from a subtree cannot have a probability higher than that of the realizations that are already in $\mathcal{L}_{N_p}^{\text{red}}$; so there is no need to consider and explore the subtree any further.

One of the difficulties of this method is that currently there is no algorithm that can return the n_{max} shortest path with a fixed number of nodes or edges. To solve this problem, the basic idea of the algorithm is to apply a dedicated algorithm based on a breadth-first search in combination with an approach to cut away parts of the search tree in order to guarantee a fixed-mode-number of n_{max} shortest path. Compared with the previous exact method, the computational burden is much smaller and the accuracy is also quite high.

3-3 Summary of Parametric Uncertainty and Switching Uncertainty

In this section, the contents of the previous two sections are summarized in Table 3-1 and 3-2.

Table 3-1: Stochastic SMPL Systems with Parametric Uncertainty

	Type of Stochastic		Stochastic Quantities	
Methods	Uniform Distribution	Normal Distribution	Expectation	Chance Constraints
Numerical Integration	Accurate Hard	Accurate Very Hard	×	×
Analytic Expression (Piecewise Polynomial Expression)	Exact {Hard (offline) Easy (online)	Exact {Hard (offline) Easy (online)	×	×
Analytic Integration (Piecewise Affine PDF)	Very Good Hard	Very Good Hard	×	×
Variability Expansion	Good Hard	Good Very Hard	×	
MC Simulation	Good Medium	Good Medium	×	×
Upper Bound (Raw Moments of Random Variable)	Good Easy	Very Good Easy	×	
Pointwise Approximation	Good Medium	Bad Hard		×
Boole's Inequality	Good Easy	Bad Hard		×
Chebyshev's Inequality	Good Easy	Bad Hard		×

* The first row in the table refers to the quality of the approximation.
** The second row in the table represents the difficulty of the computation.

Table 3-2: Stochastic SMPL Systems with Switching Uncertainty

	Type of Stochastic	
Methods	Random	Piecewise Affine
Exact Method	Accurate/Very Hard	Accurate/Very Hard
Scenario-Based Algorithm	N.A.	Very Good/Easy

* The contents in the table refer to approximation quality/calculation difficulty.

3-4 Switching Max-Plus-Linear Systems with Two Types of Uncertainties

In [15], a unified setting for stochastic SMPL systems to include both types of uncertainties has been proposed: enhance the mode switching uncertainty first, and then include the parametric

uncertainty for each mode independently, which results in a model of the form

$$x(k) = A^{(\ell(k))}\left(e(k)\right) \otimes x(k-1) \oplus B^{(\ell(k))}\left(e(k)\right) \otimes u(k) \tag{3-21}$$

$$y(k) = C^{(\ell(k))}\left(e(k)\right) \otimes x(k) \tag{3-22}$$

where $A^{(\ell(k))}\left(e(k)\right)$, $B^{(\ell(k))}\left(e(k)\right)$ and $C^{(\ell(k))}\left(e(k)\right)$ are the system matrices that correspond to mode $\ell(k)$ and where the stochastic random vector $e(k)$ represents the stochastic parametric uncertainty at the k-th event step.

Compared with Section 3-1 and 3-2, the most extensive computation problem of the system is that the model contains both two types of uncertainty, which are continuous random variable $\widetilde{e}(k)$ represents the parametric uncertainties and discrete random variable $\widetilde{\ell}(k)$. Therefore, the Joint Density Function (JDF) need to be determined.

Let $\mathcal{L}$ be the (discrete) sample space of all the mode switching sequences $\widetilde{\ell}(k)$, and $\mathcal{E}$ be the (continuous) sample space of the parametric uncertainty $\widetilde{e}(k)$. Assume that the probability density function of the uncertainty $\widetilde{e}(k)$ does not depend on the discrete mode sequence $\widetilde{\ell}(k)$, and the continuous probability density function of $\widetilde{e}(k)$ is given by $f_{\mathcal{E}}\left(\widetilde{e}(k)\right)$, then the JDF is given as:

$$f_{\mathcal{L},\mathcal{E}}\left(\widetilde{\ell}(k), \widetilde{e}(k)\right) = f_{\mathcal{E}}\left(\widetilde{e}(k)\right) \widetilde{P}\left[L = \widetilde{\ell}(k) \mid E = \widetilde{e}(k)\right] \tag{3-23}$$

where $\widetilde{P}\left[L = \widetilde{\ell}(k) \mid E = \widetilde{e}(k)\right]$ denotes the probability that we have mode switching sequence $\widetilde{\ell}(k)$, given the parametric uncertainty $\widetilde{e}(k)$, which satisfies

$$\begin{aligned}\widetilde{P}\left[L = \widetilde{\ell}(k) \mid E = \widetilde{e}(k)\right] &= \widetilde{P}\left[L = \ell(k) \mid \ell(k-1), x(k-1), \widetilde{u}(k), \widetilde{v}(k), \widetilde{e}(k)\right] \\ &= \prod_{j=0}^{N_p-1} P\left[L = \ell(k+j) \mid \ell(k+j-1),\right. \\ &\quad \left. x(k+j-1), \widetilde{u}(k+j), \widetilde{v}(k+j), \widetilde{e}(k+j)\right]\end{aligned}$$

When both the continuous probability density function $f_{\mathcal{E}}\left(\widetilde{e}(k)\right)$ and the mode switching probability $\widetilde{P}\left[L = \widetilde{\ell}(k) \mid E = \widetilde{e}(k)\right]$ are known, the expression for the expectation of the cost function $J(k)$ in MPC process can be defined as:

$$\mathbb{E}\left[J(k)\right] = \sum_{\widetilde{\ell}(k) \in \mathcal{L}} \left[\int_{\mathcal{E}} J\left(\widetilde{\ell}(k), \widetilde{e}(k)\right) f_{\mathcal{E}}\left(\widetilde{e}(k)\right) \widetilde{P}\left[L = \widetilde{\ell}(k) \mid E = \widetilde{e}(k)\right] d\widetilde{e}(k)\right] \tag{3-24}$$

If the mode switching uncertainty $\widetilde{\ell}(k)$ does not depend on the parametric uncertainty $\widetilde{e}(k)$, that is $\widetilde{P}\left[L = \widetilde{\ell}(k) \mid E = \widetilde{e}(k)\right] = \widetilde{P}\left[L = \widetilde{\ell}(k)\right]$. Represent the set $\mathcal{L}$ of all possible consecutive mode switching vectors as $\mathcal{L} = \left\{\widetilde{\ell}^1, \widetilde{\ell}^2, \cdots, \widetilde{\ell}^m\right\}$ for $M = (n_L)^{N_P}$, then

$$\begin{aligned}\mathbb{E}\left[J(k)\right] &= \mathbb{E}_{\widetilde{e},\widetilde{\ell}}[J(k)] \\ &= \sum_{\widetilde{\ell}(k) \in \mathcal{L}} \left[\widetilde{P}\left[L = \widetilde{\ell}(k)\right] \mathbb{E}_{\widetilde{e}}\left[J\left(\widetilde{\ell}(k), \widetilde{e}(k)\right)\right]\right] \\ &= \sum_{m=1}^{M} \left[\widetilde{P}\left[L = \widetilde{\ell}^m(k)\right] \mathbb{E}_{\widetilde{e}}\left[J\left(\widetilde{\ell}^m(k), \widetilde{e}(k)\right)\right]\right]\end{aligned} \tag{3-25}$$

where $\mathbb{E}_{\widetilde{e}}\left[J\left(\widetilde{\ell}^m(k), \widetilde{e}(k)\right)\right]$ is the expectation of the cost function after substitution of a given mode switching sequence $\widetilde{\ell}^m(k) \in \mathcal{L}$.

It can be observed that the problem has been decomposed into two sub-problems, involving respectively an expected value computation over a discrete stochastic variable and an expected value computation over a continuous stochastic variable. Therefore, we can deal with the mode switching uncertainty first, and then compute the expectation of the parametric uncertainty only. On the other hand, the approximation approaches discussed in the previous sections can still be applied. This significantly simplifies the computation of the expectation of the cost function.

If the mode switching uncertainty $\widetilde{\ell}(k)$ depends on the parametric uncertainty $\widetilde{e}(k)$, the computation will be quite hard. An analytical approach for this case can be found from [15].

In Chapter 4, MPC framework will be further applied to this structure.

Chapter 4

General Model Predictive Control Problem

Model Predictive Control (MPC) is an online model-based control approach that relies on a dynamic model of the process and that is capable of handling constraints on both inputs or outputs in a systematic way. It has the advantages of high efficiency and computationally tractable [18]. In [24], the MPC framework has been extended to discrete-event systems and in particular to time-invariant max-plus-linear models. Max-plus-linear MPC results in a linear programming problem, which can be solved efficiently. In this section, we will apply the MPC algorithm to the system discussed in Section 3-4 referring to [15].

4-1 Cost Function

Similar to the conventional MPC, a cost function J that reflects the input and output cost (J_{in} and J_{out} respectively) in the event period $[k, k+N_p-1]$ need to be designed, which in general should have the form

$$J(k) = J_{out}(k) + \lambda J_{in}(k) \tag{4-1}$$

where N_p is the prediction horizon and λ is a weighting parameter that makes a trade-off between the output cost function and the input cost function.

Referring to the result from [15], J_{out} and J_{in} can be chosen as

$$J_{out} = \sum_{j=0}^{N_p-1} \sum_{l=1}^{n_y} \mathbb{E}\left[\eta_i(k+j)\right] \tag{4-2}$$

$$J_{in} = -\sum_{j=0}^{N_p-1} \sum_{l=1}^{n_u} u_l(k+j) \tag{4-3}$$

where n_y and n_u are equal to the number of outputs and inputs respectively, $\mathbb{E}[\cdot]$ denotes the expectation of some random variables and $\eta_i = \max(y_i(k) - r_i(k), 0)$ denotes the i-th tardiness error. $r_i(k)$ is a vector of the reference (due date) signal.

The input cost function maximizes the input instants while the output cost function penalizes the late deliveries.

4-2 Prediction Model

As mentioned previously, Max-Plus-Linear (MPL) systems are different from the conventional time-driven system in the sense that k represents the event number rather than a specific time. So far, we have assumed that $x(k)$ is available when optimizing over the future control sequence. However, not all components of $x(k)$ are known at the same time instant since $x(k)$ contains the time instants at which the internal activities or processes of the system start for the k-th time [7]. Let t be the present time instant and define the actual current cycle k as follows:

$$k = \max_{\kappa} \{\kappa \mid x_i(\kappa - 1) \leq t \quad \forall i \in \{1, 2, \cdots, n\}\} \tag{4-4}$$

This means $x(\kappa) \leq t$ for $\kappa \leq k-1$, so these states are all known at time t. Note that parts of the states $x(\kappa) \leq t$ for $\kappa \geq k$ may also be known [2]. Define the vectors

$$\widetilde{x}(k) = \begin{bmatrix} x(k) \\ \vdots \\ x(k+N_p-1) \end{bmatrix} \widetilde{u}(k) = \begin{bmatrix} u(k) \\ \vdots \\ u(k+N_p-1) \end{bmatrix} \quad \widetilde{y}(k) = \begin{bmatrix} y(k) \\ \vdots \\ y(k+N_p-1) \end{bmatrix}$$

$$\widetilde{e}(k) = \begin{bmatrix} e(k) \\ \vdots \\ e(k+N_p-1) \end{bmatrix} \quad \widetilde{\ell}(k) = \begin{bmatrix} \ell(k) \\ \vdots \\ \ell(k+N_p-1) \end{bmatrix}$$

and the matrices

$$\widetilde{C}\left(\widetilde{\ell}(k), \widetilde{e}(k)\right) = \begin{bmatrix} \widetilde{C}_1\left(\widetilde{\ell}(k), \widetilde{e}(k)\right) \\ \vdots \\ \widetilde{C}_{N_p}\left(\widetilde{\ell}(k), \widetilde{e}(k)\right) \end{bmatrix} \quad \widetilde{D}\left(\widetilde{\ell}(k), \widetilde{e}(k)\right) = \begin{bmatrix} \widetilde{D}_{1,1}\left(\widetilde{\ell}(k), \widetilde{e}(k)\right) & \cdots & \varepsilon \\ \vdots & \ddots & \vdots \\ \widetilde{D}_{N_p,1}\left(\widetilde{\ell}(k), \widetilde{e}(k)\right) & \cdots & \widetilde{D}_{N_p,N_p}\left(\widetilde{\ell}(k), \widetilde{e}(k)\right) \end{bmatrix}$$

satisfying

$$\widetilde{y}(k) = \widetilde{C}\left(\widetilde{\ell}(k), \widetilde{e}(k)\right) \otimes x(k-1) \oplus \widetilde{D}\left(\widetilde{\ell}(k), \widetilde{e}(k)\right) \otimes \widetilde{u}(k) \tag{4-5}$$

where

$$\widetilde{C}_m\left(\widetilde{\ell}(k), \widetilde{e}(k)\right) = C_{\ell(k+m-1)}\left(e(k+m-1)\right) \otimes A_{\ell(k+m-1)}\left(e(k+m-1)\right) \otimes \cdots \otimes A_{\ell(k)}\left(e(k)\right)$$

and

$$\widetilde{D}_{m,n}\left(\widetilde{\ell}(k), \widetilde{e}(k)\right) = \begin{cases} C_{\ell(k+m-1)}\left(e(k+m-1)\right) \otimes A_{\ell(k+m-1)}\left(e(k+m-1)\right) \otimes \cdots & \\ \otimes A_{\ell(k+n)}\left(e(k+n)\right) \otimes B_{\ell(k+n-1))}\left(e(k+n-1)\right) & \text{if } m > n \\ C_{\ell(k+m-1)}\left(e(k+m-1)\right) \otimes B_{\ell(k+m-1))}\left(e(k+m-1)\right) & \text{if } m = n \\ \varepsilon & \text{if } m < n \end{cases}$$

4-3 Constraints

One of the advantages of MPC is that constraints are easy to apply to the model. The first constraint that needs to be considered is for the input signal $u(k)$. Since it corresponds to consecutive event occurrence times, the following constraint for $j = 0, \cdots, N_p - 1$ should be applied:

$$\Delta u(k+j) = u(k+j) - u(k+j-1) \geq 0 \tag{4-6}$$

which means that the start of the $(k+j)$-th event will always be later than the start of the $(k+j-1)$-th event.

Moreover, to reduce the number of decision variables and the computational complexity, control horizon N_c ($\leq N_p$) is introduced, which should satisfy

$$\Delta^2 u(k+m) = \Delta u(k+m) - \Delta u(k+m-1) = 0 \tag{4-7}$$

for $m = N_c, \cdots, N_p - 1$. This means after event step $k + N_c - 1$, the increments of $\Delta u(k)$ are assumed to become constant.

Finally, possible additional linear constraints on the inputs and the outputs can be applied, which should satisfy

$$A_c(k)\widetilde{u}(k) + B_c(k)\mathbb{E}\left[\widetilde{y}(k)\right] \leq C_c(k) \tag{4-8}$$

4-4 Optimization Problem

In summary of this section, the final optimization problem can be defined as:

$$\min_{\widetilde{u}(k)} \left(\sum_{j=0}^{N_p-1} \sum_{l=1}^{n_y} \mathbb{E}\left[\max\left(\widetilde{y}_i(k) - \widetilde{r}_i(k), 0\right)\right] - \lambda \sum_{j=0}^{N_p-1} \sum_{l=1}^{n_u} \widetilde{u}_l(k) \right)$$

subject to

$$\begin{aligned}
&\widetilde{y}(k) = \widetilde{C}\left(\widetilde{\ell}(k), \widetilde{e}(k)\right) \otimes x(k-1) \oplus \widetilde{D}\left(\widetilde{\ell}(k), \widetilde{e}(k)\right) \otimes \widetilde{u}(k) \\
&\Delta u(k+j) \geq 0 && \text{for } j = 0, \cdots, N_c - 1 \\
&\Delta^2 u(k+m) = 0 && \text{for } m = N_c, \cdots, N_p - 1 \\
&A_c(k)\widetilde{u}(k) + B_c(k)\mathbb{E}\left[\widetilde{y}(k)\right] \leq C_c(k)
\end{aligned}$$

The most difficult part of the optimization process is the expectation part with two stochastic variables $e(k)$ and $l(k)$. Based on whether these two random variables are dependent, the expectation of the cost function can be simplified into different forms, which can be found in paper [15].

Chapter 5

Printer System Modelling

As mentioned in the previous chapters, when different types of paper (length, thickness, printing method, etc.) are involved, the current scheduling algorithms commonly used in printers will not be optimal, because the controller cannot take the paper changes in subsequent print queues into consideration in advance. In order to achieve optimal scheduling, the max-plus algorithm can be applied to the printer. In this chapter, we will model the printer system as a Switching Max-Plus-Linear (SMPL) system.

5-1 Printer Model Introduction

In principle, the paper transported in the printer corresponds to the movement in the 3-dimensional space. If we assume that the paper does not deviate from its intended path and has no skew or transverse buckling, then the sheet transportation movement can be simplified to move along a straight line [1]. The schematic of the paper path in a typical printer is given in Figure 5-1.

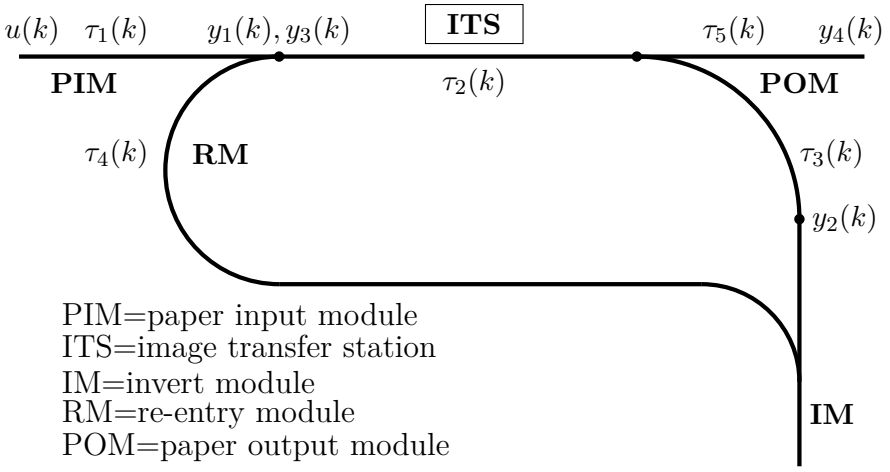
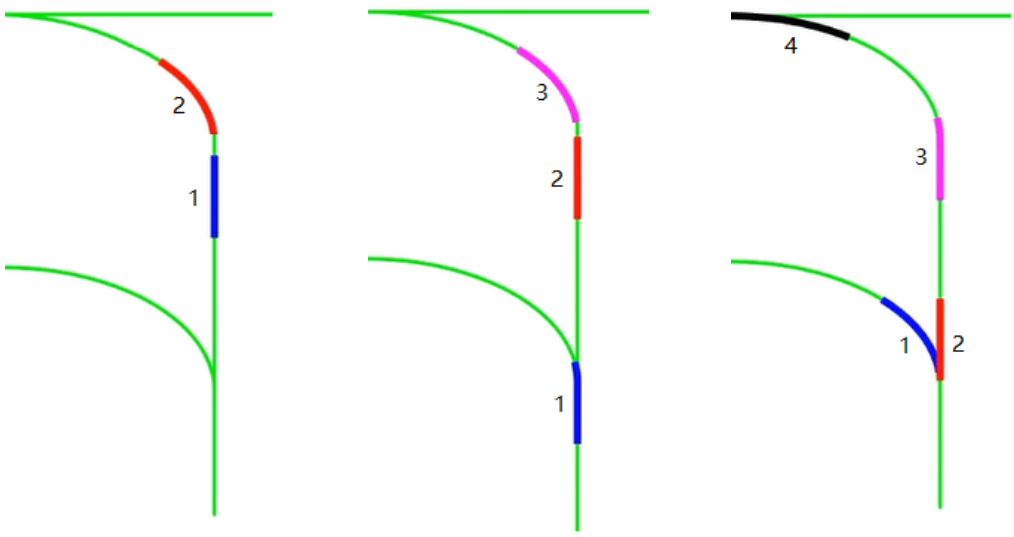


Figure 5-1: A Schematic of the Paper Path in a Typical Printer [2]

The Paper Input Module (PIM) feeds the sheets to the Image Transfer Station (ITS), where the image is printed on the paper. After that, the paper in simplex mode runs directly to the Paper Output Module (POM), and the paper in duplex mode runs via the Invert Module (IM) and Re-entry Module (RM), and then re-enters the image transfer station for backside printing [1] [2]. When the printing sequence in the duplex mode is about to end, that is, when there is no new paper input, the printer will enter the idle mode to ensure that the printer can continue to print back on the paper in the RM.

One thing to note is that in IM, the direction of paper movement is reversed. Since this process requires the paper to decelerate to 0 and then accelerate in the reverse direction, the subsequent paper should keep a certain following distance from the previous paper in this process. Therefore, the sheet is designed to decelerate first and then accelerate after entering the IM. The specific processes are shown in Figure 5-2:



(a) Sheets Approach IM (b) Sheets Enter IM (c) Sheets Leave IM

Figure 5-2: A Schematic of Inverse Module in the Printer

When the first sheet approaches IM, its speed will decrease first, and the distance between the first and second sheets will decrease. Next, the first sheet starts to accelerate into the IM and the second sheet starts to decelerate. This will cause the distance between the first two sheets to increase and the distance between the second and third sheets to decrease. When the second sheet starts to accelerate into the IM, the first sheet will just leave the IM, which can avoid overlap in the reverse process.

In general, we will use the position-time profile to describe the sheet movement in the printer, which is a one-dimensional path that represents the paper path from the printer input to the output [1]. As an example, the position-time profile for a printer working in duplex mode is given in Figure 5-3.

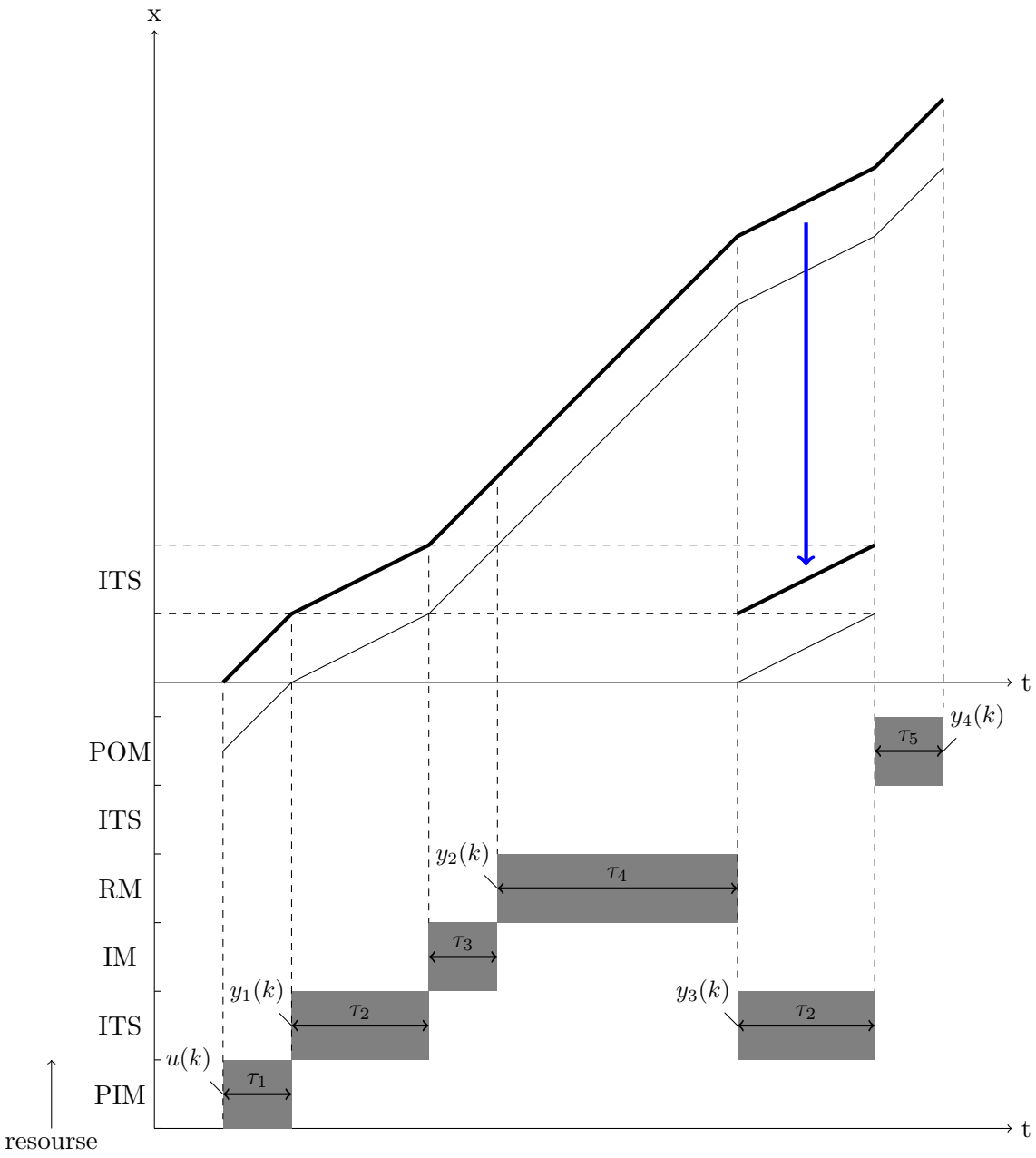


Figure 5-3: The Position Time Profile of a General Printer

In the bottom part of the figure, the position of the block represents the time when the event starts, and the length of the block represents the processing time of the event. It can be seen that the processing time of other parts is shorter than that of RM, which is mainly because the length that the sheet needs to move in RM is much longer than that of other parts. It can also be found from the upper part of the figure that the sloop in the section of ITS is smaller than the other part, the main reason is that due to the working principle of ITS, the scrolling speed of the rollers in the module is much lower than that of other parts to ensure the printing quality of the product. Note that the processing time in ITS only depends on

the length and the type of the sheets, and has nothing to do with the pattern to be printed.

Due to the different scrolling speeds of ITS and other modes, the speed of the paper entering the ITS module will slow down and be caught up by the paper behind, which will lead to a paper overlap. In order to avoid this problem, a certain buffer distance is required between every two sheets. The length of the buffer distance is the distance between the tail of one sheet and the head of the next sheet. The buffer distance should not be less than some minimum length, which is around the speed difference between the two modules multiplied by the paper processing time in ITS. In the simulation process, for the convenience of calculation, the length of the virtual paper is used, and its value is set to the length of the actual paper length plus the buffer distance.

Another graph that is used to describe the behaviour of the printer system is the dynamic graph discussed in Section 2-5, which can reflect the mode switching of the system. Figure 5-4 shows an example of a dynamic diagram of the printer's paper flow system.

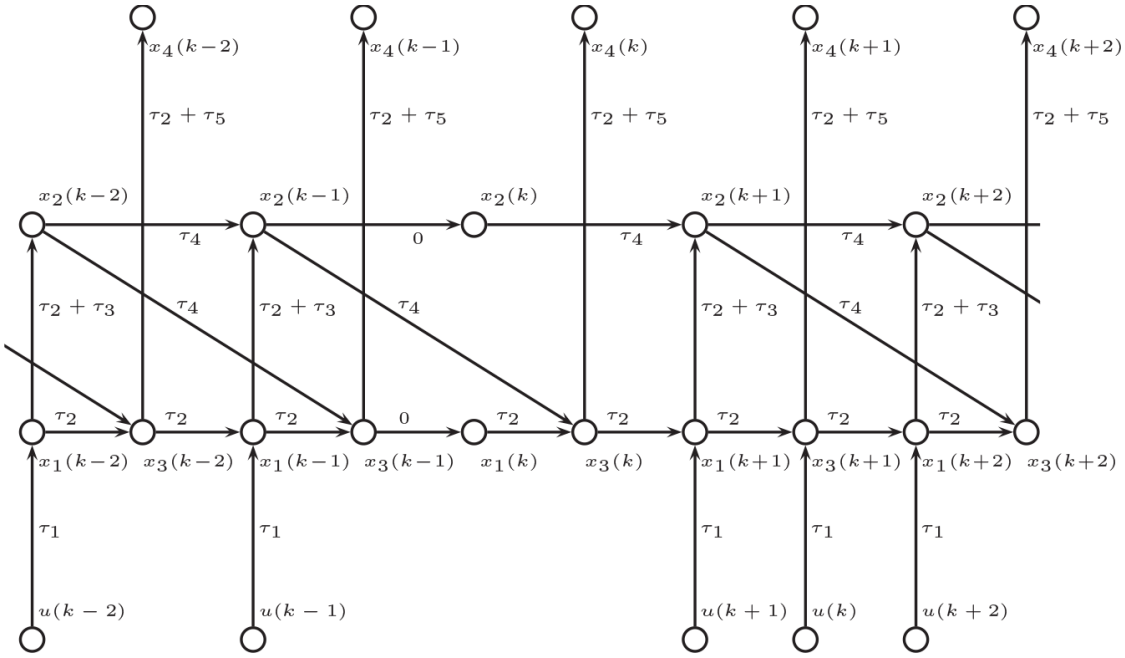


Figure 5-4: Example of a Dynamic Diagram of the Printer's Paper Flow System [2]

In this figure, the printer works in duplex mode in the cycles $k-2$, $k-1$, $k+1$, and $k+2$, but in simplex mode in cycle k.

In the printer, the movement of the paper is driven by rollers. We assume that the rollers in each module except RM and ITS can only roll at full speed and cannot be adjusted. The average rolling speed in ITS depends on the type of paper. For the same type of paper, the average rolling speed is the same. The rollers in RM can scroll at three speeds: 0, normal speed, which is the same speed as in other modes except ITS, and high speed, which is much higher than normal speed. With this setting, the paper speed in each module except ITS and RM is fixed, which means that the time intervals required for the paper to move in these modules are also fixed, regardless of the length of the paper.

5-2 Model Derivation

In the following sub-sections, the basic model of duplex mode, idle mode and simplex mode will be derived.

5-2-1 Model for Duplex Mode

The path of the paper in double-sided printing can be converted to a one-dimensional representation as shown in Figure 5-5. Let $u(k)$ be the time instant that the k-th sheet enters the PIM, $y_1(k)$ the time instant that the k-th sheet enters the ITS, $y_2(k)$ the time instant that the k-th sheet enters the IM, $y_3(k)$ the time instant that the k-th sheet enters the ITS for the second time, $y_4(k)$ the time instant that the k-th sheet leaves the printer. Assume $\tau_1(k)$, $\tau_2(k)$, $\tau_3(k)$, $\tau_4(k)$, $\tau_5(k)$, are the processing time for feeding, printing, handling in the first part of the loop, inverting, handling in the second part of the loop and stacking for the sheet k respectively.

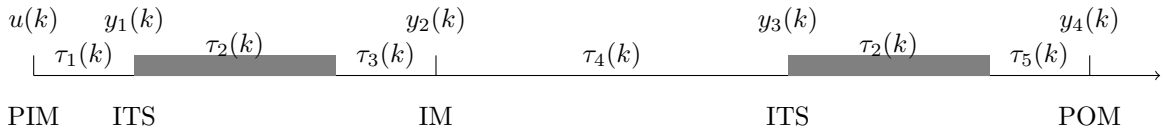


Figure 5-5: One-dimensional Schematic Diagram of Paper Movement for Double-Sided Printing

According to the assumption in Section 5-1, the processing time $\tau_2(k)$ is only determined by the type of paper, and the processing time $\tau_4(k)$ of each sheet of paper can be changed with k, which can make the paper wait for a period of time in RM or accelerate to enter ITS a little earlier. The processing time of other modules (τ_1, τ_3, τ_5) will remain constant.

Then the paper path for the duplex mode can be evaluate as follows:

$$\begin{aligned}
y_1(k) &\geq \max\left(u(k) + \tau_1(k), y_3(k - c_p(k)) + \tau_2(k - c_p(k))\right), \\
y_2(k) &\geq \max\left(y_1(k) + \tau_2(k) + \tau_3(k), y_2(k-1) + \tau_4(k-1)\right), \\
y_3(k) &\geq \max\left(y_1(k + c_p(k) - 1) + \tau_2(k + c_p(k) - 1), y_2(k) + \tau_4(k)\right), \\
y_4(k) &\geq y_3(k) + \tau_2(k) + \tau_5(k),
\end{aligned} \tag{5-1}$$

where $c_p(k)$ is a parameter affected by the printer itself, which represents the number of papers in the printer during the printing cycle k.

Equation (5-1) means that the second time that paper k enters the ITS $(y_3(k))$ is scheduled after the first time that paper $k+c_p(k)-1$ leaves the ITS $(y_1(k+c_p(k)-1)+\tau_2(k+c_p(k)-1))$. In the same way, the first time paper k enters the ITS $(x_1(k))$ will be scheduled after the second time paper $k - c_p(k)$ leaves the ITS $(y_3(k - c_p(k)) + \tau_2(k - c_p(k)))$.

Since $y_3(k)$ depends on the future event, we introduce the state x and a new input $\bar{u}$ as follows:

$$x(k) = \begin{bmatrix} y_1(k) \\ y_2(k) \\ y_3(k - c_p(k) + 1) \\ y_4(k - c_p(k) + 1) \end{bmatrix}, \qquad \bar{u}(k) = \begin{bmatrix} u(k) \\ u(k - c_p(k) + 1) \end{bmatrix}. \tag{5-2}$$

The new set of state equations become:

$$\begin{aligned}
x_1(k) &\geq \max\left(\bar{u}_1(k) + \tau_1(k), x_3(k-1) + \tau_2(k - c_p(k))\right), \\
x_2(k) &\geq \max\left(x_1(k) + \tau_2(k) + \tau_3(k), x_2(k-1) + \tau_4(k-1)\right), \\
x_3(k) &\geq \max\left(x_1(k) + \tau_2(k), x_2(k - c_p(k) + 1) + \tau_4(k - c_p(k) + 1)\right), \\
x_4(k) &\geq x_3(k) + \tau_2(k - c_p(k) + 1) + \tau_5(k - c_p(k) + 1).
\end{aligned} \tag{5-3}$$

Note that in Equation (5-1) and (5-3), an inequality sign is used instead of an equality sign. This is because the starting times of the printer system may also depend on ordering, which can delay the starting times [2]. In the following analysis, we assume that an event will take place as soon as all constraints are satisfied, which means that we now have equality in Equation (5-3) instead of inequality. Then we can write the state equations into the following standard implicit SMPL form:

$$x(k) = A_0^{\text{dup}} \otimes x(k) \oplus A_1^{\text{dup}} \otimes x(k-1) \oplus A_{cp}^{\text{dup}} \otimes x(k - c_p(k) + 1) \oplus B_0^{\text{dup}} \otimes \bar{u}(k), \tag{5-4}$$

with

$$A_0^{\text{dup}} = \begin{bmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \tau_2(k) + \tau_3(k) & \varepsilon & \varepsilon & \varepsilon \\ \tau_2(k) & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \tau_2(k - c_p(k) + 1) + \tau_5(k - c_p(k) + 1) & \varepsilon \end{bmatrix}, \; B_0^{\text{dup}} = \begin{bmatrix} \tau_1(k) & \varepsilon \\ \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \end{bmatrix},$$

$$A_1^{\text{dup}} = \begin{bmatrix} \varepsilon & \varepsilon & \tau_2(k - c_p(k)) & \varepsilon \\ \varepsilon & \tau_4(k-1) & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{bmatrix}, \; A_{cp}^{\text{dup}} = \begin{bmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \tau_4(k - c_p(k) + 1) & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{bmatrix}.$$

5-2-2 Model for Idle Mode

In the duplex mode, when the printing sequence is about to end, there is no new paper input, in other words, the new paper input time u becomes infinite. Due to the max operation in Equation (5-1), the backside printing time y_3 will also become infinite, which means that the back printing of the paper in RM is prevented. To avoid this situation, we set the idle mode, assuming that u is ε when there is no new input, and $y_1(k) = y_1(k-1)$. The paper path in this mode can be evaluated as follows:

$$\begin{aligned}
y_1(k) &= y_1(k-1), \\
y_2(k) &= y_2(k-1), \\
y_3(k) &\geq \max\left(y_3(k-1) + \tau_2(k-1), y_2(k) + \tau_4(k)\right), \\
y_4(k) &\geq y_3(k) + \tau_2(k) + \tau_5(k).
\end{aligned} \tag{5-5}$$

Using the same states and inputs defined in Equation (5-2), we have

$$\begin{aligned}
x_1(k) &= x_1(k-1), \\
x_2(k) &= x_2(k-1), \\
x_3(k) &\geq \max\left(x_3(k-1) + \tau_2(k - c_p(k)), x_2(k - c_p(k) + 1) + \tau_4(k - c_p(k) + 1)\right), \\
x_4(k) &\geq x_3(k) + \tau_2(k - c_p(k) + 1) + \tau_5(k - c_p(k) + 1).
\end{aligned} \tag{5-6}$$

Similarly, assuming that an event will take place as soon as all constraints are satisfied, the new implicit SMPL form is given as:

$$x(k) = A_0^{\text{idle}} \otimes x(k) \oplus A_1^{\text{idle}} \otimes x(k-1) \oplus A_{cp}^{\text{idle}} \otimes x(k - c_p(k) + 1) \oplus B_0^{\text{idle}} \otimes \bar{u}(k), \tag{5-7}$$

with

$$A_0^{\text{idle}} = \begin{bmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \tau_2(k - c_p(k) + 1) + \tau_5(k - c_p(k) + 1) & \varepsilon \end{bmatrix}, \; B_0^{\text{idle}} = \begin{bmatrix} \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \end{bmatrix},$$

$$A_1^{\text{idle}} = \begin{bmatrix} 0 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 0 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \tau_2(k - c_p(k)) & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{bmatrix}, \; A_{cp}^{\text{idle}} = \begin{bmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \tau_4(k - c_p(k) + 1) & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{bmatrix}.$$

5-2-3 Model for Simplex Mode

In the simplex mode, $y_1(k)$ and $y_2(k)$ in the paper path are skipped and the sheets immediately go from the input to $y_3(k)$. The paper path can be evaluate as follows:

$$\begin{aligned} &y_1(k) = y_3(k - c_p(k)), \\ &y_2(k) = y_2(k-1), \\ &y_3(k) \geq \max\left(y_1(k + c_p(k) - 1) + \tau_2(k-1), u(k) + \tau_1(k)\right), \\ &y_4(k) \geq y_3(k) + \tau_2(k) + \tau_5(k). \end{aligned} \tag{5-8}$$

With the states and input defined in Equation (5-2), we have

$$\begin{aligned} &x_1(k) = x_3(k-1), \\ &x_2(k) = x_2(k-1), \\ &x_3(k) \geq \max\left(x_1(k) + \tau_2(k - c_p(k)), \bar{u}_2(k) + \tau_1(k - c_p(k) + 1)\right), \\ &x_4(k) \geq x_3(k) + \tau_2(k - c_p(k) + 1) + \tau_5(k - c_p(k) + 1). \end{aligned} \tag{5-9}$$

Apply the same assumption as previous sub-sections, we can write the state equations6y into the following standard implicit SMPL form:

$$x(k) = A_0^{\text{simp}} \otimes x(k) \oplus A_1^{\text{simp}} \otimes x(k-1) \oplus A_{cp}^{\text{simp}} \otimes x(k - c_p(k) + 1) \oplus B_0^{\text{simp}} \otimes \bar{u}(k), \tag{5-10}$$

with

$$A_0^{\text{simp}} = \begin{bmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \tau_2(k - c_p(k)) & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \tau_2(k - c_p(k) + 1) + \tau_5(k - c_p(k) + 1) & \varepsilon \end{bmatrix},$$

$$A_1^{\text{simp}} = \begin{bmatrix} \varepsilon & \varepsilon & 0 & \varepsilon \\ \varepsilon & 0 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{bmatrix}, \; A_{cp}^{\text{simp}} = \begin{bmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{bmatrix}, \; B_0^{\text{simp}} = \begin{bmatrix} \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \\ \varepsilon & \tau_1(k - c_p(k) + 1) \\ \varepsilon & \varepsilon \end{bmatrix}.$$

5-2-4 Compact Form

Based on the results of the previous subsections, we define a set of max-plus switching binary variables $\ell(k)$ for the system in cycle k, with $\ell_1(k)$, $\ell_2(k)$ and $\ell_3(k)$ represents for duplex mode, idle mode and simplex mode respectively. If $\ell(k) = 0$, then the corresponding mode is on, and if $\ell(k) = -\infty$, the corresponding mode is off. In each cycle k, only one mode can be activated. Then the system can be written as the following compact form:

$$x(k) = A_0^{\ell(k)} \otimes x(k) \oplus A_1^{\ell(k)} \otimes x(k-1) \oplus A_{cp}^{\ell(k)} \otimes x(k - c_p(k) + 1) \oplus B_0^{\ell(k)} \otimes \bar{u}(k), \quad (5\text{-}11)$$

with

$$\begin{aligned}
A_0^{\ell(k)} &= \ell_1(k) \otimes A_0^{\text{dup}} \oplus \ell_2(k) \otimes A_0^{\text{idle}} \oplus \ell_3(k) \otimes A_0^{\text{simp}},\\
A_1^{\ell(k)} &= \ell_1(k) \otimes A_1^{\text{dup}} \oplus \ell_2(k) \otimes A_1^{\text{idle}} \oplus \ell_3(k) \otimes A_1^{\text{simp}},\\
A_{cp}^{\ell(k)} &= \ell_1(k) \otimes A_{cp}^{\text{dup}} \oplus \ell_2(k) \otimes A_{cp}^{\text{idle}} \oplus \ell_3(k) \otimes A_{cp}^{\text{simp}},\\
B_0^{\ell(k)} &= \ell_1(k) \otimes B_0^{\text{dup}} \oplus \ell_2(k) \otimes B_0^{\text{idle}} \oplus \ell_3(k) \otimes B_0^{\text{simp}}.
\end{aligned}$$

Chapter 6

Optimal Scheduling with MILP Method

In the previous chapter, we have already modelled the printer system in three different modes. The state equations are given in the form of switching max-plus-linear systems. There are two main methods to realize the optimal scheduling of the system. The first is the max-plus Model Predictive Control (MPC) method introduced in Chapter 4, and the other is to convert the system into a Mixed-Integer Linear Programming (MILP) problem. In this chapter, we will apply the second method and introduce how to solve the printer optimization scheduling problem with the MILP framework.

6-1 Conventional Algebra Transformation

Consider the Switching Max-Plus-Linear (SMPL) system defined in Equation (5-11):

$$x(k) = A_0^{\ell(k)} \otimes x(k) \oplus A_1^{\ell(k)} \otimes x(k-1) \oplus A_{cp}^{\ell(k)} \otimes x(k - c_p(k) + 1) \oplus B_0^{\ell(k)} \otimes \bar{u}(k),$$

with

$$\begin{aligned}
A_0^{\ell(k)} &= \ell_1(k) \otimes A_0^{\text{dup}} \oplus \ell_2(k) \otimes A_0^{\text{idle}} \oplus \ell_3(k) \otimes A_0^{\text{simp}},\\
A_1^{\ell(k)} &= \ell_1(k) \otimes A_1^{\text{dup}} \oplus \ell_2(k) \otimes A_1^{\text{idle}} \oplus \ell_3(k) \otimes A_1^{\text{simp}},\\
A_{cp}^{\ell(k)} &= \ell_1(k) \otimes A_{cp}^{\text{dup}} \oplus \ell_2(k) \otimes A_{cp}^{\text{idle}} \oplus \ell_3(k) \otimes A_{cp}^{\text{simp}},\\
B_0^{\ell(k)} &= \ell_1(k) \otimes B_0^{\text{dup}} \oplus \ell_2(k) \otimes B_0^{\text{idle}} \oplus \ell_3(k) \otimes B_0^{\text{simp}}.
\end{aligned}$$

Redefine the m-th row of the matrices A and B as follows:

$$A_{0_m}^{\ell(k)} = \begin{bmatrix} A_{0_{m1}}^{\ell(k)} \\ A_{0_{m2}}^{\ell(k)} \\ A_{0_{m3}}^{\ell(k)} \\ A_{0_{m4}}^{\ell(k)} \end{bmatrix}^T , \; A_{1_m}^{\ell(k)} = \begin{bmatrix} A_{1_{m1}}^{\ell(k)} \\ A_{1_{m2}}^{\ell(k)} \\ A_{1_{m3}}^{\ell(k)} \\ A_{1_{m4}}^{\ell(k)} \end{bmatrix}^T , \; A_{cp_m}^{\ell(k)} = \begin{bmatrix} A_{cp_{m1}}^{\ell(k)} \\ A_{cp_{m2}}^{\ell(k)} \\ A_{cp_{m3}}^{\ell(k)} \\ A_{cp_{m4}}^{\ell(k)} \end{bmatrix}^T , \; B_{0_m}^{\ell(k)} = \begin{bmatrix} B_{0_{m1}}^{\ell(k)} \\ B_{0_{m2}}^{\ell(k)} \end{bmatrix}^T ,$$

where $m = 1, 2, 3, 4$.

According to the nature of the max-plus system, the system can be converted into the following equation in conventional algebra:

$$\begin{aligned} x_m(k) = \max \Big(& A_{0_{m1}}^{\ell(k)} + x_1(k), A_{0_{m2}}^{\ell(k)} + x_2(k), A_{0_{m3}}^{\ell(k)} + x_3(k), A_{0_{m4}}^{\ell(k)} + x_4(k), + \cdots \\ & A_{1_{m1}}^{\ell(k)} + x_1(k-1), A_{1_{m2}}^{\ell(k)} + x_2(k-1), A_{1_{m3}}^{\ell(k)} + x_3(k-1), A_{1_{m4}}^{\ell(k)} + x_4(k-1), + \cdots \\ & A_{cp_{m1}}^{\ell(k)} + x_1(k - c_p(k) + 1), A_{cp_{m2}}^{\ell(k)} + x_2(k - c_p(k) + 1), + \cdots \\ & A_{cp_{m3}}^{\ell(k)} + x_3(k - c_p(k) + 1), A_{cp_{m4}}^{\ell(k)} + x_4(k - c_p(k) + 1), + \cdots \\ & B_{0_{m1}}^{\ell(k)} + \bar{u}_1(k), B_{0_{m2}}^{\ell(k)} + \bar{u}_2(k) \Big), \end{aligned} \tag{6-1}$$

with $m = 1, 2, 3, 4$.

Thus, for each m, the equation can be transformed as following fourteen inequalities:

$$\begin{aligned} & x_m(k) \geq A_{0_{m1}}^{\ell(k)} + x_1(k), \\ & \quad \vdots \\ & x_m(k) \geq A_{0_{m4}}^{\ell(k)} + x_4(k), \end{aligned} \qquad \begin{aligned} & x_m(k) \geq A_{cp_{m1}}^{\ell(k)} + x_1(k - c_p(k) + 1), \\ & \quad \vdots \\ & x_m(k) \geq A_{cp_{m4}}^{\ell(k)} + x_4(k - c_p(k) + 1), \end{aligned}$$

$$\begin{aligned} & x_m(k) \geq A_{1_{m1}}^{\ell(k)} + x_1(k-1), \\ & \quad \vdots \\ & x_m(k) \geq A_{1_{m4}}^{\ell(k)} + x_4(k-1), \end{aligned} \qquad \begin{aligned} & x_m(k) \geq B_{0_{m1}}^{\ell(k)} + \bar{u}_1(k), \\ & x_m(k) \geq B_{0_{m2}}^{\ell(k)} + \bar{u}_2(k), \end{aligned}$$

which can be rewritten into the following compact form:

$$\begin{cases} x_m(k) \geq A_{0_{mi}}^{\ell(k)} + x_i(k), \\ x_m(k) \geq A_{1_{mi}}^{\ell(k)} + x_i(k-1), \\ x_m(k) \geq A_{cp_{mi}}^{\ell(k)} + x_i(k - c_p(k) + 1), \\ x_m(k) \geq B_{0_{mj}}^{\ell(k)} + \bar{u}_j(k), \end{cases} \tag{6-2}$$

with $m = 1, 2, 3, 4$, $i = 1, 2, 3, 4$, $j = 1, 2$.

6-2 Binary Variable for Mode Selection

As discussed in Chapter 5, the printer has three working modes (with ℓ_1, ℓ_2 and ℓ_3 represent duplex mode, idle mode and simplex mode respectively). Therefore, we introduce conventional binary variable v_1, v_2 and v_3 with $v_1(k) + v_2(k) + v_3(k) = 1$ referring to these three modes and scalar $\beta \ll 0$. Then the inequality (6-2) can be split into the following inequalities:

$$\begin{cases} x_m(k) \geq A_{0_{mi}}^{\ell_q(k)} + x_i(k) + (1 - v_q(k)\beta)\,, \\ x_m(k) \geq A_{1_{mi}}^{\ell_q(k)} + x_i(k-1) + (1 - v_q(k)\beta)\,, \\ x_m(k) \geq A_{cp_{mi}}^{\ell_q(k)} + x_i(k - c_p(k) + 1) + (1 - v_q(k)\beta)\,, \\ x_m(k) \geq B_{0_{mj}}^{\ell_q(k)} + \bar{u}_j(k) + (1 - v_q(k)\beta)\,, \end{cases}$$

These inequalities can also be written as:

$$\begin{cases} -x_m(k) + x_i(k) + (1 - v_q(k)\beta) \leq -A_{0_{mi}}^{\ell_q(k)}, \\ -x_m(k) + x_i(k-1) + (1 - v_q(k)\beta) \leq -A_{1_{mi}}^{\ell_q(k)}, \\ -x_m(k) + x_i(k - c_p(k) + 1) + (1 - v_q(k)\beta) \leq -A_{cp_{mi}}^{\ell_q(k)}, \\ -x_m(k) + \bar{u}_j(k) + (1 - v_q(k)\beta) \leq -B_{0_{mj}}^{\ell_q(k)}, \end{cases} \tag{6-3}$$

with $i, m = 1, \cdots, 4;\ j = 1, 2;\ q = 1, 2, 3;\ v \in \mathbb{B}$, which satisfies

$$v_1 + v_2 + v_3 = 1. \tag{6-4}$$

And the inequality (6-3) can be divided into $4 \times 3 \times 14 = 168$ inequalities.

6-3 Solve the MILP Problem

Assume N is the prediction horizon of the system, y is the maximum value after N steps, i.e. $y \geq x_i(N),\ \forall i$, which represents the finishing time of the last module after N cycle. Define

$$z = \begin{bmatrix} y & x_1(1) & x_2(1) & \cdots & x_3(N) & x_4(N) & \bar{u}_1(1) & \cdots & \bar{u}_2(N) & v_1(1) & \cdots & v_3(N) \end{bmatrix}^T,$$

$$c = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & 0 \end{bmatrix}^T.$$

Then the scheduling problem defined in Equation (6-3) and (6-4) is transformed into a constrained MILP problem:

$$\begin{aligned} &\min_z c^T z \\ &s.t.\ EZ \leq F \end{aligned} \tag{6-5}$$

Chapter 7

Optimal Scheduling for Printers with Switching

In the actual printing process, due to the diversity of printed products, it is also necessary to consider the switching of multiple modes, such as the use of different types of paper for a document, or frequent switching of simplex and duplex modes. Mode switching often means a change in the routing (paper path) or ordering, which acts as an important issue for the optimal scheduling problem. In this chapter, we will discuss the use of the max-plus algorithm to solve scheduling problems involving different types of model switching.

7-1 Problem Introduction

Long-term printing (simplex mode and duplex mode) can be regarded as the steady states of the system. For the printer system, switching between different steady states cannot be completed instantaneously. It often requires many cycles of adjustment to stabilize, and this type of adjustment process is called the intermediate mode. In general, more than one intermediate mode is required, and the state equations of each intermediate mode are different. For different starting and ending steady states, the required intermediate modes are also different (such as from A4 to A3 and from single-sided A4 paper to double-sided A4 paper). This means that in the scheduling design process, a large number of intermediate modes have to be considered, and we need to parametrize the modes in a structured and generic way.

There are mainly three methods to do the parametrization [2]:

- Mode parametrization: enumerate all possible modes and use a specific mode number as a parameter.
- Integer parametrization: Use integers to describe features such as the order of operations of a specific resource (modules), or determine the route of a specific job.

- Binary parametrization: Binary variables can describe which of two operations go first for a resource.

Due to the large number of modes, it is difficult to apply mode parametrization. Therefore, we will use binary parametrization for our printer system. An important control decision, namely ordering, needs to be considered.

7-2 Ordering

In the printer system, ordering refers to which sheet enter the specific module first. We need to consider the ordering of events in different cycles, the basic idea is as follows:

Let $Z_\mu(k) \in \mathbb{R}^{4\times 4}$, $\mu = 0, 1, c_p(k)$ be order decision matrices with max-plus binary entries, where $[Z_\mu(k)]_{i,j} = 0$ if operation i in cycle k is scheduled after operation j in cycle $k+\mu$, and $[Z_\mu(k)]_{i,j} = \varepsilon$ if operation i in cycle k is scheduled before operation j in cycle $k+\mu$. Define the control vector $z_\mu(k)$ as the vector with the stacked column vectors of matrix $Z_\mu(k)$, that is $z_\mu(k) = \text{ vec}(Z_\mu(k))$. Then we can use the notation $Z_\mu(k) = Z\left(z_\mu(k)\right)$, and the ordering matrices can be defined as:

$$A_\mu^{\text{ord}}\left(z_\mu(k)\right) = Z\left(z_\mu(k)\right) \odot A_\mu^{\ell(k)}, \tag{7-1}$$

where $A_\mu^{\ell(k)}$ is defined in Equation (5-11), "$\odot$" refers to max-plus Hadamard product which satisfies:

$$(A \odot B)_{ij} = (A)_{ij} \otimes (B)_{ij} = (A)_{ij} + (B)_{ij}.$$

With ordering matrices, some unnecessary constraints previously defined can be cancelled. Then the operation ordering constraints in the system can be formulated as:

$$x(k) = A_0^{\text{ord}} \otimes x(k) \oplus A_1^{\text{ord}} \otimes x(k-1) \oplus A_{cp}^{\text{ord}} \otimes x(k - c_p(k) + 1) \oplus B_0^{\ell(k)} \otimes \bar{u}(k). \tag{7-2}$$

And all max-plus binary decision variables can be stacked into one vector:

$$b(k) = \begin{bmatrix} z_0(k) \\ z_1(k) \\ z_{cp}(k) \end{bmatrix} \in \mathbb{B}_\varepsilon^{L_{\text{bin}}}, \tag{7-3}$$

where L_{bin} is the total number of max-plus binary variables.

7-3 Intermediate Mode Design

As discussed in the previous section, the intermediate modes are in general different from the modes in steady states. This means that it is difficult for the supervisory controller to schedule the routing and ordering for intermediate modes. Therefore, we need to pre-design the intermediate modes. In the following sub-sections, we will consider the switching of modes (duplex and simplex) and paper sizes (A3 and A4 paper), and design the corresponding intermediate modes.

7-3-1 Applied Parameters

The parameters applied in this section are defined in Table 7-1:

Table 7-1: Parameters for Intermediate Modes Design

Width of the Paper Path (mm)	297
Actual Length of A4 Paper (mm)	210
Buffer distance for A4 Paper for Duplex Printing (mm)	20
Virtual Length of Duplex A4 Paper (mm)	230
Buffer distance for A4 Paper for Simplex Printing (mm)	10
Virtual Length of Duplex A4 Paper (mm)	220
Actual Length of A3 Paper (mm)	420
Buffer distance for A3 Paper for Duplex Printing (mm)	40
Virtual Length of Duplex A3 Paper (mm)	460
Buffer distance for A3 Paper for Simplex Printing (mm)	20
Virtual Length of Duplex A3 Paper (mm)	440

The conveying direction of the paper in the printer is shown in the figure below:

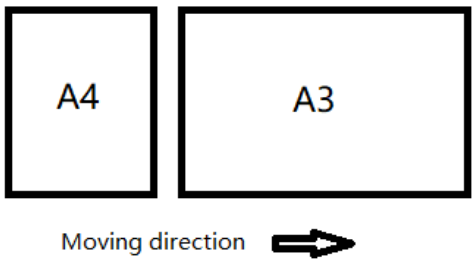


Figure 7-1: Conveying Direction of the Paper in the Printer

7-3-2 From Simplex Printing to Duplex Printing

In this sub-section, we assume that the size of the paper will not change, only the mode is switched from simplex to duplex. We use A4 paper as an example.

As shown in Figure 7-2, as the printer work in simplex mode, the back-loop (the loop with Invert Module (IM) and Re-entry Module (RM)) is empty, which means that we do not need to consider the ordering of RM. However, it can be noticed that the length of the buffer distance has changed significantly after the mode is switched. There are several reasons: first, since the double-sided printed paper will enter Image Transfer Station (ITS) twice, a longer buffer distance is required; secondly, after the first two sheets that need to be printed

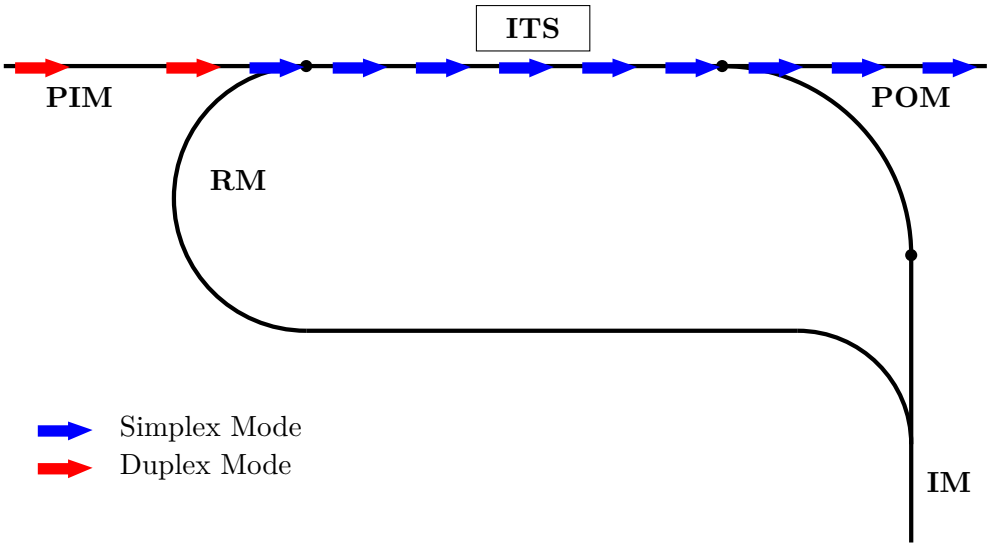


Figure 7-2: A Schematic of the Paper Path from Simplex Printing to Duplex Printing

on both sides enter RM via the loop, a new sheet of paper needs to be inserted, so a gap of at least one virtual paper length is required between the two sheets.

So the printer can switch the mode directly in this case without intermediate modes. And the buffer distance required after mode switching can be calculated as:

$$\textit{Buffer distance for A4 Paper for Duplex Printing} + \textit{Virtual Length of Duplex A4 Paper} = 250 \textit{ mm}.$$

7-3-3 From Duplex Printing to Simplex Printing

In this sub-section, we still assume that the size of the paper will not change, but the mode is switched from simplex to duplex. We also take the A4 paper as an example.

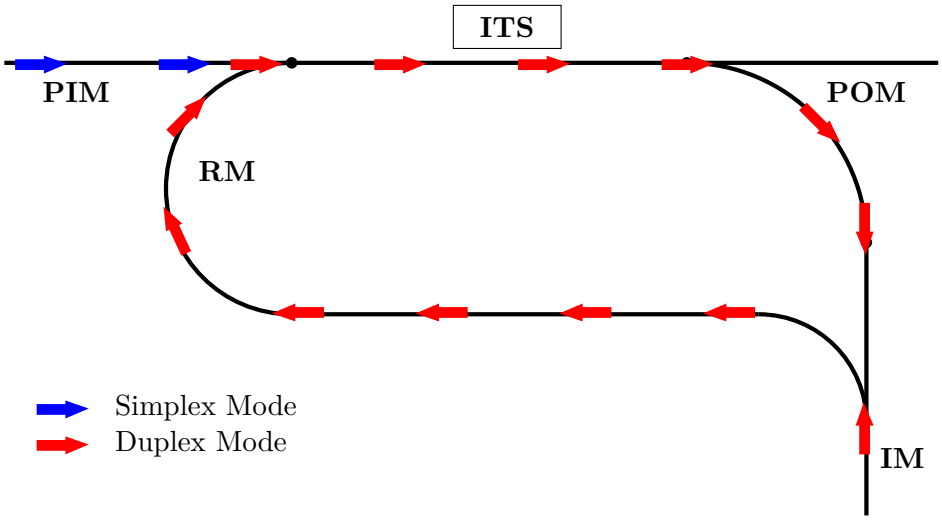


Figure 7-3: A Schematic of the Paper Path during Mode Switching

The schematic of the case is given in Figure 7-3. Compared with the paper for single-sided printing, the paper for double-sided printing passes through ITS twice, which leads to a longer buffer distance. This can be regarded as the virtual length of the input sheet is reduced, and the processing time of ITS will also be reduced. Therefore, the switching of the paper size from large to small (for example, A3 to A4) is the same case, and will not be discussed later.

The basic idea of the intermediate mode is to insert the new paper into the gap between the papers from RM, and all papers leave the printer alternately. It should be noted that for the printer system, if there are two types of paper in the loop, we need to apply the larger value of the processing time in the system matrices. Although the processing time of small sheets is overestimated and will lead to a larger buffer distance, on the contrary, applying smaller processing time prematurely on large-sized papers will cause system errors and paper overlaps. Only when all the large sheets exit the loop, a smaller processing time can be applied.

So in this case, the printer can directly switch from the duplex mode state equation given in (5-4) to the simplex mode state equation given in (5-10), but keep the processing time of ITS $\tau_2(k)$ and the buffer distance the same as double-sided printing until the last sheet of double-sided printing leaves Paper Output Module (POM). And the buffer distance required in intermediate modes is:

Buffer distance for A4 Paper for Duplex Printing+Virtual Length of Duplex A4 Paper $= 250$ *mm.*

7-3-4 From A4 Paper to A3 Paper

In this sub-section, we assume that the mode is fixed and only the paper size is modified from A4 to A3. In order to make the intermediate mode more universal, we use the more complicated double-sided printing as an example.

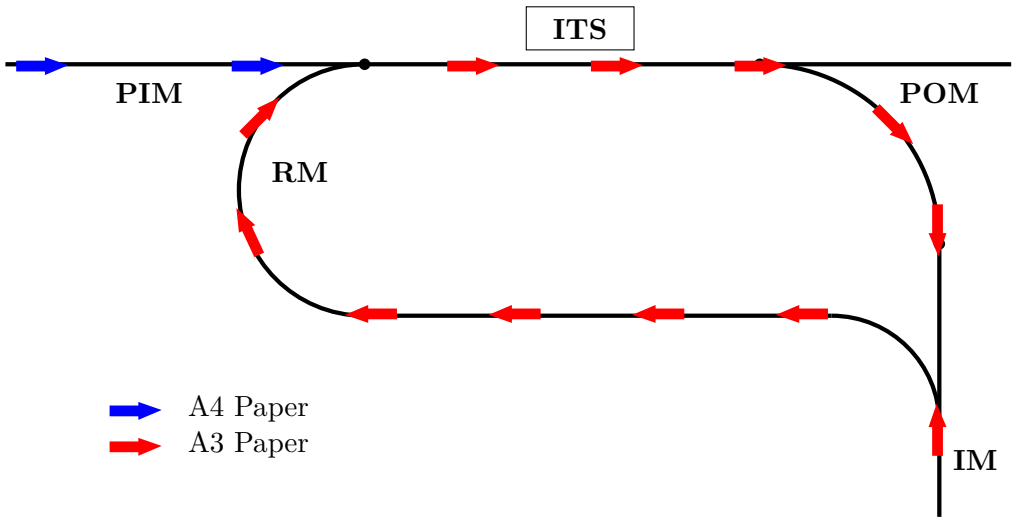


Figure 7-4: A Schematic of the Paper Path during Paper Size Switching

The schematic of the case is given in Figure 7-4. Similar to the previous sub-section, as the length of the input paper increases, the required buffer distance and the processing time of

ITS increase. To design intermediate modes for this case, the following two problems need to be considered:

1) How to leave enough space between A4 paper (the length of virtual A3 paper) so that A3 paper can be inserted into the gap?

2) In the process of adjusting the gap, how to ensure that the paper to be printed twice in the RM does not overlap with the newly entered A3 paper (the green circled part)?

To solve the question, the movement of the sheets is converted into a one-dimensional diagram:

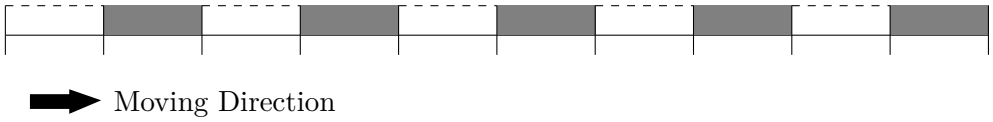


Figure 7-5: A Schematic of the Paper Path before Paper Size Switching

The movement of the paper that is about to enter Paper Input Module (PIM) before the mode switch is shown in the Figure 7-5, where the shaded part represents the position of the paper, and the dashed part represents the remaining space of the paper to be inserted from RM.

Suppose from one moment the input paper of the printer will switch to A3 after a few cycles. We will first switch the parameters in the system matrix (τ_2) from A4 paper to A3 paper. In order to improve efficiency, the buffer distance for feeding paper has been increased a little bit earlier. As shown in Figure 7-6 (a), compared with the figure above, half of the sheets are arranged to be fed in later.

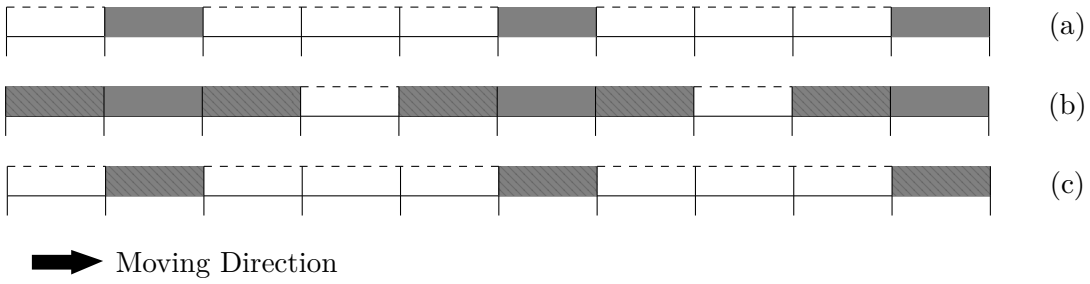


Figure 7-6: A Schematic of the Paper Path before Paper Size Switching (modified)

When this sequence is fed from PIM, it needs to be merged with the paper sequence in the back-loop (RM). Since the sequence in the back-loop has not been adjusted, the sequence is the same as that shown in Figure 7-5. The combined sequence is shown in Figure 7-6 (b). The lined part represents the paper that has been printed once from the RM, while the blank part still represents the space between the papers.

When the merged sequence passes through the ITS, the lined part of the paper will be output directly from the POM, and the previously unlined paper has been printed once and becomes the lined part, as shown in Figure 7-6 (c).

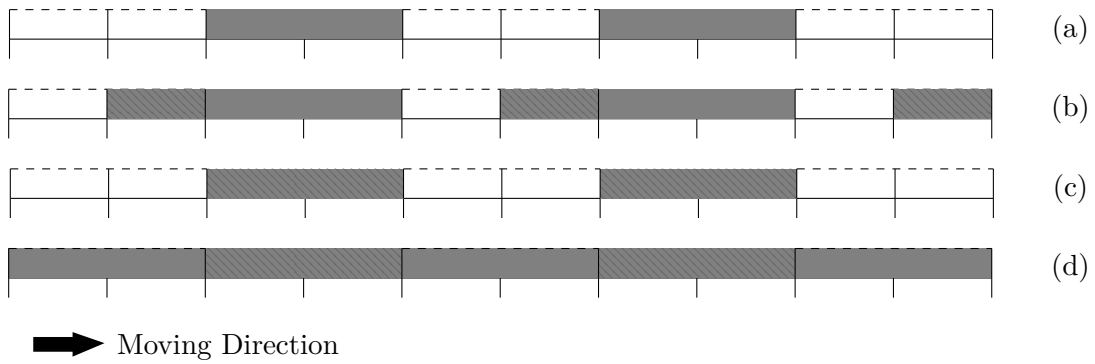


Figure 7-7: A Schematic of the Paper Path after Paper Size Switching

As mentioned in Chapter 5, the scroll speed in RM can be adjusted. Therefore, the sequence in Figure 7-6 (c) can be delayed by RM (stay at RM for a suitable time), and then merged with the A3 paper sequence in Figure 7-7 (a) into the sequence shown in Figure 7-7 (b).

After the sequence in Figure 7-7 (b) passes the ITS, all A4 papers will leave the printer through the POM. Meanwhile, the A3 papers that entered previously have completed one-sided printing, and will enter the RM through the loop (shown in Figure 7-7 (c)). It should be noted that at this time, the length of the gap between the lined paper can just be inserted into a new piece of A3 paper from PIM, as shown in Figure 7-7 (d). At this time, the printer has finished the intermediate mode and enters the double-sided printing A3 paper mode.

In summary, since the side length of A3 paper is twice that of A4 paper, the intermediate mode only needs to adjust the input sequence twice, and the result is optimal.

7-4 Compact Form for Paper Size Selection

In Section 5-2-4, we have discussed using the max-plus binary variables for working mode selection. In this section, we use a similar method to obtain a compact form for the paper size selection. We will consider duplex printing with four selectable paper sizes.

Consider the state equations for duplex mode printing given in (5-3):

$$
\begin{aligned}
x_1(k) &\geq \max\left(\bar{u}_1(k) + \tau_1(k), x_3(k-1) + \tau_2(k - c_p(k))\right), \\
x_2(k) &\geq \max\left(x_1(k) + \tau_2(k) + \tau_3(k), x_2(k-1) + \tau_4(k-1)\right), \\
x_3(k) &\geq \max\left(x_1(k) + \tau_2(k), x_2(k - c_p(k) + 1) + \tau_4(k - c_p(k) + 1)\right), \\
x_4(k) &\geq x_3(k) + \tau_2(k - c_p(k) + 1) + \tau_5(k - c_p(k) + 1),
\end{aligned}
$$

where τ and c_p are parameters depend on the mode and paper size.

Let (τ^d, c_p^d), $d = 1, 2, 3, 4$ be the pair of the printing parameter pairs for four different sizes of paper respectively. Introduce max-plus binary variables v_μ, $\mu = 1, \cdots, 4$ to control the order of the system, then the previous state equations can be transformed as:

$$
\begin{aligned}
x_1(k) \geq \max \Big(& \bar{u}_1(k) + \tau_1^1(k) + v_1, \bar{u}_1(k) + \tau_1^2(k) + v_2,\ \bar{u}_1(k) + \tau_1^3(k) + v_3, \cdots \\
& \bar{u}_1(k) + \tau_1^4(k) + v_4, x_3(k-1) + \tau_2^1(k - c_p^1(k)) + v_1, + x_3(k-1) + \cdots \\
& + \tau_2^2(k - c_p^2(k)) + v_2, x_3(k-1) + \tau_2^3(k - c_p^3(k)) + v_3, \cdots \\
& x_3(k-1) + \tau_2^4(k - c_p^4(k)) + v_4 \Big), \\
x_2(k) \geq \max \Big(& x_1(k) + \tau_2^1(k) + \tau_3^1(k) + v_1\ , x_1(k) + \tau_2^2(k) + \tau_3^2(k) + v_2, \cdots \\
& x_1(k) + \tau_2^3(k) + \tau_3^3(k) + v_3, x_1(k) + \tau_2^4(k) + \tau_3^4(k) + v_4, \cdots \\
& x_2(k-1) + \tau_4^1(k-1) + v_1, x_2(k-1) + \tau_4^2(k-1) + v_2, \cdots \\
& x_2(k-1) + \tau_4^3(k-1) + v_3, x_2(k-1) + \tau_4^4(k-1) + v_4 \Big), \\
x_3(k) \geq \max \Big(& x_1(k) + \tau_2^1(k) + v_1, x_1(k) + \tau_2^2(k) + v_2, x_1(k) + \tau_2^3(k) + v_3,\ \cdots \\
& x_1(k) + \tau_2^4(k) + v_4, x_2(k - c_p^1(k) + 1) + \tau_4^1(k - c_p^1(k) + 1) + v_1, \cdots \\
& x_2(k - c_p^2(k) + 1) + \tau_4^2(k - c_p^2(k) + 1) + v_2, x_2(k - c_p^3(k) + 1) + \cdots \\
& + \tau_4^3(k - c_p^3(k) + 1) + v_3,\ x_2(k - c_p^4(k) + 1) + \tau_4^4(k - c_p^4(k) + 1) + v_4, \Big), \\
x_4(k) \geq \max \Big(& x_3(k) + \tau_2^1(k - c_p^1(k) + 1) + \tau_5^1(k - c_p^1(k) + 1) + v_1,\ \cdots \\
& x_3(k) + \tau_2^2(k - c_p^2(k) + 1) + \tau_5^2(k - c_p^2(k) + 1) + v_2, \cdots \\
& x_3(k) + \tau_2^3(k - c_p^3(k) + 1) + \tau_5^3(k - c_p^3(k) + 1) + v_3, \cdots \\
& x_3(k) + \tau_2^4(k - c_p^4(k) + 1) + \tau_5^4(k - c_p^4(k) + 1) + v_4 \Big).
\end{aligned}
\tag{7-4}
$$

By this method, when certain paper size is selected, let the corresponding max-plus binary variable be 0 and others become ε, the state equation given in (7-4) will become the same as what we derived in Equation (5-3).

We can further apply the same method to the other two modes and convert them to the Max-Plus-Linear (MPL) form, which will provide us with a compact form of both mode and paper size selection.

Chapter 8

Optimal Control for Single Mode Printers in Stochastic Case

In this chapter, we will take the disturbance/noise into account, and assume the rolling speed of the rollers in each module becomes stochastic and following a normal distribution Probability Density Function (PDF), which will lead to a stochastic processing time in each module. We will try to analyze what requirements are needed to make the methods discussed in the previous chapters still be able to achieve optimal scheduling with overlapping probability satisfying the principle of $3\,\sigma$.

8-1 General Introduction

Recall the Switching Max-Plus-Linear (SMPL) system defined in Equation (5-11) in Chapter 5:

$$x(k) = A_0^{\ell(k)} \otimes x(k) \oplus A_1^{\ell(k)} \otimes x(k-1) \oplus A_{cp}^{\ell(k)} \otimes x(k-c_p(k)+1) \oplus B_0^{\ell(k)} \otimes \bar{u}(k) \qquad (8\text{-}1)$$

with matrices A and B consists of printer capacity c_p and the processing time of each module τ in the printer.

If the system becomes stochastic, the processing time for each sheet may be different. In order to avoid overlapping of sheets, a larger buffer distance is required. In other words, the input time of each cycle $u(k)$ should be delayed a little bit compared with the deterministic case. To create and solve a stochastic optimal scheduling problem, the basic ideas of this chapter are as follows:

1. For a certain printer system, obtain the processing time τ of each module and the movement distance d in modules in deterministic case;
2. Calculate the paper speed c in each module in deterministic case (which is the sloop of each section in the position time profile in Figure 5-3);

3. Assuming that the speed of each mode becomes stochastic, with the mean value $\bar{c} = c$. A normal distribution PDF will be applied. Let the stochastic value $\widetilde{c}$ be the actual moving speed of paper in each module.

4. Recalculate the processing time of each module $\widetilde{\tau}$ by $d/\widetilde{c}$, and use $\widetilde{\tau}$ in the system matrices in each module.

5. Solve the stochastic Mixed-Integer Linear Programming (MILP) problem with the method discussed in previous chapters.

8-2 Construct the Stochastic Speed and Time Profile

In the following sections, we assume that we already know the processing time τ and the movement distance d in each module in the deterministic case. Then the deterministic paper speed is $c = d/\tau$. We will consider a series of zero mean noise sequences that satisfies $\epsilon \sim (0, I)$. The noise sequence is somewhat correlated to ensure that the moving speed of adjacent sheets will not suddenly change greatly. Then the corresponding stochastic speed is:

$$\widetilde{c} = c + \epsilon, \quad \sim (c, I).$$

Note that in the simulation, this noise sequence can be generated by a series of zero-mean white noise signals passing through a low-pass filter.

Then the stochastic processing time for each section can be calculated by:

$$\widetilde{\tau} = \frac{d}{\widetilde{c}} = \frac{d}{c\left(1 + \frac{1}{c}\epsilon\right)}. \tag{8-2}$$

Let $f(\epsilon) = \dfrac{1}{1 + \frac{1}{c}\epsilon}$, then we have:

$$\frac{\partial f(\epsilon)}{\partial x} = -\frac{\frac{1}{c}}{\left(1 + \frac{1}{c}\epsilon\right)^2}.$$

Since ϵ is a small value with the mean of zero, $f(\epsilon)$ can be approximated by a first-order Taylor expansion around 0 as follows:

$$f(\epsilon) \approx f(0) + \frac{\partial f(0)}{\partial x}(\epsilon - 0) = 1 - \frac{\epsilon}{c}.$$

Substituting the result back into Equation (8-2), we can finally get the approximate stochastic processing time, which is given as:

$$\widetilde{\tau} = \frac{d}{c\left(1 + \frac{1}{c}\epsilon\right)} \approx \frac{d}{c}\left(1 - \frac{\epsilon}{c}\right). \tag{8-3}$$

8-3 Stochastic Case for Steady Modes

In this section, we will consider the stochastic case for steady modes (simplex and duplex printing). We will see what happens compared to the deterministic situation.

8-3-1 Simplex Mode

As discussed in Section 8-2, in order to prevent the moving speed of adjacent sheets from suddenly changing greatly, the disturbance is designed to be somewhat correlated. Therefore, the speed curves of the two adjacent sheets are similar. In other words, the difference between the acceleration and deceleration process of the two sheets of paper is small, almost negligible. In summary, this kind of random signal disturbance has little effect on the single-sided printing mode and can be ignored.

8-3-2 Duplex Mode

Similar to the previous sub-section, the influence of stochastic interference on the adjacent paper in the printer is almost negligible. But for double-sided printing, the merging of the green circle and the separation of the blue circle in the Figure 8-1 requires special attention.

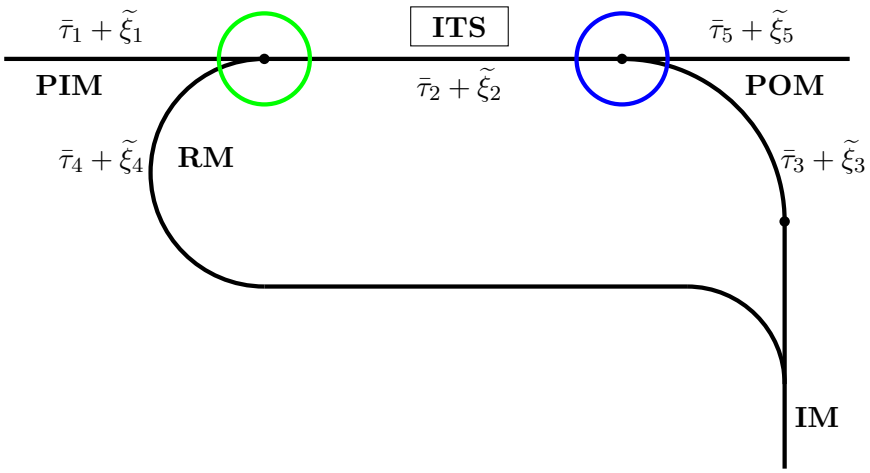
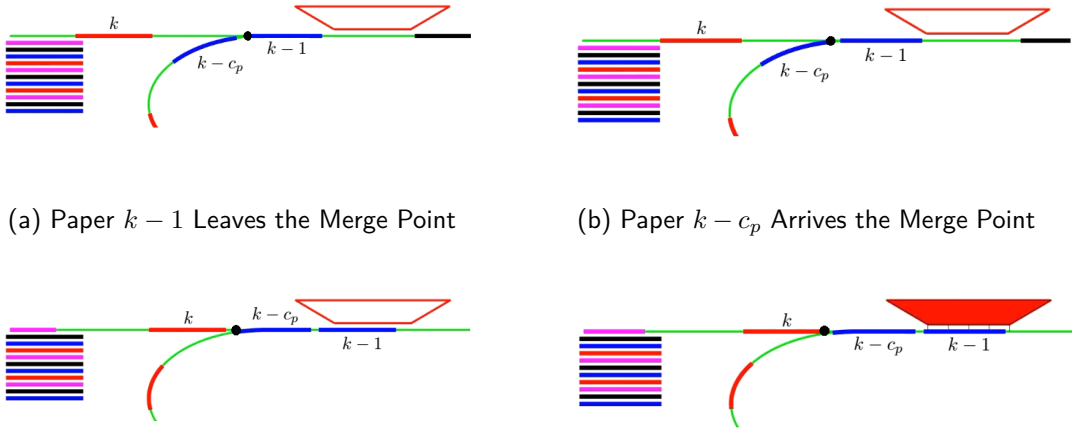


Figure 8-1: A Schematic of the Paper Path in a Stochastic Case

Consider the green circle first. There are four critical time instants in the merging process, as shown in the following figure. Suppose that the k-th paper reaches the merging point at time $t_{ma}(k)$, and the time at which it leaves the merging point is $t_{ml}(k)$. According to the figure, in order to prevent overlap, we need to meet the following conditions:

$$\begin{cases} t_{ml}(k-1) \leq t_{ma}(k-c_p) \\ t_{ml}(k-c_p) \leq t_{ma}(k) \end{cases} \tag{8-4}$$



(a) Paper $k-1$ Leaves the Merge Point

(b) Paper $k-c_p$ Arrives the Merge Point

(c) Paper $k-c_p$ Leaves the Merge Point

(d) Paper k Arrives the Merge Point

Figure 8-2: A Schematic of Critical Time Instants in the Merging Process

The actual time when the new incoming k-th paper arriving at the merging point is:

$$\begin{aligned} t_{ma}(k) &= y_1(k) \\ &= u(k) + \widetilde{\tau}_{k,1} \\ &= u(k) + \bar{\tau}_1 + \widetilde{\xi}_{k,1}, \end{aligned} \tag{8-5}$$

where $\bar{\tau}_i$ represents the mean of processing time $\widetilde{\tau}_i$, $\widetilde{\xi}_{k,i}$ represents the deviation between stochastic processing time of the k-th sheet and average processing time for $i = 1, 2, 3, 4, 5$.

Due to the accumulation of stochastic interference, the actual time when the $(k-c_p)$-th paper in Re-entry Module (RM) arriving at the merging point can be calculated as:

$$\begin{aligned} t_{ma}(k-c_p) &= y_3(k-c_p) \\ &= y_2(k-c_p) + \widetilde{\tau}_{k-c_p,4} \\ &= y_2(k-c_p) + \bar{\tau}_4 + \widetilde{\xi}_{k-c_p,1} + \widetilde{\xi}_{k-c_p,2} + \widetilde{\xi}_{k-c_p,3} + \widetilde{\xi}_{k-c_p,4}. \end{aligned} \tag{8-6}$$

Suppose the length of the paper is l, and the stochastic speed of the k-th paper in section i is $\widetilde{c}_{k,i}$ for $i = 1, 2, 3, 4, 2r, 5$, where $2r$ represents the second movement of the paper in the Image Transfer Station (ITS). Then we have:

$$\begin{aligned} t_{ml}(k-1) &= y_1(k-1) + l/\widetilde{c}_{k-1,2} \\ &= u(k-1) + \bar{\tau}_1 + \widetilde{\xi}_{k-1,1} + l/\widetilde{c}_{k-1,2}, \end{aligned} \tag{8-7}$$

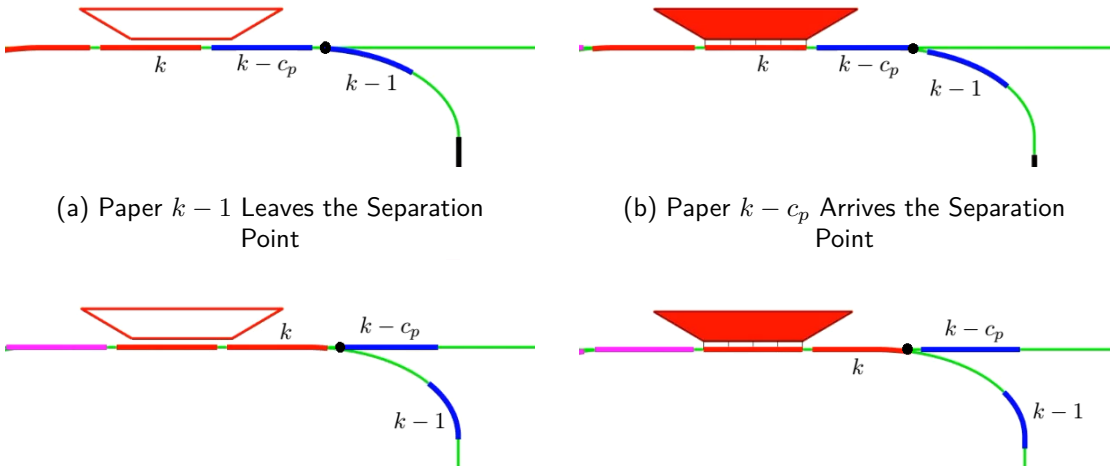
$$\begin{aligned} t_{ml}(k-c_p) &= y_3(k-c_p) + l/\widetilde{c}_{k-c_p,2r} \\ &= y_2(k-c_p) + \bar{\tau}_4 + \widetilde{\xi}_{k-c_p,1} + \widetilde{\xi}_{k-c_p,2} + \widetilde{\xi}_{k-c_p,3} + \widetilde{\xi}_{k-c_p,4} + l/\widetilde{c}_{k-c_p,2r}. \end{aligned} \tag{8-8}$$

Substitute Equation (8-5), (8-6), (8-7) and (8-8) into the constraints in (8-4), we obtain new constraints for stochastic cases. Use the set Q_m to represent the constraints from the merging point.

Since the movement of the paper in the ITS part is still disturbed by stochastic noise, it is only a necessary condition that the paper does not overlap at the merging point (the part of the green circle in Figure 8-1). In order to make it necessary and sufficient, we also need to consider the separation point (the part of the blue circle).

There are also four critical time instants in the separation process, as shown in the following figure. Suppose that the k-th paper reaches the separation point at time $t_{sa}(k)$, and the time at which it leaves the separation point is $t_{sl}(k)$. According to the figure, in order to prevent overlap, we need to meet the following conditions:

$$\begin{cases} t_{sl}(k-1) \leq t_{sa}(k-c_p) \\ t_{sl}(k-c_p) \leq t_{sa}(k) \end{cases} \tag{8-9}$$



(a) Paper $k-1$ Leaves the Separation Point

(b) Paper $k-c_p$ Arrives the Separation Point

(c) Paper $k-c_p$ Leaves the Separation Point

(d) Paper k Arrives the Separation Point

Figure 8-3: A Schematic of Critical Time Instants in the Separation Process

The actual time when the k-th paper arriving at the separation point is:

$$\begin{aligned} t_{sa}(k) &= y_1(k) + \widetilde{\tau}_{k,2} \\ &= u(k) + \bar{\tau}_1 + \bar{\tau}_2 + \widetilde{\xi}_{k,1} + \widetilde{\xi}_{k,2}. \end{aligned} \tag{8-10}$$

And the actual time when the $(k-c_p)$-th paper arriving at the separation point can be calculated as:

$$\begin{aligned} t_{sa}(k-c_p) &= y_3(k-c_p) + \widetilde{\tau}_{k-c_p,2r} \\ &= y_2(k-c_p) + \bar{\tau}_2 + \bar{\tau}_4 + \widetilde{\xi}_{k-c_p,1} + \widetilde{\xi}_{k-c_p,2} + \widetilde{\xi}_{k-c_p,3} + \widetilde{\xi}_{k-c_p,4} + \widetilde{\xi}_{k-c_p,2r}. \end{aligned} \tag{8-11}$$

In addition, the actual time for the k-th and $(k-c_p)$-th sheet to leave the separation point can also be calculated as:

$$\begin{aligned} t_{sl}(k-1) &= t_{sa}(k-1) + l/\widetilde{c}_{k-1,3} \\ &= u(k-1) + \bar{\tau}_1 + \bar{\tau}_2 + \widetilde{\xi}_{k-1,1} + \widetilde{\xi}_{k-1,2} + l/\widetilde{c}_{k-1,3}, \end{aligned} \tag{8-12}$$

$$\begin{aligned} t_{sl}(k-c_p) &= t_{sa}(k-c_p) + l/\widetilde{c}_{k-c_p,5} \\ &= y_2(k-c_p) + \bar{\tau}_2 + \bar{\tau}_4 + \widetilde{\xi}_{k-c_p,1} + \widetilde{\xi}_{k-c_p,2} + \widetilde{\xi}_{k-c_p,3} + \widetilde{\xi}_{k-c_p,4} + \widetilde{\xi}_{k-c_p,2r} + l/\widetilde{c}_{k-c_p,5}. \end{aligned} \tag{8-13}$$

Substitute Equation (8-10), (8-11), (8-12) and (8-13) into the constraints in (8-9), we obtain new constraints for stochastic cases. Use the set Q_s to represent the constraints from the separation point.

To satisfy the principle of $3\,\sigma$, we have the final constraints:

$$\mathbb{P}\left(Q_s \cap Q_s\right) \geq 99.7\% \tag{8-14}$$

Chapter 9

Conclusions and Future Work

9-1 Conclusion

In this thesis, we give a comprehensive introduction to max-plus algebra and Switching Max-Plus-Linear (SMPL) system. Then we model the printer system of three modes and discuss the optimal scheduling problem, which can be finally transformed into a Mixed-Integer Linear Programming (MILP) problem and solved by linear programming. Next, we design intermediate modes for different types of switching, and analyse the impact of stochastic systems on scheduling when noise is introduced. We find that max-plus algebra is competitive in printer scheduling.

We can now answer the questions posed in the first chapter.

Q1. *How to convert the printer system to SMPL system?*

The working status of the printer can be divided into three modes. We use the max-plus algorithm to model the three modes separately, and then convert them into a compact form through the max-plus binary variable.

Q2. *How to achieve optimal scheduling of printers with max-plus algebra?*

The optimal scheduling can be converted into a MILP problem. All pre-set constraints can be converted into the form of mixed-integer linear constraints, and then the optimal scheduling can be obtained through solving a linear programming problem.

Q3. *How to order the system when mode/paper size switching happens?*

Some unnecessary ordering can be eliminated by designing an ordering matrix. The ordering matrix can be merged with the original system matrices by the max-plus algebra to obtain new system matrices. In terms of mode/paper size switching, because of the complexity of scheduling, we need to pre-design the intermediate modes. In the report, the intermediate modes corresponding to the mutual switching between the two papers sizes and the two working modes are designed.

Q4. *What happens to the optimal scheduling problem when the system is disturbed?*

When the system is disturbed, the paper moving speed and processing time in each module become stochastics. We need to recalculate the stochastic processing time of all modules and substitute them back to the original system matrix. Then we can apply the method to deterministic situations and convert the problem to a MILP problem.

In addition to these research questions, we have also discussed the following parts in the report:

1) *How to find the places where the paper overlap is most likely to occur?*

In the absence of noise interference, adjacent papers will not overlap due to the buffer distance between the papers. In this case, what needs to be paid attention to is the sorting issue between the paper that enters Image Transfer Station (ITS) from Re-entry Module (RM) and the new incoming papers from Paper Input Module (PIM). When noise is present, in addition to the paper merging point, we also need to pay attention to the paper separation point, because the processing time of the paper in the ITS part is also stochastic. After this part of the noise is accumulated, the constraints designed at the paper merging point may not be able to ensure that the paper will not overlap before the separation point. Therefore, in this case, some new additional constraints need to be designed at the separation point.

2) *How to design intermediate modes?*

When the printer's working mode and paper size change, we have to design intermediate modes. When the working mode of the printer changes, we have to consider whether the new working mode involves double-sided printing. In this case, the paper in RM that needs to be printed twice will be inserted between the newly input paper. We, therefore, need to increase the spacing between papers in PIM. When the paper size changes, if the paper changes from large to small, we need to use the larger parameter in the system matrix to ensure that the paper does not overlap. And if the paper changes from small to large, we need to gradually increase the gap between the papers in the loop to ensure that the new large paper can be inserted into the gap between the two smaller papers.

9-2 Future Work

Regarding the research in this report, there are several topics that can be further explored, which are:

- **Restructure the matrix to speed up the optimization**: The system matrix in this report has 4 state variables, but some of them are in a coupled state. An excessive amount of states in the optimization process will slow down the calculation speed. Therefore, we can reconstruct the system matrix in the state equation, and combine and convert it into a form that is easy to optimize, such as the following triangular matrix. It may greatly reduce the constraints required for optimization and improve computational efficiency.

- **Consider a more complicated paper input situation**: In this report, we mainly considered the two paper sizes of A3 and A4. However, in the actual process of using the printer, more types of paper sizes and paper types of different materials may be involved. These factors may affect the system processing time of each module, resulting in more complex optimization problems. In addition, due to different types of paper input, the printer's working modes may also increase accordingly, and the increase in the number of switchable modes will result in a more complicated SMPL system, which can also be studied in the future.
- **Smarter ordering**: In this report, we first introduced three working modes of the printers. During the modelling process, we fixed the ordering of the printers, and then introduced the use of an ordering matrix to eliminate the previous setting in the following chapters. But this process requires manual operation. A smarter way is for the controller to automatically modify the ordering matrix according to the sequence of the input paper.

Bibliography

[1] M. Alirezaei, T. J. J. van den Boom, and R. Babuska, "Max-plus algebra for optimal scheduling of multiple sheets in a printer," in *2012 American Control Conference (ACC)*, pp. 1973–1978, 2012.

[2] T. J. J. Van den Boom, M. V. d. Muijsenberg, and B. De Schutter, "Model predictive scheduling of semi-cyclic discrete-event systems using switching max-plus linear models and dynamic graphs," *Discrete Event Dynamic Systems*, vol. 30, pp. 635–669, Dec 2020.

[3] M. Fromherz, "Constraint-based scheduling," in *Proceedings of the 2001 American Control Conference. (Cat. No.01CH37148)*, vol. 4, pp. 3231–3244 vol.4, 2001.

[4] J. Leung, *Handbook of Scheduling: Algorithms, Models, and Performance Analysis*. Chapman & Hall/CRC Computer and Information Science Series, CRC Press, 2004.

[5] C. Hanen, "Study of a np-hard cyclic scheduling problem: The recurrent job-shop," *European Journal of Operational Research*, vol. 72, pp. 82–101, Jan 1994.

[6] B. De Schutter, T. J. J. Van den Boom, J. Xu, and S. S. Farahani, "Analysis and control of max-plus linear discrete-event systems: An introduction," *Discrete Event Dynamic Systems*, vol. 30, pp. 25–54, Mar 2020.

[7] T. J. J. Van den Boom and B. De Schutter, "Modelling and control of discrete event systems using switching max-plus-linear systems," *Control Engineering Practice*, vol. 14, no. 10, pp. 1199–1211, 2006. 1199.

[8] F. Baccelli, *Synchronization and linearity : an algebra for discrete event systems*. Wiley series in probability and mathematical statistics, Chichester: Wiley, 1992.

[9] R. A. Cuninghame-Green, *Minimax Algebra*. Lecture notes in economics and mathematical systems ; 166, Berlin: Springer, 1979.

[10] T. J. J. Van den Boom and B. De Schutter, "Model predictive control for perturbed max-plus-linear systems: a stochastic approach," *International Journal of Control*, vol. 77, no. 3, pp. 302–309, 2004. 302.

[11] B. Heidergott, G. J. Olsder, and J. W. V. d. Woude, "Max plus at work : modeling and analysis of synchronized systems : a course on max-plus algebra and its applications," 2006.

[12] K. Murota, *Matrices and Matroids for Systems Analysis*. Springer Publishing Company, Incorporated, 1st ed., 2009.

[13] T. J. J. Van den Boom and B. De Schutter, "Modeling and control of switching max-plus-linear systems with random and deterministic switching," *Discrete Event Dynamic Systems : Theory and Applications*, vol. 22, no. 3, pp. 293–332, 2012. 293.

[14] K. M. Passino and K. L. Burgess, *Stability analysis of discrete event systems*. Adaptive and learning systems for signal processing, communications, and control, New York: Wiley, 1998.

[15] S. Van Loenhout, T. J. J. Van den Boom, S. Farahani, and B. De Schutter, "Model predictive control for stochastic switching max-plus-linear systems," *IFAC Proceedings Volumes*, vol. 45, no. 29, pp. 79–84, 2012. 79.

[16] T. J. J. Van den Boom and B. De Schutter, "Analytic expressions in stochastic max-plus-linear algebra and their application in model predictive control," *IEEE Transactions on Automatic Control*, pp. 1–1, 2020.

[17] T. J. J. Van den Boom, B. Heidergott, and B. De Schutter, "Complexity reduction in mpc for stochastic max-plus-linear discrete event systems by variability expansion," *Automatica*, vol. 43, no. 6, pp. 1058–1063, 2007. 1058.

[18] S. S. Farahani, T. J. J. Van den Boom, H. Van der Weide, and B. De Schutter, "An approximation method for computing the expected value of max-affine expressions," *European Journal of Control*, vol. 27, pp. 17–27, 2016. 17.

[19] P. J. . Davis and P. Rabinowitz, *Methods of numerical integration*. Computer science and applied mathematics, Orlando: Academic Press, 2nd ed. ed., 1984.

[20] J. Xu, T. J. J. Van den Boom, and B. De Schutter, "Model predictive control for stochastic max-plus linear systems with chance constraints," *IEEE Transactions on Automatic Control*, vol. 64, no. 1, pp. 337–342, 2019.

[21] V. Rostampour, D. Adzkiya, S. E. Z. Soudjani, B. De Schutter, and T. Keviczky, "Chance-constrained model predictive controller synthesis for stochastic max-plus linear systems," in *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 003581–003588, 2016.

[22] T. J. J. Van den Boom, J. Xu, and B. De Schutter, "Corrections to "model predictive control for stochastic max-plus linear systems with chance constraints,"," *IEEE Transactions on Automatic Control*, vol. 65, no. 2, pp. 905–906, 2020.

[23] T. J. J. Van den Boom and B. De Schutter, "Model predictive control for randomly switching max-plus-linear systems using a scenario-based algorithm," in *49th IEEE Conference on Decision and Control (CDC)*, pp. 2298–2303, 2010.

[24] B. De Schutter and T. J. J. Van den Boom, "Model predictive control for max-plus-linear discrete event systems," *Automatica*, vol. 37, no. 7, pp. 1049 – 1056, 2001.

Glossary

List of Acronyms

ITS	Image Transfer Station
PIM	Paper Input Module
POM	Paper Output Module
RM	Re-entry Module
IM	Invert Module
DESs	Discrete Event Systems
SMPL	Switching Max-Plus-Linear
MILP	Mixed-Integer Linear Programming
MPL	Max-Plus-Linear
MC	Monte Carlo
MPC	Model Predictive Control
RSMPL	Randomly Switching Max-Plus-Linear
JDF	Joint Density Function
PDF	Probability Density Function