Energy-Efficient Technologies for High-Performance Manufacturing Industries

Cao Vinh Le

NATIONAL UNIVERSITY OF SINGAPORE

2013

Energy-Efficient Technologies for High-Performance Manufacturing Industries

Cao Vinh Le

B.Eng. (Hons.), Nanyang Technological University, 2009

A DISSERTATION SUBMITTED

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

NATIONAL UNIVERSITY OF SINGAPORE

2013

Declaration

I hereby declare that this thesis is my original work

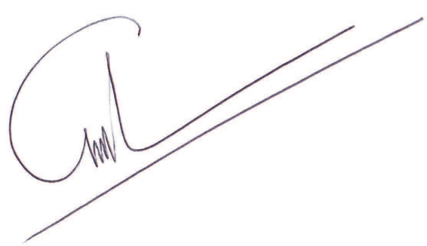
and it has been written by me in its entirety.

I have duly acknowledged all the sources of

information which have been used in the thesis.

This thesis has also not been submitted for any

degree in any university previously.



Le Cao Vinh

14 October 2013

Acknowledgements

Having always borne in mind that pursuing a Ph.D. is a long and tough journey which really tests one's endurance, self-determination, and tenacity, but I still nearly gave up. There are a number of people without whom this dissertation might not have been written, and to whom I greatly indebted.

First and foremost, I sincerely thank my sole dissertation advisor Prof. Pang Chee Khiang, Justin for his great supervision, patience, and motivation. I have been lucky enough to know and to work with such a great advisor and teacher. I am grateful to every single thing he has taught and his enthusiasm in grooming me into an independent researcher. I truly admire his diligence and passion for high-quality and high-impact research which has been always a source of inspiration to me. I also wish to thank him for forgiving and resolving so many troubles I have made along the way. May God bless him with good health and happiness, and I hope to learn a lot and a lot more from him.

My special thanks should go out to Dr. Oon Peen Gan, Ms. Danhong Zhang, Dr. Ming Luo, Dr. Hian Leng Ian Chan, and Dr. Junhong Zhou of Manufacturing Execution and Control Group, A*STAR Singapore Institute of Manufacturing Technology for their hospitality and support during my attachment. I am grateful to Prof. Frank L. Lewis of Automation and Robotics Research Institute, The University of Texas at Arlington for offering me valuable comments and suggestions in supervisory control of discrete-event systems. I also deeply appreciate Dr. Greg R. Hudas and Mr. Dariusz G. Mikulski of The U.S. Army Tank Automotive Research, Development and Engineering Center for the great collaboration.

I am greatly thankful to my parents Mr. Trong Toi Le and Mrs. Thi Anh Nga Cao for their nurture, continued love, emotional support, inspiration, and valuing my dreams. They have always been a great role model of resilience, strength, and character since my childhood. I am proud to dedicate this dissertation to them. I also want to thank all the members of my research group, Mr. Tan Yan Zhi, Mr. Yan Weili, Mr. Yan Hengchao, and Mr. Zhu Haiyue, for the fruitful discussions during our weekly research forums.

Last but not least, I would like to thank the Department of Electrical and Computer Engineering, National University of Singapore for providing me financial support in the form of a research scholarship. My gratitude also goes to all the staffs and students of Manufacturing Execution and Control, A*STAR Singapore Institute of Manufacturing Technology and Advanced Control Technology Laboratory, Department of Electrical and Computer Engineering, National University of Singapore who had helped me in many ways.

Abbreviations

AC	Air conditioner
ACO	Ant-colony optimization
ADEC	Augmented discrete event control
AR	Auto regression
B&B	Branch-and-bound
B&R	Branch-and-reduce
BTU	British thermal unit
CAPEX	Capital expenditure
CAPP	Computer-aided process planing
CBM	Condition-based maintenance
CO_2	Carbon dioxide
CP	Convex programming
CR	Completed rescheduling
DAP	Deadlock avoidance policie
DEC	Discrete event control

DSS	Decision support system
DP	Dynamic programming
EA	Evolutionary algorithm
EBayes	Empirical Bayesian
EMA	Energy Market Authority
EIA	Energy Information Administration
FCFS	First come first served
FCM	Fuzzy c-means
FMS	Flexible manufacturing system
FSM	Finite-state machine
FTC	Fault tolerant control
GA	Genetic algorithm
GJ/t	Gigajoule/tonne
GM	Geometric median
GUI	Graphic user interface
HDD	Hard disk drive
IEA	International Energy Agency
IID	Independent and identically distributed
ITL	Information-theoretic learning
LEC	Least energy cost first
LP	Linear programming

MAD	Mean absolute deviation
ME	Mean-entropy
MINLP	Mixed integer nonlinear program
MP	Manufacturing process
MTME	Max-throughput-min-energy
MV	Mean-variance
NP-hard	Non-deterministic polynomial-time hard
OPEX	Operational expenditure
PDF	Probability distribution function
PN	Petri net
PR	Partial rescheduling
PSO	Particle-swarm optimization
SEC	Specific energy consumption
SG	Savitzky-Golay
SPT	Shortest processing time first
SVM	Support vector machine
RAM	Random-access memory
RBF	Radial basis function
R&D	Research and development
RG	Reachability graph
RUDOLF	Rudolf R-DPA96A digital power analyzer

RV	Random variable
VCM	Voice coil motor
W-C	Worst-case
WSN	Wireless sensor network
WTPN	Weighted p-timed Petri net

Contents

Acknowledgements **ii**

Abbreviations **iv**

Summary **xiv**

List of Tables **xvii**

List of Figures **xix**

List of Symbols **xxii**

1 Introduction **1**

1.1 Background 2

1.1.1 Energy Consumption of Manufacturing Industries 3

1.1.2 Energy Saving Potentials through Energy-Efficient Technologies 8

1.2 Literature Review on Energy-Efficient Technologies 9

1.2.1 Systems Level 10

1.2.2 Process Level 12
1.2.3 Facility Level 13
1.2.4 Equipment Level 14
1.3 Motivation of Dissertation 15
1.4 Contributions and Organization 17

2 Descriptions and Modeling of Flexible Manufacturing Systems 20

2.1 Descriptions 21
2.2 Finite-State Machine Models of Manufacturing Processes 28
2.3 Weighted P-Timed Petri Net Models of Flexible Manufacturing Systems 30
2.3.1 Petri Nets 30
2.3.2 Weighted P-Timed Petri Nets 32
2.4 Augmented Discrete Event Control Models of Flexible Manufacturing Systems 36
2.4.1 Matrices and Vectors 38
2.4.2 Logical State Equation 42
2.5 Summary 45

3 Energy Data-Driven Process State Identification for High-Performance Decision Support 47

3.1 Background 48

3.2 Process Identification Framework 50
3.2.1 Signal Segmentation 50
3.2.2 Segment Clustering 53
3.3 Industrial Applications 57
3.3.1 Experiment Setup 57
3.3.1.1 Injection Moulding Process 60
3.3.1.2 Stamping Process 61
3.3.2 Experiment Results 63
3.3.2.1 Identification Results with Sufficient Training Data . 66
3.3.2.2 Identification Results with Limited Training Data . . 69
3.3.3 Discussions with Related Works 71
3.4 Energy Data-Driven Decision Support System 72
3.4.1 Architecture 73
3.4.2 Decision-Making Models 77
3.5 Summary 81

4 Scheduling of Flexible Manufacturing Systems under Power Consumption Uncertainties 82

4.1 Background 83
4.2 Dynamic Scheduling Under Power Consumption Uncertainties 86
4.2.1 Mathematical Model of Power Consumption Uncertainties . . 86

4.2.2 Problem Description 88
4.3 Fast Reactive Scheduling 90
4.3.1 Solution Overview 90
4.3.2 Reduction of Model Complexity 93
4.3.3 Choice Set 94
4.3.4 Min-Throughput-Max-Energy Reactive Scheduling 96
4.4 Industrial Application 99
4.4.1 Energy Analysis of Stamping Process 100
4.4.2 Augmented Discrete Event Control Models of Stamping System 103
4.4.3 Experiment Results 109
4.4.4 Scalability 112
4.4.5 Discussions with Related Works 116
4.5 Summary 117

5 Total Energy Optimization of Flexible Manufacturing Systems Using Dynamic Programming 119

5.1 Background 120
5.2 Problem Formulation with Mathematical Programming 123
5.2.1 Formulation of Constraints 124
5.2.2 Objective Function and Convexity Analysis 127
5.3 Energy-Optimal Path Computation Using Dynamic Programming . . 130

5.3.1 Formulation of Dynamic Programming 131
5.3.2 Computation of Energy-Optimal Path 134
5.3.3 Error Analysis 139
5.4 Industrial Application 143
5.4.1 Weighted P-Timed Petri Net Models of Industrial Stamping System 143
5.4.2 Experiment Results 145
5.4.3 Discussions with Related Works 148
5.5 Summary 150

6 Robust Total Energy Optimization of Flexible Manufacturing Systems Based on Renyi Mean-Entropy Criterion 152

6.1 Background 153
6.2 Robust Energy Optimization Based on Renyi Mean-Entropy Criterion 156
6.2.1 Brief Overview on Robust Shortest Path Problem 156
6.2.1.1 Models of Uncertainties 157
6.2.1.2 Robustness Measures 158
6.2.2 Renyi Mean-Entropy Criterion 159
6.2.3 Non-Parametric Estimation of Edge Costs 162
6.3 Simulations 166
6.3.1 Probability Distributions 167

6.3.2 Simulation Setup and Results 168

6.4 Industrial Application 171

6.4.1 Robust Energy Analysis of Stamping Process 172

6.4.2 Results and Discussions 174

6.5 Summary 177

7 Conclusion and Future Work **178**

Bibliography **184**

List of Publications **210**

Summary

The manufacturing industries have shifted towards a "green" paradigm due to increase of dangerous climate change, emergence of new energy legislation and regulations, and consumers' growing trend in buying green products and services, where manufacturers will compete in energy efficiencies and carbon footprints of manufactured products. This dissertation proposes novel technologies for improving manufacturing energy efficiencies with specific applications to manufacturing processes (MPs) and flexible manufacturing systems (FMSs).

After a brief introduction of current energy consumption in manufacturing industries, literature review on state-of-the-art energy-efficient technologies, and motivations of this dissertation, mathematical modeling of MPs and FMSs using different languages will be detailed.

First, a novel approach is proposed to reduce the number of required sensors in process state tracking by identifying the operational states of MPs using useful information and features in energy data. Finite-state machines (FSMs) are used to model MPs, and a two-stage framework for online classification of real-time energy data in terms of MP operational states is proposed using Haar transform and em-

pirical Bayesian (EBayes) threshold for segmentation of time series of power data and support vector machines (SVMs) for clustering of power segments into groups according to underlying MP operational states. Based on obtained results, we design an energy data-driven decision support system (DSS), which uses real-time energy measurements and process operational states to make effective decisions, enabling high-performance manufacturing.

Next, the reduction of energy consumption is studied in scheduling and operational control of FMSs. A dynamic scheduling problem which minimizes the sum of energy cost and tardiness penalty under power consumption uncertainties is studied. An integrated control and scheduling framework is proposed including two modules, namely, an augmented discrete event control (ADEC) and a max-throughput-min-energy (MTME) reactive scheduling model.

A total energy optimization problem is studied next, which aims to minimize both productive and idle energy consumption optimally subjected to the general production constraints, using the weighted p-timed Petri net (WTPN) models of FMSs. The considered problem is proven to be a nonconvex mixed integer nonlinear program (MINLP). A new reachability graph (RG)-based discrete dynamic programming (DP) approach is proposed for generating near energy-optimal schedules within adequate computational time.

Extending the total energy optimization problem to deal with uncertainties in energy measurement process, a robust energy optimization problem is studied where both productive and idle powers are random variables (RVs). The robust energy-optimal schedule is determined by searching the robust shortest path of WTPN RG based on a novel Renyi mean-entropy (ME) criterion. It is shown that DP can be applied with Renyi ME criterion to construct the robust shortest path efficiently.

This dissertation presents novel energy-efficient technologies to fulfill the emerging green demands for high-performance manufacturing industries, which require manufactured products not only to be free of flaws but also to be environmentally sustainable. In addition to necessary simulations, our proposed energy-efficient technologies are verified with energy data logged from industrial manufacturing plants, making our contributions readily applicable for high-performance manufacturing industries.

List of Tables

2.1 Part Type π_1 of FMS Example–Rule Bases 39

2.2 Part Type π_2 of FMS Example–Rule Bases 39

3.1 Outlier Detection Results 66

3.2 Cluster Label for Injection Moulding and Stamping Operational States 67

3.3 Number of Validated Segments with Sufficient Training Data 68

3.4 Number of Validated Segments with Limited Training Data 69

3.5 Energy Audit for Arburg A220 S 150–60 77

3.6 Machine Clustering of Arburg A220 S 150–60 and Arburg A420 S 1000–150 79

4.1 Machine Performance and Efficiency 101

4.2 Part Type π_1–Rule Bases 105

4.3 Part Type π_2–Rule Bases 106

4.4 Mean and Variance of Power Consumption Uncertainties $\mu_{ij}^q, \left(\sigma_{ij}^q\right)^2$. 110

4.5 Comparison of $T_{\text{mean}}(s)$ under Different Probability Distributions . . 113

4.6 $T_{\mathrm{mean}}(s)$ of MTME with Different FMS Sizes 115

5.1 Performance Comparisons of B&R, PSO, ACO, and DP 147

6.1 Fully FMS Sizes for Simulation Test Cases 170

6.2 Performance Comparisons of W-C Analysis, MV, and Renyi ME Criteria175

List of Figures

1.1 Delivered energy consumption by sector 1980–2040. 3

1.2 Global energy consumption 1990–2035. 4

1.3 Annual changes in world industrial and all other end-use energy consumption 2007–2011. 6

1.4 Energy consumption *per capita* for selective developed countries in 2006. 7

2.1 Power consumption profile of injection moulding process using Arburg A220 S 150–6 machine tool. 22

2.2 An example of FMSs with two part types, eight jobs, and eight resources including five machines and three material routing robots. . . 26

2.3 FSM models of industrial injection moulding process. 29

2.4 WTPN models of FMS example. 34

3.1 Arburg A220 S 150–60 injection moulding machine. 58

3.2 Arburg A420 S 1000–150 injection moulding machine. 58

3.3 A screenshot of GUI developed in LabVIEW for online energy monitoring. 59

3.4 A comparative example between a normal and an abnormal power segments from *Stamping* state: (top) normal segment and (bottom) *Abnormal* segment. 62

3.5 The discrete-state time series of power data of industrial processes: (top) injection moulding and (bottom) stamping. 63

3.6 FSM models of industrial stamping process. 64

3.7 An illustrated example of signal segmentation using the our proposed framwork: a) time series of power data, b) wavelet coefficients with EBayes threshold (dashed line), and c) detected change points. 65

3.8 An example of outlier detection of *Moulding* state. 66

3.9 Energy data-driven DSS architecture for high-performance manufacturing industries. 74

4.1 Simplified flowchart of our proposed framework. The ADEC replicates the discrete-event dynamics of the system jobs and resources. The MTME decides the local optimal schedule of active jobs and resources. 92

4.2 PN models of example part type. 95

4.3 PN-equivalent ADEC models of example part type. 95

4.4 An example of VCM yokes. 101

4.5 Typical power profile of stamping process. 102

4.6 Deviation from Pareto optimality under Weibull distribution. 112

4.7 Deviation from Pareto optimality under truncated normal distribution. 113

4.8 Deviation from Pareto optimality under exponential distribution. . . 114

5.1 A simple marked WTPN models example. 136

5.2 The full 3-stage RG of WTPN models example. 137

5.3 The reduced 3-stage RG of WTPN models example. 138

5.4 Layout of the stamping system. 145

5.5 WTPN models of the stamping system. 146

6.1 Marked WTPN models of a fully FMS. 169

6.2 Mean deviation of three robustness measures under Gaussian distribution. 171

6.3 Mean deviation of three robustness measures under uniform distribution.172

6.4 Mean deviation of three robustness measures under bimodal distribution.173

6.5 Histogram of c_{113} with 120 observations. 174

7.1 The nano-satellite swarm concept. 182

List of Symbols

Σ_F	Set of symbols of finite-state machine
S_F	Set of states of finite-state machine
s_{F0}	Initial state of finite-state machine
δ_F	State-transition function of finite-state machine
F_F	Set of final states of finite-state machine
R	Set of resources of finite-state machine
Π	Set of part types of finite-state machine
$\lvert \bullet \rvert$	Cardinality of a set
r_i	Resource i of flexible manufacturing system
$C(r_i)$	Capacity of resource i
π_q	Part type q of flexible manufacturing system
$\varphi(\pi_q)$	Number of type-π_q parts
ω_q	Job sequence of part type q
V	Set of jobs of flexible manufacturing system
V_z	Set of choice jobs of flexible manufacturing system

V_{nz}	Set of non-choice jobs of flexible manufacturing system
v_j^q	Job j of part type pi_q
$R\left(v_j^q\right)$	Set of resources which can perform v_j^q
v_{in}^q	Input buffer of part type π_q
v_{out}^q	Output buffer of part type π_q
$\mathbf{A}$	Productive power matrix of flexible manufacturing system
$\mathbf{A}^q$	Productive power matrix of part type π_q
$\mathbf{b}$	Idle power vector of flexible manufacturing system
$\mathbf{D}$	Processing time matrix of flexible manufacturing system
$\mathbf{D}^q$	Processing time matrix of part type π_q
a_{ij}^q	Productive power of r_i to perform v_j^q
b_i	Idle power of r_i
d_{ij}^q	Processing time of r_i to perform v_j^q
$\mathbb{R}^+$	Set of nonnegative real numbers
χ	A weighted p-timed Petri net
P	Set of places.
T	Set of transitions.
I	Set of input arcs of weighted p-timed Petri net
O	Set of output arcs of weighted p-timed Petri net
α	A node of weighted p-timed Petri net
${}^\bullet\alpha$	Pre-set of α

$\alpha^{\bullet}$	Post-set of α
$\mathbf{W}$	Incident matrix of weighted p-timed Petri net
$\mathbf{x}$	State vector of weighted p-timed Petri net
$\mathbf{u}$	Control vector of weighted p-timed Petri net
p_i	Place i of weighted p-timed Petri net
t_j	Transition j of weighted p-timed Petri net
$M(p_i)$	Marking of place p_i
c_i	Sojourn cost per time unit of place p_i
h_i	Minimal sojourn time of place p_i
P_R	Set of resource places of weighted p-timed Petri net
P_V	Set of job places of weighted p-timed Petri net
P_{in}	Set of input buffer places of weighted p-timed Petri net
P_{out}	Set of output buffer places of weighted p-timed Petri net
$\mathbb{Z}^+$	Set of nonnegative integers
$\mathbf{x}_0$	Initial state of weighted p-timed Petri net
$\mathbf{x}_{\lvert K \rvert}$	Final state of weighted p-timed Petri net
$\lvert K \rvert$	Total number of firing epochs of weighted p-timed Petri net
G	Set of rules of augmented discrete event control
g_i^q	Rule i of part type π_q
$supp(\bullet)$	Support of a vector
$\mathbf{F}_v$	Job sequence matrix of augmented discrete event control

$\mathbf{F}_v^q$	Job sequence matrix of part type π_q
$\mathbf{F}_r$	Conjunctive resource assignment matrix of augmented discrete event control
$\mathbf{F}_r^q$	Conjunctive resource assignment matrix of part type π_q
$\mathbf{F}_u$	Input matrix of augmented discrete event control
$\mathbf{F}_u^q$	Input matrix of part type π_q
$\mathbf{F}_{ud}$	Deadlock resolution matrix of augmented discrete event control
$\mathbf{F}_{rd}$	Disjunctive resource assignment matrix of augmented discrete event control
$\mathbf{F}_{rd}^q$	Disjunctive resource assignment matrix of part type π_q
$\mathbf{S}_v$	Job start matrix of augmented discrete event control
$\mathbf{S}_v^q$	Job start matrix of part type π_q
$\mathbf{S}_y$	Output matrix of augmented discrete event control
$\mathbf{S}_y^q$	Output matrix of part type π_q
$\mathbf{g}$	Rule vector of augmented discrete event control
$\mathbf{v}_c$	Job completion vector of augmented discrete event control
$\mathbf{r}_c$	Resource available vector of augmented discrete event control
$\mathbf{u}$	Input vector of augmented discrete event control
$\mathbf{u}_d$	Deadlock resolution vector of augmented discrete event control
$\mathbf{v}_s$	Job start vector of augmented discrete event control
$\mathbf{y}$	Output vector of augmented discrete event control
$\mathbf{g}$	Logical rule state vector of augmented discrete event control
$\mathbf{p}$	Time series of power data

$\mathbf{a}$	Approximate coefficient vector of Haar wavelet
$\mathbf{d}$	Wavelet detail coefficient vector of Haar wavelet
μ	Distribution mean vector of empirical Bayesian threshold
ε	Noise vector of empirical Bayesian threshold
f_{prior}	Prior distribution of empirical Bayesian threshold
w	Probabilistic variable of empirical Bayesian threshold
δ_0	Dirac function
σ	Symmetric heavy-tailed probability density function
ϕ	Standard Gaussian probability density function
β	Regression coefficient vector of auto regressive model
ω	Intercept variable of auto regressive model
ς	Noise parameter of auto regressive model
$\mathbf{s}$	Feature vector of support vector machine
$\mathbf{o}$	Orientation vector of support vector machine hyperplane
ξ	Slack variables of support vector machine
$k(\cdot,\cdot)$	Kernel function of support vector machine
α_s	Nonvanishing coefficients of support vector machine
N_s	Number of support vectors of support vector machine
(C,γ)	Kernel parameter pair of support vector machine
θ_k	Real parameters of a step function
B_k	Intervals of a step function

χ_B	Indicator function of interval B
ν	Weight parameter
G_a	Choice set in rule domain
G_z	Set of choice rules of augmented discrete event control
G_{nz}	Set of nonchoice rules of augmented discrete event control
R_f	Set of resources to accomplish rule set G_a
R_c	Set of available resources
R_a	Choice set in resource domain
$\mathbf{F}_{sd}$	Submatrix of $\mathbf{F}_{rd}$
$\mathbf{F}_{sr}$	Submatrix of $\mathbf{F}_r$
τ_k	Time instance of firing epoch k
D	Due date
$\mathbf{x}_{c0}$	State vector of critical subsystems of weighted p-timed Petri net
$\mathbf{c}$	Cost vector of weighted p-timed Petri net
f_{dp}	Sampling frequency of dynamic programming
$\tilde{c}_i^q$	A realization of random variable c_i
$\mathrm{E}\left[\cdot\right]$	Expected value of a random function
$\mathrm{Var}\left[\cdot\right]$	Variance of a random function
$H_2\left[\cdot\right]$	Quadratic Renyi entropy of a random function
$\sup\left\{\cdot\right\}$	Supremum of a set
$f_{c_j}\left(\tilde{c}_j\right)$	Probability density function of continuous random variable c_j

Chapter 1

Introduction

Improving energy efficiencies is the most important step for achieving security of energy supply, environmental protection, and economic growth. A large portion of global energy consumption and carbon dioxide (CO_2) emissions are attributable to manufacturing industries, especially the primary material industries such as chemicals and petrochemicals, iron and steel, cement, paper, and aluminium. While impressive improvements of energy efficiencies have already been achieved in the past two decades, energy consumption and CO_2 emissions in manufacturing industries could be still further reduced significantly, if effective energy-efficient technologies are to be applied.

1.1 Background

Climate change is an emerging challenge of our time. The scientific evidence of its occurrence, its derivation from human energy consumption, and its potentially devastating effects accumulate [1]. Sea levels have risen by 15–20 cm, on average, over the last century and this increase has accelerated over the last decade [2]. Oceans are warming and becoming more acidic, while the rate of ice-sheet loss is increasing. The Arctic provides a particularly clear illustration, with the area of ice covering the Arctic Ocean in the summer diminishing by half over the last 30 years to a record low level in 2012. There has also been an increase in the frequency and intensity of heat waves, resulting in more of the world being affected by droughts, harming agricultural production [3].

Global awareness of the phenomenon of climate change is increasing and political action is underway to try and tackle the underlying causes, both at national and international levels. Governments, based on the results of scientific research [4, 5], agreed at the United Nations Framework Convention on Climate Change Conference of the Parties in Cancun, Mexico in 2010 that the average global temperature increase, compared with pre-industrial levels, must be held below 2 degrees Celsius, and that means greenhouse-gas emissions must still be reduced significantly. This new global climate agreement will come into effect in 2020. But although overcoming the challenge of climate change will be a long-term endeavour, urgent actions are re-

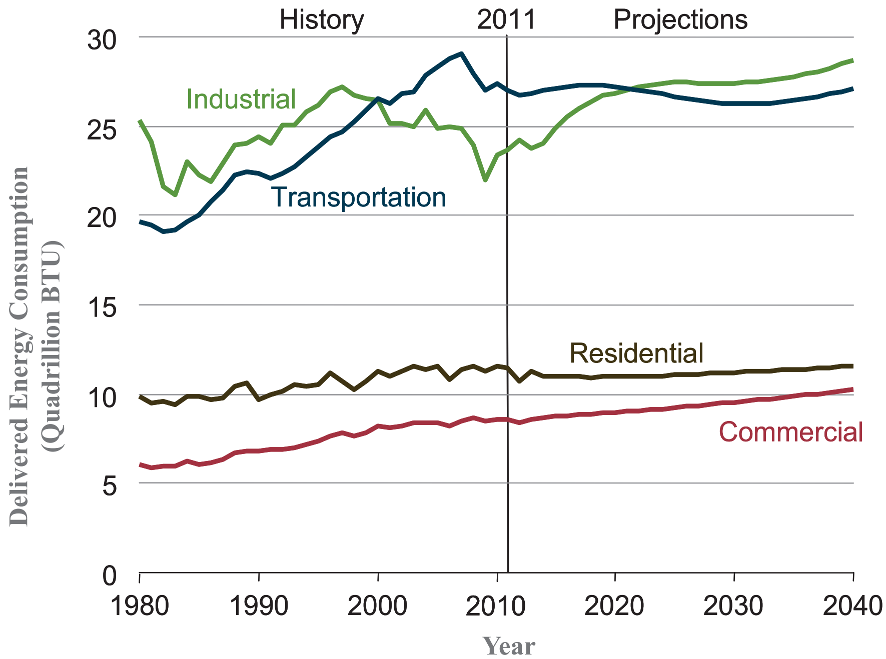


Figure 1.1: Delivered energy consumption by sector 1980–2040 [6].

quired, well before 2020, in order to keep open a realistic opportunity for an efficient and effective international agreement from that date.

1.1.1 Energy Consumption of Manufacturing Industries

Global CO_2 emissions from fossil-fuel combustion increased again in 2012, reaching a record high of 31.6 gigatonnes, according to some preliminary estimates [7]. Furthermore, under business-as-usual assumptions, the U.S. Energy Information Administration (EIA) projects worldwide energy consumption of primary sectors to be constantly increased in next twenty-five years as shown in Figure 1.1, as the global

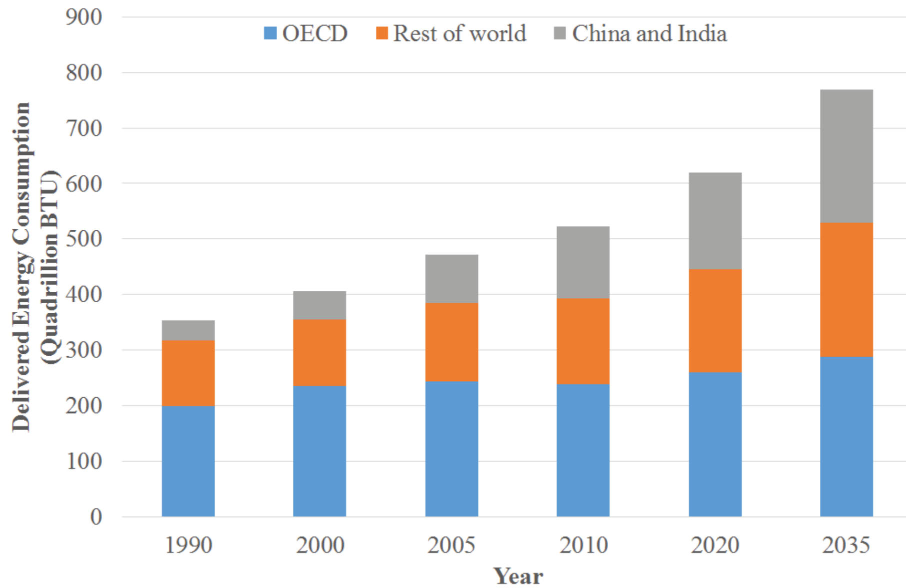


Figure 1.2: Global energy consumption 1990–2035 [8].

recovery from the 2008–2009 worldwide economic recession continues to advance [6].

Two nations were least affected by the recession are China and India. Strong economic growth leads China and India to more than double their combined energy demand by 2035, accounting for one-half of the world's energy growth as shown in Figure 1.2. EIA projects that China and India together will consume 31% of the world's energy in 2035, up from 21% in 2008. China, which surpassed the United States as the world's largest energy consumer in 2009, is the predominant driver of growing energy demand. By 2035, China's projected energy consumption is 68% higher than U.S. energy consumption. Global energy consumption grows 53% between 2008 and 2035, representing an average annual growth rate of 1.6%.

Among major national sectors including transportation, residential, and commercial, the industrial sector has been constantly responsible for the largest percentage of energy consumption as shown in Figure 1.3. The worldwide industry makes up diverse sub-sectors including manufacturing, agriculture, mining, and construction, *etc.* Of these sub-sectors, manufacturing is the most energy-intensive. Manufacturing's energy consumption is projected to grows from 191 quadrillion British thermal units (BTUs) in 2008 to 288 quadrillion BTUs in 2035 with the energy demand increasing by an average of 1.5% per year. The industrial sector experienced a significant reduction in energy usage in 2009 due to the global economic recession, which caused substantial cutbacks in manufacturing outputs demand. In the long term, national economic growth rates return to a constant increase and so does the industrial energy consumption.

The energy consumption of Singapore is overseen and regulated by Singapore Energy Market Authority (EMA), which is a statutory board under the Ministry of Trade and Industry. EMA's main goals are to ensure a reliable and secure energy supply, promote effective competition in the energy market, and develop a dynamic energy sector. Among major sectors in Singapore, industrial sector is the largest gas consumer, accounting for 79.9% of total gas consumption. For electricity consumption, industrial sector is also the second-largest consumer, accounting for 34% of total electricity consumption [9].

Over the past eight years or so, Singapore industrial sector's consumption of en-

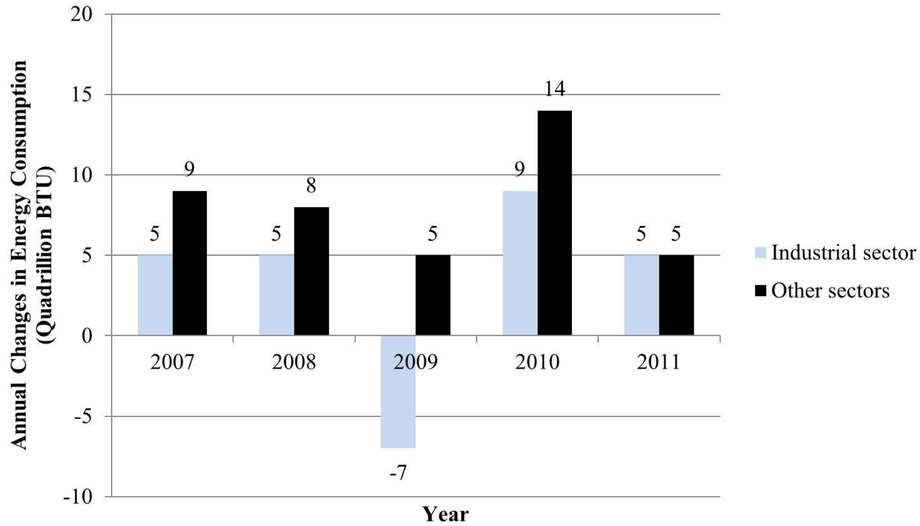


Figure 1.3: Annual changes in world industrial and all other end-use energy consumption 2007–2011.

ergy has increased by a whopping 27% [10]. Its share of total energy consumption is expected to rise further, especially with expansion of the energy-intensive petrochemical industries. Oil refining, petrochemicals, and wafer fabrication have the highest energy consumption. Apart from the oil refining and petrochemical subsectors for which electricity accounts for less than half of total energy costs, most manufacturing companies consume energy mainly in the form of electricity. For some industries, energy constitutes a small proportion of total operating costs but their absolute total energy costs are actually relatively high due to high production output. The energy is consumed for space cooling purposes and to drive various MPs. There is tremendous potential to save energy in industrial sector and increase economic competitiveness

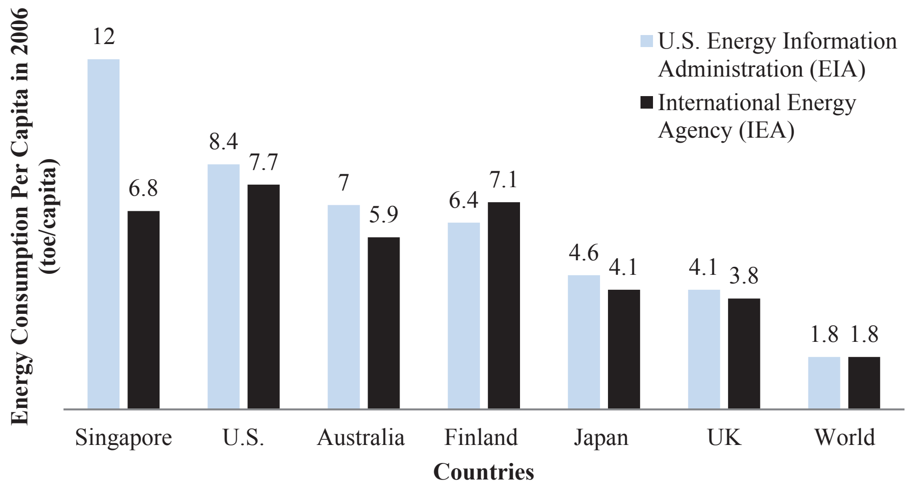


Figure 1.4: Energy consumption *per capita* for selective developed countries in 2006.

through improvements of energy efficiencies, but rising industrial energy efficiencies has not proven to be easy.

As compared other developed countries, Singapore is a highly energy-intensive country. The energy consumption *per capita* for selective developed countries in 2006 is reported in Figure 1.4 based on statistics from EIA and International Energy Agency (IEA) [6, 8], where Singapore is seen to have high energy consumption *per capita* according to both data sources.

1.1.2 Energy Saving Potentials through Energy-Efficient Technologies

Incrementally optimizing the systems in industrial facilities' operations is usually the most cost-effective way to improve energy efficiencies. This entails applying best practices and a progressive investment in equipment and technological upgrades. For example, an intelligent energy audit technology can quickly determine what systems within the plant use the most energy. Plant managers can then estimate the costs of these systems, determine the payback, and make the case for capital expenditures.

U.S. has the world's largest manufacturing economy, responsible for 18.2% of global manufactured products. To compete more effectively in the challenging manufacturing marketplace, the U.S. industrial sector continues to search for ways to become more productive. The reduction of energy presents significant opportunities for manufacturing industries to maximize efficiencies and productivity, cut expenses, create jobs, reduce emissions, and enhance competitiveness [11]. Energy-efficient technologies have always perceived as a key to energy saving capabilities [12–14]. Increased adoption of energy-efficient technologies is projected to reduce energy consumption by an additional 4.7 quadrillion BTU per year, which is almost 27% of the current energy consumption. As such, U.S. manufacturing industries aim to double their current energy efficiencies by 2020 [15].

Singapore's manufacturing industries had been significantly improving their energy efficiencies over the past years, and aim to reduce the energy intensity output by 35% as compared to 2005 levels. Energy-efficient technologies are now one of the key focuses of the Government to meet this target [9]. To encourage more industrial facilities to invest in energy-efficient equipment and technologies, Singapore government provides a grant for energy-efficient technologies to companies to help offset part of the investment cost. The grant was launched in November 2008 and is now co-administered by the National Environment Agency and the Economic Development Board of Singapore.

1.2 Literature Review on Energy-Efficient Technologies

Research literature is quickly adapting to this emerging green trend in green manufacturing industries, where novel energy-efficient methods have been frequently proposed in recent years [16]. The existing energy-efficient methods can be categorized into three main directions including [17]

1. energy policy, in which the governmental bodies set legislation, taxation, and penalties on energy consumption;
2. energy management such as energy audits and reporting, courses and training programs, and energy housekeeping, *etc.*; and
3. energy-efficient technologies, which directly improve manufacturing plants' en-

ergy efficiencies.

This section is intended not to provide a broad survey of general energy-efficient research and development (R&D), but to focus only on direction 3. Energy-efficient technologies are the most technical and directive approaches for the next generation of energy-efficient manufacturing. They involve multiple engineering disciplines, *e.g.*, chemical, mechanical, control and automation, electronic, and mechatronics, *etc.* Each technology is at a different point in the development or commercialization, indeed, many of them still need further R&D to evaluate costs and performances. In this chapter, energy-efficient technologies are reviewed according to four different approaching levels, namely systems, process, facility, and equipment.

1.2.1 Systems Level

At the systems level, energy-efficient technologies can be facilitated through the appropriate planning and scheduling of machines, tools, materials, people, and information to produce energy-efficient workflows and resource assignments. Planning is the procedure of selecting among different processing possibilities (for a specific product), each of these possibilities poses different advantages and limitations, these are, functions of both geometries and lots size of to-be-manufactured products; while scheduling is the procedure of assigning resources for specific instances to selected process plans, which is in fact an optimization process by which resources are allocated among parallel and sequential jobs.

Energy efficiencies were early adopted into computer-aided process planing (CAPP) by Sheng and colleagues [18,19], where a feature-based multi-objective model was proposed considering environmental metrics such as process energy, process time, and fluid coated on chips, *etc.* This model was further detailed in [20] based on micro-planning and macro-planning case studies of industrial cutting process. R&D on energy-efficient CAPP was continued in [21] to support green manufacturing, where optimization of energy consumption was considered as part of the planning process. Similar approaches can also be found in [22,23]. Altogether, these researches provided a basis for future R&D in energy-oriented and multi-objective CAPP combining both micro/macro-decisions with mathematical rigors.

Energy consumption was just recently synthesized into the FMS scheduling. The energy-efficient shop scheduling problems were studied by [24, 25], where multi-objective mixed-integer programming and preference vector ant colony system were employed for decision-making, respectively. The energy consumption reduction was investigated through effective scheduling of machine startup and shutdown, where machines were assumed to have Bernoulli reliability model [26]. The control strategy for a closed-loop flow shop was designed to coordinate running of the machines and motion of pallets to minimize energy consumption in idle machines [27]. The robotic manufacturing systems were considered in [28], where energy optimal trajectories were generated for a range of execution times for the individual operations based on only a single simulation.

1.2.2 Process Level

Manufacturing industries comprise of many distinct MPs such as grinding, milling, injection moulding, and stamping, *etc.* Each process includes a unique procedure of chemical or mechanical steps to aid in the manufacture of a product. Last decade has witnessed an overwhelming research interest in energy-efficient MPs.

Adjustment of cutting conditions was proven capable of improving energy efficiencies by numerical experiments [29]. This work was extended at the same time by [30] and [31], where calculation of optimal cutting parameters was mathematically proposed. Through an indirect method, a simulation-based technique was proposed to predict cutting forces that result in minimum energy consumption [32]. The energy consumption behavior of milling process was studied in [33]. Baker and McKenzie described practical methods for benchmarking the energy consumption of the industrial dryers, and suggested a number of techniques for implementing energy saving. [34]. Sun *et al.* studied theoretical minimum and actual specific energy consumption (SEC) of typical MPs [35]. The results showed that typical MPs had a theoretical minimum SEC of 6.74 Gigajoule/tonne (GJ/t) and an additional SEC of 19.32 GJ/t, which accounted for 25.88% and 74.12% of the actual SEC, respectively. Palamutcu investigated unit electric energy consumption of cotton textile processing stages using real-time msuring method [36]. Actual and estimated SEC values for electric energy were calculated in the cotton textile processing stages of spinning, warping-sizing, weaving, wet processing and clothing manufacturing. It was found that actual energy consumption per unit textile product is higher than the estimation of each involved textile processing stage.

1.2.3 Facility Level

Manufacturing facilities include various utilities which support manufacturing operations such as wireless sensor networks (WSNs), air conditioners (ACs), lighting and heating systems, and databases, *etc.* Of these facilities, ACs are biggest energy consumers. In recent years, traditional ACs have been modified in several ways to improve energy efficiencies. The solar-assisted ACs were frequently studied in [37–40], where fuzzy logics were applied as controllers. Other researches on modified energy-improved ACs include ground-assisted for direct evaporative cooling [41], split-type [42], and domestic hot water supply [43], *etc.*

Besides ACs, WSNs have also been frequently investigated. Current approaches to energy-efficient WSNs concentrate on optimal routing, planning, and forwarding methods. An energy-aware routing for real-time and reliable communication in wireless industrial sensor networks was studied in [44]. A scalable offline planning approach was discussed in [45]. A distributed topology control technique for low interference and energy efficiency in industrial WSNs was also proposed in [46].

1.2.4 Equipment Level

Manufacturing equipment such as machines, tools, and robots, *etc.,* are the primary components in manufacturing industries. The energy costs needed to operate these equipment throughout their useful life can easily exceed the original purchasing costs. Although, improvements in equipment design and optimization can improve the energy efficiencies, daily operations and machine maintenance play even more important roles in reducing environmental impacts and costs. Energy efficiencies and equipment reliability were shown to be closely correlated [47].

Mouzon and Yildirim reported several works on operational methods for sustainable production planning of manufacturing equipment [48–50]. Neugebauera and colleagues discussed the structure principles of machine tools [51], and proposed an energy-efficient tool designs using virtual reality simulation [52]. On the other hand, energy efficiencies were improved through product designs using a softcomputing techniques [53]. Lightweight component designs were discussed comprehensively, considering direct and indirect effects of mass reduction on energy consumption [54]. A self-optimization approach to energy-efficient equipment was also discussed in [55]. Other equipment that have been studied include press-brakes [56], injection moulding machines [57], milling machines [58], and translatory feed axes [59], *etc.*

A specific equipment that has drawn much attention from researchers is industrial motor. New optimal current control methods for energy-efficient synchronous motors were proposed in [60]. A novel energy-efficient single-phase induction motor with three series-connected windings and two capacitors was studied in [61]. Concepts and principles of energy-efficient motors were summarised in [62], and a comprehensive comparison between induction and permanent magnet motors was given in [63].

1.3 Motivation of Dissertation

Green manufacturing, a concept rarely heard before 2008, now occupies a prominent position in the discourse of international economic institutions. Expanding economic activity has been accompanied by growing concerns about climate change, energy security, and scarcity of natural resources. Although governmental regulations and managerial policies have helped to reduce energy consumption, they are typically not the most efficient way and do not offer enough incentives to innovate beyond end-of-pipe solutions. As such, energy-efficient technologies have been perceived as central for a global paradigm shift towards green manufacturing.

At the systems level, idle energy consumption was usually omitted or assumed to be trivial in energy-efficient operational control and scheduling of FMSs. This assumption may not be applicable in many realistic FMSs, where idle energy is observed to be significant as compared to total energy consumption [64]. In addition, most existing energy-efficient technologies for scheduling of FMSs presented in Section 1.2.1 often deal with deterministic manufacturing environments, where energy consumption is assumed to be deterministic and there is no uncertainty that would influence the established schedule. Real-world manufacturing is, however, dynamic and subjected to a wide range of uncertainties. Common sources of uncertainties in dynamic manufacturing environments have been classified into two categories, namely, resource-related uncertainties such as machine breakdown, machine degradation, tool wears, and job-related uncertainties such as rush jobs, job cancellation, stochastic processing time [65]. In general, the FMS scheduling problem is non-deterministic polynomial-time hard (NP-hard) in computational complexity theory, but consideration of uncertainties further aggravates its complexity, which impedes the effectiveness of a scheduling algorithm in handling uncertainties.

In addition, most existing energy-efficient technologies presented in Section 1.2 and the newly proposed technologies in this dissertation will need both energy and state data of MPs to make energy-efficient decisions. For example, the recently proposed technologies in [48, 50] continually require real-time information of both energy data and operational states (busy or idle) of involved resources. In other technologies, energy and state data of MPs are explored more thoroughly. An entire cycle of industrial milling process was divided into nine operational states (*e.g.,* run-up, spindle running, chipping milling, *etc.*) in a generic energy optimization model [33]. An industrial plastics welding process was also dismantled into six operational states (*e.g.,* start-up, stand-by, waiting, and processing, *etc.*) used for an energy-efficient planning methodology called EnergyBlocks [23].

Traditionally, energy monitoring and process state tracking are carried out separately using either extraneous number of sensors or accounting exercises, which are expensive for large-scale FMSs. To reduce the number of required sensors for energy monitoring, theoretical estimations of energy consumption for particular industrial processes were derived based on the processing parameters [66, 67]. Such theoretical estimations could be precise but are computationally intensive to implement as many process parameters must be known *a priori*.

1.4 Contributions and Organization

This dissertation concentrates on the study and development of energy-efficient technologies using operations research and artificial intelligence principles for the next generation of high-performance manufacturing industries. Technically, we shall focus on operational control and scheduling of FMSs with and without uncertainties in energy data for enhanced energy efficiencies, as well as time series analysis of energy data for real-time intelligent energy monitoring and process state tracking.

The original contributions of this dissertation are as follow:

1. Proposes a novel approach to reduce the number of required sensors in process state tracking by identifying the operational states of MPs by extracting useful information and features in energy data. Finite-state machines (FSMs) are used to model MPs, and a two-stage framework for online classification of real-time energy data in terms of MP operational states is proposed. To justify

our proposed framework, comparative experiments with an existing framework are evaluated on two industrial applications, an injection moulding system and a stamping system. Based on the obtained results, an energy data-driven decision support system (DSS) is designed to use real-time energy measurements and process operational states for effective decision-making, enabling high-performance manufacturing.

2. Proposes an integrated control and scheduling framework, which includes two modules: the ADEC and a novel MTME, to optimize the sum of energy cost and tardiness penalty in FMSs under power uncertainties due to machine degradation. Our proposed framework is applied to an industrial stamping system with power consumption uncertainties formulated using three different probability distributions to verify it effectiveness as compared the related work in current literature in terms of deviation from Pareto optimality and mean interrupted time.

3. Formulates a total energy optimization problem for FMSs using WTPN and proposes a new RG-based DP scheduling algorithm. The resulted schedules are obtained with low deviation from global optimality and within adequate computational time as compared to the related works in current literature.

4. Extends the deterministic total energy optimization problem with its robust counterpart to deal with uncertainties in energy measurements. A novel robustness measure is proposed, called Renyi ME criterion, using Renyi quadratic

entropy for searching the robust shortest path of WTPN RG. The effectiveness of Renyi ME criterion is compared with the related works in current literature in terms of computational complexity and deviation from global optimality.

The rest of the dissertation is organized as follow:

- Chapter 2 describes MPs and FMSs and introduces their mathematical modeling languages.
- Chapter 3 proposes a novel approach to identify the operational states of MPs by extracting useful information and features in energy data.
- Chapter 4 details the use of ADEC and MTME to minimize the sum of energy cost and tardiness penalty under power consumption uncertainties due to machine degradation.
- Chapter 5 explores the use of WTPN and DP to optimize total energy consumption in FMSs.
- Chapter 6 extends the total energy optimization problem presented Chapter 5 with its robust counterpart to deal with uncertainties in energy measurement process and proposes the Renyi ME criterion for robustness measure.
- Chapter 7 summarises the findings and results of this dissertation, and presents some possible future research directions.

Chapter 2

Descriptions and Modeling of Flexible Manufacturing Systems

Manufacturing is the backbone of any industrialized nation. Manufacturing staffs work with the various manufacturing processes (MPs) including materials being processed, tools and equipments for manufacturing different components and products, and process planning to efficiently meet production requirements. In addition, the operational skills of flexible manufacturing systems (FMSs) in term of scheduling machines, robots, conveyer belts, *etc.*, and routing parts from raw materials to finished products are also crucial. This chapter is used to introduce the basic descriptions of MPs and FMSs and their mathematical modeling languages.

2.1 Descriptions

A MP is the process followed in a manufacturing plant for converting semi-finished parts or raw materials into finished parts with application of different types of tools, equipments, and machines. Prior to executing a MP, careful process planning is often required [18,22]. This consists of selection of means of production (machines, cutting tools, presses, jigs, fixtures, measuring tools, *etc.*), establishing the efficient sequence of operation, determination of changes in form and dimension, or finish of the machine tools in addition to the specification of the actions of the operator. It includes the calculation of the machining time, as well as the required skill of the operator [20]. It also establishes an efficient sequence of manufacturing steps for minimizing material handling which ensures that the work will be done at the minimum cost and at maximum productivity. Examples of MPs may include machining, casting, forging, sheet metal forming, assembling, and heat treatment, *etc.*

The entire MP cycle can be often decomposed into a finite number of operational states, which are linked to the status of machine components [33]. During the MP cycle, the machine switches from one state to another at different instances in time. The energy consumption of a MP is therefore given as the sum of the consumption of individual components of the machine. This, in turn, is determined by the operational states, which define which components are active and thus consuming a certain amount of basic power, as well as the transitions of a MP executed by the machine.

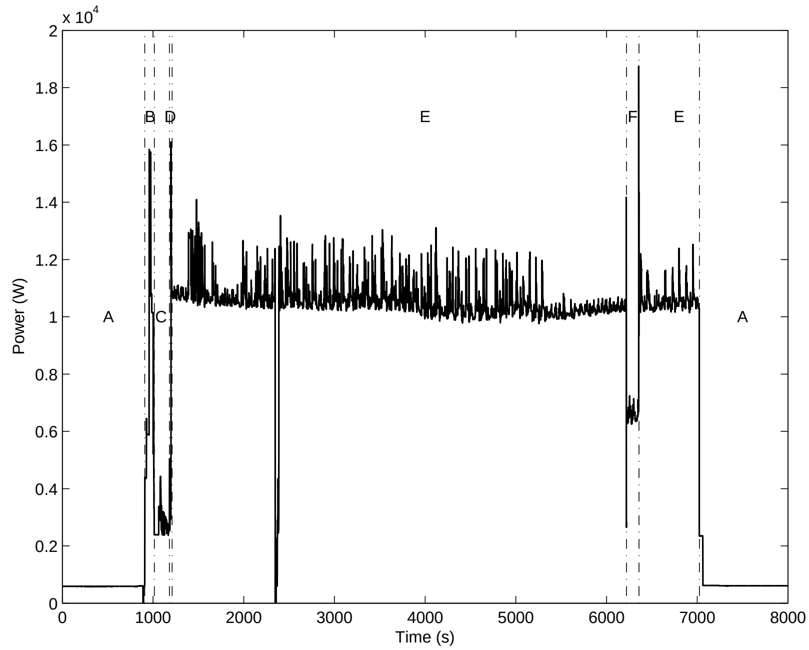


Figure 2.1: Power consumption profile of injection moulding process using Arburg A220 S 150–6 machine tool [68].

Example 2.1 To illustrate the state-based analyses of MPs, let us demonstrate the injection moulding process considered in [68]. Figure 2.1 shows a typical power consumption profile of Arburg A220 S 150–60 hydraulic injection moulding machine tool, which is used for a wide variety of applications and can be individually adapted for operation in conjunction with all familiar injection moulding techniques within a clamping force range from 125 to 5000 kilonewton. Therein, the actual input power at the main connection of the machine is plotted over time. As highlighted in Figure 2.1, the total energy intake for the execution of the milling operation is equal to the integral of the power over the entire operation of the machine. Darker areas of the graph indicate a fast oscillation of the power drain. Several operational states can be clearly observed during the execution of the injection moulding cycle including (A) *Switch off*, (B) *Warm up*, (C) *Idle*, (D) *Start up*, (E) *Moulding*, and (F) *Pump/heat*.

On the other hand, a manufacturing system usually consists of resources, each of them can perform one or more MPs, working together to produce finished parts. In an era of intensive competition, most manufacturing systems have migrated from conventional fixed-hardware sequential or batch production with dedicated workstations in the shop floor into an FMS. A typical FMS is characterized by the following four major components [69]:

1. a set of machines, robots, fixtures, or work stations,
2. an automated material handling system that allows flexible part routing,
3. distributed buffer storage sites where the parts may be temporarily placed during processing, and
4. a computer-based supervisory controller for monitoring the status of jobs and directing part routing and machine job selections.

This section covers the descriptions of a class of FMSs with shared resources and flexible part routes. Mathematical models for computer-based supervisory controller of FMSs are also detailed with insights from discrete-event analyses.

The standard assumptions that define the sort of discrete-part manufacturing systems are: (i) no preemption – Once assigned, a resource cannot be removed from a job until it is completed; (ii) mutual exclusion – A single resource can be used for only one job at a time; (iii) hold while waiting – A process holds the resources already allocated to it until it has all the resources required to perform a job; and (iv) no resource failures.

The class of FMSs, investigated herein, has the following properties [70]: a) each part type has a strictly defined sequence of jobs; b) each job in the system requires one and only one resource; c) there are choice jobs (the term choice job is used when the parts can be processed by alternative machines) and shared resources (the term shared resource is used when the resources can perform different jobs); d) resource allocation and part routing are flexible; and e) there are no assembly jobs. Such configuration can be encountered in many realistic manufacturing flowlines, job shops, and material handling systems, *etc.*

An FMS consists of $|R|$ types of resources, denoted by $R = \{r_i, i = 1, 2, \ldots, |R|\}$, to manufacture $|\Pi|$ types of parts, where $|\bullet|$ is a standard term to denote the cardinality of a set. Each resource can be a machine, a conveyor, a robotic arm, an automated guided vehicle, *etc.* In large-scale FMSs, r_i can denote a pool of similar resources. The capacity of r_i is denoted by $C(r_i)$, which indicates the maximal number of parts that r_i is able to hold simultaneously. Resources which can perform multiple jobs are called *shared* resources, while resources which can perform only one job are called non-shared resources.

The set of part types is denoted by $\Pi = \{\pi_q, q = 1, 2, \ldots, |\Pi|\}$, and $\varphi(\pi_q)$ is the number of type-π_q parts to be manufactured. Each π_q has a strictly defined sequence of jobs $\omega_q = v_1^q v_2^q \dots v_{|\omega_q|}^q$, where v_j^q is the j^{th} job in ω_q and $|\omega_q|$ is the length of ω_q. The set of jobs is denoted by $V = \left\{v_j^q, q = 1, 2, \dots, |\Pi|, j = 1, 2, \dots, |\omega_q|\right\}$. In an FMS with flexible part routing, choice jobs are ubiquitous. Therefore, V can be partitioned into two disjoint sets, $V = V_z \cup V_{nz}$, where V_z and V_{nz} denote the sets of choice and non-choice jobs, respectively. Let $R\left(v_j^q\right)$ be the set of resources which can perform v_j^q. Obviously, $\left|R\left(v_j^q\right)\right| > 1$ if $v_j^q \in V_z$ and $\left|R\left(v_j^q\right)\right| = 1$ if $v_j^q \in V_{nz}$. For each $v_j^q \in V_z$, assume that $v_{j-1}^q \in V_{nz}$ is a routing job and $r_i \in R(v_{j-1}^q)$ is a resource which routes parts. A routing resource is often a robotic arm, an automatic guided vehicle, and a conveyor, *etc.* The routing resources act as some kinds of crossroads where scheduling decisions regarding part routing are made. The role of routing resources is important in any kind of Petri net (PN) modelling of FMSs with choice jobs, as they significantly reduce the size of supervisory controllers [70]. For each π_q, ω_q is associated with two fictitious jobs v_{in}^q and v_{out}^q, called input buffer and output buffer, which represent the storage of raw and finished parts, respectively. v_{in}^q and v_{out}^q do not require any resources, thus $R\left(v_{in}^q\right) = R\left(v_{out}^q\right) = \emptyset$.

In an FMS where energy consumption is concerned, energy data are often logged and documented in some convenient forms. In this dissertation, the productive powers to manufacture π_q are archived in a $|R| \times |\omega_q|$ matrix $\mathbf{A}^q$, where element a_{ij}^q denotes the productive power of r_i to perform v_j^q. $a_{ij}^q = 0$ if r_i cannot perform v_j^q. The idle powers are archived in a $|R| \times 1$ vector $\mathbf{b}$, where element b_i denotes the idle power

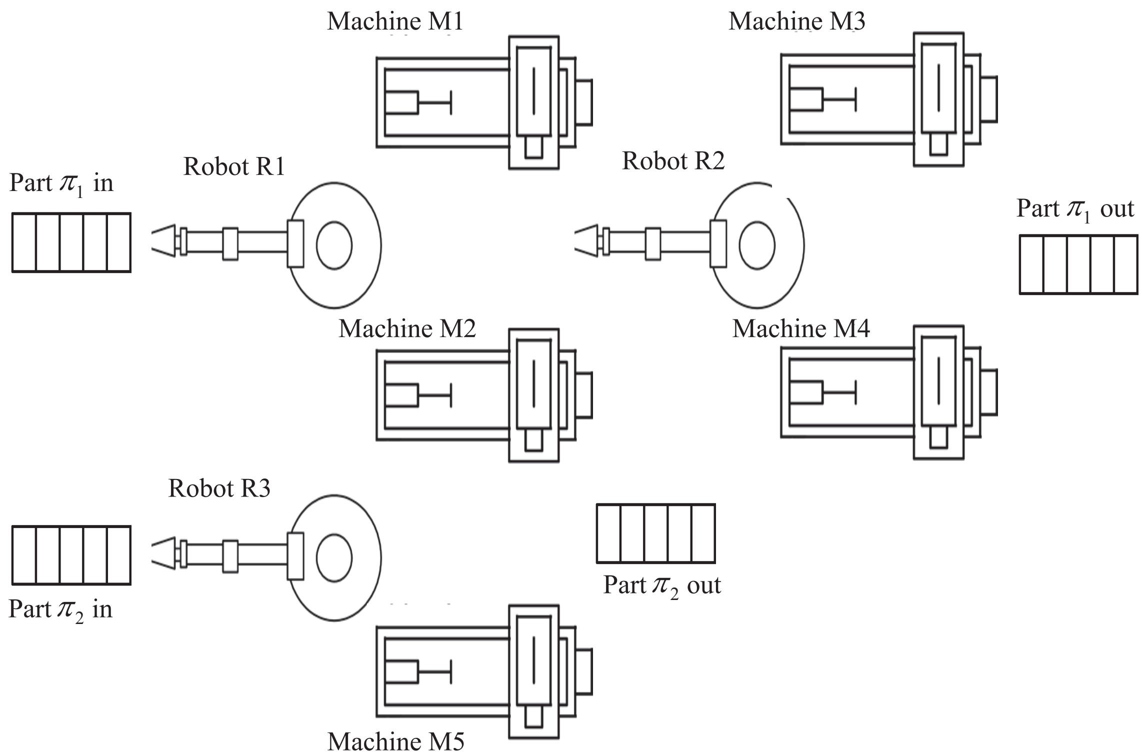


Figure 2.2: An example of FMSs with two part types, eight jobs, and eight resources including five machines and three material routing robots.

of r_i. In a $|R| \times |\omega_q|$ matrix $\mathbf{D}^q$, element d_{ij}^q denotes the processing time of r_i to perform v_j^q. $d_{ij}^q = 0$ if r_i cannot perform v_j^q. $\left\{a_{ij}^q, b_i, d_{ij}^q\right\} \subseteq \mathbb{R}^+$, where $\mathbb{R}^+$ denotes the set of nonnegative real numbers.

Example 2.2 An example of FMSs is shown in Figure 2.2. There are two part types π_1 and π_2, a total of eight resources including five machines denoted by $\mathrm{M1} - \mathrm{M5}$ and three material routing robots denoted by $\mathrm{R1} - \mathrm{R3}$. Part π_1 has a job sequence $\omega_1 = v_{in}^1 v_1^1 v_2^1 v_3^1 v_4^1 v_{out}^1$, where v_2^1 and v_4^1 are stamping jobs, and v_1^1 and v_3^1

are routing jobs. This implies that both v_2^1 and v_4^1 are choice jobs. Part π_2 has a job sequence $\omega_2 = v_{in}^2 v_1^2 v_2^2 v_{out}^2$, where v_2^2 is a choice stamping job, while v_1^2 is a routing job.

For each choice job, there is an associated routing resource which routes parts. For instance, choice job v_2^1 can be processed by M1 or M2, and is associated with routing resource R1; choice job v_4^1 can be processed by M3 or M4, and uses routing resource R2 for part routing; and choice job v_2^2 can be processed by M2 or M5, and is routed by routing resource R3, *etc.* All routing resources are non-shared. All stamping machines are non-shared resources, except for M2. The productive power matrices $\mathbf{A}^q$ (kW), the idle power vector $\mathbf{b}$ (kW), and the processing time matrices $\mathbf{D}^q$ (kW) of FMS example can be constructed as shown in (2.1)–(2.5).

$$(\mathbf{A}^1)^T = \begin{array}{c} \\ v_1^1 \\ v_2^1 \\ v_3^1 \\ v_4^1 \end{array} \begin{array}{c} \begin{array}{cccccccc} \mathrm{M1} & \mathrm{M2} & \mathrm{M3} & \mathrm{M4} & \mathrm{M5} & \mathrm{R1} & \mathrm{R2} & \mathrm{R3} \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 2.5 & 0 & 0 \\ 3.2 & 4.6 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2.5 & 0 \\ 0 & 0 & 3.7 & 5.8 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array}, \tag{2.1}$$

$$(\mathbf{D}^1)^T = \begin{array}{c} \\ v_1^1 \\ v_2^1 \\ v_3^1 \\ v_4^1 \end{array} \begin{array}{c} \begin{array}{cccccccc} \mathrm{M1} & \mathrm{M2} & \mathrm{M3} & \mathrm{M4} & \mathrm{M5} & \mathrm{R1} & \mathrm{R2} & \mathrm{R3} \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 3.5 & 0 & 0 \\ 2.6 & 5.8 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 3.5 & 0 \\ 0 & 0 & 4.2 & 6.1 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array}, \tag{2.2}$$

$$
(\mathbf{A}^2)^T = \begin{array}{c} \\ v_1^2 \\ v_2^2 \end{array} \begin{array}{c} \begin{array}{cccccccc} \mathrm{M1} & \mathrm{M2} & \mathrm{M3} & \mathrm{M4} & \mathrm{M5} & \mathrm{R1} & \mathrm{R2} & \mathrm{R3} \end{array} \\ \left[\begin{array}{cccccccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2.5 \\ 0 & 3.4 & 0 & 0 & 5.2 & 0 & 0 & 0 \end{array} \right] \end{array}, \tag{2.3}
$$

$$
(\mathbf{D}^2)^T = \begin{array}{c} \\ v_1^2 \\ v_2^2 \end{array} \begin{array}{c} \begin{array}{cccccccc} \mathrm{M1} & \mathrm{M2} & \mathrm{M3} & \mathrm{M4} & \mathrm{M5} & \mathrm{R1} & \mathrm{R2} & \mathrm{R3} \end{array} \\ \left[\begin{array}{cccccccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3.5 \\ 0 & 2.7 & 0 & 0 & 3.6 & 0 & 0 & 0 \end{array} \right] \end{array}, \tag{2.4}
$$

$$
\mathbf{b}^T = \begin{array}{c} \begin{array}{cccccccc} \mathrm{M1} & \mathrm{M2} & \mathrm{M3} & \mathrm{M4} & \mathrm{M5} & \mathrm{R1} & \mathrm{R2} & \mathrm{R3} \end{array} \\ \left[\begin{array}{cccccccc} 2.1 & 1.2 & 1.4 & 1.7 & 1.8 & 0.8 & 0.8 & 0.8 \end{array} \right] \end{array}. \tag{2.5}
$$

The productive power and processing time of M2 to perform job v_2^1 are 4.6 kW and 5.8 s, respectively, while M1 only requires 3.2 kW and 2.6 s. Similar observation can be made for other choice jobs. The idle powers of machines are also not identical. The differences in productive powers and processing times required to perform the same choice job by different machines are central to the study of FMS scheduling to be presented in Chapters 4–6.

2.2 Finite-State Machine Models of Manufacturing Processes

An FSM is a mathematical model commonly used to represent discrete-event and logicial systems [71]. An FSM is generally represented mathematically by a quintuple $(\Sigma_F, S_F, s_{F0}, \delta_F, F_F)$, where:

- Σ_F is the input alphabet in a finite and non-empty set of symbols;
- S_F is a finite and non-empty set of states;
- s_{F0} is the initial state of an element in S_F;
- δ_F is the state-transition function $\delta : S_F \times \Sigma_F \rightarrow S_F$; and
- F_F is the set of final states and a (possibly empty) subset of S_F.

An FSM is also represented graphically using a directed graph with states, transitions, and actions (or triggers). From the observations of the power consumption profiles in Figure 2.1, an FSM energy consumption model in the form of diagraph can be built as shown in Figure 2.3.

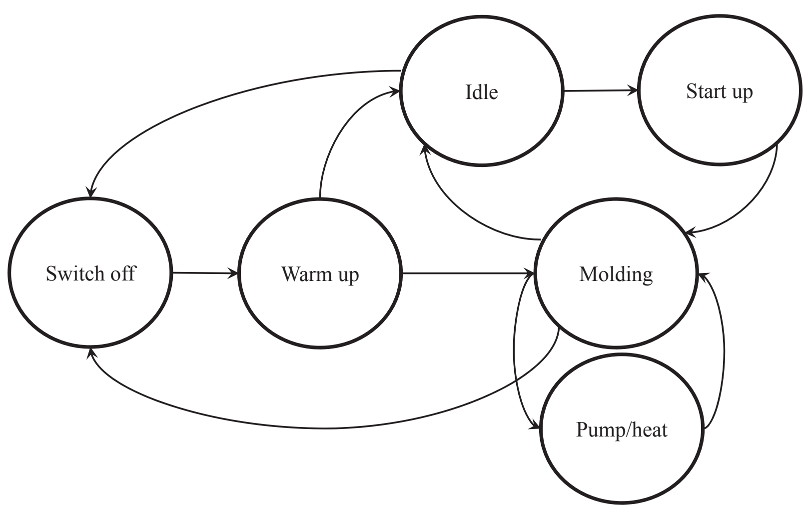


Figure 2.3: FSM models of industrial injection moulding process.

Each operation state is defined according to the functionality of the machine. The states are linked by transitions which define the possible operation states which a machine can switch from its present state. Transitions are represented by unidirectional arrows. Certain conditions may have to be satisfied for a transition to be activated, *e.g.*, the production state requires raw material to be input to the machine.

2.3 Weighted P-Timed Petri Net Models of Flexible Manufacturing Systems

This section briefs the notations of PN and introduces the WTPN models of FMSs for energy-optimal scheduling. For analysis of structural properties of PN, interested readers are referred to [70, 72] for more details.

2.3.1 Petri Nets

A PN is a bipartite (having two sorts of nodes) digraph denoted by $\chi = (P, T, I, O)$, where P and T are finite and disjoint sets. P is the set of places, T is the set of transitions, I is the set of (input) arcs from places to transitions, and O is the set of (output) arcs from transitions to places. In a PN, given a node $\alpha \in P \cup T$ (α is either a place or a transition), define by ${}^\bullet\alpha$, called pre-set of α, the set of nodes that have arcs to α, and by $\alpha^\bullet$, called post-set of α, the set of nodes that have arcs from α.

It is common in PN theory to represent I and O as matrices. Thus, element i_{ij} of the Boolean input incidence matrix $\mathbf{I}$ is equal to 1 if place p_j is an input to transition t_i. Similarly, the output incidence matrix $\mathbf{O}$ is defined. Then, the PN incidence matrix is defined as

$$\mathbf{W} = \mathbf{O} - \mathbf{I}. \tag{2.6}$$

A column vector $\mathbf{x}$ indexed by P, *i.e.,*

$$\mathbf{x}^T = \begin{bmatrix} x_1 & x_2 & \dots & x_{|P|} \end{bmatrix}, \tag{2.7}$$

is called the PN state or marking vector. A column vector $\mathbf{u}$ indexed by T, *i.e.,*

$$\mathbf{u}^T = \begin{bmatrix} u_1 & u_2 & \dots & u_{|T|} \end{bmatrix}, \tag{2.8}$$

is called the PN control or firing vector.

Definition 2.1 (Marking) Given a PN, the PN state (or marking) is the number of tokens in each place. Given a place $p_j \in P$, the state of p_j is represented by an element x_i of $\mathbf{x}$, denoted by $x_i = M(p_j)$. Thus, $\mathbf{x}$ is a state vector of the individual places. If a PN is analyzed in some domains, *e.g.,* discrete-event domain, it is common to add a subscript to $\mathbf{x}$, *i.e.,* $\mathbf{x}_k$ and its elements x_{ik} denote the PN state at event k. A PN χ with an initial marking $\mathbf{x}_0$ is called a marked PN, denoted by $(\chi, \mathbf{x}_0)$.

Definition 2.2 (Firing) $\mathbf{u}_k$ denotes which transitions are fired at k, where element $u_j = n_j$ if the j^{th} transition is fired n_j times. In terms of the PN incidence matrix, one can write the PN state equation as

$$\mathbf{x}_k = \mathbf{x}_{k-1} + \mathbf{W}^T\mathbf{u}_k, \tag{2.9}$$

which means a transition is said to be enabled and can be fired, if all its input places are marked. Firing a transition once removes one token from each of its input places and adds one token to each of its output places.

2.3.2 Weighted P-Timed Petri Nets

To improve the description ability of PNs for formulation of the total energy optimization problem, the PNs are extended such that each $p \in P$ is associated with a pair $\{c, h\} \subseteq \mathbb{R}^+$ written by $p\langle c, h\rangle$, where h denotes the minimal token sojourn time and c is the sojourn cost per time unit. This means tokens are forced to spend at least h time units in p, immediately after their arrival, with the incurred cost is c per time unit.

In our modelling, the places represent resources and jobs, and the transitions represent decisions or rules for the starting and completion of jobs, which also involve allocation and release of resources. In particular, $P = P_V \cup P_R \cup P_{in} \cup P_{out}$, with the places in P_R, P_V, P_{in}, and P_{out}, representing the availability of resources, jobs (*i.e.*, jobs on parts carried out by the resources), input buffers, and output buffers, respectively.

A $v_j^q \in V_{nz}$ with $R\left(v_j^q\right) = r_i$ is represented by a place $p_{qj}\langle c_{qj}, h_{qj}\rangle \in P_V$, where $c_{qj} = a_{ij}^q$ and $h_{qj} = d_{ij}^q$. A $v_j^q \in V_z$ with $R\left(v_j^q\right) = \{r_i, r_{i'}, \ldots\}$ is represented by a set of places

$$\{p_{qji}\langle c_{qji}, h_{qji}\rangle, p_{qji'}\langle c_{qji'}, h_{qji'}\rangle, \ldots\} \subset P_V, \tag{2.10}$$

where $c_{qji} = a^q_{ij}$, $c_{qji'} = a^q_{i'j}$, $h_{qji} = d^q_{ij}$, and $h_{qji'} = d^q_{i'j}$.

Resource places always occur off part-paths. A resource r_i is represented by a place $p_i \langle c_i, h_i \rangle \in P_R$, where $c_i = b_i$ and $h_i = 0$. It is worth noting that h_i is the release time of resource r_i, which is assumed to be zero in this dissertation. As such, it is convenient to simplify $p_i \langle c_i, h_i \rangle$ to $p_i \langle c_i \rangle$ for all $p_i \in P_R$. For input and output buffers, places $p^q_{in} \in P_{in}$ and $p^q_{out} \in P_{out}$ are used to represent v^q_{in} and v^q_{out}, respectively, which have neither minimal token sojourn time nor the sojourn cost per time unit. Since transitions are not important in our analysis, they are simply labeled in numerical order according to their order of appearance. Usually, index i (j, q) is replaced with job or resource notation, *e.g.*, p_{M1} stands for the place that corresponds to resource M1.

An available resource or an ongoing job is indicated by tokens in the respective places. It is assumed that places in P_{in} are always marked (*i.e.*, there is always a part ready to enter) and those in P_{out} are always empty (*i.e.*, finished product is pulled out immediately). The marking $\mathbf{x} : P \to \mathbb{Z}^+$, with $\mathbb{Z}^+$ as the set of nonnegative integers, gives the distribution of tokens. $\{\chi, \mathbf{x}\}$ denotes a marked PN. For the FMS shown in Example 2.2, the WTPN models can be constructed as shown in Figure 2.4.

For the analyses of WTPN models of FMSs, the following definitions should be detailed.

Definition 2.3 (Initial State) The initial state of the WTPN models of FMSs is

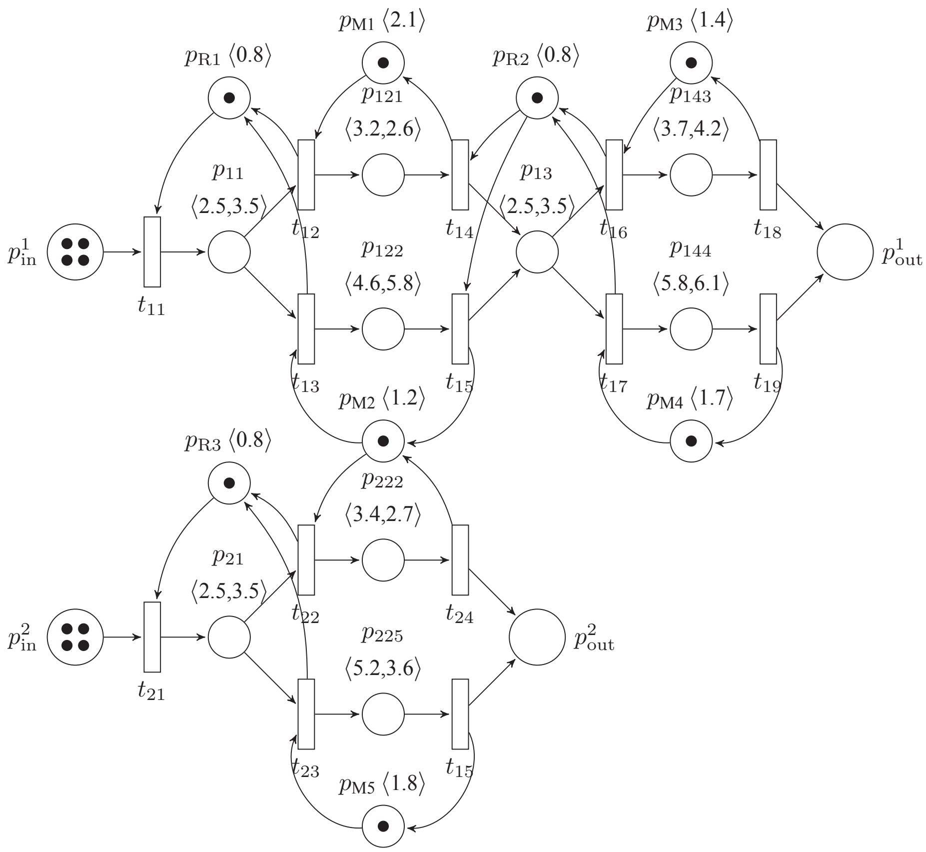


Figure 2.4: WTPN models of FMS example.

defined by

$$x_{i0} = \begin{cases} C(r_i) \text{ if } x_i = M\left(p_i\right), p_i \in P_R; \\ \varphi\left(\pi_q\right) \text{ if } x_i = M\left(p_{in}^q\right), p_{in}^q \in P_{in}; and \\ 0 \text{ otherwise,} \end{cases} \tag{2.11}$$

which means tokens only distribute in the input and resource places initially.

Definition 2.4 (Final State) The final state of the WTPN models of FMSs is defined as

$$x_{i|K|} = \begin{cases} C(r_i) \text{ if } x_i = M\left(p_i\right), p_i \in P_R; \\ \varphi\left(\pi_q\right) \text{ if } x_i = M\left(p_{out}^q\right), p_{out}^q \in P_{out}; and \\ 0 \text{ otherwise,} \end{cases} \tag{2.12}$$

which means tokens only distribute in the output buffer and resource places finally. $|K|$ denotes the total number of firing epoches.

Definition 2.5 (Split Places and Scheduling) In a PN, $|{}^{\bullet}t| = |t^{\bullet}| = 1, \forall t \in T$. When $p \in P$, $|p^{\bullet}| > 1$, p is called a split place. If $p \in P_R$ is a split place, then p represents a shared resource. If $p \in P_J$ is a split place, then p represents a routing job and the next job is a choice job. In an FMS with shared resources and flexible part routes, split places are ubiquitous, which lead to simultaneous firings of their output transitions.

Obviously, these simultaneous firings are not possible in reality, and they are central to the study of resource allocation (for shared resources) and part routing (for choice jobs) considered in this disseration. There are two ways to inhibit the illegal simultaneous transition firings. One is to add additional arcs and places which are called control places. This method was used in [70,72]. Another way is to control the firing timing of transitions with timing constraints [73]. Herein, the latter is adopted as it retains the brevity of the WTPN models and it is sufficient for our scheduling purpose.

2.4 Augmented Discrete Event Control Models of Flexible Manufacturing Systems

The ADEC has been proposed recently [74, 75], proving to be very efficient in modeling and controlling the large-scale discrete-event dynamics of typical manufacturing systems. In particular, it reduces the model complexity when modelling large-scale FMSs as compared to the traditional conjunctive supervisory tools, such as the discrete event control (DEC) [16, 70], and PN [76].

Let us first consider an FMS with part type π_q is characterized a job sequence ω_q properly predefined and a set of available resources R. It is convenient to describe the production flow of π_q using a finite set of linguistic IF-THEN rules denoted by G^q, and $\cup_{q=1}^{|\Pi|} G^q = X$. Each rule $\mathrm{g}_i^q \in G^q$ has the form:

IF (job 1 is finished **AND** job 2 is finished **AND**...)

AND (resource 1 is free **AND** resource 2 is free **AND**...)

AND (resource 3 is free **OR** resource 4 is free **OR**...)

THEN (start job 3 **AND** start job 4 **AND**...)

AND (release resource 5).

In the IF-part, the sets of preceding jobs, required resources, and part inputs needed to activate each rule are predefined. The THEN-part of each rule specifies the consequent jobs to be performed and the part outputs in the next dispatching epoch. As compared the conjunctive supervisory tools such as the DEC [16, 70] and PN [76], the ADEC contains additional add-on disjunctive rule bases (the **OR** operators) in the IF-part.

The ubiquity of choice jobs has a serious impact on the conjunctive supervisory tools. Specifically, the starting of a choice job which can be processed by p alterative resources must be described by p conjunctive IF-THEN rules [70]. This exhibits an incompetency, which is called the *rule explosion*, to model a large-scale FMSs which possesses a large number of choice jobs. This will be expatiated further in Section 4.3.2. In the ADEC, the rule explosion is overcome by proposing a novel add-on disjunctive reasoning into the rule bases, where the IF-part now contains **OR** operators. As such, the starting of a choice job is described by one and only one rule regardless of the number of disjunctive resources [74, 75].

In the ADEC models, the system sets of rules G, jobs V, and resources R are represented in a compact form using Boolean matrices and vectors. The following Boolean vectors are defined: a job vector $\mathbf{v}$, a resource vector $\mathbf{r}$, and a rule vector $\mathbf{g}$ that represent the sets of jobs, resources, and rules, respectively, corresponding to their "1" elements. The set represented by $\mathbf{a}$ (for $\mathbf{a}$ be $\mathbf{v}$, $\mathbf{r}$, or $\mathbf{g}$) is called the *support* of $\mathbf{a}$, denoted by $supp(\mathbf{a})$; *e.g.,* given $\mathbf{v} = \begin{bmatrix} v_1 & v_2 & \dots & v_q \end{bmatrix}^T$, $v_j = 1$ if and only if $v_j \in supp(\mathbf{v})$.

2.4.1 Matrices and Vectors

Let us first focus on the ADEC matrices and vectors of a single part type π_q, and then obtain the global ADEC models of FMSs with multiple part types. To map the set of preceding jobs to the set of rules, job sequence matrix $\mathbf{F}_v^q$ is defined such that element $f_{ij}^{vq} = 1$ if the completion of job v_j^q is required to activate rule g_i^q. Analogously, job start matrix $\mathbf{S}_v^q$ has element $s_{ij}^{vq} = 1$ such that job v_i^q is started if rule g_j^q is activated. To map the set of conjunctive resources to the set of rules, conjunctive resource assignment matrix $\mathbf{F}_r^q$ is defined such that element $f_{ij}^{rq} = 1$ if the availability of resource r_j is required to activate rule g_i^q. $\mathbf{F}_u^q$ is the input matrix which maps the set of input parts to the set of rules, having element $f_{ij}^{uq} = 1$ if the presence of input u_j^q is required to activate rule g_i^q. Output matrix $\mathbf{S}_y^q$ has the (i, j) element set to "1" if output y_i^q is released if rule g_j^q is activated.

The rule set of G^q is represented by vector $\mathbf{g}^q$ having element g_i^q stand for rule g_i^q. If all antecedences (IF part) required for rule g_i^q are met, then $g_i^q = 1$ (true). $\mathbf{v}_c^q$ is the job completed vector having element $v_j^{cq} = 1$ if job v_j^q is completed. $\mathbf{r}_c^q$ is the resource available vector having element $r_j^{cq} = 1$ if resource r_j is available. $\mathbf{u}^q$ is the input vector having element $u_j^q = 1$ if part input u_j^q occurs. Entries of "1" in vectors $\mathbf{v}_s^q$ denote the starting jobs and in vectors $\mathbf{y}^q$ imply that finished parts are out.

Deadlock is avoided by the ADEC using the deadlock resolution matrix $\mathbf{F}_{ud}^q$ and vector $\mathbf{u}_d^q$ [70, 72]. Matrix $\mathbf{F}_{ud}^q$ has as many columns as the number of jobs performed by shared resources, *i.e.*, the number of columns of $\mathbf{F}_r^q$ having multiple "1s". Element $f_{ij}^{udq} = 1$ if job v_j^q is a preceding job needed to activate rule g_i^q. Then, element $u_j^{dq} = 1$ determines the inhibition of logic state g_i^q (whether rule g_i^q can be activated). Depending on the way one selects the conflict resolution strategy to generate vector $\mathbf{u}_d^q$, deadlock can be avoided. On the other hand, possible assignments of available disjunctive resources to choice jobs are captured using the disjunctive resource assignment matrix $\mathbf{F}_{rd}^q$, which has entry $f_{ij}^{rdq} = 1$ if resource r_j can accomplish rule g_i^q. $\mathbf{F}_{rd}^q$ essentially captures information about which available resources can be used for each rule, such that only one of the possible resources listed in row i of $\mathbf{F}_{rd}^q$ is required to activate rule g_i^q. As such, $\mathbf{F}_{rd}^q$ maps the set of resources R to the set of rules G.

Table 2.1: Part Type π_1 of FMS Example–Rule Bases

Rule	Notation	Description
Rule 1	g_1^1	**IF** v_{in}^1 *is ready* **AND** R1 *is free* **THEN** *start* v_1^1
Rule 2	g_2^1	**IF** v_1^1 *is done* **AND** (M1 *is free* **OR** M2 *is free*) **THEN** *start* v_2^1
Rule 3	g_3^1	**IF** v_2^1 *is done* **AND** (R2 *is free* **OR THEN** *start* v_3^1
Rule 4	g_4^1	**IF** v_3^1 *is done* **AND** (M3 *is free* **OR** M4 *is free*) **THEN** *start* v_4^1
Rule 6	g_5^1	**IF** v_4^1 *is done* **THEN** *release* v_{out}^1

Table 2.2: Part Type π_2 of FMS Example–Rule Bases

Rule	Notation	Description
Rule 1	g_1^2	**IF** v_{in}^2 *is ready* **AND** R3 *is free* **THEN** *start* v_1^2
Rule 2	g_2^2	**IF** v_1^2 *is done* **AND** (M2 *is free* **OR** M5 *is free*) **THEN** *start* v_2^2
Rule 3	g_3^2	**IF** v_2^2 *is done* **THEN** *release* v_{out}^2

Now, let consider an FMS with several part types are prescribed, with part type π_q having its own job sequence's ordering given by $\mathbf{F}_v^q$, its required resources is given by $\mathbf{F}_r^q$ and $\mathbf{F}_{rd}^q$, *etc*. The global matrices $\mathbf{F}_v$, $\mathbf{F}_r$, and $\mathbf{F}_{rd}$, *etc*., of the FMS are then given by

$$\mathbf{F}_v = \begin{bmatrix} \mathbf{F}_v^1 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_v^2 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{F}_v^{|\Pi|} \end{bmatrix}, \mathbf{F}_r = \begin{bmatrix} \mathbf{F}_r^1 \\ \mathbf{F}_r^2 \\ \vdots \\ \mathbf{F}_r^{|\Pi|} \end{bmatrix}, \mathbf{F}_r = \begin{bmatrix} \mathbf{F}_{rd}^1 \\ \mathbf{F}_{rd}^2 \\ \vdots \\ \mathbf{F}_{rd}^{|\Pi|} \end{bmatrix},$$
$$\mathbf{F}_{ud} = \begin{bmatrix} \mathbf{F}_{ud}^1 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_{ud}^2 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{F}_{ud}^{|\Pi|} \end{bmatrix}, \ \mathbf{F}_u = \begin{bmatrix} \mathbf{F}_u^1 \\ \mathbf{F}_u^2 \\ \vdots \\ \mathbf{F}_u^{|\Pi|} \end{bmatrix}, \tag{2.13}$$

and similarly matrices $\mathbf{S}_v$ and $\mathbf{S}_y$ can be derived. $\mathbf{0}$ denotes a null matrix. The global vectors $\mathbf{g}$, $\mathbf{v}_c$, $\mathbf{r}_c$, and $\mathbf{u}$, *etc*., of the FMS are given by

$$\mathbf{g} = \begin{bmatrix} \mathbf{g}^1 \\ \mathbf{g}^2 \\ \vdots \\ \mathbf{g}^{|\Pi|} \end{bmatrix}, \mathbf{v}_c = \begin{bmatrix} \mathbf{v}_c^1 \\ \mathbf{v}_c^2 \\ \vdots \\ \mathbf{v}_c^{|\Pi|} \end{bmatrix}, \mathbf{r}_c = \begin{bmatrix} \mathbf{r}_c^1 \\ \mathbf{r}_c^2 \\ \vdots \\ \mathbf{r}_c^{|\Pi|} \end{bmatrix}, \tag{2.14}$$

and similarly vectors $\mathbf{u}$, $\mathbf{u}_d$, $\mathbf{v}_s$, $\mathbf{y}$ can be derived. It is worth noting that the job sequences of different part types are independent, each using its own jobs, so that $\mathbf{F}_v$ is block diagonal. However, all the job sequences use the same pool of resources available in the FMS, and so have commensurate columns of their resource assignment matrices $\mathbf{F}_r$ and $\mathbf{F}_{rd}$.

For the FMS shown in Example 2.2, the ADEC models's rule bases of part types π_1 and π_2 can be constructed as shown in Tables 2.1 and 2.2, respectively. The rule bases in Tables 2.1 and 2.2 are now represented by means of ADEC matrices $\mathbf{F}_v^q$, $\mathbf{F}_r^q$, $\mathbf{F}_{rd}^q$, $\mathbf{F}_u^q$ as shown in (2.15)–(2.16) and (2.17)–(2.18), respectively. It is noted that the contents of matrices $\mathbf{S}_r^q$ and $\mathbf{S}_y^q$ are omitted for brevity.

$$\mathbf{F}_v^1 = \begin{array}{c} \\ \mathrm{g}_1^1 \\ \mathrm{g}_2^1 \\ \mathrm{g}_3^1 \\ \mathrm{g}_4^1 \\ \mathrm{g}_5^1 \end{array} \begin{array}{c} \begin{array}{cccc} v_1^1 & v_2^1 & v_3^1 & v_4^1 \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{array}, \quad \mathbf{F}_u^1 = \begin{array}{c} \\ \mathrm{g}_1^1 \\ \mathrm{g}_2^1 \\ \mathrm{g}_3^1 \\ \mathrm{g}_4^1 \\ \mathrm{g}_5^1 \end{array} \begin{array}{c} u_1^1 \\ \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \end{array}, \tag{2.15}$$

$$\mathbf{F}_r^1 = \begin{array}{c} \\ \mathrm{g}_1^1 \\ \mathrm{g}_2^1 \\ \mathrm{g}_3^1 \\ \mathrm{g}_4^1 \\ \mathrm{g}_5^1 \end{array} \begin{array}{c} \mathrm{M1M2M3M4M5R1R2R3} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array}, \quad \mathbf{F}_{rd}^1 = \begin{array}{c} \\ \mathrm{g}_1^1 \\ \mathrm{g}_2^1 \\ \mathrm{g}_3^1 \\ \mathrm{g}_4^1 \\ \mathrm{g}_5^1 \end{array} \begin{array}{c} \mathrm{M1M2M3M4M5R1R2R3} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array}, \tag{2.16}$$

$$\mathbf{F}_v^2 = \begin{array}{c} \\ \mathrm{g}_1^1 \\ \mathrm{g}_2^1 \\ \mathrm{g}_3^1 \end{array} \begin{array}{c} \begin{array}{cc} v_1^1 & v_2^1 \end{array} \\ \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \end{array}, \quad \mathbf{F}_u^2 = \begin{array}{c} \\ \mathrm{g}_1^1 \\ \mathrm{g}_2^1 \\ \mathrm{g}_3^1 \end{array} \begin{array}{c} u_1^1 \\ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \end{array}, \tag{2.17}$$

$$\mathbf{F}_r^2 = \begin{array}{c} \\ \mathrm{g}_1^1 \\ \mathrm{g}_2^1 \\ \mathrm{g}_3^1 \end{array} \begin{array}{c} \mathrm{M1M2M3M4M5R1R2R3} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array}, \quad \mathbf{F}_{rd}^2 = \begin{array}{c} \\ \mathrm{g}_1^1 \\ \mathrm{g}_2^1 \\ \mathrm{g}_3^1 \end{array} \begin{array}{c} \mathrm{M1M2M3M4M5R1R2R3} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array}. \tag{2.18}$$

2.4.2 Logical State Equation

At each dispatching epoch, the ADEC receives the vectors $\mathbf{v}_c$, $\mathbf{r}_c$, $\mathbf{u}$, and $\mathbf{u}_d$. The ADEC's main function at a supervisory level is to determine which rules can be activated, which jobs to be started, and the part outputs for which a release command to be sent to the FMS. These functions are processed by means of two different sets of logical equations, the former is used for checking the conditions for the activation of rules, and the latter is used for defining the consequent controller outputs. The updated value of the logical rule vector is computed with the following logical state equation

$$\overline{\mathbf{g}_{k+1}} = \mathbf{F}_v \otimes \overline{\mathbf{v}_{ck}} \oplus \mathbf{F}_r \otimes \overline{\mathbf{r}_{ck}} \oplus \overline{\mathbf{F}_{rd} \otimes \mathbf{r}_{ck}} \oplus \mathbf{F}_u \otimes \overline{\mathbf{u}_k} \oplus \mathbf{F}_{ud} \otimes \overline{\mathbf{u}_{dk}}, \tag{2.19}$$

where k denotes the dispatching epoch. The overbar in (2.19) denotes a vector negation. Given a natural number vector $\mathbf{a}$, its negation is such that $\overline{a_i} = 0$ if $a_i > 0$, and $\overline{a_i} = 1$ otherwise. $\otimes$ and $\oplus$ denote the *and/or* multiplication and addition, respectively. $\mathbf{C} = \mathbf{A} \otimes \mathbf{B}$ is defined by $c_{ij} = (a_{i1} \wedge b_{1j}) \vee (a_{i2} \wedge b_{2j}) \vee \cdots$, and $\mathbf{C} = \mathbf{A} \oplus \mathbf{B}$ is defined by $c_{ij} = (a_{ij} \vee b_{ij})$. $\wedge$ and $\vee$ are symbols for logical AND and OR operations, respectively. $\mathbf{g}_{k+1}$ essentially provides the information of which rules can be activated without causing a deadlock in dispatching epoch $k+1$.

Denote by V_i the set of jobs that are required as immediate precursors to rule g_i, by R_i the conjunctive set of resources that are all required to fire rule g_i, by U_i the set of inputs that are all required to fire rule g_i, and by R_{d_i} the disjunctive set of additional resources, any one of which can accomplish rule g_i in addition to all the required resources $r_j \in R_i$. The properness of ADEC is ensured by Theorem 2.1 [74, 75].

Theorem 2.1 (Conjunctive and Disjunctive Rule-Bases) *The i^{th} rule (i.e., i^{th} row) of (2.19) is equivalent to*

$$g_i = \bigwedge_{v_j \in V_i} v_j \wedge \bigwedge_{r_j \in R_i} r_j \wedge \bigwedge_{u_j \in U_i} u_j \wedge \left(\bigvee_{r_j \in R_{d_i}} r_j \right), \tag{2.20}$$

i.e., *rule state g_i is true (equal to 1) if all job vector elements v_j required for rule* g_i *are true, all resource vector elements r_j required for rule* g_i *are available, and all input vector elements u_j required for rule* g_i *are true, while any of the resources in R_{d_i} is available.*

Proof: Using matrix operations in the OR algebra, we have

$$\overline{g_i} = \left(\bigvee_{j=1}^{|V_i|} f_{ij}^v \wedge \overline{v_j} \right) \vee \left(\bigvee_{j=1}^{|R_i|} f_{ij}^r \wedge \overline{r_j} \right) \vee \left(\bigvee_{j=1}^{|U_i|} f_{ij}^u \wedge \overline{u_j} \right) \vee \overline{\left(\bigvee_{j=1}^{|R_{d_i}|} f_{ij}^{rd} \wedge r_j \right)} \vee \left(f_{ij}^{ud} \wedge \overline{u_j^d} \right). \tag{2.21}$$

Successive application of the de Morgans theorem yields

$$\begin{aligned} g_i &= \overline{\left(\bigvee_{j=1}^{|V_i|} f_{ij}^v \wedge \overline{v_j} \right) \vee \left(\bigvee_{j=1}^{|R_i|} f_{ij}^r \wedge \overline{r_j} \right) \vee \left(\bigvee_{j=1}^{|U_i|} f_{ij}^u \wedge \overline{u_j} \right) \vee \overline{\left(\bigvee_{j=1}^{|R_{d_i}|} f_{ij}^{rd} \wedge r_j \right)} \vee \left(f_{ij}^{ud} \wedge \overline{u_j^d} \right)} \\ &= \overline{\left(\bigvee_{j=1}^{|V_i|} f_{ij}^v \wedge \overline{v_j} \right)} \wedge \overline{\left(\bigvee_{j=1}^{|R_i|} f_{ij}^r \wedge \overline{r_j} \right)} \wedge \overline{\left(\bigvee_{j=1}^{|U_i|} f_{ij}^u \wedge \overline{u_j} \right)} \wedge \left(\bigvee_{j=1}^{|R_{d_i}|} f_{ij}^{rd} \wedge r_j \right) \wedge \overline{\left(f_{ij}^{ud} \wedge \overline{u_j^d} \right)} \\ &= \left(\bigwedge_{j=1}^{|V_i|} \overline{f_{ij}^v} \vee v_j \right) \wedge \left(\bigwedge_{j=1}^{|R_i|} \overline{f_{ij}^r} \vee r_j \right) \wedge \left(\bigwedge_{j=1}^{|U_i|} \overline{f_{ij}^u} \vee u_j \right) \vee \left(\bigvee_{j=1}^{|R_{d_i}|} f_{ij}^{rd} \wedge r_j \right) \wedge \left(\overline{f_{ij}^{ud}} \vee u_j^d \right). \end{aligned} \tag{2.22}$$

Now, $f_{ij}^v = 0$ if task v_j is not needed to fire rule xi. As such $\overline{f_{ij}^v} = 1$ so that for those elements one has $\overline{f_{ij}^v} \vee v_j = 1$ whether the corresponding task element is true or not. On the other hand, $f_{ij}^v = 1$ if task v_j is needed to fire rule g_i. This makes $\overline{f_{ij}^v} = 0$. One has $\overline{f_{ij}^v} \vee v_j = 1$ if the corresponding task element is true. A similar reasoning applies to f_{ij}^r, f_{ij}^u, and f_{ij}^{ud}. Likewise, $f_{ij}^{rd} = 0$ if resource r_j is not able to accomplish rule g_i. As such, one has $f_{ij}^{rd} \wedge r_j = 0$ regardless whether the corresponding resource element is true or not. Elements $f_{ij}^{rd} = 1$ if resource r_j is able to accomplish rule g_i. One now has $f_{ij}^{rd} \wedge r_j = 1$ if and only if the corresponding resource element is true. As such, the last equation in (2.22) is equivalent to

$$g_i = \bigwedge_{v_j \in V_i} v_j \wedge \bigwedge_{r_j \in R_i} r_j \wedge \bigwedge_{u_j \in U_i} u_j \wedge \left(\bigvee_{r_j \in R_{d_i}} r_j \right) \wedge u_j^d. \tag{2.23}$$

∎

On the ground of the current value of $\mathbf{g}_{k+1}$, the scheduling decisions are included by the MTME resulting in the final state vector $\mathbf{g}_{pk+1}$, which describes the set of eventually activated rules at dispatching epoch $k+1$. The computation of $\mathbf{g}_{p_{k+1}}$ will be presented later in Chapter 4. Based on the value of $\mathbf{g}_{p_{k+1}}$, the ADEC decides which jobs to be started and which outputs to be released by means of the following output equations

$$\mathbf{v}_{sk+1} = \mathbf{S}_v \otimes \mathbf{g}_{pk+1}, \tag{2.24}$$

$$\mathbf{y}_{k+1} = \mathbf{S}_y \otimes \mathbf{g}_{pk+1}. \tag{2.25}$$

(2.19) and (2.24)–(2.25) represent a conjunctive and disjunctive rule-based supervisory control for any class of discrete event systems. It is worth noting that all matrices and vectors are Boolean, making real-time computations easy even for large-scale FMSs.

2.5 Summary

In this chapter, general descriptions and mathematical modeling languages of MPs and FMSs were described in a rigorous manner. The FSM models of MPs were introduced, which is conceived as a graphical abstract language that contains a finite number of states. The FSM is in only one state at a time; the state it is in at any given time is called the current state. It can change from one state to another when initiated by a triggering event or condition, and the MPs often exhibit a distinct level of power consumption in each state. Beyond MPs, the behavior of FSMs can be observed in many other industrial systems as well, which perform a predetermined sequence of actions depending on a sequence of events with which they are presented.

Next, two mathematical modelling languages for FMSs, namely, the WTPN and ADEC, were detailed and discussed. The WTPN, or PN in general, is a very simple mathematical model. It is a graphical language, yet the semantics are clear. It provides a convenient way to analyze the structural properties of event-triggered and concurrent systems such as reachability, liveness, boundedness, deadlock and blocking, *etc* [77]. On the other hand, a key attractiveness of the ADEC is its minimal model complexity, portability, and ease of implementation on any platform using any programming language, *e.g.*, MATLAB, LabVIEW, and C, *etc.* [72, 78]. ADEC's command-and-control structure facilities industries adapting quickly to fast-evolving and dynamic manufacturing environments and relieves much human supervisory requirements by enabling efficient automation of job execution and resource allocation *online*.

In the following chapter, a novel approach is proposed to reduce the number of required sensors in process state tracking through identifying the operational states of MPs by extracting useful information and features in energy data.

Chapter 3

Energy Data-Driven Process State Identification for High-Performance Decision Support

To reduce energy consumption for high-performance manufacturing industries, continual energy monitoring and state tracking of industrial manufacturing processes (MPs) are essential. In this chapter, we introduce a novel framework to reduce the number of required sensors in process state tracking based on finite-state machine (FSM) models of MPs and energy data. To justify our proposed framework, comparative experiments with an existing framework are evaluated on two industrial applications, an injection moulding system and a stamping system.

3.1 Background

In recent years, many energy-efficient technologies for high-performance manufacturing industries have been proposed in the literature. Most of the reported techniques require both energy data and operational states of MPs to make energy-efficient decisions. In particular, several energy optimization technologies recently proposed [50, 79, 80] require real-time information of both energy data and operational states (busy or idle) of involved resources. In other technologies, energy and state data of MPs are explored more thoroughly. An entire cycle of industrial milling process was divided into nine operational states (*e.g.,* run-up, spindle running, chipping milling, *etc.*) in a generic energy optimization model [33]. An industrial plastics welding process was also dismantled into six operational states (*e.g.,* start-up, stand-by, waiting, and processing, *etc.*) used for an energy-efficient planning methodology called EnergyBlocks [23]. Other relevant energy-efficient technologies also used both energy data and operational states includes [25, 81].

Traditional approaches to energy monitoring and process state tracking relied either on sensors or accounting exercises, which are expensive for large-scale FMSs. To reduce the number of required sensors for energy monitoring, theoretical estimations of energy consumption for particular MPs were derived based on the processing parameters [66, 67]. The existing approaches could be precise but are computationally intensive to implement because many process parameters must be known. In an opposite way, the number of required sensors for process state tracking is reduced by inferring the operational states based on energy consumption patterns using complex event processing methods [82].

In addition, an effective energy monitoring and process state tracking is also crucial to integrate with novel energy-efficient technologies in a unified decision support system (DSS) to align the sheer amount real-time feedback energy and state data from MPs more closely with business and production requirements in high-performance manufacturing. In many large-scale information and realistic manufacturing systems, data is usually logged and collected from the various sources of systems and sensors at a fixed sampling rate in a big data matrix. This impedes the effectiveness of a DSS in extracting useful information and features from the data matrix [16], and hence make existing data more powerful to support the decision making of energy-efficient technology.

In this chapter, we propose a softsensing approach for online identification of operational states based on energy consumption patterns. We first use FSMs to model MPs. FSMs consist of a finite number of states, transitions, and actions, and have been applied to model MPs [83, 84], where they are proven to be very useful for energy audit and reporting in large-scale FMSs. To be applicable to FMSs where a large amount of energy data exists, we propose a two-stage framework for online classification of real-time energy data into different energy consumption patterns in the context of the FSM models of MPs. Our proposed framework uses a Haar transform and Ebayes threshold as well as support vector machine (SVM) for segmentation and clustering energy data into different operational states, respectively. Our implementation results on energy data logged from industrial injection moulding and stamping machines reveal our proposed framework's efficiencies in process state identification based on energy consumption patterns. Based on the obtained results, we design an energy data-driven DSS, which uses real-time energy measurements and process operational states to make effective decisions, enabling high-performance manufacturing.

3.2 Process Identification Framework

In this section, an intelligent framework for identification of process operational states based on time series of power data is proposed. Structurally, our proposed framework consists of two consecutive layers. The first layer uses Haar transform and an EBayes threshold to segregate the time series of power data into segments. The second layer includes feature extraction and a two-stage SVM to sort the time series segments into clusters, each cluster indicates an operational state.

3.2.1 Signal Segmentation

To segment the time series of power data, Haar transform is first used to compute the wavelet coefficients. The computed wavelet coefficients are then passed through an EBayes threshold level, where the cross-over coefficients indicate the change points of the process operational states.

Haar transform can be interpreted as a dyadic multi-rate filter bank. It uses both scaling functions and wavelet functions, which are associated with the low-pass filter and high-pass filter, respectively. The scaling function generates the approximation coefficients, while the wavelet function computes the detail coefficients. Wavelet transforms are recursive, where the wavelet coefficients computed by the previous iteration are the inputs for the subsequent iteration. A non-overlapping rectangular window is used to sample the time series, where the window width is two for the initial iteration and is doubled at each subsequent iteration during the wavelet decomposition.

Consider a time series of power data $\mathbf{p} = \{p_i : i \in \mathbb{Z}^+\}$. Let us also denote the wavelet approximation and detail coefficients by $\mathbf{a} = \{a_\mathrm{i} : i \in \mathbb{Z}^+\}$ and $\mathbf{d} = \{d_\mathrm{i} : i \in \mathbb{Z}^+\}$, respectively. As such, the Haar transform of $\mathbf{p}$ is computed as follows [85]

$$a_i = \frac{p_i + p_{i+1}}{\sqrt{2}}, \tag{3.1}$$

$$d_i = \frac{p_i - p_{i+1}}{\sqrt{2}}. \tag{3.2}$$

In our industrial applications, a level-five wavelet decomposition is used, as it provides sufficient wavelet resolution to detect significant change points in the time series of power data. Obviously, a higher-level wavelet decomposition provides better results, but also requires more computational efforts.

In time series segmentation using wavelet transforms, selecting an appropriate threshold method is crucial. An unnecessarily large threshold will segregate too many coefficients resulting in over-segmentation, and *vice versa.* Recently, an empirical EBayes threshold for wavelet decomposition has been proposed in [86], which was proven to outperform existing threshold methods for different data sets, *e.g.,* the modified universal threshold, the sureshrink and false discovery rate techniques, and the block thresholdings such as neighBlock or neighCoeK [87, 88].

Consider the wavelet coefficients $\mathbf{d}$ inundated with noise ε which can be written as

$$d_i = \mu_i + \varepsilon_i, \tag{3.3}$$

where μ is the distribution mean. As such, finding a threshold value using EBayes includes three main steps.

- Step 1: the parameters μ are modeled as having independent prior distributions f_{prior} each given by

  $$f_{prior}(\mu) = (1 - w)\delta_0(\mu) + w\sigma(\mu), \tag{3.4}$$

  where w, $0 < w < 1$, is probabilistic variable. δ_0 is the Dirac function, and σ is a symmetric heavy-tailed density such as Laplace or Cauchy density. σ is assumed to be a fixed unimodal symmetric density. While σ is commonly a normal density in current literature, a heavier-tailed density is used here.

- Step 2: the probability w is estimated by defining the marginal maximum likelihood estimator $\hat{w}$ of w to be the maximizer of the marginal log-likelihood as follows

$$\hat{w} = \arg\max_{w} \sum_{i=1}^{m} \log \{(1 - w)\, \phi\,(d_i) + wg\,(d_i)\}, \tag{3.5}$$

where $g = \sigma \star \phi$, and $\star$ denotes convolution. m denotes the length of time series. To avoid confusion with the scaling function of the wavelet family, ϕ is used to denote the standard normal density.

- Step 3: an estimation for μ is found by substituting $\hat{w}$ back into (3.4) and taking the posterior median of μ.

3.2.2 Segment Clustering

One now wishes to extract useful features from the obtained power segments. To effectively eliminate outliers, which apparently correspond to the segments with many sharp transitional spikes, the mean absolute deviation (MAD) is used. Consider a time series segment $\mathbf{p}$, the MAD is defined as the median of the absolute deviations from the segment's median as follows

$$\mathrm{MAD}\,(\mathbf{p}) = \mathrm{median}_i\,(\ |p_i - \mathrm{median}_j\,(p_j)|\ )\,. \tag{3.6}$$

Next, the geometric median (GM) is computed to measure the central tendency of the power segments. Our purpose is to determine the amplitude level, which is relatively constant after removing outliers. GM is used instead of the mean, because the power segments possibly contain abnormal nadirs at both ends. Formally, the GM of a power segment $\mathbf{p}$ is computed as follows

$$\mathrm{GM}\left(\mathbf{p}\right) = \arg\min_{y} \sum_{i=1}^{m} \left\|p_i - y\right\|_2. \tag{3.7}$$

To explore the dynamic characteristics of power segments, the third-order auto regression (AR) features are used. A third-order AR model of a power segments $\mathbf{p}$ is defined as follows

$$p_i = \omega + \sum_{t=1}^{n=3} \beta_t p_{i-t} + \varsigma_i, \tag{3.8}$$

where β_t denotes the regression coefficients, ω is the intercept variable, and ς is a noise parameter. β_t, ω, and ς can be estimated by various step-wise least-squares algorithms. Two algorithms, the Levinson–Durbin and Burg algorithms, are widely used to estimate the coefficients of a AR model [89, 90]. An exact value of n for a given power segment is not known *a priori*; it is desirable to reduce the computational complexity by choosing the minimal n such that the AR model is of satisfactory performance. In our industrial applications, $n = 3$ is chosen as it provides a sufficient fit to the power segments. As a result, three AR features β_1, β_2, and β_3 are obtained. In summary, a total of five features has been extracted for segment clustering including MAD, GM, β_1, β_2, and β_3.

Once features are extracted, each power segment is now represented by a feature vector $\mathbf{s} \in \mathbb{R}^5$ with element s_j denotes the j^{th} feature. To ensure the segment clustering is of satisfactory accuracy, the first stage is an unsupervised SVM designated to eliminate outliers from the set of power segments. Unsupervised SVM was first proposed by Schoelkopf *et al.* in [91] for identifying outliers in data. It contains two classes and does not require the class labels *a priori*. The outlier refers to a feature vector and can be defined as an observation which is numerically distant from the rest of the data.

Unsupervised SVM basically separates all the data points from the origin (in feature space) and maximizes the distance from this hyperplane to the origin. This results in a binary function which captures regions in the input space where the probability density of the data locates. To separate outliers by unsupervised SVM, the binary function returns +1 for the training data and −1 for the outlier data. This can be achieved by solving the following constrained optimization problem

$$\min_{\mathbf{o},b,\xi_i} \frac{1}{2}\|\mathbf{o}\|^2 + \frac{C}{U}\sum_{i=1}^{U}\xi_i - \rho, \tag{3.9}$$

s.t.,

$$\mathbf{o}\cdot\mathbf{s}_i \geq \rho - \xi_i, \xi_i \geq 0,\ \forall i = 1,\ldots,U, \tag{3.10}$$

where vector $\mathbf{o}$ and ρ determine the orientation of the hyperplane, and $\xi_i \leq 0$ are slack variables. U denotes the number of power segments. The constant C is an upper bound on the fraction of outliers.

The second stage is a supervised SVM used to cluster the remaining power segments into groups, each of which indicates to an operational state. The second-stage SVM deals with a supervised learning problem where the labels of all power segments are available. The whole set of power segments is segregated into a "training" set and "validation" set–the former used to train the underlying clustering mechanism, and the latter used to test if the trained SVMs are of satisfactory accuracy. The SVM algorithm is a well-known classification method that has been applied to various engineering applications [92]. Its objective is to find the separating hyperplane for which the distance between the clusters, measured along a line perpendicular to the hyperplane, is maximized. This can be achieved by solving the following constrained optimization problem

$$\min_{\mathbf{o},b,\xi_i} \frac{1}{2}\|\mathbf{o}\|^2 + C\sum_{i=1}^{U}\xi_i, \tag{3.11}$$

s.t.,

$$y_i\left(\mathbf{o}\cdot\mathbf{s}_i - b\right) \geq 1-\xi_i,\ \xi_i \geq 0,\ \forall i = 1,\ldots,U, \tag{3.12}$$

where vector $\mathbf{o}$ and b determine the orientation of the hyperplane, C denotes a positive smoothness constant that specifies the tradeoff between margin and misclassification error.

For a nonlinear decision surface, this optimization problem can be addressed by the concept of using kernel function k. Their decision function has the following form

$$y(\mathbf{s}) = \sum_{i=1}^{N_s}\alpha_{si}k(\mathbf{s},\mathbf{s}_i) + b, \tag{3.13}$$

where $k\left(\cdot,\cdot\right)$ represents the kernel, which can be shown to compute the dot products in associated feature spaces $\mathbb{R}^5$, *i.e.*, $k\left(\mathbf{s},\mathbf{s}'\right) = \left\langle\Phi\left(\mathbf{s}\right),\Phi\left(\mathbf{s}'\right)\right\rangle$. The function Φ : $\mathbf{s}\in\mathbb{R}^5, \mathbf{s}\rightarrow\Phi\left(\mathbf{s}\right)$ maps the feature vectors into kernel space. The SVM decision hyperplane is determined by $\psi = \sum_{i=1}^{N_s} \alpha_{si} \Phi(\mathbf{s}_i)$, with N_s support vectors $\mathbf{s}_i$ and nonvanishing coefficients α_{si}. In this chapter, the most common type of kernel, the radial basis function (RBF), is used where $k(\mathbf{s}, \mathbf{s}') = \exp\left(-\gamma \|\mathbf{s} - \mathbf{s}'\|^2\right)$. The RBF kernel is often designed by tuning the kernel parameter pair (C, γ) using the cross-validation method. It is worth noting that cross-validation method can not be used to train the SVM in the case of unsupervised learning problem due to the lack of segment labels. In this case, the kernel parameter pair (C, γ) is obtained by an iterative method [93].

3.3 Industrial Applications

The performance of our proposed process identification framework is evaluated on two industrial applications. In the first application, the energy consumption of two industrial injection moulding machines is investigated, namely, Arburg A220 S 150–60 and Arburg A420 S 1000–150 as depicted in Figs. 3.1 and 3.2, respectively. In the second application, eight stamping machines denoted by M1–M8 at an industrial stamping system are studied.

3.3.1 Experiment Setup

In both industrial applications, three-phase electrical parameters are continually measured using RUDOLFs. In our experiments, only real power is used, but other elec-

Figure 3.1: Arburg A220 S 150–60 injection moulding machine.

Figure 3.2: Arburg A420 S 1000–150 injection moulding machine.

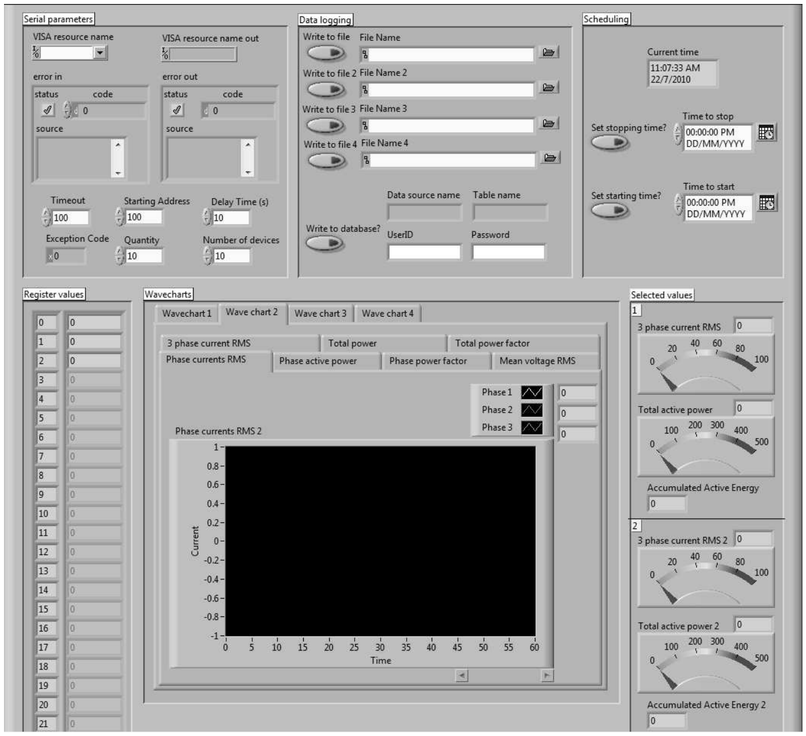


Figure 3.3: A screenshot of GUI developed in LabVIEW for online energy monitoring.

tric variables such as reactive power, apparent power, and power factor, *etc.*, are all measured. To interface RUDOLFs with computers or handled devices, a GUI has been developed in LabVIEW as shown in Figure 3.3 to convert serial data from Rudolf power analyzers to real power consumption data. The sampling frequency of RUDOLFs is set to 1 Hz. The energy consumption produced by manufacturing seventeen injection moulding parts and ten stamping parts are logged. For each part, numerous work-pieces are produced.

3.3.1.1 Injection Moulding Process

Injection moulding is a typical MP for producing parts from plastic materials. Materials are fed into a heated barrel, mixed, and forced into a mold cavity, where they are cooled and hardened to the shape of the cavity. The entire injection moulding cycle can be segregated into six operational states:

1. *Switch off*: In this state, control panel, heater, and hydraulic pump are off;
2. *Warm up*: Machines require warm-up before it is ready to be used. In this state, it executes a rapid motion from starting position and causes high narrow pulses in the power consumption profile;
3. *Idle*: In this state, the machine is on but not in production. The power consumption profile is low, but not as low as when the machine is in *Switch off* mode;
4. *Pump/heat*: This is the preparation state when the hydraulic pump is used to inject heating oil into the mold, and the heater is warming up the oil;
5. *Start up*: This state is the transition between the *Idle* and *Moulding* modes; and
6. *Moulding*: In this state, workpieces are mass-produced. Moulding state typically consists of numerous injection moulding cycles. An injection moulding cycle to produce one workpiece includes seven sub-states, *i.e.*, mould closing, mould filling, mould packing, part cooling, mould opening, ejection, and mould

cleaning, as described by [94]. However, an injection moulding cycle is usually very short (as fast as 14 seconds) in modern injection moulding machines as depicted in [95]. As such, it is more meaningful to identify the entire production state instead of each sub-state, which is very short and repetitive.

The power consumption profiles reveal a number of different operation states which are linked to the status of machine components and power consumption. During the MP, the machine switches from one state to another at different instances in time. It is worth noting that similar observations can be made for other types of industrial machines in large-scale manufacturing systems as well.

3.3.1.2 Stamping Process

Stamping includes a variety of sheet-metal forming processes, *e.g.*, punching, coining, blanking, piercing, and bending, *etc.* During operation, the stamping die is placed into a reciprocating stamping press. As the press moves up, the top die moves with it, which allows the material to feed. When the press moves down, the die closes and performs the stamping operation. With each stroke of the press, a completed part is removed from the die [96].

The entire stamping cycle can be decomposed into five operational states, namely, *Switch off, Warm up, Idle, Start up*, and *Stamping*. The energy consumption of the first four states are relatively similar to injection moulding process. The *Stamping* state specially includes many spikes. Each spike is observed every time the stamp-

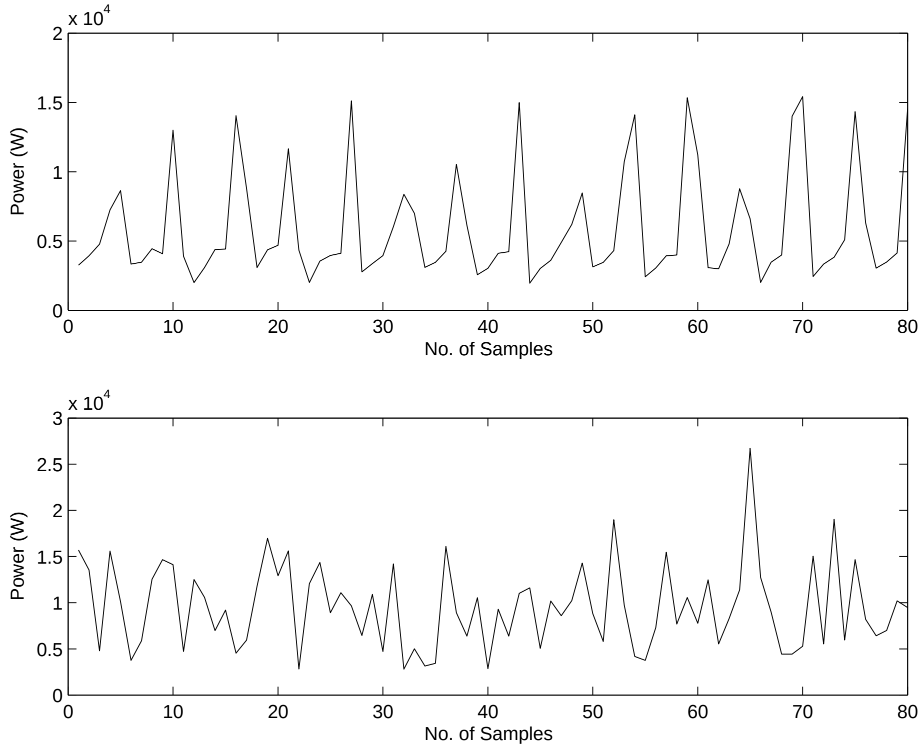


Figure 3.4: A comparative example between a normal and an abnormal power segments from *Stamping* state: (top) normal segment and (bottom) *Abnormal* segment.

ing press moves down to perform stamping operations. Among the stamping machines, M8 is oldest and has relatively different patterns in time series of power data. Thus, all power segments extracted from M8 are labelled as *Abnormal* state. An illustrated comparison between segments from *Stamping* and *Abnormal* states is shown in Figure 3.4, where different patterns can be clearly observed.

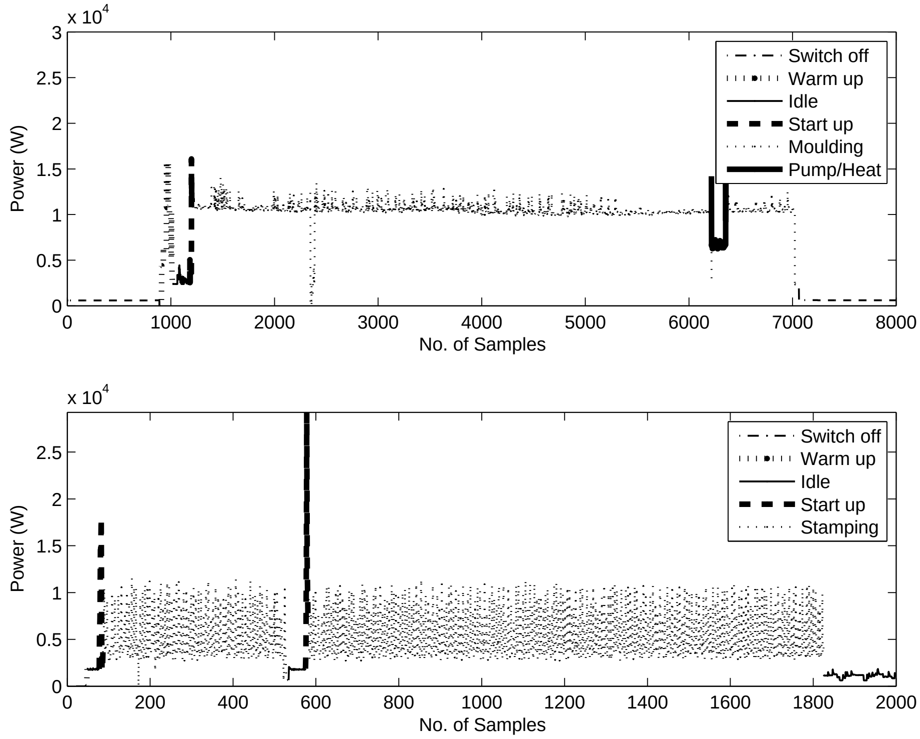


Figure 3.5: The discrete-state time series of power data of industrial processes: (top) injection moulding and (bottom) stamping.

3.3.2 Experiment Results

In this section, the performance of our proposed framework is compared with an existing framework reported by [97]. It is worth noting that there are two main differences between the two frameworks. First, our proposed framework does not include the Savitzky-Golay (SG) filter for preprocessing energy measurements and the EBayes threshold is used instead of the universal threshold. Second, SVM is used to cluster the power segments instead of the fuzzy c-means (FCM) algorithm.

Examples of time series of power data of injection moulding process and stamping process are shown in Figure 3.5, where the process operational states are clearly distinguished. From the observations of the power consumption profiles, an FSM energy consumption model in the form of diagraph can be built for injection moulding and stamping process as shown in Figures 2.3 and 3.6, respectively.

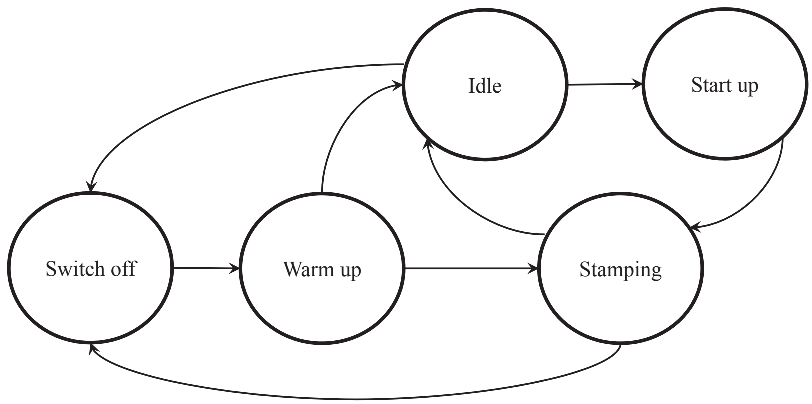


Figure 3.6: FSM models of industrial stamping process.

It can be seen that each operational state exhibits a relatively distinct level of magnitude in the time series of power data. Both process identification frameworks are implemented in MATLAB. The time series of power data of both data sets are segmented as discussed. An example of the signal segmentation using our proposed framework on a time series of power data from Arburg A220 S 150–60 is shown in Figure 3.7. It can be seen that the time series of power data is first transformed into wavelet coefficients. Wavelet coefficients are then filtered by an EBayes threshold (the dashed line), where only cross-over coefficients are accentuated indicating the change

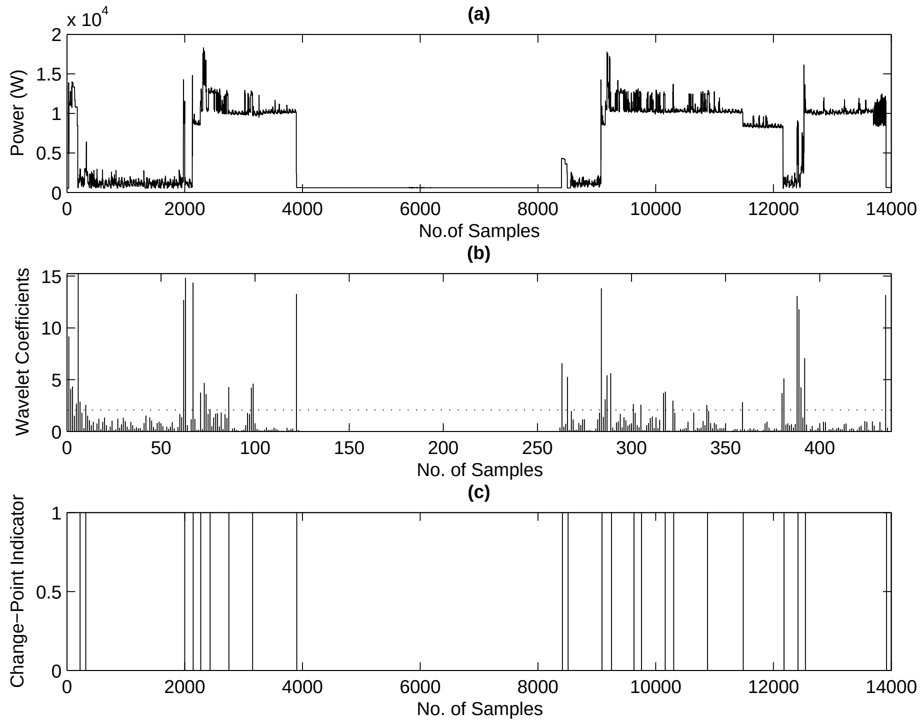


Figure 3.7: An illustrated example of signal segmentation using the our proposed framwork: a) time series of power data, b) wavelet coefficients with EBayes threshold (dashed line), and c) detected change points.

points of the process operational states.

Five useful features including the MAD, GM, and three AR parameters are extracted from each power segment. The first-stage SVM is now used to detect "outlier" power segments. For illustration, an example of detected "outlier" segments in *Moulding* state is shown in Figure 3.8. The outlier detection results is reported in Table 3.1. The best SVM kernel parameters $\gamma = 0.71$ and $\gamma = 0.96$ are obtained for the injection and stamping data sets, respectively.

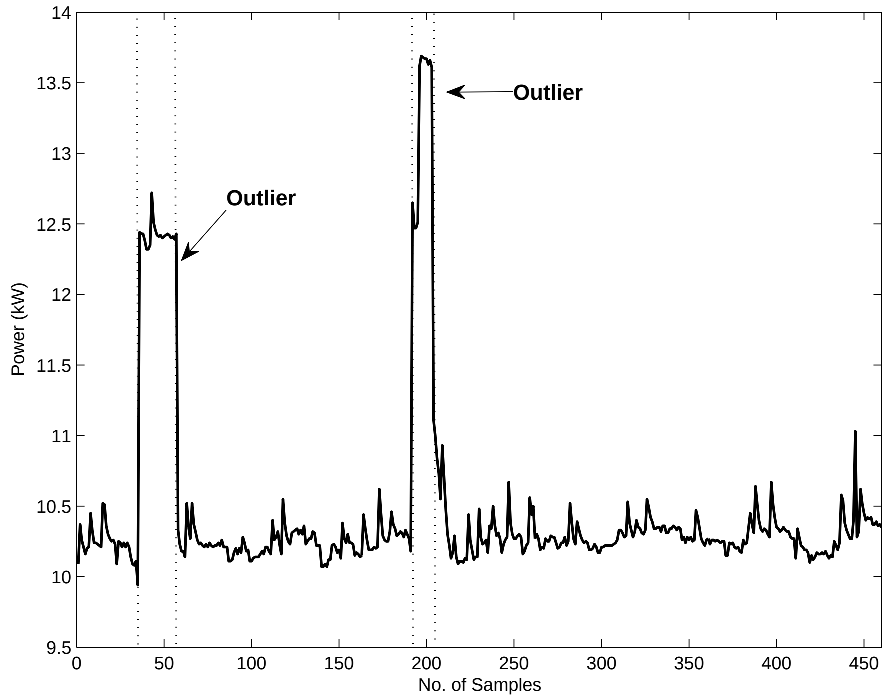


Figure 3.8: An example of outlier detection of *Moulding* state.

Table 3.1: Outlier Detection Results

Data set	SVM kernel parameter γ	Percentage of outlier segments
Injection	0.71	6.4%
Stamping	0.96	12.23%

3.3.2.1 Identification Results with Sufficient Training Data

We now wish to cluster the remaining power segments according to their underlying process operational states. In this section, the clustering performance of both frameworks are evaluated with sufficient training data, where both data sets are segregated into 50% for training and 50% for testing. For convenience, the operational states of both injection moulding and stamping processes are numerically labelled in Table 3.2.

Table 3.2: Cluster Label for Injection Moulding and Stamping Operational States

Injection data set		Stamping data set	
Cluster label	Operational state	Cluster label	Operational state
1	Switch off	1	Switch off
2	Warm up	2	Warm up
3	Idle	3	Idle
4	Start up	4	Start up
5	Pump/heat	5	Stamping
6	Moulding	6	Abnormal

To tune the parameter pair $(C;\gamma)$ of the second-stage SVM, a coarse grid-search is first performed using cross-validation method. Various pairs of $(C;\gamma)$ values are tested and the one with the best cross-validation accuracy is picked. An exponentially growing sequence of $(C;\gamma)$ is examined, *e.g.*, $C = (2^{-4}, 2^{-3}, \ldots, 2^{15})$ and $\gamma = (2^{-14}, 2^{-13}, \ldots, 2^{5})$. Our results show that the best $(C;\gamma)$ for injection and stamping data sets are $(2^5; 2^{-5})$ and $(2^4; 2^{-3})$ with the corresponding cross-validation rate are 84% and 79.5%, respectively. Thus, finer grid searches on the neighborhood of $(2^5; 2^{-5})$ and $(2^4; 2^{-3})$ are conducted. Better cross-validation rates 84.699% at $(2^{4.77}; 2^{-4.51})$ and 80.300% at $(2^{4.48}; 2^{-2.78})$ are obtained for injection and stamping data sets, respectively.

The validation results using the our proposed framework for identification of the operational states are reported in Table 3.3. For injection data set, our proposed

Table 3.3: Number of Validated Segments with Sufficient Training Data

Injection data set			Stamping data set		
Cluster	Correct	Wrong	Cluster	Correct	Wrong
1	72	0	1	50	0
2	36	0	2	25	0
3	106	2	3	79	1
4	73	0	4	48	0
5	30	0	5	105	2
6	150	5	6	45	3

framework classifies 467 out of 474 different segments correctly, thereby yielding an accuracy of 98.52% in identification of the operational states. For the stamping data set, 352 out of 358 segments are identified correctly with an accuracy of 98.32%. In particular, it can be seen from Table 3.3 that the proposed method is able to classify *Switch off*, *Idle* and *Warm up* states very accurately. This is because the energy consumption of these two states are completely or almost flat with few fluctuations. Minor classification errors occur for the *Start up* state, as its energy consumption patterns is similar to *Warm up* state both having sharp increases from low levels in time series of power data. It can also be seen that most error cases arise from erroneous classification of the *Moulding* state (a prediction error of 3.22%) due to the many different sub-states. There is also a prediction error of 6.25% in *Abnormal* state, because some power segments in *Switch off* and *Idle* states of M8 are similar to other machines. The our proposed framework accurately classifies segments from *Stamping, Start up*, and *Warm up* states of M8 as *Abnormal* segments. With the same experiment setups, the existing framework yields an accuracy of 97.08%

and 96.68% in identification of the process operational states for the injection moulding and stamping data sets, respectively. It can be seen that our proposed framework slightly outperforms in the case of sufficient training data.

3.3.2.2 Identification Results with Limited Training Data

Next, it is desired to evaluate the performance of both frameworks in the case of limited training data. Both available data sets are segregated into 30% for training and 70% for validation. To tune the SVM parameter pair $(C;\gamma)$, the same cross-validation method is applied. The best cross-validation rates are obtained as 89.735% at $(2^{3.35}; 2^{-2.57})$ and 83.373% at $(2^{5.58}; 2^{-3.64})$ for injection and stamping data sets, respectively. The validation results using the trained SVMs for state classification are shown in Table 3.4.

Table 3.4: Number of Validated Segments with Limited Training Data

Injection data set			Stamping data set		
Cluster	Correct	Wrong	Cluster	Correct	Wrong
1	101	0	1	71	0
2	48	3	2	32	3
3	144	8	3	106	7
4	103	0	4	68	0
5	40	2	5	146	8
6	208	10	6	59	9

It can be seen our proposed framework still shows a reliable performance in the case of limited training data. In particular, it classifies 644 out of 667 different segments of injection data set correctly, thereby yielding an accuracy of 96.55%. For stamping data set, 482 out of 509 segments are classified correctly yielding an accuracy of 94.69%. *Switch off* and *Idle* states are still classified very correctly. Minor classification errors occur for the *Warm up* and *Start up* states, as their energy consumption patterns are similar with both having a sharp increase in energy consumptions from low power levels. However, this can be easily resolved by checking the developed FSM models if previous states are different but *not always.* According to the traversal in the FSM state diagrams, identifying the errors can be simply done by creating a "memory/buffer" to store the previous states. For example, if the previous state is *Warm up*, the current state must be either *Idle* or *Moulding, Stamping* as shown in Figures 2.3 and 3.6. If the result of the two-stage framework is not either Idle or Production, it is detected as an error. This clearly explains the connection between the developed FSM models and the two-stage classification framework. It can also be seen that the classification accuracy of *Moulding, Stamping*, and *Abnormal* states become worse but still be acceptable, where the corresponding classification errors are 4.80%, 5.47%, and 15.25%, respectively. Most of misclassified segments of *Abnormal* state belong to *Switch off, Idle* and *Warm up* states of stamping machine M8, while all segments belonging to *Stamping* state are correctly classified. Since *Switch off, Idle* and *Warm up* states are relatively short as compared to *Stamping* state, the *Abnormal* state can be quickly identified. The identification results using the existing framework are of 89.33% and 88.38% accuracies for the injection moulding and stamping data sets, respectively. It can be seen that the performance of the existing framework has dropped significantly in the case of limited training data.

3.3.3 Discussions with Related Works

There are several reasons for the advantages of our proposed process identification framework as compared to the existing framework. Instead of preprocessing the raw time series of power data, our proposed framework takes into account the noisy effects using the EBayes threshold method. This completely avoids losing important power features of the data distribution such as relative maxima, minima, and width, *etc.*, which are potentially flattened during noise filtering or signal smoothing.

Another benefit of our proposed framework is the reduction of time delay. As the SG filter is not causal and relies on future data, the existing framework causes extraneous time delay for online applications. The usage of a SG filter with window size $w = w_L + w_R + 1$ in series with a level-p Haar transform delays the wavelet coefficients by $2^p + w_R$ samples from real-time. In the SG filter, the calculated central sample of the fitted polynomial curve is the latest filtered sample. Without loss of generality, let us assume that the real-time sample is currently at index p_i, while the SG window still lags behind and only covers up to sample p_{i-w_R}. Furthermore, it can be seen from (3.1) and (3.2) that the window size of a level-p Haar transform is 2^p, which means a wavelet coefficient can only be calculated for every 2^p smoothed samples. Therefore, the latest wavelet coefficient only represents sample $p_{i-w_R-2^p}$.

It is also worth noting that the FCM is an unsupervised learning algorithm, where the SVM is of supervised learning type. One uses the FCM to cluster data which their labels are not known *a priori*, however, the SVM is firstly trained with the labeled data and then is used to classify the unlabeled data. In our industrial applications, the segment labels as well as the number of operational states are all available. This implies that it is more suitable to use the SVM rather than the FCM.

It also can be seen that our proposed framework is effective for imbalanced datasets, as both injection and stamping datasets used in this chapter are relatively imbalanced, where classes *moulding* and *stamping* dominate the datasets. For the case of highly imbalanced datasets, the classification accuracy of our proposed framework is expected to drop down to 80% at most.

3.4 Energy Data-Driven Decision Support System

A crucial step towards high-performance manufacturing requires a unified DSS to align the sheer amount of real-time energy and state data of MPs logged from various sensors more closely with state-of-the-art energy-efficient technologies [98–100]. In this view, a manufacturing work cell can no longer work in silos within shop floors, but rather an integral part. Sensory data need not to be consolidated manually and locally, but must be shared and synchronized across the company, resulting in a sheer variety and volume of data. In many large-scale information and realistic engineering systems, data is usually logged and collected from the various sources of systems and sensors at a fixed sampling rate in a big data matrix. This impedes the effectiveness of a DSS in extracting useful information and features from the data matrix for making effective decisions [16].

3.4.1 Architecture

DSS is an information system that support decision-making processes and problem solving activities. As a concept, DSS has been proliferated and evolved over the past few decades [101]. With advancing information and communications technology, DSS is nowadays widely implemented in global industries. Most existing DSS architectures are highly specific, which focus to solve one particular problem, such as supplier selection [102], reconfiguration product design [103], and machine selection [104], *etc.*

In general, there is no universally accepted taxonomy of DSS models as different researchers propose different classifications. Herein, we follow [98] to discuss three generic approaches to develop the DSS.

- Data-driven DSS. A data-driven DSS emphasizes access to and manipulation of a time series logged from various sources of sensors and meters. Data analytic and artificial intelligent techniques can be used for decision-making model.
- Knowledge-based DSS. A knowledge-based DSS provides specialized problem-solving expertise stored as facts, rules, procedures, *etc.* Knowledge-driven DSS is developed based on engineering and management expertise and experiences.
- Model-based DSS. A model-based DSS use data and parameters provided by

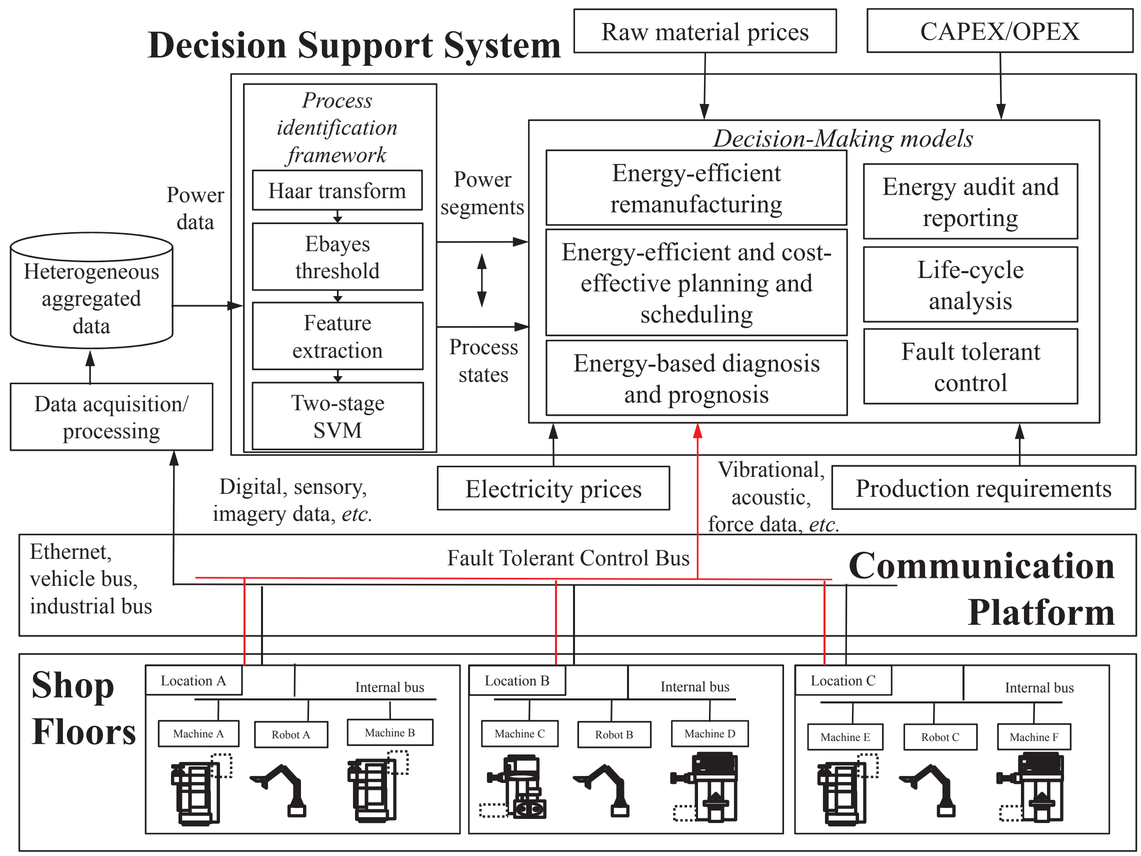


Figure 3.9: Energy data-driven DSS architecture for high-performance manufacturing industries.

users to assist decision makers in analyzing a situation. Model-driven DSS can be built using various statistical, optimization, or simulation models in current literature.

The developed DSS must be able for the extraction and mash-up of heterogeneous data using artificial intelligence and data-mining principles to combine the data and human expertise in creating new services, experiences, decisions, and maintenance rule-bases, *etc.* It must also integrate data from different sources and formats seamlessly using a data-aware correlation engine, making existing data more powerful for technical and professional users with a more potent decision-making and predictive capability. The architecture of our proposed DSS is shown in Figure 3.9.

In today's information technology era, the amount of digital, sensory, imagery, and audio data, *etc.*, from various sources such as simulation systems, control systems, inspection process, *etc.*, is expanding at an explosive rate. As such, the proposed DSS architecture must include a centralized and distributed monitoring network to enable the holistic access and analysis of a large variety of data from manufacturing shop floors at different locations. The monitoring network often comprises of various types of meters and sensors. At the shop-floor level, the local monitoring systems communicates via internal buses. The shop-floor monitoring systems are interfaced to the DSS over an communication platform, which includes two bus systems, namely data bus and fault tolerant control (FTC) bus, respectively. The key advantage of such distributed bus network is the reduction of required bus cables and wires. Although computer networking protocols such as Ethernet for local area networks and transmission control protocol/internet protocol for the Internet are most suitable for the data bus, industrial computer network protocols (*e.g.,* Fieldbus) and vehicle bus protocols (*e.g.,* controller area network bus and local interconnect network bus) can also be applied [105].

In addition, other data transmission equipments such as data acquisition systems and transducers may also be required. From the data bus, heterogeneous data are aggregated, processed (if necessary), and stored. FTC bus is highly important for our proposed DSS architecture to prevent unpredicted downtime due to machinery failures. FTC data could be in various forms such as vibrational, acoustic, and force data, *etc.*, depending on specific systems and applications. Various FTC schemes in current literature can be applied, one popular FTC scheme is the fault detection and isolation based on analytical redundancy [106].

The next important task is to provide effective monitoring of operational states of MPs. This task is accomplished by an intelligent process identification framework, whose mathematical rigour was detailed in Chapter 3.2. Real-time process operational states, energy measurements, and other related data (if any) are collated and synthesized to indicate the amount of energy consumption during specific process operational states. This encompasses not only the amount of energy required to run industrial machines, but would possibly include other facilities' energy consumption such as lighting, heating, ventilation, and air conditioning, *etc.*, which can contribute as much as 30% of the total energy consumption in a shop floor. A popular energy model in the current literature to the analyze MPs and facilities is the state-based model, where the entire process cycle is divided into a finite number of discrete operational states [23, 82]. Such energy model is based on integrated measurements over time to determine time-based power consumption in accordance with the underlying operational states, *e.g.*,, the amount of energy being consumed during *Stamping* state, versus the amount of energy being consumed when the resource is in *Idle* state.

3.4.2 Decision-Making Models

The obtained process operational states and energy measurements are then fed into several useful decision-making models. In addition, other production and business requirements such as electricity prices, raw material prices, operational expenditure (OPEX), and capital expenditure (CAPEX), *etc.,* may also be required. The objective of the decision-making models is to make energy efficient, cost effective, reliable decisions for high-performance manufacturing. Life-cycle analysis is also partially supported herein.

Table 3.5: Energy Audit for Arburg A220 S 150–60

State	Product I			Product II		
	Ave. power (kW)	Max. power (kW)	Time (s)	Ave. power (kW)	Max. power (kW)	Time (s)
Switch off	0.2	0.21	–	0.2	0.21	–
Warm up	5.63	8	50	1.63	3.57	55
Idle	0.47	1.72	–	0.4	1.1	–
Start up	0.47	1.72	–	0.4	1.1	–
Moulding	3.95	6.34	1800	4.32	6.34	2500
Pump/heat	–	–	–	2.85	2.93	400

The proposed decision-making models are summarised as follows.

- *Energy audit and reporting.* Energy audit and reporting is the inspection, analysis, and documentation of energy consumption in the shop floors. This is often carried out on weekly, monthly, and yearly basis. Over the past decade, industrial energy audit and reporting have exploded as the demand to lower increasingly expensive energy costs and move towards a sustainable manufacturing. A knowledge-based approach such as expert systems and case-based reasoning systems are most suitable for this model. Energy consumption of each machine state can be acquired, which provides an insight of how energy consumption is distributed among the different defined operation states which are usually not available.

An example of energy audit for the Arburg A220 S 150–60 injection moulding machine using the FSM models described in Table 3.5. It can be seen from Table 3.5 that both products I and II are produced by the same amount of workpieces. The metrics can also be used for auditing average power, maximum power, and completion time, *etc.*, for each operation state. Machines are also clustered into groups of high, medium, and low energy efficiency when manufacturing a particular product, and are given priority in manufacturing based on their groups. Machines clustering can also be based on various schemes, *e.g.*, average power consumption of production state. An example of a machine clustering result for the Arburg A220 S 150–60 and Arburg A420 S 1000–150 injection moulding machine is presented in Table 3.6, where the machines are clustered into three groups with the average power consumption of production state being high, medium, and low, respectively.

- *Energy-based diagnosis and prognosis.* Energy-based diagnosis and prognosis

Table 3.6: Machine Clustering of Arburg A220 S 150–60 and Arburg A420 S 1000–150

	Arburg A220 S 150–60			Arburg A420 S 1000–150		
	High	Medium	Low	High	Medium	Low
Product I	x				x	
Product II		x				x
Product III	x			x		
Product IV			x		x	
Product V			x			x
Product VI	x			x		

aims at investigating the relationship between energy consumption and machine faults, and hence using energy consumption as an indirect condition monitoring for industrial machines. This model effectively prevent not only production costs but also excessive energy consumption due to machinary faults, which are common in today's dynamic manufacturing environment. In current literature, diagnosis and prognosis have been developed using knowledged-based, data-driven, and model-based systems. However, data-driven system is a growing research trend, especially in prognosis and remanding useful life estimation.

- *Energy-based remanufacturing.* Energy-based remanufacturing is the next logical step of energy-based diagnosis and prognosis, which decides to reuse, repair, refurbishing, or recycle faulty machines in an energy-optimal and cost-effective way. Remanufacturing is a relatively new research area. Most existing works often considered only cost effectiveness and customer satisfaction, while energy consumption was hardly studied. As a new research area, all data-driven, model-based, and knowledge-based systems are applicable for remanufacturing.

- *Energy-efficient process planing and scheduling.* Energy-efficient process planing and scheduling can be defined as the arrangements and operations of machines, tools, materials, people, and information to produce energy-efficient workflows and resource assignments. This model may also include cost effectiveness as an optimization objective and find a Pareto optimal solution. Model-based systems are often applied the process planing and scheduling. Our proposed energy-efficient technologies in Chapters 4–6 are applicable for this decision-making model.
- *Fault tolerant control.* Fault tolerant control system ensures the manufacturing system to continue operating properly in the event of the failure of (or one or more faults within) some machines. This model prevents production and energy costs due to unexpected downtime due to machinery failures. Data-driven and model-based systems are often considered in this model.
- *Life-cycle analysis.* Our proposed DSS can contribute as a part of whole life-cycle analysis of the manufactured products, where energy consumption during production cycles are logged and documented. This model supports the environmental sustainability in manufacturing.

The decision-making models described herein provide useful suggestions towards high-performance manufacturing. Although the proposed decision-making models are fully computerized and autonomous, they can also be combined with engineering expertise.

3.5 Summary

In this chapter, we proposed an intelligent framework which identifies the MP operational states based on energy measurements. To justify our proposed framework, comparative experiments with an existing framework in current literature were evaluated on two industrial applications, an injection moulding system and a stamping system. The experimental results showed that our proposed framework achieved the accuracies of 98.52% and 98.32% in the case of sufficient training data, and 96.55% and 94.69% in the case of limited training data, respectively, which outperformed an existing framework in current literature. Based on the obtained results, an energy data-driven DSS was proposed, which targeted to make energy-efficient, cost-effective, and reliable decisions for the next generation of high-performance manufacturing. In addition, life-cycle analysis could be partially supported, as energy consumption during the production stage of a product's life-cycle was accurately logged and documented.

In the next chapter, we consider a dynamic scheduling problem which minimizes the sum of energy cost and tardiness penalty under power consumption uncertainties due to machine degradation. An integrated operational control and fast reactive scheduling framework will be proposed to solve the problem.

Chapter 4

Scheduling of Flexible Manufacturing Systems under Power Consumption Uncertainties

Motivated by the need to deal with uncertainties in energy optimization of flexible manufacturing systems (FMSs), this chapter considers a dynamic scheduling problem which minimizes the sum of energy cost and tardiness penalty under power consumption uncertainties. An integrated control and scheduling framework is proposed including two modules, namely, the augmented discrete event control (DEC) and the max-throughput-min-energy (MTME) reactive scheduling model. The ADEC is in charge of inhibiting jobs which may lead to deadlocks, and sequencing active jobs and resources. The MTME ensures the fulfillment of the innate constraints and decides the local optimal schedule of active jobs and resources. Our proposed framework is applied to the industrial stamping system presented in Chapter 3 with power consumption uncertainties formulated using three different probability distributions. The obtained schedules are compared with three dispatching rules and two rescheduling approaches. The experiment results verify that MTME outperforms three dispatching rules in terms of deviation from Pareto optimality and reduces interrupted time significantly as compared to rescheduling approaches. In addition, ADEC and MTME are programmed using the same matrix language, providing easy implementation for industrial practitioners.

4.1 Background

A common source of uncertainties in energy optimization of FMSs is resource degradation, which obviously increases the energy consumption of resources. Unlike the uncertainties in energy measurement process to be considered later in Chapter 6, uncertainties due to resource degradation is difficult to predict at the time an offline schedule is executed. The FMS scheduling problem is NP-hard in computational complexity theory, but consideration of uncertainties in resource degradation (dynamic scheduling) further aggravates its complexity. The existing energy-efficient technologies for dynamic scheduling of FMSs can be classified into three categories, namely, the reactive, the proactive, and the predictive-reactive. Each category has its own pros and cons [65].

Predictive-reactive scheduling is a scheduling/rescheduling process, in which the baseline schedules are generated offline and the active schedules are revised online in response to real-time uncertainties. The most common predictive-reactive scheduling include completed rescheduling (CR) and partial rescheduling (PR) [107, 108]. In theory, CR provides the optimal schedules, but these schedules are rarely achievable in practice and require prohibitive computational time. In addition, it can result in instability and disruption in manufacturing flows, leading to tremendous production costs. In PR, only jobs and resources affected by the uncertainties are rescheduled. On the other hand, the reactive scheduling is characterized by its capability of real-time decision-making, in which no baseline schedules are generated offline, and decisions are quickly made online using real-time information. Dispatching rules are typical examples of reactive scheduling, in which jobs are selected by sorting them according to some predefined criteria. Dispatching rules are still the most preferred scheduling approaches in industry due to their ease of implementation, low computational cost, and guarantee of schedule stability and feasibility [109–112]. The main weakness of reactive scheduling is that they cannot globally optimize the overall performance of generated schedules. Proactive scheduling focuses on building a predictive schedule which minimizes the effects of real-time uncertainties [113]. Baseline schedules are generated offline and will not be revised online. The main difficulty of these approaches is modelling of uncertainties. Computational cost is also an issue, since the stochastic search space is usually huge.

In this chapter, an FMS dynamic scheduling problem which minimizes the sum of energy cost and tardiness penalty is considered under power consumption uncertainties. Uncertainties in productive powers are realistic in a dynamic manufacturing environment, as power consumption was verified to be dependent on uncertain factors including machine conditions, tool conditions, and workloads [17]. The minimization of energy cost and tardiness penalty is a practical problem which was considered by [114] under static environment. Such tradeoff happens when a resource requires shorter time but higher energy to perform a job as compared to others.

To solve the formulated dynamic scheduling problem, this chapter proposes a matrix-based integrated control and scheduling framework for a class of FMSs described in Chapter 2 The proposed framework can be viewed as an aggregation of two interacting modules, the ADEC and the MTME. The ADEC has been described in Chapter 2, proving to be very efficient in modeling and controlling the large-scale discrete-event dynamics of typical manufacturing systems. The proposed MTME resembles a reactive scheduling approach, which dispatches the imminent jobs and resources quickly and online using real-time power consumption of resources. It includes two 0–1 LP submodels, the former maximizes the production throughput and the latter minimizes the energy cost at every dispatching epoch. Both ADEC and MTME are programmed using the same matrix language and function during operational control as a whole, which provide easy implementation for industrial practitioners.

Our proposed framework is tested with industrial energy data logged from the stamping system presented earlier in Chapter 3. The schedules obtained by our proposed framework are compared with three dispatching rules, CR, and PR approaches. The experiment results with different batch sizes verify that MTME outperforms the three dispatching rules for all test cases in terms of deviation from Pareto optimality. The PR outperforms MTME when the batch size is small (short schedules), but the reverse is observed when the batch size is larger than 60 parts (long schedules). In terms of mean interrupted time, MTME achieves less than 1 s for all test cases, while the PR and CR cause prohibitive interrupted time (instability) for the FMSs.

4.2 Dynamic Scheduling Under Power Consumption Uncertainties

In this section, a mathematical model of power consumption uncertainties is presented and the dynamic scheduling problem is formally defined.

4.2.1 Mathematical Model of Power Consumption Uncertainties

A practical mathematical model of power consumption uncertainties is selected purely for performance evaluation of MTME and the related works presented later in Section 4.4. In practice, power consumption uncertainties due to machine degradation may follow different kinds of mathematical models.

Most importantly, industrial practitioners need not to model power consumption uncertainties prior to using our proposed framework. As a reactive scheduling approach, MTME does not consider the uncertainties in generating schedules, but finding the effective ways to react to uncertainties at every dispatching epoch. With this reactive capability, our proposed framework can be adapted to any mathematical model of power consumption uncertainties. The reactive nature of our proposed framework also differs with predictive-reactive approaches such as CR and PR in the sense that our framework is triggered by job completions instead of changes in energy consumption.

The power consumption $a_{ij}^q(t)$ can be conveniently modelled by a step function, which is often defined in literature as [115]

$$a_{ij}^q(t) = \sum_{k=0}^{n} \theta_k \chi_{B_k}(t), \tag{4.1}$$

where $n \geq 0$ is the number of times that $a_{ij}^q(t)$ increases, θ_k are real numbers, B_k are intervals, and χ_B, is the indicator function of B defined by

$$\chi_B(t) = \begin{cases} 1 & \text{if } \ t \in B, \\ 0 & \text{if } \ t \notin B. \end{cases} \tag{4.2}$$

In this definition, the intervals B_k have the following two properties, namely, $B_k \cap B_{k'} = \emptyset$ for $k \neq k'$ and $\cup_{i=0}^{n} B_k = [0, \infty)$. To incorporate the uncertainties into $a_{ij}^q(t)$, B_k is defined by

$$B_k = \begin{cases} [0, b_0) & \text{if } k = 0, \\ \left[\sum_{z=1}^{k-1} b_z, \sum_{z=1}^{k} b_z\right) & \text{if } k \notin \{0, n\}, \\ \left[\sum_{z=1}^{n-1} b_z, \infty\right) & \text{if } k = n, \end{cases} \tag{4.3}$$

where the values of $\{b_z, 0 \leq z \leq n-1\}$ follow a probability distribution function denoted by $g\left(b; \mu_{ij}^q, \left(\sigma_{ij}^q\right)^2\right)$ with μ_{ij}^q and $\left(\sigma_{ij}^q\right)^2$ are the distribution mean and variance, respectively. Three probability distributions widely used in reliability engineering and life data analysis are investigated, namely, the Weibull, exponential, and truncated normal distributions [113]. Obviously $a_{ij}^q(t)$ must be monotonically increasing, as such $\theta_k = 1.05\theta_{k-1}$ is assumed for simplicity, *i.e.*, a_{ij}^q constantly grows by 5% every time it increases. Lastly, it is worth noting $a_{ij}^q(0) = \alpha_0$ always.

4.2.2 Problem Description

The FMS scheduling problem that minimizes the sum of energy cost and tardiness penalty under power consumption uncertainties can be formulated as

$$\min_{s \in \Theta} J(s) = \sum_{r_j \in R} \sum_{\pi_q \in \Pi} \sum_{v_i^q \in \omega_q} \sum_{m=1}^{\varphi(\pi_q)} y_{ijm}^q a_{ij}^q(t) d_{ij}^q + \sum_{\pi_q \in \Pi} w_q \tau_q, \tag{4.4}$$

where τ_q and w_q denotes tardiness and penalty of time unit tardiness of π_q, respectively. $J(s)$ denotes the weighted sum of energy consumption and makespan to be minimized. y_{ijm}^q are decision variables such that $y_{ijm}^q = 1$ if job v_i^q on part m ($m = 1, 2, \ldots, \varphi(\pi_q)$) is assigned to resource r_j; $y_{ijm}^q = 0$, otherwise. Let Θ denote the set of feasible schedules. A feasible schedule s should satisfy the following production constraints:

$$\sum_{r_j \in R} y_{ijm}^q = 1, \tag{4.5}$$

$$t_{(i+1)m}^q \geq t_{im}^q + \sum_{r_j \in R} y_{ijm}^q d_{ij}^q, \tag{4.6}$$

$$y_{ijm}^q + y_{i'jm'}^{q'} \geq 1 + \left(\theta_{im}^q + \theta_{i'm'}^{q'}\right), \tag{4.7}$$

$$t_{im}^q \geq t_{i'm'}^{q'} + \sum_{r_j \in R} y_{i'jm'}^{q'} d_{i'j}^{q'} - M(1 - \theta_{i'm'}^{q'}), \tag{4.8}$$

$$t_{i'm'}^{q'} \geq t_{im}^q + \sum_{r_j \in R} y_{ijm}^q d_{ij}^q - M(1 - \theta_{im}^q), \tag{4.9}$$

$$\tau_q = t_{|\omega_q|\varphi(\pi_q)}^q + \sum_{r_j \in R} y_{|\omega_q|j\varphi(\pi_q)}^q d_{|\omega_q|j}^q - D^q, \tag{4.10}$$

$$\forall \pi_q \in \Pi,\ \forall v_i^q \in \omega_q,\ m = 1, 2, \ldots, \varphi(\pi_q), \theta_{im}^q \in \{0, 1\}, \tag{4.11}$$

where t_{im}^q denotes the starting time of v_i^q on part m, θ_{im}^q is a dummy variable, M is a large number for big M method, and D^q denotes the due date of part type π_q. (4.5) ensures that each job needs only one machine at a time. (4.6) specifies the precedence constraints due to the order in which the jobs need to be done for each part. (4.7)–(4.9) guarantee that each resource can process at most one job at a time and jobs cannot be preempted once started. (4.7) functions as an indicator such that if $y_{ijm}^q = y_{i'jm'}^{q'} = 1$ then (4.9) and (4.10) will work in such a way that only one of them will hold. Finally, (4.10) defines the tardiness of each part type. It is worth noting that though the energy cost and tardiness penalty are weighted equally in this chapter, decision makers can adjust the weights based on the specific economic situations.

4.3 Fast Reactive Scheduling

At a given dispatching epoch, the ADEC determines in (2.19) which rules can be activated without causing a deadlock. When multiple uninhibited rules are ready to be activated and multiple disjunctive resources are available to be assigned, the MTME must be adopted as tie-break rule to select the most effective schedules to execute, such that the predefined performance criteria are optimized. Based on the ADEC models of FMSs, this section develops a fast reactive scheduling model that optimizes the throughput and energy cost of the FMS at every dispatching epoch. Throughput is adopted here as an objective function as throughput maximization was verified to effectively reduce the tardiness penalty of FMS [116]. For generality, it should be noted that any kind of reactive scheduling approaches, with any kind of optimization criteria and constraints, can be easily combined with the ADEC in such a way to be described as follows.

4.3.1 Solution Overview

From a global viewpoint, it is convenient to view the framework architecture as an aggregation of two interacting modules, the ADEC and the MTME, both are programmed using the same matrix language. The framework provides a complete description of the discrete event dynamics of an FMS; and is used 1) as a means to track active job/resource statues and sequence deadlock-free imminent jobs and outputs, and 2) to identify the optimal schedule of jobs and resources at each distancing epoch.

As shown in Figure 4.1, its inputs are the acknowledgement messages from the FMS sensors for resource availability and job completion (vectors $\mathbf{v}_c$ and $\mathbf{r}_c$), the information about the arrival of new part inputs (vector $\mathbf{u}$), and the deadlock avoidance control (vector $\mathbf{u}_d$) computed by a DAP. Using these information, the logic conditions of control rules (vector $\mathbf{g}$) is computed by the ADEC.

The interaction of the ADEC and the MTME includes two phases. First, the ADEC computes and passes the deadlock-free search space (choice sets) to the MTME (resource set R_a and rule set G_a). Second, the MTME identifies the optimal assignment of rules and resources (matrices $\mathbf{F}_{sr}$ and $\mathbf{F}_{sd}$) that optimizes (4.4) without violating the required production constraints. It is worth noting that the inputs to the MTME are also real-time power consumption and processing time of resources (matrices $\mathbf{A}(t)$ and $\mathbf{D}$) obtained by a real-time energy monitoring network. The framework's outputs are vectors describing the conditions of the jobs to be start (vector $\mathbf{v}_s$) and the part output to be released (vector $\mathbf{y}$). All the mentioned tasks of the two modules are performed by means of matrix equations. The FMS sensors returns acknowledgements for job completion, for the subsequent release of the resource, and about the arrival of new part inputs.

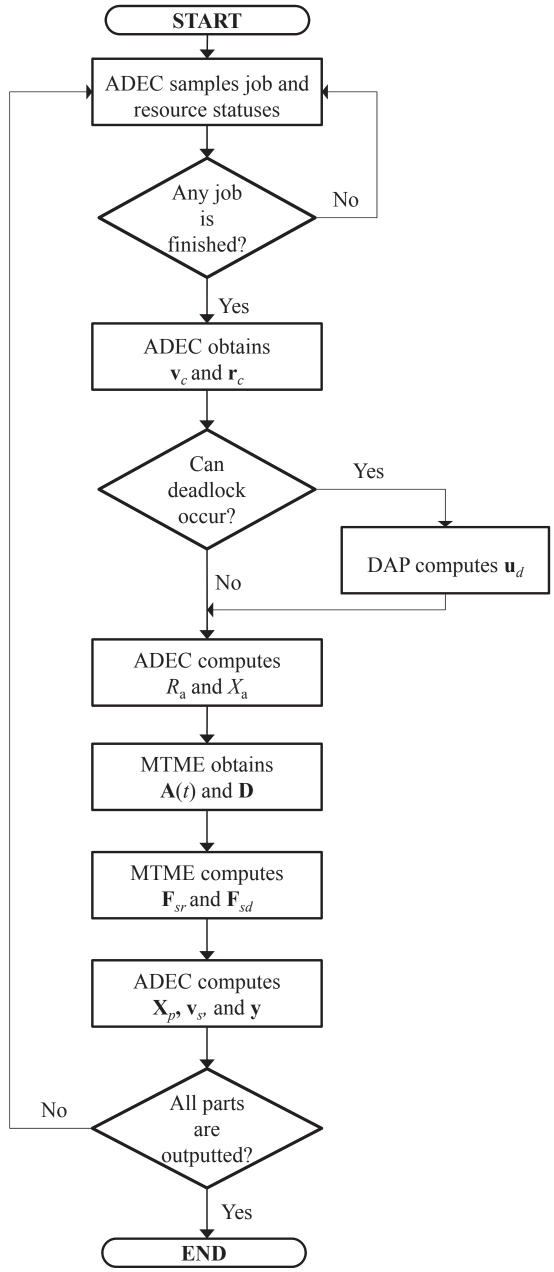


Figure 4.1: Simplified flowchart of our proposed framework. The ADEC replicates the discrete-event dynamics of the system jobs and resources. The MTME decides the local optimal schedule of active jobs and resources.

4.3.2 Reduction of Model Complexity

The key benefit provided by using the ADEC is reduction of the model complexity for implementing large-scale FMSs. If one compares our framework with the conjunctive DEC, then usage of our framework requires less memory. It should be noted that one dimension of all DEC (ADEC) matrices is $|G|$ (*e.g.,* $|G| \times |J|$ for $\mathbf{F}_v$, $|V| \times |G|$ for $\mathbf{S}_v$, $|G| \times |R|$ for $\mathbf{F}_r$, *etc.*), where $|G|$, $|V|$, and $|R|$ are the numbers of rules, jobs, and resources, respectively. Recall that the conjunctive DEC needs p rules to describe the starting of a choice job, which can be performed by p different resources. This drastically increases $|G|$. In our framework, a new matrix $\mathbf{F}_{rd}$ (dimensions of $|G| \times |R|$) is included to keep $|G|$ minimized. Since $|G| \gg |R|$ in large-scale FMSs, the reduction in the model complexity can be significant.

It is also worth noting that the ADEC described in (2.19) is more general than a PN. In fact, the first two and last two terms are equivalent to a PN, while the middle term allows additional **OR** reasoning in the rule bases. To further exemplify this, let consider the FMS recently presented in [76]. Although, this system was considered as a place-timed PN, the timing is ignored in this dissertation.

In part type π_2 of this FMS, there are three machining jobs (p_{22}, p_{24}, and p_{26}), two buffering jobs (p_{23} and p_{25}), three resources (r_1–r_3), an input p_{21}, and an output p_{27}. This part type is therefore characterized by a job sequence $\omega_2 = \{p_{22}, p_{23}, \ldots, p_{26}\}$ and a set of resources $R = \{r_1, r_2, r_3\}$. Job p_{22} is not a choice job, while job p_{24} and job p_{26} are. Choice job p_{24} can be done by either resource r_2 or resource r_3, while choice job p_{26} can be done by either resource r_1 or resource r_3. The PN models of this part type is presented in Figure 4.2. It can be seen that the PN, which only contains **AND** reasoning, requires p branches (each branch contains two transitions and one place noted by dashed circles) to represent a choice job that is processed by p disjunctive resources.

For example, consider choice job p_{24}. The branch containing place p_{24} presents the case where this choice job is processed by resource r_2, while the token will flow to the branch containing place p_{242} if resource r_3 is assigned instead. To switch between these branches (resource routing), controlled places are added to the PN accordingly.

Next, a PN-equivalent ADEC models of this part type is presented in Figure 4.3. It can be seen that the ADEC only needs one branch to represent a choice job regardless of the number of processable resources. The resource routing is decided by switching corresponding resources (noted by dashed circles) not branches. This significantly reduces the model complexity for modelling large-scale FMSs.

4.3.3 Choice Set

Prior to formulating the MTME, it is needed to identify its *search space* or *choice set*. It can be seen that (2.19) determines the set of deadlock-free rules, denoted by $G_{a_{k+1}}$, which can be activated at dispatching epoch $k+1$, where $G_a = supp(\mathbf{g})$. G_a can always be partitioned into two disjoint subsets $G_a = G_z \cup G_{nz}$; where G_z denotes

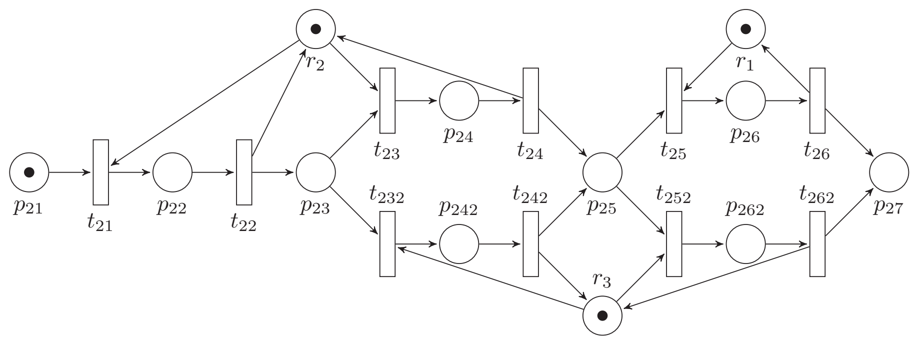


Figure 4.2: PN models of example part type.

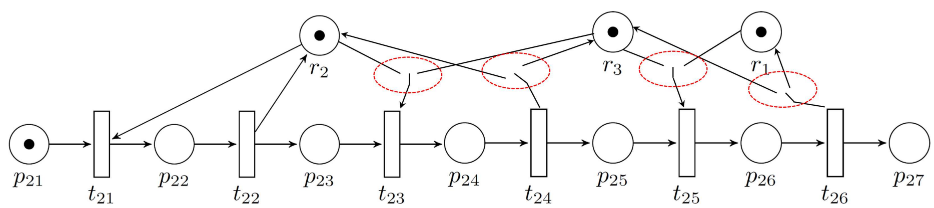


Figure 4.3: PN-equivalent ADEC models of example part type.

the set of choice rules, *i.e.*, rules which start choice jobs, and G_{nz} denotes the set of nonchoice rules.

In the resource domain, denote by R_{fk+1} as a set of resources which accomplish the rule set G_{ak+1}. A resource vector that represents R_{fk+1} is calculated by

$$\mathbf{r}_{fk+1}^{T} = \mathbf{g}_{k+1}^{T} \otimes \left(\mathbf{F}_r \oplus \mathbf{F}_{rd}\right), \tag{4.12}$$

where $R_f = supp\left(\mathbf{r}_f\right)$. In addition, denote the set of available resources by R_{ck+1}, where $R_c = supp\left(\mathbf{r}_c\right)$. Let $R_a = R_c \cap R_f$. It can be clearly seen that the search space of MTME, which includes all possible schedules at dispatching epoch $k+1$, is defined by a resource set R_{ak+1} and a rule set G_{ak+1}.

4.3.4 Min-Throughput-Max-Energy Reactive Scheduling

To compress the scheduling decisions into convenient matrix forms, at dispatching epoch k define $\mathbf{F}_{rd}$'s submatrix $\mathbf{F}_{sdk}$; such that in the case of multiple entries of "1" (choice job) in a row of $\mathbf{F}_{rd}$, submatrix $\mathbf{F}_{sdk}$ comprises at most one "1" referring to the resource selected to process the corresponding choice job, and in the case of multiple entries of "1" (shared resources) in a column of $\mathbf{F}_{rd}$, submatrix $\mathbf{F}_{sdk}$ comprises at most one "1" referring to the rule selected to be fired. Analogously, define $\mathbf{F}_r$'s submatrix $\mathbf{F}_{srk}$; such that in the case of multiple entries of "1" in a column of $\mathbf{F}_{rd}$, submatrix $\mathbf{F}_{srk}$ comprises at most one "1". As such, $\mathbf{F}_{srk}$ and $\mathbf{F}_{sdk}$ are assembled in the ADEC logical state equation by

$$\overline{\mathbf{g}_{pk+1}} = \mathbf{F}_v \otimes \overline{\mathbf{v}_{ck}} \oplus (\mathbf{F}_r \circ \mathbf{F}_{srk}) \otimes \overline{\mathbf{r}_{ck}} \oplus \mathbf{F}_u \otimes \overline{\mathbf{u}_k} \oplus \overline{(\mathbf{F}_{rd} \circ \mathbf{F}_{sdk}) \otimes \mathbf{r}_{ck}} \oplus \mathbf{F}_{ud} \otimes \overline{\mathbf{u}_{dk}}, \tag{4.13}$$

where $\circ$ denotes the Hadamard product (piecewise multiplication) with $\mathbf{C} = \mathbf{A} \circ \mathbf{B}$ is defined by $c_{ij} = a_{ij} \times b_{ij}$. G_{pk}, $G_p = supp\,(\mathbf{g}_p)$, presents a set of rules that include the scheduling decisions and will be eventually activated at dispatching epoch $k+1$.

The MTME at dispatching epoch k is presented as follows, where index k is dropped in all mathematical notations for brevity, *i.e.,* f^{sd}_{qijk} is simplified to f^{sd}_{qij}, where f^{sd}_{qijk} is the element of $\mathbf{F}_{sdk}$. The MTME comprises of two 0–1 LP submodels. The former computes the maximum throughput achievable; and the latter, among solutions of the former, decides the one with least energy cost. The first submodel is given by

$$\max \quad \delta = \sum_{\pi_q \in \Pi} \sum_{\mathrm{g}_j^q \in G_a} \sum_{r_j \in R_a} \left(f_{qij}^{sd} + f_{qij}^{sr}\right), \tag{4.14}$$

s.t.,

$$\sum_{r_j \in R_a} f_{qij}^{sd} \leq 1, \forall \mathrm{g}_j^q \in G_z, \tag{4.15}$$

$$\sum_{\mathrm{g}_j^q \in G_z} f_{qij}^{sd} + \sum_{\mathrm{g}_j^q \in G_{nz}} f_{qij}^{sr} \leq 1, \forall r_j \in R_a, \tag{4.16}$$

$$f_{qij}^{sd}, f_{qij}^{sr} \in \{0, 1\}, \forall (\mathrm{g}_j^q, r_j) \in G_a \times R_a, \tag{4.17}$$

where (4.14) is the cost function of throughput to be maximized. (4.15) essentially constraints the solution to select one and only one resource for each rule of $\mathrm{g}_j^q \in G_{zk}$, while (4.16) avoids shared-resource conflicts (if any). (4.17) is a mapping constraint which implies how resources and rules are indexed.

With a solution of the first submodel, it is now proceeded to express the second submodel by

$$\min \quad \sum_{\pi_q \in \Pi} \sum_{\mathrm{g}_j^q \in G_a} \sum_{r_j \in R_a} a_{ij}^q d_{ij}^q \left(f_{qij}^{sd} + f_{qij}^{sr}\right), \tag{4.18}$$

s.t.,

$$\sum_{\pi_q \in \Pi} \sum_{\mathrm{g}_j^q \in G_a} \sum_{r_j \in R_a} \left(f_{qij}^{sd} + f_{qij}^{sr}\right) = \nu \delta_{\max}, \tag{4.19}$$

(4.15)-–(4.17),

where (4.18) is the cost function of energy cost to be minimized. (4.19) depicts the constraint of the minimum throughput must be achieved, where $c \in \mathbb{R}_{0 \le \nu \le 1}$ is a weight parameter. As the energy cost and tardiness penalty are weighted equally in (4.4), $\nu = 0.5$ is chosen herein. ν can be adjusted depending on how the dynamic scheduling problem is formulated. $a_{ij}^{q}(t)$ are real-time productive power measurements from the meters. Finally, f_{ijk}^{sdq} and f_{qijk}^{srq} denote the elements of $\mathbf{F}_{sdk}^{q}$ and $\mathbf{F}_{srk}^{q}$, respectively.

The MTME is formulated as a standard 0-1 (binary) LP, which is classified as NP-hard in computational complexity theory. Advanced algorithms for solving 0-1 LP include cutting-plane method, B&B, branch and cut, branch and price, *etc.*, each method has its own pros and cons. In this chapter, a specialized B&B algorithm known as Balas additive algorithm is chosen as solution method, which is widely available in commercial solvers [117]. Branching is done similarly with other B&B algorithms by letting each decision variable take on only one of two values 0 or 1. Bounding is done differently as compared to other B&B algorithms. Balas algorithm does not perform look-ahead to complete the solution or its simplified counterpart. Instead, the bounding function of Balas algorithm optimizes at the cost of the next cheapest solution that might provide a feasible solution.

In general, B&B algorithms have exponential worst-case (W-C) complexity on the problem size, but the average-case complexity is significantly lower. Figuring out the average-case complexity is much more difficult than figuring out either the worst-case or best-case because we have to identify a given probability distribution for input data. For example, in [118], it is assumed that p_0, which is the probability that each edge of B&B search tree has zero cost, is known *a priori*. Let B be the branching factor of B&B algorithm which is the number of children at each node of B&B search tree. Let d_B be the depth of the B&B search tree. The average-case complexity was proven to be linear in d_B if $Bp_0 > 1$ and to be exponential in d_B if $Bp_0 < 1$.

At every dispatching epoch, the MTME generates online schedules in a local Pareto optimal way, and the global Pareto front can be calculated in certain scenarios. Unlike deterministic multi-objective optimization problems, whose Pareto optimal solutions are commonly generated using evolutionary algorithm (EA) or GA In addition, since our proposed framework functions as an online scheduler, it is not possible to include advanced algorithms such as EA and GA, which require long computation times and induce disruptions for the production flow of FMSs.

4.4 Industrial Application

An industrial application is carried out to verify the usability of our prosed scheduling method. An application related to stamping system is selected for the experiment. The energy data are monitored at a stamping company in the Republic of Singapore. This stamping system can be characterized by the class of FMSs described in Chapter 2. Each stamping part type has a predetermined sequence of jobs, with some jobs can be processed by more than one resource, and some resources can perform more than one job. At this stamping system, the scheduling task is primarily decided based on human decisions.

4.4.1 Energy Analysis of Stamping Process

In this application, input parts are raw metal sheets, while output parts are various types of voil coil motor (VCM) yokes used in commercial hard disk drive (HDD) actuators. A typical VCM comprises of a coil rotatable about a predetermined axis; a pair of yokes opposing each other with a predetermined distance; and a permanent magnet between the pair of yokes. VCM yokes are used to harness the strong permanent magnets. VCM yokes are usually manufactured massively by progressive stamping systems. To avoid even tiny particles penetrating into HDDs, the VCM components must be then assembled in extreme clean rooms. An example of VCM yokes is shown in Figure 4.4. The stamping system comprises of eight stamping machines, which are only named by M1–M1 due to confidential restrictions. Energy consumption is continually monitored using RUDOLFs. To interface RUDOLFs with computers or handled devices, a graphic user interface (GUI) has been developed in LabVIEW.

The stamping machines are of different working conditions as well as energy consumption profiles, and their performances and efficiencies are summarised in Table 4.1. It can be seen that there is a wide range in average stamping power, even for different machines of the same model. This is due to a multitude of factors, *e.g.,* tooling, machine loading, machine degradation, machine age, *etc.* The entire stamping cycle can be divided into three main states, namely, productive, idle, and off as shown Fig-

Figure 4.4: An example of VCM yokes.

ure 4.5. In idle and off states, the power data are observed to be relatively constant. In productive state, many spikes are generated, and each spike is observed every time the stamping press moves down to perform stamping operations.

Table 4.1: Machine Performance and Efficiency

Machine ID	Rated tonnage (tonnes)	Motor rated power (tonnes)	Actual max load (tonnes)	Average stamping power (kW)
M1	200	22	168	11.96
M2	300	37	238	4.45
M3	300	37	250	7.60
M4	300	37	183	6.19
M5	300	37	176	5.37
M6	300	37	198	6.46
M7	300	37	202	7.84
M8	300	37	–	12.23

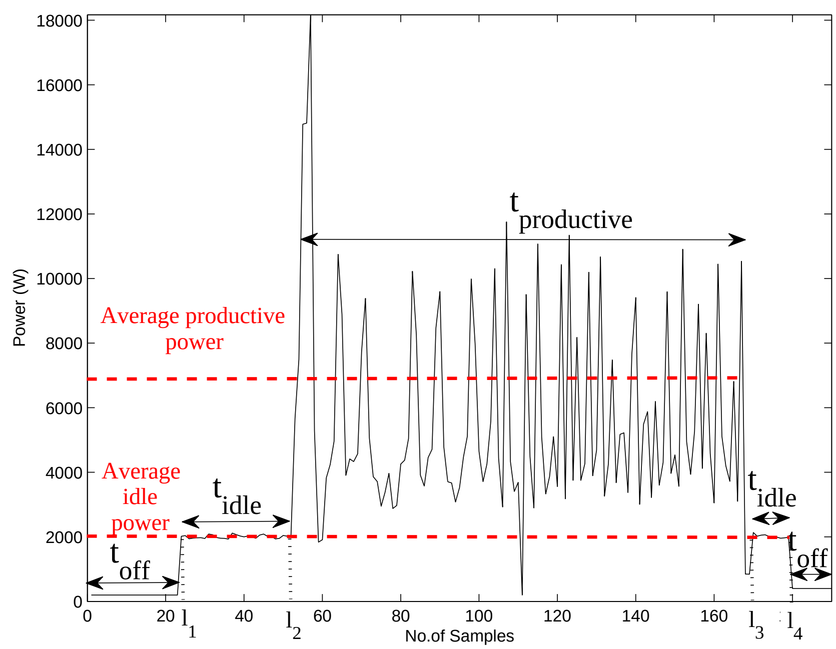


Figure 4.5: Typical power profile of stamping process.

Using the measured power data, the productive power matrices $\mathbf{A}^q$, the idle power vector $\mathbf{b}$, and the processing time matrices $\mathbf{D}^q$ can be constructed as follows. Let P_m, where m is the number of samples, be the power profile shown in Figure 4.5 measured on resource r_i when performing job v_j^q. As such, one has

$$a_{ij}^q = \frac{1}{l_3 - l_2} \sum_{m=l_3}^{l_2} P_m, \tag{4.20}$$

$$b_i = \frac{1}{l_2 - l_1 + l_4 - l_3} \left(\sum_{m=l_1}^{l_2} P_m + \sum_{m=l_3}^{l_4} P_m \right), \tag{4.21}$$

$$d_{ij}^q = \frac{l_3 - l_2}{f_s}, \tag{4.22}$$

where l_1–l_4 denote the instances that the state is changed from *off* to *idle*, *idle* to *productive*, *productive* to *idle*, and *idle* to *off*, respectively. f_s is the sampling frequency of RUDOLFs.

4.4.2 Augmented Discrete Event Control Models of Stamping System

From (4.20) and (4.22), the initial productive power matrices $\mathbf{A}^q(0)$ (kW) and the processing time matrices $\mathbf{D}^q$ (s) of the stamping system are obtained as shown in (4.23)–(4.26).

$$(\mathbf{A}^1)^T = \begin{array}{c} \\ v_1^1 \\ v_2^1 \\ v_3^1 \\ v_4^1 \\ v_5^1 \\ v_6^1 \\ v_7^1 \end{array} \begin{array}{c} \begin{array}{ccccccccccccc} \mathrm{M1} & \mathrm{M2} & \mathrm{M3} & \mathrm{M4} & \mathrm{M5} & \mathrm{M6} & \mathrm{M7} & \mathrm{M8} & \mathrm{B1} & \mathrm{B2} & \mathrm{B3} & \mathrm{B4} & \mathrm{B5} \end{array} \\ \begin{bmatrix} 10.72 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2.40 & 0 & 0 & 0 & 0 \\ 0 & 4.32 & 7.45 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2.40 & 0 & 0 & 0 \\ 0 & 0 & 7.73 & 6.22 & 5.24 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2.40 & 0 & 0 \\ 0 & 4.86 & 0 & 6.06 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array}, \tag{4.23}$$

$$
(\mathbf{D}^1)^T = \begin{array}{c} \\ v_1^1 \\ v_2^1 \\ v_3^1 \\ v_4^1 \\ v_5^1 \\ v_6^1 \\ v_7^1 \end{array} \begin{array}{c} \begin{array}{ccccccccccccc} \mathrm{M1} & \mathrm{M2} & \mathrm{M3} & \mathrm{M4} & \mathrm{M5} & \mathrm{M6} & \mathrm{M7} & \mathrm{M8} & \mathrm{B1} & \mathrm{B2} & \mathrm{B3} & \mathrm{B4} & \mathrm{B5} \end{array} \\ \begin{bmatrix} 4.76 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 3.89 & 3.48 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 4.26 & 5.68 & 4.37 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 5.21 & 0 & 4.46 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array}, \tag{4.24}
$$

$$
(\mathbf{A}^2)^T = \begin{array}{c} \\ v_1^2 \\ v_2^2 \\ v_3^2 \\ v_4^2 \\ v_5^2 \end{array} \begin{array}{c} \begin{array}{ccccccccccccc} \mathrm{M1} & \mathrm{M2} & \mathrm{M3} & \mathrm{M4} & \mathrm{M5} & \mathrm{M6} & \mathrm{M7} & \mathrm{M8} & \mathrm{B1} & \mathrm{B2} & \mathrm{B3} & \mathrm{B4} & \mathrm{B5} \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2.4 & 0 \\ 4.68 & 0 & 0 & 0 & 0 & 6.86 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2.4 \\ 0 & 0 & 0 & 0 & 0 & 0 & 7.84 & 12.78 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 5.64 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array}, \tag{4.25}
$$

$$
(\mathbf{D}^2)^T = \begin{array}{c} \\ v_1^2 \\ v_2^2 \\ v_3^2 \\ v_4^2 \\ v_5^2 \end{array}
\begin{array}{c}
\begin{array}{ccccccccccccc} \mathrm{M1} & \mathrm{M2} & \mathrm{M3} & \mathrm{M4} & \mathrm{M5} & \mathrm{M6} & \mathrm{M7} & \mathrm{M8} & \mathrm{B1} & \mathrm{B2} & \mathrm{B3} & \mathrm{B4} & \mathrm{B5} \end{array} \\
\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
3.73 & 0 & 0 & 0 & 0 & 4.30 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 5.27 & 5.52 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 5.29 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\end{array}, \tag{4.26}
$$

The jobs and possible resource assignments of part types π_1 and π_2 are reported in Figure 5.4. The IF-THEN rule bases of π_1 and π_2 are reported in Tables 4.2 and 4.3, respectively.

Table 4.2: Part Type π_1–Rule Bases

Rule	Notation	Description
Rule 1	g_1^1	**IF** v_{in}^1 *is ready* **AND** M1 *is free* **THEN** *start* v_1^1
Rule 2	g_2^1	**IF** v_1^1 *is done* **AND** B1 *is free* **THEN** *start* v_2^1
Rule 3	g_3^1	**IF** v_2^1 *is done* **AND** (M2 *is free* **OR** M3 *is free*)
		THEN *start* v_3^1
Rule 4	g_4^1	**IF** v_3^1 *is done* **AND** B2 *is free* **THEN** *start* v_4^1
Rule 5	g_5^1	**IF** v_4^1 *is done* **AND** (M3 *is free* **OR** M4 *is free*
		OR M5 *is free*) **THEN** *start* v_5^1
Rule 6	g_6^1	**IF** v_5^1 *is done* **AND** B3 *is free* **THEN** *start* v_6^1
Rule 7	g_7^1	**IF** v_6^1 *is done* **AND** (M2 *is free* **OR** M4 *is free*)
		THEN *start* v_7^1
Rule 8	g_8^1	**IF** v_7^1 *is done* **THEN** *release* v_{out}^1

The rule bases of two part types are now represented by means of Boolean matrices $\mathbf{F}_v^q$, $\mathbf{F}_r^q$, $\mathbf{F}_{rd}^q$, $\mathbf{F}_u^q$ as shown in (4.27)–(4.29) and (4.30)–(4.32), respectively. It is

Table 4.3: Part Type π_2–Rule Bases

Rule	Notation	Description
Rule 1	g_1^2	**IF** v_{in}^2 *is ready* **AND** B4 *is free* **THEN** *start* v_1^2
Rule 2	g_2^2	**IF** v_1^2 *is done* **AND** (M1 *is free* **OR** M6 *is free*)
		THEN *start* v_2^2
Rule 3	g_3^2	**IF** v_2^2 *is done* **AND** B5 *is free* **THEN** *start* v_3^2
Rule 4	g_4^2	**IF** v_3^2 *is done* **AND** (M7 *is free* **OR** M8 *is free*)
		THEN *start* v_4^2
Rule 5	g_5^2	**IF** v_4^2 *is done* **AND** (M5 *is free* **THEN** *start* v_5^2
Rule 6	g_6^2	**IF** v_5^2 *is done* **THEN** *release* v_{out}^2

noted that the contents of matrices $\mathbf{S}_r^q$ and $\mathbf{S}_y^q$ are omitted for brevity.

$$\mathbf{F}_v^1 = \begin{array}{c} \\ \mathrm{g}_1^1 \\ \mathrm{g}_2^1 \\ \mathrm{g}_3^1 \\ \mathrm{g}_4^1 \\ \mathrm{g}_5^1 \\ \mathrm{g}_6^1 \\ \mathrm{g}_7^1 \\ \mathrm{g}_8^1 \end{array} \begin{array}{c} \begin{array}{ccccccc} v_1^1 & v_2^1 & v_3^1 & v_4^1 & v_5^1 & v_6^1 & v_7^1 \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{array}, \quad \mathbf{F}_u^1 = \begin{array}{c} \\ \mathrm{g}_1^1 \\ \mathrm{g}_2^1 \\ \mathrm{g}_3^1 \\ \mathrm{g}_4^1 \\ \mathrm{g}_5^1 \\ \mathrm{g}_6^1 \\ \mathrm{g}_7^1 \\ \mathrm{g}_8^1 \end{array} \begin{array}{c} u_1^1 \\ \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \end{array}, \tag{4.27}$$

$$
\mathbf{F}_r^1 = \begin{array}{cc}
 & \begin{array}{ccccccccccccc} \mathrm{M1} & \mathrm{M2} & \mathrm{M3} & \mathrm{M4} & \mathrm{M5} & \mathrm{M6} & \mathrm{M7} & \mathrm{M8} & \mathrm{B1} & \mathrm{B2} & \mathrm{B3} & \mathrm{B4} & \mathrm{B5} \end{array} \\
\begin{array}{c} \mathrm{g}_1^1 \\ \mathrm{g}_2^1 \\ \mathrm{g}_3^1 \\ \mathrm{g}_4^1 \\ \mathrm{g}_5^1 \\ \mathrm{g}_6^1 \\ \mathrm{g}_7^1 \\ \mathrm{g}_8^1 \end{array} &
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\end{array}, \tag{4.28}
$$

$$
\mathbf{F}_{rd}^1 = \begin{array}{cc}
 & \begin{array}{ccccccccccccc} \mathrm{M1} & \mathrm{M2} & \mathrm{M3} & \mathrm{M4} & \mathrm{M5} & \mathrm{M6} & \mathrm{M7} & \mathrm{M8} & \mathrm{B1} & \mathrm{B2} & \mathrm{B3} & \mathrm{B4} & \mathrm{B5} \end{array} \\
\begin{array}{c} \mathrm{g}_1^1 \\ \mathrm{g}_2^1 \\ \mathrm{g}_3^1 \\ \mathrm{g}_4^1 \\ \mathrm{g}_5^1 \\ \mathrm{g}_6^1 \\ \mathrm{g}_7^1 \\ \mathrm{g}_8^1 \end{array} &
\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\end{array}, \tag{4.29}
$$

$$
\mathbf{F}_v^2 = \begin{array}{c} \\ \mathrm{g}_1^2 \\ \mathrm{g}_2^2 \\ \mathrm{g}_3^2 \\ \mathrm{g}_4^2 \\ \mathrm{g}_5^2 \\ \mathrm{g}_6^2 \end{array}
\begin{array}{c}
\begin{array}{ccccc} v_1^2 & v_2^2 & v_3^2 & v_4^2 & v_5^2 \end{array} \\
\begin{bmatrix}
0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1
\end{bmatrix}
\end{array},\
\mathbf{F}_u^2 = \begin{array}{c} \\ \mathrm{g}_1^2 \\ \mathrm{g}_2^2 \\ \mathrm{g}_3^2 \\ \mathrm{g}_4^2 \\ \mathrm{g}_5^2 \\ \mathrm{g}_6^2 \end{array}
\begin{array}{c}
u_1^2 \\
\begin{bmatrix}
1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0
\end{bmatrix}
\end{array}, \tag{4.30}
$$

$$
\mathbf{F}_r^2 = \begin{array}{c} \\ \mathrm{g}_1^2 \\ \mathrm{g}_2^2 \\ \mathrm{g}_3^2 \\ \mathrm{g}_4^2 \\ \mathrm{g}_5^2 \\ \mathrm{g}_6^2 \end{array}
\begin{array}{c}
\begin{array}{ccccccccccccc} \mathrm{M1} & \mathrm{M2} & \mathrm{M3} & \mathrm{M4} & \mathrm{M5} & \mathrm{M6} & \mathrm{M7} & \mathrm{M8} & \mathrm{B1} & \mathrm{B2} & \mathrm{B3} & \mathrm{B4} & \mathrm{B5} \end{array} \\
\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\end{array}, \tag{4.31}
$$

$$
\mathbf{F}_{rd}^2 = \begin{array}{c} \\ \mathrm{g}_1^2 \\ \mathrm{g}_2^2 \\ \mathrm{g}_3^2 \\ \mathrm{g}_4^2 \\ \mathrm{g}_5^2 \\ \mathrm{g}_6^2 \end{array}
\begin{array}{c}
\begin{array}{ccccccccccccc} \mathrm{M1} & \mathrm{M2} & \mathrm{M3} & \mathrm{M4} & \mathrm{M5} & \mathrm{M6} & \mathrm{M7} & \mathrm{M8} & \mathrm{B1} & \mathrm{B2} & \mathrm{B3} & \mathrm{B4} & \mathrm{B5} \end{array} \\
\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\end{array}. \tag{4.32}
$$

Concatenating these mission matrices, the overall system's matrix description is obtained by

$$
\mathbf{F}_v = \begin{bmatrix} \mathbf{F}_v^1 & 0 \\ 0 & \mathbf{F}_v^2 \end{bmatrix}, \mathbf{F}_r = \begin{bmatrix} \mathbf{F}_v^1 \\ \mathbf{F}_v^2 \end{bmatrix}, \mathbf{F}_{rd} = \begin{bmatrix} \mathbf{F}_{rd}^1 \\ \mathbf{F}_{rd}^2 \end{bmatrix}, \mathbf{F}_u = \begin{bmatrix} \mathbf{F}_u^1 & 0 \\ 0 & \mathbf{F}_u^2 \end{bmatrix}. \tag{4.33}
$$

4.4.3 Experiment Results

In this section, we will show the test results of applying the proposed framework and demonstrate the solution quality by comparing with the schedules obtained by three dispatching rules, CR, and PR approaches. The considered dispatching rules include shortest processing time first (SPT), least energy cost first (LEC), and first come first served (FCFS) [111]. The SPT rule sequences the jobs so that the job which takes the shortest time to process is first to be performed. The LEC rule gives the priority to the job which has the least energy cost to be scheduled first. The FCFS rule sequences the jobs starting with the current time period and working forward. In predictive-reactive approaches, the baseline schedule and the reschedules are generated using particle swarm optimization [114]. In CR approach, a totally new schedule is regenerated, while only the jobs and resources which are affected by the power consumption uncertainties are rescheduled in PR approach.

Since jobs are not preemptive, the reschedules are only applied if all ongoing jobs from the previous schedules have been completed. The rescheduling is triggered if any $a_{ij}^{q}(t)$ increases. On the contrary, MTME and dispatching rules are triggered if any job is completed and there are sets of jobs and resources to be dispatched. These approaches are used to solve (4.4) and the obtained objective values are compared under three probability distributions of $g\left(b; \mu_{ij}^{q}, \left(\sigma_{ij}^{q}\right)^{2}\right)$, namely, the Weibull, exponential, and truncated normal distributions. The distribution mean and variance are reported in Table 4.4.

Table 4.4: Mean and Variance of Power Consumption Uncertainties μ_{ij}^{q}, $\left(\sigma_{ij}^{q}\right)^{2}$

	M1	M2	M3	M4	M5	M6	M7	M8
v_1^1	70,140	–	–	–	–	–	–	–
v_3^1	–	40,80	50,100	–	–	–	–	–
v_5^1	–	–	80,160	60,120	70,140	–	–	–
v_7^1	–	50,100	–	70,140	–	–	–	–
v_2^2	50,100	–	–	–	–	60,120	–	–
v_4^2	–	–	–	–	–	–	80,160	90,180
v_5^2	–	–	–	–	70,140	–	–	–

Two performance metrics are of interest, which are the mean interrupted time to the production flow and deviation from Pareto optimality. It is well known in current literature that CR provides the Pareto optimal solutions for dynamic scheduling problems, but these schedules are rarely achieved in practice due to prohibitive disruptions to the FMS. As such, the deviation from Pareto optimality of a scheduling method is computed as follows

$$\mathrm{dev} = \frac{J - J_{\mathrm{CR}}}{J_{\mathrm{CR}}} \times 100\%, \tag{4.34}$$

where J_{CR} is the optimal cost obtained by the CR. The mean interrupted time is computed as follows

$$T_{\mathrm{mean}} = \frac{1}{n}\sum_{i=1}^{n} T_i, \tag{4.35}$$

where T_i is the computational time to generate the i^{th} reschedule, and n denotes the number of reschedules. For all scheduling approaches, the results are obtained after 20 test runs. This experiment is carried on a digital computer equipped with Intel Core i7 processor and 32 gigabyte RAM. All computations are done using MATLAB.

The deviation from Pareto optimality of the schedules generated by three dispatching rules, CR, PR, and MTME in the Weibull distribution, truncated normal distribution, and exponential distribution are reported in Figures 4.6–4.8, respectively, while test results of mean interrupted time are provided in Table 4.5. For simplicity, $\varphi(\pi_1) = \varphi(\pi_2)$ and $w_1 = w_2 = 4.75$ are assumed, and suitable due dates are selected for each test cases. $n = 40$ is set, *i.e.*, each power consumption will increase for 40 times during the entire production.

In terms of deviation from Pareto optimality, MTME outperforms the three dispatching rules for all test cases. The PR outperforms MTME when the batch size is

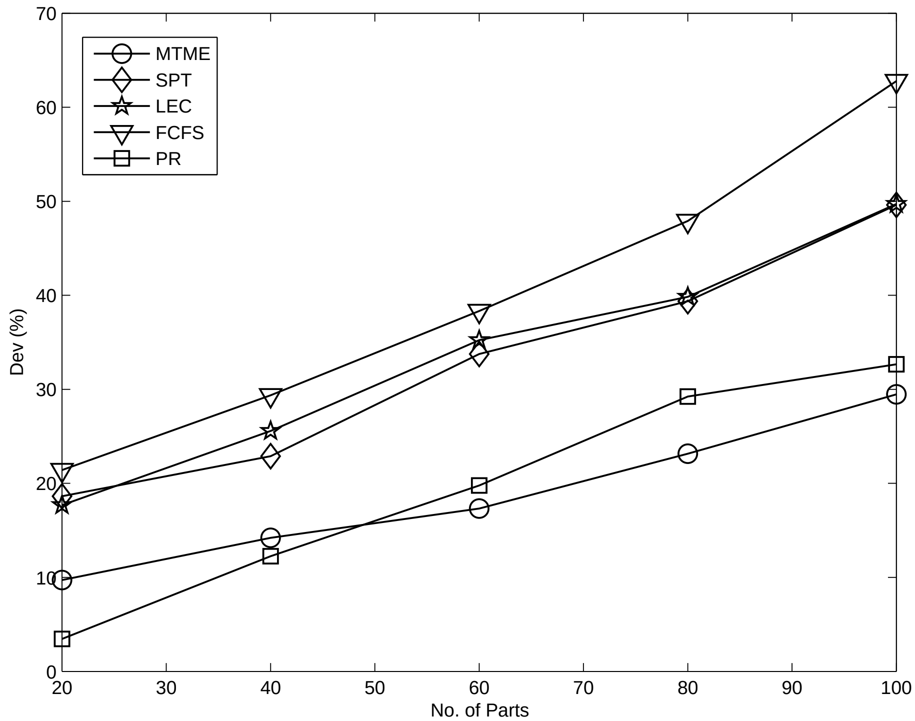


Figure 4.6: Deviation from Pareto optimality under Weibull distribution.

small (short schedules), but the reverse is observed when the batch size is larger than 60 parts (long schedules). In terms of mean interrupted time, MTME achieves less than 1 s for all distributions and batch sizes, while the PR and CR cause prohibitive interrupted time.

4.4.4 Scalability

In this section, the usability of our proposed MTME is verified with different sizes of FMS, *i.e.*, different numbers of jobs and resources. Let rewrite the MTME in a

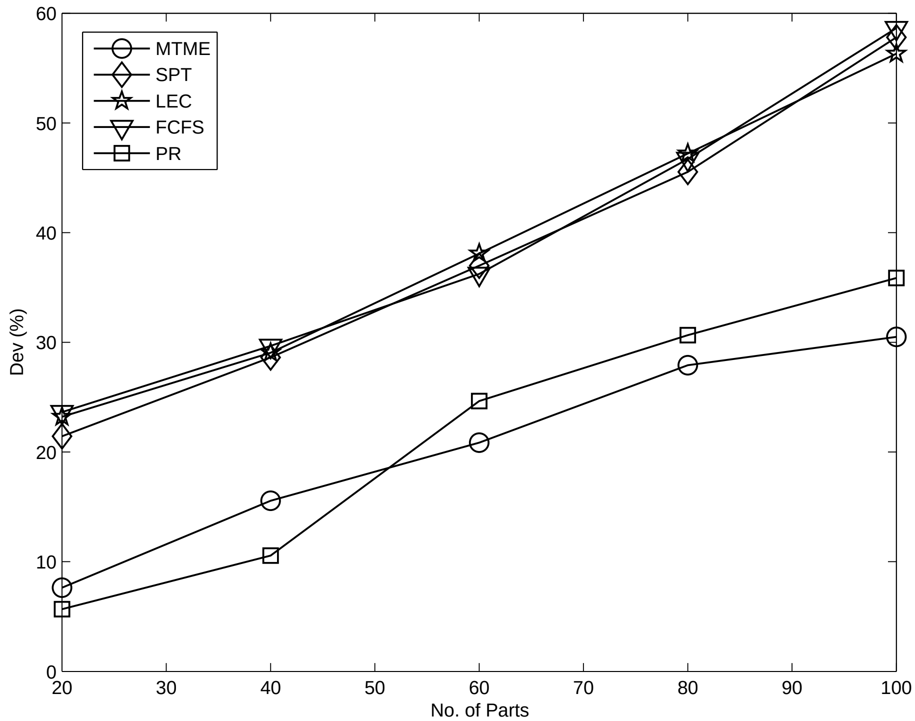


Figure 4.7: Deviation from Pareto optimality under truncated normal distribution.

Table 4.5: Comparison of $T_{\text{mean}}(s)$ under Different Probability Distributions

Parts	Exponential			Weibull			Truncated Normal		
$\varphi\,(\pi)$	MTME	PR	CR	MTME	PR	CR	MTME	PR	CR
20	<1	3.49	23.73	<1	5.23	27.73	<1	6.74	33.23
40	<1	12.73	47.24	<1	16.46	57.24	<1	18.93	68.83
60	<1	20.12	> 100	<1	20.85	> 100	<1	26.54	> 100
80	<1	28.89	> 100	<1	29.48	> 100	<1	31.34	> 100
100	<1	32.73	> 100	<1	37.63	> 100	<1	44.38	> 100

standard 0–1 LP model as follows.

$$\max\ \mathbf{w}^T\zeta, \tag{4.36}$$

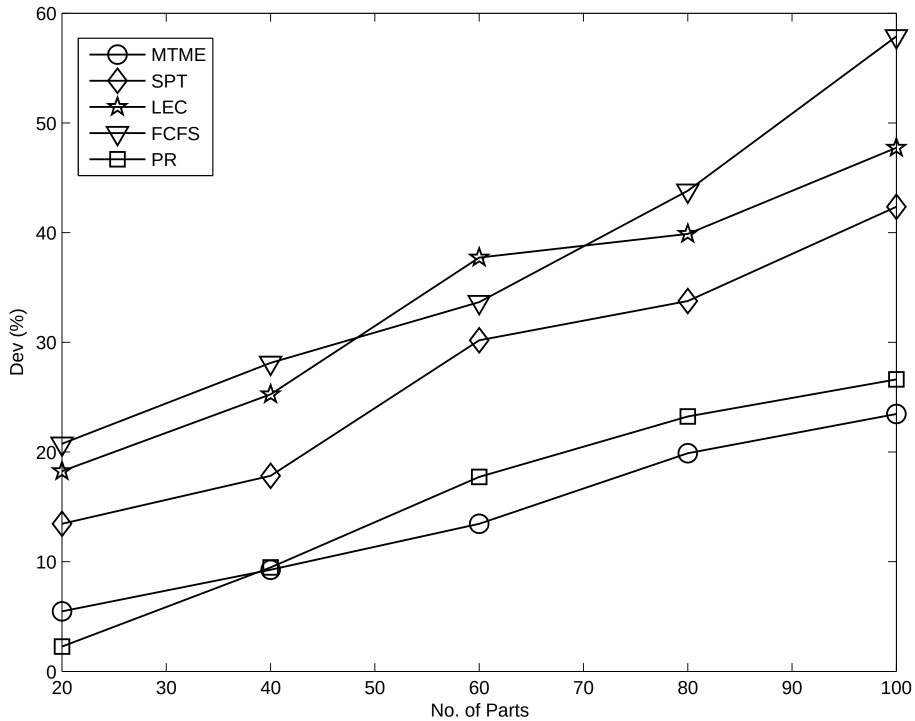


Figure 4.8: Deviation from Pareto optimality under exponential distribution.

s.t.,

$$\mathbf{L}\zeta \leq \mathbf{h}, \tag{4.37}$$

$$\zeta_i \in \{0, 1\}, \forall \zeta_i, \tag{4.38}$$

where ζ represents the vector of variables, $\mathbf{w}$ and $\mathbf{h}$ are vectors of coefficients, and $\mathbf{L}$ is a matrix of coefficients. Computational results for each 0-1 submodel are reported in Table 4.6, where λ is the number of variables, γ is the number of constraints, and σ is the density (the ratio of the number of non-zero elements to the total number of elements) of matrix $\mathbf{L}$. Obviously, the values of λ and d depend on the numbers of

Table 4.6: $T_{\text{mean}}(s)$ of MTME with Different FMS Sizes

$\lambda \times \gamma$	σ	$T_{\text{mean}}(s)$
30×5	0.2	0.039
	0.4	0.054
	0.6	0.065
	0.8	0.147
50×2	0.2	0.112
	0.4	0.091
	0.6	0.130
	0.8	0.609
75×5	0.2	0.197
	0.4	0.394
	0.6	0.630
	0.8	0.775
100×5	0.2	0.240
	0.4	0.679
	0.6	0.589
	0.8	1.535
30× 10	0.8	0.255
50× 10	0.8	1.367
75× 10	0.8	1.832
100× 10	0.8	3.712

jobs and resources in the FMS.

It can be seen that in most problem sizes, each submodel takes less than 2 s of computational time. In addition, the ADEC and the MTME are programmed in the same language of Boolean matrices and vectors, which allows fast deployment of scheduling decisions. These advantages make our framework computationally efficient even for large-scale FMSs.

4.4.5 Discussions with Related Works

At a conceptual level, our proposed framework achieves less deviation from global Pareto optimality as compared to the dispatching rules, as it incorporates optimization into the reactive dispatching while still keeping computational time sufficiently small. As compared to predictive-reactive approaches such as PR and CR, our proposed framework achieves infinitesimal interruption to the FMSs as the MTME's problem size only depends on the currency of the FMSs instead of the batch size. Let MTME, PR, and CR be solved using the same method such as B&B algorithm. The interrupted time of MTME is independent on the batch size, and its problem size (search space) is confined to a set of rules X_{ak+1} and a set of available resources R_{ak+1}, *i.e.*, the degree of concurrency of FMSs. For PR and CR, the problem size is the (partial or complete) set of unfinished jobs and resources by instance the rescheduling is triggered, which is obviously dependent on the batch size. To compare with proactive scheduling, our proposed framework is more convenient in the sense that industrial practitioners need not to explicitly model the power consumption uncertainties *a priori.*

In addition to tardiness penalty and energy cost, our formulation can be potentially applied to other two objectives, such that there is a trade-off between them. For example, the MTME can be used directly to minimize makespan and energy cost. It was shown that the makespan objective is closely related to the throughput objective [119]. Heuristics that tend to minimize the makespan in a machine environment with a finite number of jobs also tend to maximize the throughput rate when there is a constant flow of parts overtime.

Another potential application is minimization of throughput and flow time [120]. Ideally, an FMS should have both a high throughput and a low flow time or low wip. Unfortunately, these objectives are conflicting and can not both be met simultaneously. If a high throughput is required, machines should always be busy. As from time to time disturbances like machine failures happen, buffers between two consecutive machines are required to make sure that the second machine can still continue if the first machine fails (or *vice versa*). For a high throughput many lots are needed in the manufacturing system, *i.e.*, wip needs to be high. As a result, if a new lot starts in the system it has a large flow time, since all lots that are currently in the system need to be completed first.

4.5 Summary

In this chapter, an integrated control and reactive scheduling framework was proposed for improving energy efficiencies in FMSs subjected to power consumption uncertainties. Our proposed framework was rigorously justified with mathematical formulation and its effectiveness was evaluated based on an industrial stamping system, where the stamping parts were various types of VCM yokes used in commercial HDD actuators. The obtained schedules were compared with three dispatching rules and two rescheduling approaches. The experiment results verified that MTME outperforms three dispatching rules in terms of deviation from Pareto optimality and reduces interrupted time significantly as compared to rescheduling approaches.

In the next chapter, we consider an optimal scheduling problem to minimize both productive and idle energy consumption subjected to the general production constraints.

Chapter 5

Total Energy Optimization of Flexible Manufacturing Systems Using Dynamic Programming

Schedule optimization is crucial to reduce energy consumption of flexible manufacturing systems (FMSs) with shared resources and route flexibility. Based on the weighted p-timed Petri net (WTPN) models of FMSs, this chapter considers a scheduling problem that minimizes both productive and idle energy consumption subject to the general production constraints. The considered problem is proven to be a non-convex mixed integer nonlinear program (MINLP). A new reachability graph (RG)-based discrete dynamic programming (DP) approach is proposed for generating near energy-optimal schedules within adequate computational time. The non-convex MINLP is sampled, and the reduced RG is constructed such that only reachable paths are retained for energy-optimal path computation. Each scheduling subproblem is linearized, and each optimal substructure is computed to store in a routing table. It is proven that the sampling-induced error is bounded, and this upper bound can be reduced by increasing the sampling frequency. Experiment results on an industrial stamping system show the effectiveness of our proposed scheduling algorithm in terms of computational complexity and deviation from optimality.

5.1 Background

FMSs are modern production facilities that possess high flexibility of resource allocation and part routing. A resource is capable of performing multiple jobs, and multiple resources can be used to perform the same job on a part [16,121–123]. If one monitors the energy consumption of FMSs, it is not uncommon to see that different resources require different productive powers and processing times to perform the same job and their idle powers are often varied from each other. This variation is due to a multitude of factors, whether predicted or unpredicted, including the resource types, its operating conditions, process parameters, and part types [17]. To reduce energy consumption of FMSs, it is crucial to develop efficient scheduling algorithms that generate energy-optimal schedules in consideration of the production constraints.

The traditional scheduling of FMSs often optimizes time-critical objectives such as makespan [73,124,125], mean tardiness and earliness [126], and mean flowtime [127], *etc.* Owing to the current looming economic situation and rising energy prices, energy consumption has been recently considered in the FMS scheduling problems either in the multi-objective function or in a constraint [25, 79, 80, 114]. However, idle energy consumption was usually omitted or assumed to be trivial by the existing works in current literature. This assumption may not be applicable in many realistic FMSs, where idle energy is observed to be significant as compared to total energy consumption [64]. To bridge this gap, the scheduling problem herein considers both idle and productive energy consumption in the objective function. Instead of a Pareto optimal schedule of energy consumption and makespan, an energy-optimal schedule is of interest and the required makespan is formulated as a production constraint. The considered problem is proven to be a non-convex MINLP.

A wide range of problems arising in practical applications can be formulated as MINLPs. Although all MINLPs are NP-hard in general, non-convex MINLPs are much more difficult to solve than convex ones, in both theory and practice. This is because the continuous relaxation of a non-convex MINLP is itself a global optimization problem, and therefore likely to be NP-hard [128]. For the general case of non-convex MINLPs, existing solution methods in current literature are still limited. To the best of our knowledge, the branch-and-reduce (B&R) algorithm [128] and its variants [129, 130] are the only available methods that provide exact solutions of non-convex MINLPs. Its key concept is to replace each non-convex function with a convex under-estimator, and solve the estimated problem using either linear programming (LP) or convex programming (CP) relaxations.

Besides, there are also proposed metaheuristic methods, which are designed to find good, but not provably optimal, solutions quickly. For example, a metaheuristic based on tabu search algorithm was proposed for certain non-convex MINLP instances arising in integrated systems and process control design [131]. A particle-swarm optimization (PSO) approach was presented in [132], an enhanced genetic algorithm (GA) was studied in [133], and an extended ant-colony optimization (ACO) was considered in [134]. Another two recent works are an integration of neighborhood search, local branching, sequential quadratic programming, and branch-and-bound (B&B) [135], and a large neighborhood local search method by rounding the fractional solution from a relaxation [136].

In this chapter, the WTPN is used to model energy-efficient FMSs described in Chapter 2. Based on the WTPN models, a non-convex scheduling problem is formulated, where the optimization criterion is to minimize both productive and idle energy consumption. The considered production constraints include deadlock avoidance, job precedence, minimal part holding time, and maximal makespan. A new RG-based discrete DP method for generating very near-optimal schedules within adequate computational time is presented. The full RG is reduced, such that only reachable paths are retained for energy-optimal path computation. Each subproblem is now linearized, and the optimal substructures are computed and stored in a routing table. It is proven that the induced error by sampling the non-convex MINLP is bounded, and this upper bound can be reduced by increasing the sampling frequency. As compared to the related works, our scheduling method is especially useful for large-scale FMSs (*i.e.,* either the numbers of resources, jobs, or parts are large) with high degree of flexibility.

The proposed scheduling method is tested on an industrial system at a stamping company in the Republic of Singapore. Energy consumption from each stamping machine is continually monitored using a Rudolf R-DPA96A digital power analyzer (RUDOLF). RUDOLFs are interfaced with computers via LabVIEW environment. Our proposed scheduling method is then used to generate near energy-optimal schedules. Comparisons are made with the schedules generated by other related techniques in the current literature [128, 132, 134]. The interested performance metrics are computational time and deviation from optimality.

5.2 Problem Formulation with Mathematical Programming

The WTPN models of FMSs was established in Section 2.3 and this section continues to formulate the total energy optimization problem with mathematical programming. To properly formulate a scheduling problem, it is convenient to write the problem in terms of mathematical programming, which includes a cost function to be minimized or a utility function to be maximized, and a set of constraints, equalities or inequalities, to be satisfied. In a discrete-event domain, we define a finite horizon $k = 0, 1, \ldots, K$ such that after K firing epochs the WTPN models will traverse from the initial state $\mathbf{x}_0$ in (2.11) to the final state $\mathbf{x}_K$ in (2.12).

The WTPN models of FMSs investigated herein have the following structural properties, whose proofs are omitted for brevity.

1. *Property 1:* $\mathbf{x}_K$ is always reachable from $\mathbf{x}_0$, denoted by $\mathbf{x}_K \in \mathrm{Re}\,(\chi, \mathbf{x}_0)$.
2. *Property 2:* $K = \sum_{q=1}^{|\Pi|} \varphi\left(\pi_q\right)\left(|\omega_q| + 1\right)$.
3. *Property 3:* The WTPN models of FMSs is bounded and its RG is finite.

These properties enable us to formulate the following mathematical program with the integer decision variables $\mathbf{x}_k$ and $\mathbf{u}_k$.

5.2.1 Formulation of Constraints

In PN theory, the information of job precedences and resource assignments (defined by the set of arcs) is included in the incidence matrix $\mathbf{W}$. To satisfy these logical constraints, $\mathbf{x}_k$ and $\mathbf{u}_k$ must comply with the following equations

$$\mathbf{x}_k = \mathbf{x}_{k-1} + \mathbf{W}^T \mathbf{u}_k, k = 1, 2, \ldots, K, \tag{5.1}$$

$$\mathbf{x}_{k-1} \geq \mathbf{O}^T \mathbf{u}_k, k = 1, 2, \ldots, K. \tag{5.2}$$

(5.1) specifies that the evolution of $\mathbf{x}_k$ due to $\mathbf{u}_k$ must satisfy the logical constraints embedded in $\mathbf{W}$. (5.2) provides the condition for firing transitions such that illegal markings of the individual places are forbidden.

As mentioned, the scheduling decisions for the split places are made by controlling the firing timing of their output transitions. Different firing instances are assigned to the transitions such that only one transition is enabled at a firing epoch, *i.e.*,

$$\sum_{j=1}^{|T|} u_{jk} = 1, k = 1, 2, \ldots, K. \tag{5.3}$$

To model the firing timing of transitions, we introduce the intermediate continuous variables $\tau_k \in \mathbb{R}^+$, such that τ_k is the time instance of firing epoch k. The following timing constraints are imposed on τ_k

$$\tau_k < \tau_{k+1}, k = 1, 2, \ldots, K-1, \tag{5.4}$$

$$\tau_K \leq D, \tag{5.5}$$

where D, $D \in \mathbb{R}^+ \backslash \{0\}$, is a due date of production orders required by customers. D is assumed to be sufficiently large to avoid trivial solutions.

There is a number of deadlock avoidance policies (DAPs) have been proposed in current literature. In our application, a PN-based DAP that ensures maximal work in progress is used [70,72,121,137]. Let $C = \{c\}$ and $\mathbf{x}_{c0}$ be the set of circular waits in an FMS and the state vector of the resources in the critical subsystem of c, respectively. To prevent deadlocks, the following constraint must be satisfied

$$\mathbf{x}_{ck} \leq \mathbf{x}_{c0} \ \forall c \in C, k = 1, 2, \ldots, K-1. \tag{5.6}$$

Remark 5.1 Circular waits are ubiquitous in an FMS with shared resources, and deadlock can occur in general if the firing of transitions are not properly taken care.

In fact, (5.5) can eliminate such a deadlock, because a solution that causes a deadlock has $\tau_K \to \infty$, which violates (5.5). However, (5.6) is still included as a deadlock avoidance constraint to speed up the search process, as it inhibits searching deadlock solutions wastefully.

We now wish to formulate the constraints for minimal token sojourn time in places. For each $p \in P_J$, let τ_k and τ_k' denote the firing instances of an input transition t and an output transition t', respectively. Let the firing of t and t' be determined by u_j and $u_{j'}$, respectively. A token entering p must sojourn in p for a minimal duration of h. Since the input transitions and output transitions of a place can be fired more than once if $\varphi(\pi_q) \geq 2$, the following count variables $\delta_{\bullet pk}$ is necessarily formulated as

$$\delta_{\bullet pk} = \sum_{g=1}^{k} \sum_{t_j \in {}^\bullet p} u_{jg}, k = 1, 2, \ldots, K, \tag{5.7}$$

where $\delta_{\bullet pk}$ denotes the total number of times that input transitions of p has already been fired up to k. Similarly $\delta_{p\bullet k}$ can be defined. Obviously, $0 \leq \delta_{\bullet pk}, \delta_{p\bullet k} \leq \varphi(\pi_q)$. As such, the following inequalities hold for each input and output transitions pair $\langle t, t' \rangle$ of p

$$u_{jk} + u_{j'k'} \leq z_{jk} + z_{j'k'} + (\delta_{\bullet pk} - \delta_{p\bullet k'})^2, \tag{5.8}$$

$$\tau_{k'} - \tau_k \geq h + A(z_{jk} + z_{j'k'} - 2), \tag{5.9}$$

$$\forall p \in P_J, t \in {}^\bullet p, t' \in p^\bullet, k = 1, 2, \ldots, K,$$

where z_{jk} and $z_{j'k'} \in \{0, 1\}$ are dummy variables and A is a large number for big M method. (5.8) functions as an indicator such that $z_{jk} = z_{j'k'} = 1$ if $u_{jk} = u_{j'k'} = 1$ and $\delta_{\bullet pk} = \delta_{p\bullet k'}$, *i.e.*, t and t' are fired for the same ordinal number at epoch k and k', respectively. In such a case, (5.9) is activated to ensure that $\tau_{k'} - \tau_k \geq h$. It is worth noting that (5.8) is a nonlinear quadratic constraint due to the term $(\delta_{\bullet pk} - \delta_{p\bullet k'})^2$, which is widely known to be convex.

5.2.2 Objective Function and Convexity Analysis

A column vector indexed by the set of places P, *i.e.,*

$$\mathbf{c}^T = \begin{bmatrix} c_1 & c_2 & \dots & c_{|P|} \end{bmatrix}, \tag{5.10}$$

is called the cost vector, where element c_i denotes the token sojourn cost per time unit of $p_i \in P$. As such, the total energy consumption of an FMS can be formulated as

$$J(\mathbf{x}_k, \tau_k) = \sum_{k=0}^{K-1} (\tau_{k+1} - \tau_k)\, \mathbf{c}^T \mathbf{x}_k + (D - \tau_K)\, \mathbf{c}^T \mathbf{x}_K. \tag{5.11}$$

Unlike dedicated manufacturing systems where equipment are dedicated to one product and often switched off when needed, a FMS is typically used for manufacturing multiple part types and products, which will operate idly if certain part types are finished before the due date [138]. This industrial practice is indicated by the second term of (5.11) accordingly.

It can be seen that $J(\mathbf{x}_k, \tau_k)$ is a nonlinear mixed integer function, as it contains the products of different decision variables, the integer variables $\mathbf{x}_k$ and the continuous variables τ_k. In operations research, a mixed integer (linear or nonlinear) problem is often solved using continuous relaxation of the integer variables. For a nonlinear case, it is required to examine the convexity of the problem.

Theorem 5.1 *The continuous relaxation of $J(\mathbf{x}_k, \tau_k)$ is a non-convex function.*

Proof: From (5.11), $J(\mathbf{x}_k, \tau_k)$ can be written as

$$\begin{aligned}
J(\mathbf{x}_k, \tau_k) &= (\tau_1 - \tau_0)\, \mathbf{c}^T \mathbf{x}_0 + (\tau_2 - \tau_1)\, \mathbf{c}^T \mathbf{x}_1 + \ldots + (D - \tau_K)\, \mathbf{c}^T \mathbf{x}_K \\
&= \tau_1 \left(c_1 x_{10} + \ldots + c_{|P|} x_{|P|0}\right) + (\tau_2 - \tau_1) \left(c_1 x_{11} + \ldots + c_{|P|} x_{|P|1}\right) + \ldots \\
&\quad + (D - \tau_K) \left(c_1 x_{1K} + \ldots + c_{|P|} x_{|P|K}\right). \qquad (5.12)
\end{aligned}$$

First of all, it worth noting that a multivariate function $f\left(x_1, x_2, \ldots, x_n\right)$ is not convex if it is not convex in its sub-domain, *i.e.,* $f\left(x_1, x_2, \ldots, x_q\right)$ is not convex with any $q < n$. We prove the non-convexity of the continuous relaxation of $J(\mathbf{x}_k, \tau_k)$ by induction. It would be trivial to consider the base case of $K = 1$ as a job needs at least two transitions to be fired. As such, let us consider the base case of $K = 2$.

At $K = 2$: J is written as $J(\mathbf{x}_k, \tau_k, 2)$. One has

$$\begin{aligned}
J(\mathbf{x}_k, \tau_k, 2) = \tau_1 \left(c_1 x_{10} + \ldots + c_{|P|} x_{|P|0}\right) + (\tau_2 - \tau_1) \left(c_1 x_{11} + \ldots + c_{|P|} x_{|P|1}\right) \\
+ (D - \tau_2) \left(c_1 x_{12} + \ldots + c_{|P|} x_{|P|2}\right). \qquad (5.13)
\end{aligned}$$

To prove that $J(\mathbf{x}_k, \tau_k, 2)$ is not convex, another mathematical induction is used on $|P|$. It would be trivial to consider the base case of $|P| = 1$ as an FMS needs at least four places. As such, let us consider the base case of $|P| = 4$.

At $|P| = 4$: $J(\mathbf{x}_k, \tau_k, 2)$ is written as $J(\mathbf{x}, \tau_k, 2, 4)$, and one has

$$
\begin{aligned}
J(\mathbf{x}_k, \tau_k, 2, 4) = \tau_1 \left(c_1 x_{10} + \ldots + c_4 x_{40}\right) + (\tau_2 - \tau_1) \left(c_1 x_{11} + \ldots + c_4 x_{41}\right) \\
+ (D - \tau_2) \left(c_1 x_{12} + \ldots + c_4 x_{42}\right). \qquad (5.14)
\end{aligned}
$$

The Hessian matrix of $J(\mathbf{x}, \tau, 2, 4)$ has both negative and positive eigenvalues. As such, $J(\mathbf{x}, \tau, 2, 4)$ is not either convex or concave.

At $|P| = s+1$: Let us assume that $J(\mathbf{x}, \tau, 2)$ is not convex up to $J(\mathbf{x}, \tau, 2, s)$. We now wish to prove that $J(\mathbf{x}, \tau, 2, s+1)$ is not convex too. One has

$$
\begin{aligned}
J(\mathbf{x_k}, \tau_k, 2, s+1) &= J(\mathbf{x}_k, \tau_k, 2, s) + \tau_1 c_{s+1} x_{(s+1)0} \\
&\quad + (\tau_2 - \tau_1)\, c_{s+1} x_{(s+1)1} + (D - \tau_2)\, c_{s+1} x_{(s+1)2} \\
&= J(\mathbf{x}_k, \tau_k, 2, s) + F. \qquad (5.15)
\end{aligned}
$$

$J(\mathbf{x}_k, \tau_k, 2, s+1)$ is convex if F is able to convexify $J(\mathbf{x}_k, \tau_k, 2, s)$. This is not possible as the projection of F on the domain of $J(\mathbf{x}_k, \tau_k, 2, s)$ is $F^* = 0$. As such, one has $J(\mathbf{x}_k, \tau_k, 2)$ is not convex.

At $K = q+1$: Now, let us assume that $J(\mathbf{x}_k, \tau_k)$ is not convex up to $J(\mathbf{x}_k, \tau_k, q)$. We now wish to prove that $J(\mathbf{x}_k, \tau_k, q+1)$ is not convex too. From (5.12), one has

$$
\begin{aligned}
J(\mathbf{x}_k, \tau_k, q+1) &= J(\mathbf{x}_k, \tau_k, q) + (\tau_{q+1} - D)(c_1 x_{1q} + \ldots + c_{|P|} x_{|P|q}) \\
&\quad + (D - \tau_{q+1})(c_1 x_{1q+1} + \ldots + c_{|P|} x_{|P|q+1}) \\
&= J(\mathbf{x}_k, \tau_k, q) + Q. \qquad (5.16)
\end{aligned}
$$

$J(\mathbf{x}_k, \tau_k, q+1)$ is convex if Q is able to convexify $J(\mathbf{x}_k, \tau_k, q)$. This is not possible as the projection of Q on the domain of $J(\mathbf{x}_k, \tau_k, q)$ is $Q^* = -D(c_1 x_{1q} + \ldots + c_{|P|} x_{|P|q})$, which is a linear function. The summation of a non-convex function with a linear function results a non-convex function, which implies that $J(\mathbf{x}_k, \tau_k, q+1)$ is not convex. As such, the continuous relaxation of $J(\mathbf{x}_k, \tau_k)$ is not convex. ∎

The cost function (5.11) and the constraints (5.1)–(5.9) form a non-convex MINLP to be solved.

Remark 5.2 The scheduling problem considered in this chapter only includes some general production constraints (5.1)–(5.9), which are practical in many realistic FMSs. We acknowledge that other constraints may also be necessary, which were included by other excellent works, *e.g.,* maximal part holding time [73], line balancing [139], and resource utilization [140], *etc.* These additional constraints can be easily included in the formulated non-convex MINLP as well.

5.3 Energy-Optimal Path Computation Using Dynamic Programming

DP is a powerful mathematical technique for making a sequence of interrelated decisions. Bellman formalized the term DP and used it to describe the process of solving problems where one needs to find the best decision one after another [141]. It provides a systematic procedure for determining the optimal combination of decisions which takes much less time than naive methods. In contrast to other optimization techniques, such as mathematical programming, DP does not provide a standard mathematical formulation of the algorithm. Rather, DP requires a problem to be ably formulated in a (forward or backward) recursive form, which is popularly known as Bellman equation.

5.3.1 Formulation of Dynamic Programming

To formulate DP, the non-convex MINLP is sampled in discrete time at sampling frequency f_{dp}. Owing to constraint (5.5), we define a finite discrete-time horizon $n = 0, 1, \ldots, H$, where $H = D \times f_{\mathrm{dp}}$ and $\mathbf{x}_H = \mathbf{x}_K$.

In timed extensions of PNs, there are two types of controls, namely, timed and discrete. A discrete control represents the effect of firing a discrete transition, which has been discussed so far. On the other hand, a timed control only increases the elapsed sojourn time of each token by one and does not affect the current PN state. Thus, the state evolution of the WTPN models is formally defined as follows.

Definition 5.1 (State Evolution of the WTPN Models)

$$\mathbf{x}_n = \begin{cases} \mathbf{x}_{n-1} + \mathbf{W}^T \mathbf{u}_n, \\ \mathbf{x}_{n-1}, \end{cases} \tag{5.17}$$

where the upper holds if $\mathbf{u}_n$ is discrete and the lower holds if $\mathbf{u}_n$ is timed.

Let U be the set of admissible controls which satisfy the set of constraints (5.1)–(5.3) and (5.6)–(5.9). It can be seen that a timed control is always admissible. This leads to the definition of lazy (non-urgent) behaviour in the general class of timed PNs, which was introduced in [142, 143].

Definition 5.2 (Lazy- and Forced-Timed Controls) A fired timed control $\mathbf{u}_n$ is called forced, if there is no discrete control admissible at n. On the contrary, a lazy control indicates that one may choose to "let time pass" instead of firing an admissible discrete control.

Let us denote the timed control by $\mathbf{u}_0$, which is a null $|T| \times 1$ vector. We now wish to justify the recursive form of $J(\mathbf{x}_k, \tau_k)$ in (5.11). Let us denote by $J^*(\mathbf{x}_n)$ the optimal cost-to-go from $\mathbf{x}_n$ to $\mathbf{x}_H$ and by $J(\mathbf{x}_n, \mathbf{u}_{n+1})$ the cost-to-go from $\mathbf{x}_n$ to the next stage. The following backward recursive formula can be derived.

Theorem 5.2 *The Bellman recursive formula holds by*

$$J^*(\mathbf{x}_n) = \min_{\mathbf{u}_{n+1} \in U} \{J(\mathbf{x}_n, \mathbf{u}_{n+1}) + J^*(\mathbf{x}_{n+1})\}. \tag{5.18}$$

Proof: From Definition 5.1, $J(\mathbf{x}_k, \tau_k)$ is approximated by

$$J = \frac{1}{f_{\mathrm{dp}}} \sum_{n=1}^{H} \mathbf{c}^T \mathbf{x}_n = \frac{1}{f_{\mathrm{dp}}} \sum_{n=1}^{H} \mathbf{c}^T \left(\mathbf{x}_{n-1} + \mathbf{W}^T \mathbf{u}_n\right), \tag{5.19}$$

where $\mathbf{u}_n$ can be either discrete or timed. We denote by $\mathbf{u}_{[n_1,n_2]}$ the control vector sequence within the range $n_1 \leq n \leq n_2$, such that $\mathbf{u}_{[n_1,n_2]} = \mathbf{u}_{n_1}\mathbf{u}_{n_1+1} \ldots \mathbf{u}_{n_2}$. The cost-to-go from $\mathbf{x}_0$ to $\mathbf{x}_H$ is computed by expanding the summation in (5.19) as

$$\begin{aligned} J \times f_{\mathrm{dp}} &= \mathbf{c}^T \mathbf{x}_1 + \mathbf{c}^T \mathbf{x}_2 + \ldots \\ &= \mathbf{c}^T \mathbf{x}_0 + \mathbf{c}^T \mathbf{W}^T \mathbf{u}_1 + \mathbf{c}^T \mathbf{x}_1 + \mathbf{c}^T \mathbf{W}^T \mathbf{u}_2 + \ldots \\ &= 2\mathbf{c}^T \mathbf{x}_0 + 2\mathbf{c}^T \mathbf{W}^T \mathbf{u}_1 + \mathbf{c}^T \mathbf{W}^T \mathbf{u}_2 + \ldots \end{aligned} \tag{5.20}$$

As such, one can consider J as a function of the initial state $\mathbf{x}_0$, and the admissible control sequence $\mathbf{u}_{[1,H]} \in U$. We write this as $J\left(\mathbf{x}_0, \mathbf{u}_{[1,H]}\right)$. It can be seen that the optimal value of J, denoted by J^*, is also the function of $\mathbf{x}_0$, because the optimal control vector $\mathbf{u}^*{}_{[1,H]} \in U$ is determined based on the value of $\mathbf{x}_0$. We write this as $J^*(\mathbf{x}_0)$. As such, one has

$$J^*(\mathbf{x}_0) = \min_{\mathbf{u}_{[1,H]} \in U} J(\mathbf{x}_0, \mathbf{u}_{[1,H]}). \tag{5.21}$$

Similarly, cost-to-go from any state $\mathbf{x}_\gamma$, where $1 \leq \gamma \leq H$, is computed by

$$J^*(\mathbf{x}_\gamma) = \min_{\mathbf{u}_{[\gamma+1,H]} \in U} J(\mathbf{x}_\gamma, \mathbf{u}_{[\gamma+1,H]}), \tag{5.22}$$

One now considers an initial state $\mathbf{x}_\mu$, where $\gamma \leq \mu \leq H$. The optimal cost-to-go is denoted by $J^*(\mathbf{x}_\mu)$. This implies that $J\left(\mathbf{x}_\gamma, \mathbf{u}_{[\gamma+1,\mu]}\right)$ is independent of $\mathbf{u}_{[\mu+1,H]}$. As such, (5.22) is equivalent to

$$\begin{aligned} J^*\left(\mathbf{x}_\gamma\right) &= \min_{\mathbf{u}_{[\gamma+1,\mu]} \in U} \left\{ J\left(\mathbf{x}_\gamma, \mathbf{u}_{[\gamma+1,\mu]}\right) + \min_{\mathbf{u}_{[\mu+1,H]} \in U} J\left(\mathbf{x}_\mu, \mathbf{u}_{[\mu+1,H]}\right) \right\} \\ &= \min_{\mathbf{u}_{[\gamma+1,\mu]} \in U} \left\{ J\left(\mathbf{x}_\gamma, \mathbf{u}_{[\gamma+1,\mu]}\right) + J^*\left(\mathbf{x}_\mu\right) \right\}. \end{aligned} \tag{5.23}$$

Now let $\mu = \gamma + 1$ and replace γ with index n, one has (5.23) rewritten in the form of Bellman's famous equation as

$$J^*\left(\mathbf{x}_n\right) = \min_{\mathbf{u}_{n+1} \in U} \left\{ J\left(\mathbf{x}_n, \mathbf{u}_{n+1}\right) + J^*\left(\mathbf{x}_{n+1}\right) \right\}. \tag{5.24}$$

∎

The recursive formula (5.18) is the basis of the optimization principle which is based on the backward direction. That is to say, in the optimal control sequence $\mathbf{u}^*_{[1,H]}$, the first half of the control sequence $\mathbf{u}^*{}_{[1,g]}$ may be of any value, and the last half of the control sequence in relation to $\mathbf{x}_g$ which has been produced by the first half of the control sequence, produces the optimal control sequence $\mathbf{u}^*{}_{[g+1,H]}$.

5.3.2 Computation of Energy-Optimal Path

Algorithm 1 Compute energy-optimal schedule

Require: H **and** $(\chi, \mathbf{x}_0)$ **and** $\mathbf{x}_H$
Ensure: $J^*(\mathbf{x}_0)$ **and** routing table

$J^*(\mathbf{x}_H) = 0$ **and** $J^*(\mathbf{x}_n^i) = \infty, \forall n \neq H$
Create nodes $\mathbf{x}_1$ **and** $\mathbf{x}_H$
for $\mathbf{u}_1^j \in U \setminus \{\mathbf{u}_0\}$ **do**
    Create nodes $\mathbf{x}_1^j = \mathbf{x}_0 + \mathbf{W}^T \mathbf{u}_1^j$
    if $\exists\ j \neq j' : \mathbf{x}_1^j = \mathbf{x}_1^{j'}$ **then**
        Merge $\mathbf{x}_1^j$ with $\mathbf{x}_1^{j'}$
    end if
end for
for $n = 2 \to H - 1$ **do**
    for all $\mathbf{x}_n^i$ **do**
        for all $\mathbf{u}_n^j \in U$ **do**
            Create node $\mathbf{x}_n^{ij} = \mathbf{x}_{n-1}^i + \mathbf{W}^T \mathbf{u}_n^j$
        end for
        if $\exists\ i \neq i'$ **or** $j \neq j' : \mathbf{x}_n^{ij} = \mathbf{x}_n^{i'j'}$ **then**
            Merge $\mathbf{x}_n^{ij}$ with $\mathbf{x}_n^{i'j'}$
        end if
    end for
end for
for all $\mathbf{x}_{H-1}^i, !\exists \mathbf{u}_H^j \in U : \mathbf{x}_H = \mathbf{x}_{H-1}^i + \mathbf{W}^T \mathbf{u}_H^j$ **do**
    Delete $\left(\mathbf{x}_{H-1}^i\right)$
end for
for $n = H \to 1$ **do**
    $J^*\left(\mathbf{x}_{n-1}^i\right) = \min_{\mathbf{u}_n \in U} \left\{ J\left(\mathbf{x}_{n-1}^i, \mathbf{u}_n^j\right) + J^*\left(\mathbf{x}_n^j\right) \right\}$
    $\mathbf{u}^*\left(\mathbf{x}_{n-1}^i\right) = \arg\left\{ J^*\left(\mathbf{x}_{n-1}^i\right) \right\}$ ▷ Update routing table
end for

In PN theory, the RG is a directed graph of all possible states reachable from $\mathbf{x}_0$ in a net $(\chi, \mathbf{x}_0)$, denoted by $\mathrm{Re}\,(\chi, \mathbf{x}_0)$. A node represents a reachable state and an edge represents a control which drives the PN from its tail to head states.

To search for the energy-optimal path from $\mathbf{x}_0$ to $\mathbf{x}_H$, Theorem 5.2 suggests to perform the DP algorithm on the H-stage RG, *i.e.,* $\mathbf{x}_H$ must be reached from $\mathbf{x}_0$ within H edges. DP has a node-table routing architecture in which the routing table is stored at each hop of the route. The destination of the header flit will be checked, and it will decide the routing direction among several possible next hops at each stage of the route based on the table entries. The RG-based DP algorithm on WTPN models is presented in Algorithm 1.

Algorithm 1 performs two main tasks recursively: construction of a H-stage RG (lines 3–21) and evaluation of the energy-optimal cost-to-go from each constructed state to the final state (lines 22–25). At stage n, all legal states $\mathbf{x}_n^i$ were constructed from the previous iteration. For each $\mathbf{x}_n^i$, Algorithm 1 finds all admissible controls $\mathbf{u}_{n+1}^j \in U$ which result the legal states $\mathbf{x}_{n+1}^{ij}$ at stage $n+1$. To find the set of admissible controls, the following procedures are done. Given a state $\mathbf{x}_n^i$ and a control vector $\mathbf{u}_{n+1}^j$, it is direct to validate whether $\mathbf{u}_{n+1}^j$ satisfies constraints (5.1)–(5.3) and (5.6). Let $\mathbf{u}_{n+1}^j$ represent the firing of transition $t' \in p^{\bullet}$. To ensure constraints (5.8)–(5.9) satisfied, it is required to trace back to the root $(\mathbf{x}_0)$ of the path containing $\mathbf{x}_n^i$ to determine the count variables $\delta_{p^{\bullet}n}$ and $\delta_{^{\bullet}pn}$.

As compared to a full RG, the RG constructed by Algorithm 1 is reduced by the

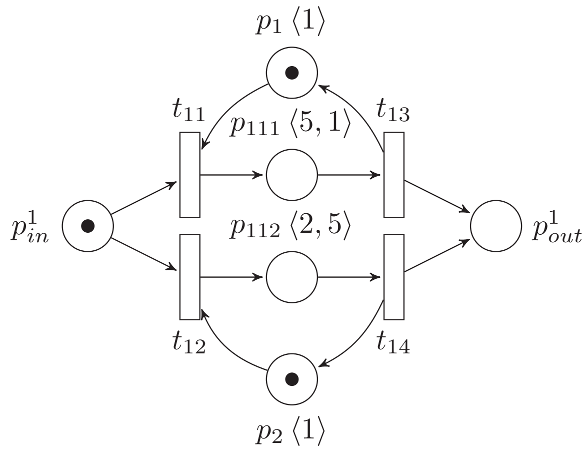


Figure 5.1: A simple marked WTPN models example.

following reasons.

1. It can be seen that, at stage $H-1$ only the paths which are able to reach $\mathbf{x}_H$ are of interest. This implies that there is a number of paths, which are not able to reach $\mathbf{x}_H$. To avoid wasteful evaluations, these paths are pruned at stage $H-1$ using function Delete, as outlined in lines 19–21.

2. The effect of lazy behaviour of a timed PN is considered. This lazy behaviour has a negative impact, as it creates redundant paths in the RG. As discussed, a lazy control $\mathbf{u}_0$ is always admissible at any state. It will be proven in Theorem 5.3 that the paths initiated by some lazy controls are redundant. Thus, they are pruned, as outlined in lines 3–8.

Theorem 5.3 *The reduced RG of WTPNs constructed by Algorithm 1 consists of only but all reachable paths from* $\mathbf{x}_0$ *to* $\mathbf{x}_H$ *within* H *edges.*

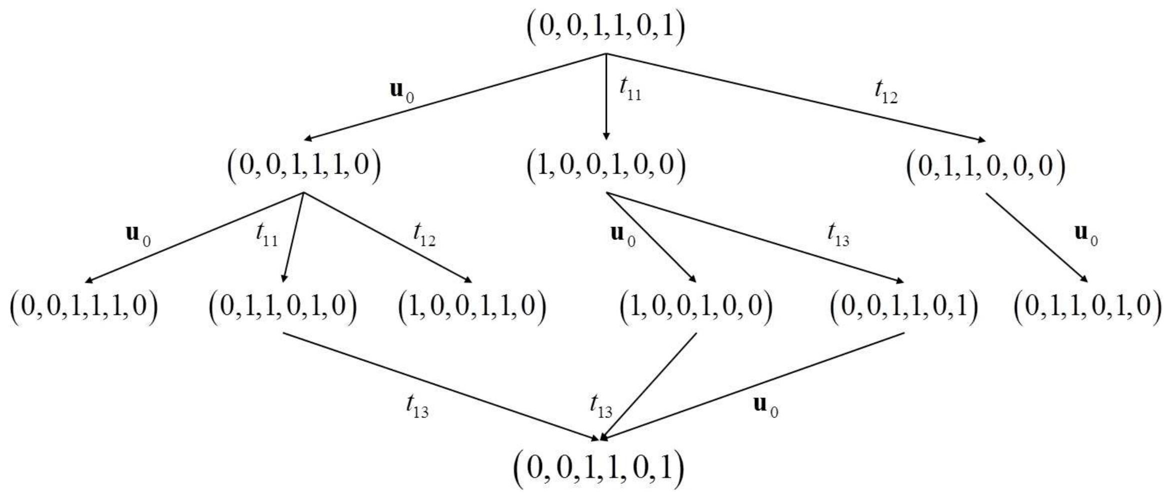


Figure 5.2: The full 3-stage RG of WTPN models example.

Proof: The proof of this theorem is obvious from the generation algorithm for the RG, if one can prove that all paths which are initiated by some lazy timed controls, called lazy paths, are redundant. In other words, there are always some other paths that are equivalent to the lazy ones.

A H-stage reachable lazy path is characterized by a state sequence of the form $\mathbf{x}_0\mathbf{x}_1\cdots\mathbf{x}_\delta\mathbf{x}_{\delta+1}\ldots\mathbf{x}_{H-1}\mathbf{x}_H$, where $0 \le \delta \le H$ and $\mathbf{x}_0 = \mathbf{x}_1 = \ldots = \mathbf{x}_\delta$. It can be seen that the non-lazy counterpart that has the form $\mathbf{x}_0\mathbf{x}_1\cdots\mathbf{x}_{H-\delta}\mathbf{x}_{H-\delta+1}\ldots\mathbf{x}_{H-1}\mathbf{x}_H$ is also reachable, where $\mathbf{x}_{H-\delta} = \ldots = \mathbf{x}_{H-1} = \mathbf{x}_H$. It is direct to see that these two paths are equivalent, if one notes the little-realized fact that

$$J(\mathbf{x}_0, \mathbf{u}_0) = J(\mathbf{x}_H, \mathbf{u}_0). \tag{5.25}$$

∎

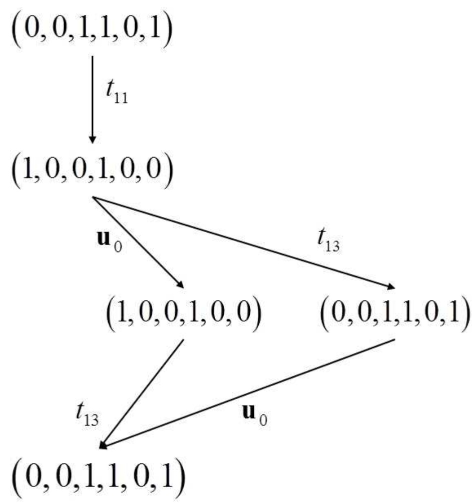


Figure 5.3: The reduced 3-stage RG of WTPN models example.

Example 5.1 For clarification, let us consider the simple marked WTPN models of FMSs shown in Figure 5.1. There is one choice job represented by places p_{111} and p_{112}. There are two resources represented by places p_1 and p_2, respectively. Thus, the initial state is $\mathbf{x}_0 = \begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}^T$, and the final state is $\mathbf{x}_H = \begin{bmatrix} 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}^T$, where $D = 3$ is given and $f_{\text{dp}} = 1$ is chosen. The full 3-stage RG is constructed in Figure 5.2. It can be seen that there are only three paths are able to reach $\mathbf{x}_H$ within three edges. Among these reachable paths, the lazy path $\mathbf{u}_0 \rightarrow t_{11} \rightarrow t_{13}$ is equivalent to its non-lazy counterpart $t_{11} \rightarrow t_{13} \rightarrow \mathbf{u}_0$. The reduced 3-stage RG generated by Algorithm 1 is shown in Figure 5.3, where only two paths are retained for computation of the energy-optimal path.

To find the energy-optimal path, it can be seen that each node involves $O(|\mathcal{A}|)$ additions and comparisons, where $|\mathcal{A}|$ is the largest number of nodes at one stage. Note that the number of additions corresponds to the number of adjacent nodes, and $|\mathcal{A}|$ is an upper bound, which corresponds to the configuration of a fully connected RG. As such, the W-C complexity is $O(H|\mathcal{A}|)$, as compared to the exponential complexity of exhaustive searches or B&B algorithm. However, in modern computers with multi-core processors and parallel execution, $|\mathcal{A}|$ additions can be executed in parallel. Each computational unit can simultaneously compute the new expected cost for all neighboring nodes. Therefore, the solution time becomes the time for the updated value to be distributed to every other node, and the computational complexity can be reduced to $O(H)$.

5.3.3 Error Analysis

In this subsection, we study the error induced by sampling the non-convex MINLP problem in discrete time at sampling frequency f_{dp}. Denote the optimal solution of the non-convex MINLP model by pairs $\langle \mathbf{x}_k^*, \tau_k^* \rangle$, where $\mathbf{x}_0$ and $\mathbf{x}_K$ are known *a priori*. Let the optimal cost be $J_{\text{opt}} = J(\mathbf{x}_k^*, \tau_k^*)$ and the cost obtained by DP in discrete time be J_{dp}.

If $\exists n : \tau_k^* = \frac{n}{f_{\text{dp}}}, \forall k$, *i.e.*, all time instances τ_k^* are sampled, then $\langle \mathbf{x}_k^*, \tau_k^* \rangle$ is feasible in discrete time. According to Bellman's principle of optimality, $J_{\text{dp}} - J_{\text{opt}} = 0$. Otherwise, some τ_k^* are not sampled, then $\exists n : \frac{n-1}{f_{\text{dp}}} < \tau_k^* < \frac{n}{f_{\text{dp}}}$. Let $\Delta_k = \frac{n}{f_{\text{dp}}} - \tau_k^*$. Obliviously, $0 \leq \Delta_k \leq \frac{1}{f_{\text{dp}}}$.

It is worth noting the minimal token sojourn time constraint imposes on all pairs of consecutive transitions $\langle t, t' \rangle$ from a certain part route, such that $\tau_{k'}^* - \tau_k^* \geq h$ if t and t' are fired at τ_k^* and $\tau_{k'}^*$, respectively. Let us denote by τ_k^d the discrete time instances that $\mathbf{x}_k^*$ are activated. In other words, the continuous optimal solution $\langle \mathbf{x}_k^*, \tau_k^* \rangle$ is mapped into a discrete solution $\left\langle \mathbf{x}_k^*, \tau_k^d \right\rangle$, where $\tau_{k'}^d - \tau_k^d \geq h$ still must hold for all pairs $\langle t, t' \rangle$.

Theorem 5.4 *The error induced by sampling the non-convex MINLP at sampling frequency f_{dp} is bounded by*

$$J_{\text{dp}} - J_{\text{opt}} < \sum_{k=1}^{K-1} \frac{1}{f_{\text{dp}}} \mathbf{c}^T \mathbf{x}_k^*. \tag{5.26}$$

The following Lemmas are needed to prove Theorem 5.4.

Lemma 5.1 $\tau_k^d \geq \tau_k^*, k = 2, 3, \ldots, K.$

Proof: Proof by induction.

At $k = 2$: One has $\tau_2^* - \tau_1^* \geq h_2$, and it is always true that $\tau_1^d = \tau_1^* = 0$. To ensure that $\tau_2^d - \tau_1^d \geq h_2$ holds, one has $\tau_2^d \geq \tau_2^*$.

At $k = s$: Assume that $\tau_k^d \geq \tau_k^*$ holds for $k = 2, \ldots, s-1$. We now wish to prove that $\tau_s^d \geq \tau_s^*$. The minimal token sojourn time constraint imposed on τ_s^* is

$$\tau_s^* - \tau_{s'}^* \geq h_s, \text{ for } s' < s. \tag{5.27}$$

According to the hypothesis,

$$\tau_{s'}^d \geq \tau_{s'}^*. \tag{5.28}$$

From (5.27) and (5.28), $\tau_s^d \geq \tau_s^*$ must hold to ensure $\tau_s^d - \tau_{s'}^d \geq h_s$. ∎

From Lemma 5.1, the position of τ_k^d can be investigated further.

Lemma 5.2 *For each pair $\langle t, t' \rangle$ such that $\tau_{k'}^* - \tau_k^* \geq h$. Assume that $\tau_k^d = \tau_k^* + \Delta_k$, one has*

$$\tau_{k'}^d = \begin{cases} \tau_{k'}^* + \Delta_{k'} \ \textit{if}\ \Delta_{k'} \geq \Delta_k \\ \tau_{k'}^* + \Delta_{k'} + \frac{1}{f_{\text{dp}}} \ \textit{if}\ \Delta_{k'} < \Delta_k \end{cases}. \tag{5.29}$$

Proof: Since $\tau_{k'}^d - \tau_k^d \geq h$ must always hold, one has

1. $\Delta_{k'} \geq \Delta_k$: $\tau_{k'}^d = \tau_{k'}^* + \Delta_{k'}$ ensures $\tau_{k'}^d - \tau_k^d \geq h$, as

$$\tau_{k'}^d - \tau_k^d = (\tau_{k'}^* - \tau_k^*) + (\Delta_{k'} - \Delta_k) \geq h + 0 \geq h, \tag{5.30}$$

   i.e., $\tau_{k'}^*$ is sampled at the next sampling point to the right.

2. $\Delta_{k'} < \Delta_k$: Similarly, it can be proven that not $\tau_{k'}^d = \tau_{k'}^* + \Delta_{k'}$ but $\tau_{k'}^d = \tau_{k'}^* + \Delta_{k'} + \frac{1}{f_{\text{dp}}}$ ensures $\tau_{k'}^d - \tau_k^d \geq h$, *i.e.*, $\tau_{k'}^*$ is sampled at the next two sampling points to the right.

∎

The proof of Theorem 5.4 is derived as follows.

Proof: Without loss of generality, the worst error case is considered when $\Delta_{k'} < \Delta_k$, $\forall\ \langle t, t' \rangle$, and the continuous optimal solution only comprises of one part route (unlikely).

In this case, the error is propagated serially throughout the optimal state sequence. From Lemma 5.1 and 5.2, one has $\Delta_{k+1} < \Delta_k$, $\forall k$ and

$$\tau_2^d = \tau_2^* + \Delta_2 \tag{5.31}$$

$$\tau_3^d = \tau_3^* + \Delta_3 + \frac{1}{f_{\text{dp}}} \tag{5.32}$$

$$\tau_4^d = \tau_4^* + \Delta_4 + \frac{2}{f_{\text{dp}}} \tag{5.33}$$

$$\vdots \tag{5.34}$$

From (5.11), the worst error induced by sampling at f_{dp} is

$$\begin{aligned} J(\mathbf{x}_k^*, \tau_k^d) - J_{\text{opt}} &= \sum_{k=1}^{|K|-1} \mathbf{c}^T \mathbf{x}_k^* \left(\Delta_{k+1} - \Delta_k + \frac{1}{f_{\text{dp}}}\right) + \Delta_1 \mathbf{c}^T \mathbf{x}_0 - \left(\Delta_K + \frac{K-2}{f_{\text{dp}}}\right) \mathbf{c}^T \mathbf{x}_K \\ &= \sum_{k=1}^{|K|-1} \mathbf{c}^T \mathbf{x}_k^* \left(\Delta_{k+1} - \Delta_k + \frac{1}{f_{\text{dp}}}\right) - \left(\Delta_K - \Delta_1 + \frac{K-2}{f_{\text{dp}}}\right) \mathbf{c}^T \mathbf{x}_K \\ &< \sum_{k=1}^{|K|-1} \frac{1}{f_{\text{dp}}} \mathbf{c}^T \mathbf{x}_k^*. \end{aligned} \tag{5.35}$$

Since Bellman's principle of optimality ensures that $J_{\text{dp}} \leq J(\mathbf{x}_k^*, \tau_k^d)$, one has

$$J_{\text{dp}} - J_{\text{opt}} < \sum_{k=1}^{K-1} \frac{1}{f_{\text{dp}}} \mathbf{c}^T \mathbf{x}_k^*. \tag{5.36}$$

∎

It can be seen that a certain part route indeed consists of several pairs $\langle t, t' \rangle$. From Lemma 5.2, if a pair $\langle t, t' \rangle$ has $\Delta_{k'} < \Delta_k$, then τ_k^* must sampled at the next two points to the right, which shifts the subsequent transitions' sampling points on all part routes which contain $\langle t, t' \rangle$ by one sampling point to the right as well. This phenomenon is called *error propagation*. Theorem 5.4 suggests that the induced error can be reduced by increasing the sampling frequency f_{dp}. Obviously, $J_{\text{dp}} - J_{\text{opt}} \to 0$

if $f_{\mathrm{dp}} \to \infty$. It is also worth noting that there is always a tradeoff between optimality and computational complexity.

5.4 Industrial Application

The same industrial application presented in Chapter 4 is used to verify the practicality of considered robust problem and the feasibility of our proposed integrated framework. Using the measured power data, the productive power matrix $\mathbf{A}(t)$ and the processing time matrix $\mathbf{D}$ can be constructed as described in (4.20) and (4.22).

5.4.1 Weighted P-Timed Petri Net Models of Industrial Stamping System

This stamping system is now formulated using the WTPN models. There are two part types π_1 and π_2, eight stamping machines denoted by $\mathrm{M1} - \mathrm{M8}$, and five material routing resources denoted by $\mathrm{B1} - \mathrm{B5}$. Part π_1 has a job sequence $\omega_1 = v^1_{in} v^1_1 v^1_2 \ldots v^1_7 v^1_{out}$, where v^1_1, v^1_3, v^1_5, and v^1_7 are stamping jobs, and v^1_2, v^1_4, and v^1_6 are routing jobs. This implies that v^1_3, v^1_5, and v^1_7 are choice jobs and v^1_1 is a non-choice job. Likewise, part π_2 has a job sequence $\omega_2 = v^2_{in} v^2_1 v^2_2 \ldots v^2_5 v^2_{out}$, where v^2_5 is a non-choice stamping job, v^2_2 and v^2_4 are choice stamping jobs, and hence v^2_1 and v^2_3 are routing jobs.

For each choice job, there is an associated routing resource which routes parts.

For instance, choice job v_3^1 can be processed by M2 or M3, and is associated with routing resource B1; choice job v_5^1 can be processed by M3, M4, or M5, and uses routing resource B2 for part routing; and choice job v_4^2 can be processed by M7 or M8, and is routed by routing resource B5, *etc.* All routing resources are non-shared. All stamping machines are shared resources, except for M6 − M8. Capacities of all resources are one.

The layout of this stamping system is shown in Figure 5.4, where the solid and dashed lines represent the part routes of part π_1 and π_2, respectively. Since energy data of material routing resources are not available in our application, they are assumed to be identical. This assumption does not affect the scheduling decisions, as all routing resources are non-shared without routing choices. The state vectors $\mathbf{x}_n$ and control vectors $\mathbf{u}_n$ are arranged as described in (2.7) and (2.8). From (4.20)–(4.22), the productive power matrices $\mathbf{A}^q$ (kW), the processing time matrices $\mathbf{D}^q$(s), and the idle power vector $\mathbf{b}$ (kW) of the stamping system can be constructed as shown in (4.23)–(4.26) and (5.37), respectively.

$$\begin{array}{c} \begin{array}{ccccccccccccc} \text{M1} & \text{M2} & \text{M3} & \text{M4} & \text{M5} & \text{M6} & \text{M7} & \text{M8} & \text{B1} & \text{B2} & \text{B3} & \text{B4} & \text{B5} \end{array} \\ \mathbf{b}^T = \begin{bmatrix} 3.14 & 1.88 & 2.25 & 2.05 & 1.93 & 2.14 & 2.56 & 3.23 & 1.2 & 1.2 & 1.2 & 1.2 & 1.2 \end{bmatrix}. \end{array} \tag{5.37}$$

All FMS information is embedded in the WTPN models shown in Figure 5.5. In reality, the required makespan D is given by customers, and it should be sufficiently large corresponding to the number of parts $\varphi(\pi_q)$. For simplicity but without loss of generality, we let $\varphi(\pi_q) = \varphi(\pi), \forall \pi_q \in \Pi$, and a suitable required makespan D is chosen for each test case.

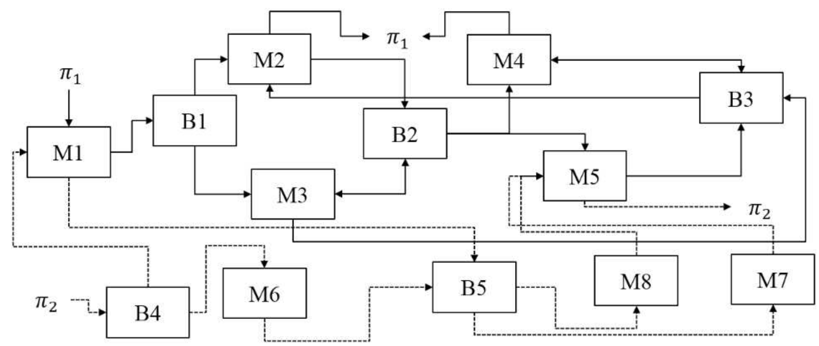


Figure 5.4: Layout of the stamping system.

5.4.2 Experiment Results

This computational test is carried on a digital computer equipped with Intel Core i7 processor and 32 gigabyte random-access memory (RAM). This configuration is widely available in modern computers nowadays. Our proposed scheduling method is programmed in MATLAB. To verify its effectiveness, existing scheduling methods are implemented for comparisons. Firstly, the B&R algorithm, which solves non-convex optimization problems to global optimality, is implemented using BARON software [128]. Secondly, two metaheuristic algorithms, an extended ACO [134] and PSO [132], are also included. Two performance metrics are considered, which are computational time and deviation from optimality. In our experiment, the deviation

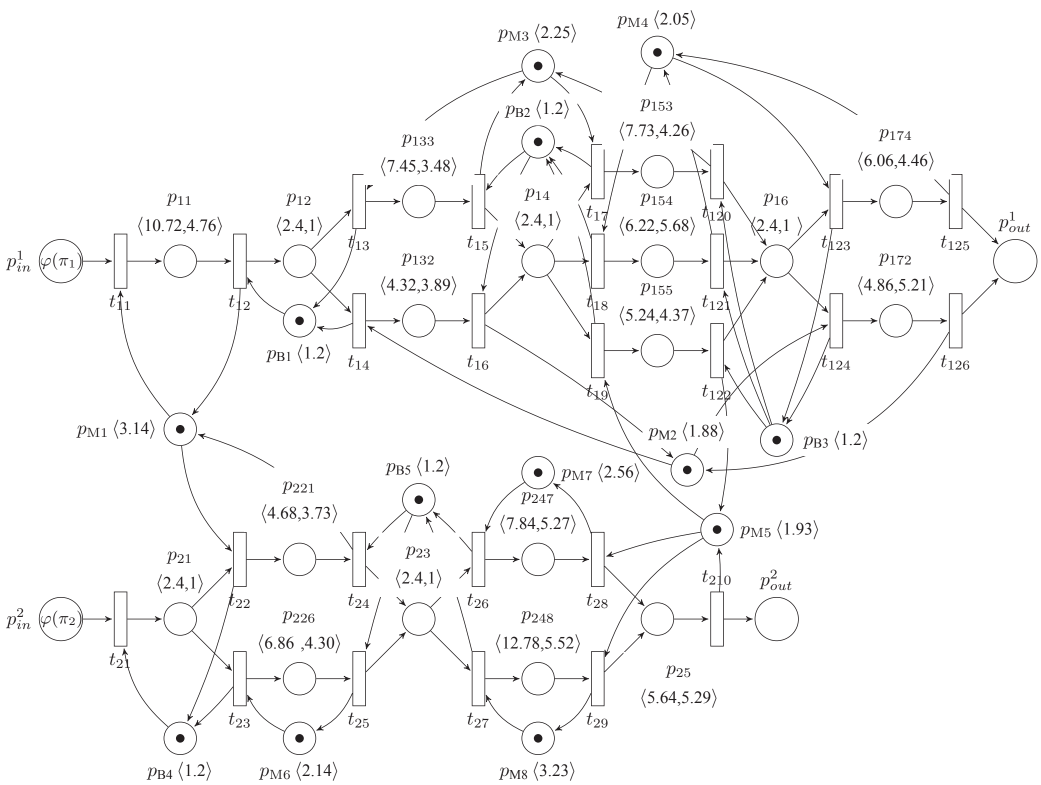


Figure 5.5: WTPN models of the stamping system.

Table 5.1: Performance Comparisons of B&R, PSO, ACO, and DP

Parts	Mean deviation (%)			Mean time (s)			
$\varphi\,(\pi)$	DP	ACO	PSO	B&R	DP	ACO	PSO
5	<1	3.41	3.41	7.91	2	<1	<1
10	<1	12.36	16.36	26.89	7.73	<1	<1
15	<1	18.14	21.41	56.78	15.96	<1	1.16
20	1.21	25.47	24.32	126.78	36.37	2.57	5.73
30	1.81	28.90	29.41	548.38	69.73	8.12	17.15
40	2.38	29.87	30.37	>1000	112.69	20.45	25.54
50	2.47	32.17	34.21	>1000	298.79	28.62	31.48
80	2.77	37.03	40.37	>1000	684.79	49.89	56.48
100	2.83	42.03	49.78	>1000	>1000	60.73	78.48

from optimality of a scheduling method is computed as follows

$$\text{dev} = \frac{J - J_{\text{B\&R}}}{J_{\text{B\&R}}} \times 100\%, \tag{5.38}$$

where $J_{\text{B\&R}}$ is the optimal cost obtained by the B&R algorithm. The mean deviation and mean computational time of ACO and PSO are obtained after forty test runs. To select the sampling frequency, f_{dp} is tuned over a wide range of values. The range $\frac{10}{h_{\min}} \leq f_{\text{dp}} \leq \frac{20}{h_{\min}}$, where $h_{\min}$ is the minimal processing time of all jobs, is observed to yield satisfactory performance within adequate computational time. In this computational test, $f_{\text{dp}} = 10$ Hz is used. We set the time limit for the test as 1000 s in the DP run.

Numerical results are presented in Table 5.1, where it can be seen that our proposed scheduling method achieves infinitesimal deviation from optimality as compared to two metaheuristic algorithms. The deviations of two metaheuristic algorithms vary in a great range, and they even grow up to more than 40% in some test cases. On the contrary, our proposed scheduling method keeps the deviation less than 3% for all test cases. In terms of computational time, the B&R algorithm exceeds 1000 s from the test case of forty parts, while our proposed scheduling method is still computationally efficient up to the test case of one hundred parts.

5.4.3 Discussions with Related Works

As discussed in Section 5.1, our scheduling problem is more practical than the related energy-efficient scheduling works [25, 79, 80, 114], in which idle energy consumption is usually omitted or assumed to be trivial. However, this assumption can be a gap from research to industry, as the idle power consumption can be significant in realistic FMSs. An example is the industrial stamping system in Section 5.4, where the idle power is equivalent up to 43% of the productive power for stamping machines. In fact, it can be seen that the consideration of idle energy consumption is the main reason which makes the scheduling problem non-convex.

To solve the general case of non-convex MINLP, our proposed method efficiently reduces the computational time with low deviation from optimality as compared to the B&R techniques [128–130]. Although the reduced RG of WTPN models could be larger than the size of the B&R enumeration tree at high sampling frequencies, the evaluation of the scheduling subproblems by our proposed method is much more efficient. The B&R algorithm can be performed using either LP or CP relaxations of integer variables. The evaluation of a scheduling subproblem involves three time-consuming procedures: before a subproblem is solved, its constraints are checked if the domain of any variables can be reduced without losing any feasible solutions; after a subproblem is solved, sensitivity information is used to check if the domain of any variables can be reduced without losing any optimal solutions; and the convex underestimators are generated after domain reduction. On the contrary, it can be seen that the scheduling subproblems considered by our proposed method are linearized. Each subproblem is solved only once and the optimal substructure is stored in the routing table, thus reducing the number of computations.

From a theoretical viewpoint, our proposed scheduling method and the B&R algorithm are both applicable to all FMS scales. However, our method is especially more useful for large-scale FMSs, *i.e.,* $|V|, |R|$, and $\varphi(\pi)$ are large, with high degree of overlapping subproblems. This happens when the FMSs have a high degree of flexibility and the number of choice jobs dominates the number of system jobs, *i.e.,* $|V_z| \gg |V_{nz}|$.

Metaheuristic algorithms such as ACO [134], PSO [132], and GA [133] are designed to find good solutions quickly, which are more suitable for online applications where computational time is very critical. The performances of the metaheuristic algorithms highly depend on the convex structure of the search space, and an optimal solution can never be found. On the contrary, Bellman's principle of optimality guarantees the solution found by our proposed scheduling method is optimal in discrete-time space, and the deviation from optimality is indeed the bounded sampling-induced error. This explains why our proposed method achieves low deviation from optimality as compared to the metaheuristic algorithms.

Beyond the energy-optimal scheduling application considered in this chapter, it is worth noting that our proposed scheduling method can be applied to any industrial problems that involve the costs per time unit for generality. Some realistic examples include the labor costs in labor management and scheduling [144], the penalty costs in scheduling aircraft landings [145], the crash costs in project management [146], the rental fees in resource rental optimization [147], *etc.*

The key concern for such table-based routing mechanics like our proposed scheduling method is the routing-table size, which requires allocation of RAM. However, with recent advances in computer and data storage technologies, this is no longer a main hindrance. Our scheduling method and B&B related algorithms both suffer the "curse of dimensionality", when the dimension of the state vector $\mathbf{x}$ is large [141]. Solving the "curse of dimensionality" without scarification of optimality is still an open issue.

5.5 Summary

In this chapter, the energy-optimal scheduling problem for a class of FMSs described in Chapter 2 was studied. Based on the WTPN models of FMSs, a RG-based DP algorithm was proposed for generation of near energy-optimal schedules within adequate computational time. Our proposed scheduling algorithm was rigorously justified with mathematical formulations and derivations. The practicality of the considered problem and the feasibility of our proposed scheduling algorithm were verified based on an industrial stamping system, in which the stamping parts are various types of VCM yokes used in commercial HDD actuators. Our results showed that our proposed scheduling algorithm achieved less than 3% deviation from optimality for all test cases with significant reduction in computational time as compared to related works in current literature.

In the next chapter, we extend the problem presented in this chapter with its robust counterpart, where the productive and idle powers of resources are now considered as random variables.

Chapter 6

Robust Total Energy Optimization of Flexible Manufacturing Systems Based on Renyi Mean-Entropy Criterion

Motivated by the need to deal with uncertainties in energy-optimal scheduling of flexible manufacturing systems (FMSs), this chapter considers a robust total energy optimization problem, where both productive and idle powers are considered as random variables (RVs). Uncertainties in energy measurement process are realistic, which can be induced by machine faults, sensor faults, and many other sources of disturbances. In practical cases, the probability distribution of each RV is often unknown, while only a finite number of observations are attainable. Based on the available observations and weighted p-timed Petri net (WTPN) models of FMSs, a scheduling method is developed by seeking the robust shortest path of WTPN reachability graph (RG) based on Renyi quadratic entropy. The practicality of the considered problem and the usability of our proposed scheduling algorithm are verified with simulations and also with industrial energy data logged from the same stamping system presented in Chapters 4 and 5.

It is worth noting that parametric family of Renyi entropies was introduced by Alfred Renyi in the mid 1950s as a mathematical generalization of Shannon entropy. Renyi wanted to find the most general class of information measure that preserved the additivity of statistically independent systems. Renyi's entropy was shown to be more flexible than Shannon and includes Shannon as a special case [160].

6.1 Background

In recent literature, energy-optimal scheduling of FMSs has been frequently addressed. For example, the energy consumption reduction was investigated through effective scheduling of machine startup and shutdown, where machines were assumed to have Bernoulli reliability model [26]. The control strategy for a closed-loop flow shop was designed to coordinate running of the machines and motion of pallets to minimize energy consumption in idle machines [27]. The robotic manufacturing systems were considered in [28], where energy optimal trajectories were generated for a range of execution times for the individual operations based on only a single simulation. In Chapter 5, the total energy optimization problem for a class of FMSs, where both idle and productive energy consumption were minimized. The system workflows were modelled using the WTPN models of FMSs and an effective scheduling algorithm was developed based on finding the shortest path of WTPN RG.

Most of the existent energy-efficient technologies for scheduling of FMSs often deal with deterministic manufacturing environments, where energy consumption is assumed to be deterministic and there is no uncertainty that would influence the established schedule. Real-world manufacturing is, however, dynamic and subjected to a wide range of uncertainties. In general, common sources of uncertainties in dynamic manufacturing environments have been classified into two categories, namely, resource-related uncertainties such as machine breakdown, machine degradation, tool wears, and job-related uncertainties such as rush jobs, job cancellation, stochastic processing time [108, 148]. In particular energy measurement process, uncertainties can be induced by machine faults, sensor faults, and other sources of disturbances [17].

Furthering the total energy optimization problem considered in Chapter 5, we consider its robust counterpart with resources' productive and idle powers are random variables (RVs). The WTPN models were conveniently used to model energy-efficient FMSs and finding the shortest path of WTPN RG was proven to be an effective solution for the total energy optimization problem as compared to the related algorithms such as branch and reduce and heuristic search. As such, the solution method for the robust problem considered in this paper will be developed based the WTPN models of FMSs. We now wish to solve the robust shortest path problem (RSPP) of WTPN RG, where the edge costs are RVs. We shall first review existent models of uncertainties for RSSP, and then select the finite observation model, which is most suitable for our industrial data. We then propose a novel robustness measure, called Renyi ME criterion, to enable computation of robust shortest path. While mean (expectation) is the most conventional adaption from deterministic to robust optimization, which basically gives the expected cost of a solution path, Renyi quadratic entropy is used to quantify the uncertainties. It will be shown that entropy provides a more effective uncertainty measure than the popularly used variance in the case of non-Gaussian distributed RVs, and the robust shortest path in Renyi mean-entropy criterion can be computed efficiently using Bellman's DP [141].

Simulations are carried out to evaluate the performance of Renyi ME criterion with related robustness measures. In the first simulation case, productive and idle powers are drawn from a truncated Gaussian distribution, while non-Gaussian distributions are used in the subsequent simulation cases. The practicality of considered problem is verified by industrial energy data logged from the same stamping system presented in Chapter 5. Both simulation and experiment results verify the effectiveness of Renyi ME criterion in terms of computational complexity and mean deviation from optimality.

6.2 Robust Energy Optimization Based on Renyi Mean-Entropy Criterion

This section extends the deterministic problem described in Chapter 5 with its robust counterpart, where token sojourn costs of places $\{c_j\}$ (productive and idle powers) are considered as RVs. The minimal token sojourn times of places $\{h_j\}$ (processing times of jobs on resources) are assumed to be deterministic variables. We shall first review only a part of literature on robust shortest path problem, which is most related and applicable to our industrial application. For a broad survey of robust shortest path problem, the readers are referred to [149] and the references therein for more details. Then, we proceed to introduce the Renyi ME criterion as a robustness measure.

6.2.1 Brief Overview on Robust Shortest Path Problem

The robust shortest path problem, also known as reliable or stochastic shortest path problem, aims to find the robust shortest path from a source node to a destination node of a directed graph, where the edge costs are RVs. The study of robust shortest path problem is crucial to deal with uncertainties in many real-world applications. Existent works on robust shortest path problem can be categorized by a) models of uncertainties which describe how the random edge costs are formulated and b) robustness measures which determine how uncertainties are quantified.

6.2.1.1 Models of Uncertainties

In general, there are three popular models of uncertainties for robust shortest path problem often considered in current literature, namely, the finite set of scenarios, interval data, and finite observation models. In the finite set of scenarios model, a scenario is associated with a unique value of each RV [150–152]. The finite set of scenarios model is suitable for applications where the interdependence of RVs are known *a priori*. On the other hand, the interval data model associates each RV with an interval (infinite set), which represents all possible values of the corresponding RV. It is worth noting that all combinations of values of RVs are allowed, thus making this model suitable for applications where there is a total independence between RVs [153–155].

In the finite observation model, each RV c_j is characterized solely by a finite set of numerical observations, denoted by $C = \left\{\tilde{c}_j^l; l = 1, 2, \ldots, N\right\}$, where any observation value $\tilde{c}_j^l$ is a realization value of c_j [156–159]. It is worth noting that c_j may have a single Dirac-δ distribution, in which case c_j is in fact a deterministic variable (a degenerate RV). In many applications, $\tilde{c}_j^l$ is the value of c_j in the l^{th} measurement, and therefore $\tilde{c}_j^l = \tilde{c}_j^{l'}$ with $l \neq l'$ is possible. In this chapter, the finite observation model is used, as it is most suitable for our industrial application. Throughout the chapter, all c_j are assumed to be continuous RVs and their observation instances are obtained by an independent and identically distributed (IID) sampling process.

6.2.1.2 Robustness Measures

To the best of our knowledge, W-C analysis (min-max regret) and mean-variance (MV) criterion are most applicable for the finite observation model.

In W-C analysis, a path is evaluated on the basis of its worst situation [149, 153, 155]. For computation of the robust shortest path on WTPN RG, the objective function of W-C analysis is formulated by

$$J^{*}\left(\mathbf{x}_0\right) = \min_{\mathbf{u}_{[1,H]} \in U} \max_{\mathbf{c} \in C} J\left(\mathbf{x}_0, \mathbf{u}_{[1,H]}, \mathbf{c}\right). \tag{6.1}$$

(6.1) indicates that computation of the robust shortest path of WTPN RG is performed by simply replaced each RV c_j by the greatest observation. In other words, computation of robust shortest path in W-C analysis is equivalent to computation of a deterministic shortest path.

In theory, W-C analysis requires least computation and provides an absolute guarantee on the lower bound of the solution value. However, it has been criticized by many researchers for its conservatism [151, 152]. Since it exclusively relies on the worst situation, making its solution very pessimistic, W-C value often hides the solution values on the other situations.

MV criterion for robust shortest path problem was first proposed by [156] and subsequently extended in [157–159], which seeks to minimize the weighted sum of the expected cost (mean) and variance of the solution paths. While minimization of expected cost is the most straightforward adaption from a deterministic to robust optimization, variance is adopted as an measure of robustness or reliability. As such, the objective function of MV criterion is formulated by

$$J^*(\mathbf{x}_0) = \min_{\substack{\mathbf{u}_{[1,H]}\in U \\ \mathbf{c}\in C}} \left\{ \mathrm{E}\left[J\left(\mathbf{x}_0, \mathbf{u}_{[1,H]}, \mathbf{c}\right)\right] + \nu \mathrm{Var}\left[J\left(\mathbf{x}_0, \mathbf{u}_{[1,H]}, \mathbf{c}\right)\right] \right\}, \tag{6.2}$$

where $\nu \in \mathbb{R}^+$ denote a weight parameter representing the importance of robustness, and $\mathrm{E}[\cdot]$ and $\mathrm{Var}[\cdot]$ are standard terms to denote expected value and variance, respectively. (6.2) indicates that among all admissible paths of WTPN RG, the robust shortest path in MV criterion should minimize the weighted sum of expected energy cost and variance. It is worth noting from (5.19) that the cost of a path $J\left(\mathbf{x}_0, \mathbf{u}_{[1,H]}, \mathbf{c}\right)$ is a linear combination of RVs c_j, whose expected value and variance can be computed easily. The extreme cases $\nu = 0$ and $\nu \to \infty$ indicate situations where one is only concerned with minimization of either expected cost or variance, respectively. Although, MV criterion requires more computational efforts to estimate the expected value and variance of a solution path from observation data, it compensates the conservatism of W-C analysis.

6.2.2 Renyi Mean-Entropy Criterion

In information theory, Renyi entropy is a generalized entropy that quantifies the diversity and uncertainties of a RV. In this chapter, the concept of Renyi entropy is applied to robust shortest path problem, resulting a novel Renyi ME criterion for robustness measure, which minimizes the weighted sum of the expected cost and accumulative Renyi entropy of the solution paths. The objective function of Renyi

ME criterion is formulated by

$$J^*(\mathbf{x}_0) = \min_{\substack{\mathbf{u}_{[1,H]} \in U \\ \mathbf{c} \in C}} \left\{ \mathrm{E}\left[J\left(\mathbf{x}_0, \mathbf{u}_{[1,H]}, \mathbf{c}\right)\right] + \nu \sum_{n=1}^{H} H_2\left[J\left(\mathbf{x}_{n-1}, \mathbf{u}_n, \mathbf{c}\right)\right] \right\}, \tag{6.3}$$

where $H_2[\cdot]$ denotes Renyi quadratic entropy [160], which is defined by

$$H_2[c_j] = -\log \int_{-\infty}^{\infty} f_{c_j}^2(\tilde{c}_j) d\tilde{c}_j, \tag{6.4}$$

and ν denotes a weight parameter for linear scalarization. (6.3) indicates that among all admissible paths of WTPN RG, the robust shortest path in Renyi mean-entropy criterion should minimize the weighted sum of expected energy cost and Renyi entropy. Decision makers can adjust ν based on the specific economic situations. With different parameter ν, different Pareto optimal solutions are produced. For simplicity but without loss of generality, the expected energy cost and uncertainty are weighted equally for both MV and Renyi ME criteria in all simulations and industrial application of this paper.

Using Renyi entropy as a measure of dispersion and uncertainties compensates a major drawback of MV criterion. It is worth noting that usage of variance or standard deviation, which are second-order statistical central moments, relies heavily on the Gaussianity and linearity assumptions. From a statistical viewpoint, entropy can be interpreted as a measure of disparity of a probability distribution function (PDF) from the uniform. On the other hand, variance measures dispersion of a PDF from the mean. Although both entropy and variance are measures of dispersion and uncertainties, they have quite substantial and subtle differences. The readers are referred to [160, 161] and the references therein for more details on comparison between variance and entropy.

Remark 6.1 Our proposed Renyi ME criterion uses accumulative entropy instead of absolute entropy, which is written by $H_2\left[J\left(\mathbf{x}_0, \mathbf{u}_{[1,H]}, \mathbf{c}\right)\right]$. Our idea is to formulate a sufficient robustness measure for scheduling purposes, such that the robust shortest path can be computed efficiently using DP instead of exhaustive or heuristic search algorithms.

We have the following results.

Theorem 6.1 *Computation of $J^*(\mathbf{x}_0)$ in (6.3) has a recursive formulation which satisfies Bellman's principle of optimality.*

Proof: Straightforward computations give

$$\begin{aligned}
\mathrm{E}\left[J\left(\mathbf{x}_0, \mathbf{u}_{[1,H]}, \mathbf{c}\right)\right] &= \mathrm{E}\left[\frac{1}{f_{\mathrm{dp}}}\sum_{n=1}^{H}\mathbf{c}^T\mathbf{x}_n\right] \\
&= \mathrm{E}\left[\frac{1}{f_{\mathrm{dp}}}\sum_{n=1}^{H}\mathbf{c}^T\left(\mathbf{x}_{n-1} + \mathbf{W}^T\mathbf{u}_n\right)\right] \\
&= \sum_{n=1}^{H}\mathrm{E}\left[\frac{1}{f_{\mathrm{dp}}}\mathbf{c}^T\left(\mathbf{x}_{n-1} + \mathbf{W}^T\mathbf{u}_n\right)\right] \\
&= \sum_{n=1}^{H}\mathrm{E}\left[J\left(\mathbf{x}_{n-1}, \mathbf{u}_n, \mathbf{c}\right)\right]. \qquad (6.5)
\end{aligned}$$

As such, (6.3) is equivalent to

$$\begin{aligned}
J^*(\mathbf{x}_0) &= \min_{\substack{\mathbf{u}_{[1,H]}\in U \\ \mathbf{c}\in C}} \sum_{n=1}^{H}\left\{\mathrm{E}\left[J\left(\mathbf{x}_{n-1}, \mathbf{u}_n, \mathbf{c}\right)\right] + \gamma H_2\left[J\left(\mathbf{x}_{n-1}, \mathbf{u}_n, \mathbf{c}\right)\right]\right\} \\
&= \min_{\substack{\mathbf{u}_{[1,H]}\in U \\ \mathbf{c}\in C}} \sum_{n=1}^{H} J'\left(\mathbf{x}_{n-1}, \mathbf{u}_n, \mathbf{c}\right). \qquad (6.6)
\end{aligned}$$

It can be seen that (6.6) has exactly the same additive formation with (5.19), which was proven to satisfy Bellman's principle of optimality in Theorem 5.2. In other words, the additivity of (6.6) and (5.19) indicate that the cost of a solution path is the sum of the costs of its edges, where the cost-to-go (arc cost) from $\mathbf{x}_n$ to next stage is now defined by $J'(\mathbf{x}_{n-1}, \mathbf{u}_n, \mathbf{c})$. ∎

It is worth noting that computation of robust shortest path in W-C analysis also complies with Bellman's principle of optimality. In current literature, the robust shortest path problem in W-C analysis has been often solved using DP, Dijkstra's algorithm, Bender's decomposition, *etc.* [149]. From Theorem 6.1, it is obvious that all of these algorithms can be applied to Renyi ME criterion as well.

Remark 6.2 The robust shortest path problems, whose objective functions contain nonlinear components such as the absolute variance $\mathrm{Var}\left[J\left(\mathbf{x}_0, \mathbf{u}_{[1,H]}, \mathbf{c}\right)\right]$ or absolute entropy $H_2\left[J\left(\mathbf{x}_0, \mathbf{u}_{[1,H]}, \mathbf{c}\right)\right]$, are widely known in current literature as non-additive robust shortest path problems, where the cost of a solution path is not the sum of the costs of its edges. It was proven that the non-additivity violates Bellman's principle of optimality, and non-additive robust shortest path problems are often solved by B&B algorithms with Lagrangian relaxations [162, 163].

6.2.3 Non-Parametric Estimation of Edge Costs

The edge costs including $\mathrm{E}\left[J\left(\mathbf{x}_{n-1}, \mathbf{u}_n, \mathbf{c}\right)\right]$ and $H_2\left[J\left(\mathbf{x}_{n-1}, \mathbf{u}_n, \mathbf{c}\right)\right]$ should be estimated directly from the observations in a non-parametric way, since PDFs of RVs c_j

are not known *a priori* in the finite observation model.

It is worth noting that Shannon entropy was effectively used as a robustness measure, especially in the fields of finance and economics [164–166]. In these reported works, the PDFs or types of fuzzy membership function of RVs are known *a priori*, and hence Shannon entropy of RVs can be easily computed.

On the other hand, Renyi entropy is chosen in this paper as it provides easy non-parametric estimation and overcomes the difficulty in computing Shannon entropy directly from finite observation data. With the recent invention of information-theoretic learning (ITL) [160], a non-parametric estimator of Renyi entropy $H_2\left[c_j\right]$ with computational complexity $O\left(N\right)$ was proposed, where N is the number of observations of c_j. ITL provided a convenient way to estimate Renyi entropy from finite observation data by skipping the PDF estimation. The traditional estimation of Shannon entropy from observation data will follow the route: data $\rightarrow$ PDF estimation $\rightarrow$ entropy estimation. Notice that entropy is a scalar, but as an intermediate step one has to fit the PDF of a RV, which is much more computational intensive. Using ITL, the estimation of Renyi entropy skips the PDF curve fitting step and follows the direct route: data $\rightarrow$ entropy estimation.

Based on the results of ITL, we now wish to estimate $\mathrm{E}\left[J\left(\mathbf{x}_{n-1}, \mathbf{u}_n, \mathbf{c}\right)\right]$ and $H_2\left[J\left(\mathbf{x}_{n-1}, \mathbf{u}_n, \mathbf{c}\right)\right]$ non-parametrically. Recall the definition of Renyi entropy given in (6.4), the kernel (Parzen) estimate of f_{c_j} using Gaussian kernel function $G_\sigma\left(\cdot\right)$

given by

$$f_{c_j}\left(\tilde{c}_j\right) \approx \frac{1}{N} \sum_{l=1}^{N} G_\sigma\left(\tilde{c}_j - c_j^l\right), \tag{6.7}$$

where σ is the kernel size or bandwidth parameter and

$$G_\sigma\left(\tilde{c}_j - c_j^l\right) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2\sigma^2}\left(\tilde{c}_j - c_j^l\right)^2}. \tag{6.8}$$

We have the following results.

Theorem 6.2 *The following estimation holds.*

$$\mathrm{E}\left[J\left(\mathbf{x}_{n-1}, \mathbf{u}_n, \mathbf{c}\right)\right] \approx \frac{1}{N} \sum_{x_{jn} \neq 0} \sum_{l=1}^{N} \tilde{c}_j^l x_{jn}. \tag{6.9}$$

Proof: It is obvious that $J\left(\mathbf{x}_{n-1}, \mathbf{u}_n, \mathbf{c}\right)$ is a multivariate function of RVs c_j, since

$$J\left(\mathbf{x}_{n-1}, \mathbf{u}_n, \mathbf{c}\right) = \sum_{x_{jn} \neq 0} c_j x_{jn}. \tag{6.10}$$

Let us denote by $f_{c_1,\ldots,c_{|P|}}\left(\tilde{c}_1, \ldots, \tilde{c}_{|P|}\right)$ the joint PDF for continuous RVs $\{c_j, j = 1, 2, \ldots, |P| : x_{jn} \neq 0\}$. By definition, one has

$$\mathrm{E}\left[J\left(\mathbf{x}_{n-1}, \mathbf{u}_n, \mathbf{c}\right)\right] = \int \cdots \int_{-\infty}^{\infty} J\left(\mathbf{x}_{n-1}, \mathbf{u}_n, \mathbf{c}\right) f_{c_1,\ldots,c_{|P|}}\left(\tilde{c}_1, \ldots, \tilde{c}_{|P|}\right)\, d\tilde{c}_1 \cdots d\tilde{c}_{|P|}. \tag{6.11}$$

Since c_j are IID RVs, (6.11) is equivalent to

$$\begin{aligned}
\mathrm{E}\left[J\left(\mathbf{x}_{n-1}, \mathbf{u}_n, \mathbf{c}\right)\right] &= \int \cdots \int_{-\infty}^{\infty} \sum_{j=1}^{|P|} f_{c_1}\left(\tilde{c}_1\right) \ \cdots f_{c_{|P|}}\left(\tilde{c}_{|P|}\right) d\tilde{c}_1 \cdots d\tilde{c}_{|P|} \\
&= \sum_{j=1}^{|P|} \int_{-\infty}^{\infty} \tilde{c}_j x_{jn} f_{c_j}\left(\tilde{c}_j\right) d\tilde{c}_j \\
&\approx \sum_{j=1}^{|P|} \int_{-\infty}^{\infty} \tilde{c}_j x_{jn} \frac{1}{N} \sum_{l=1}^{N} G_\sigma\left(\tilde{c}_j - c_j^l\right) d\tilde{c}_j \\
&= \frac{1}{N} \sum_{j=1}^{|P|} \sum_{l=1}^{N} x_{jn} \int_{-\infty}^{\infty} \tilde{c}_j G_\sigma\left(\tilde{c}_j - c_j^l\right) d\tilde{c}_j \\
&= \frac{1}{N} \sum_{x_{jn} \neq 0} \sum_{l=1}^{N} \tilde{c}_j^l x_{jn}.
\end{aligned} \tag{6.12}$$

∎

Theorem 6.3 *The following estimation holds.*

$$H_2\left[J\left(\mathbf{x}_{n-1}, \mathbf{u}_n, \mathbf{c}\right)\right] \approx -\log\left\{\prod_{x_{jn} \neq 0} \frac{1}{N^2} \sum_{l=1}^{N} \sum_{g=1}^{N} G_{\sigma\sqrt{2}}\left(\tilde{c}_j^g - \tilde{c}_j^l\right)\right\}. \tag{6.13}$$

Proof: From (6.4) and (6.10), one has

$$
\begin{aligned}
& H_2\left[J\left(\mathbf{x}_{n-1}, \mathbf{u}_n, \mathbf{c}\right)\right] \\
& = -\log \int \cdots \int_{-\infty}^{\infty} f^2_{c_1,\ldots,c_{|P|}}\left(\tilde{c}_1, \ldots, \tilde{c}_{|P|}\right)\ d\tilde{c}_1 \cdots d\tilde{c}_{|P|} \\
& = -\log \int \cdots \int_{-\infty}^{\infty} f^2_{c_1}\left(\tilde{c}_1\right)\ \cdots f^2_{c_{|P|}}\left(\tilde{c}_{|P|}\right) d\tilde{c}_1 \cdots d\tilde{c}_{|P|} \\
& = -\log \prod_{j=1}^{|P|} \int_{-\infty}^{\infty} f^2_{c_j}\left(\tilde{c}_j\right) d\tilde{c}_j \\
& \approx -\log \prod_{j=1}^{|P|} \int_{-\infty}^{\infty} \left\{\frac{1}{N} \sum_{l=1}^{N} G_\sigma\left(\tilde{c}_j - c_j^l\right)\right\}^2 d\tilde{c}_j \\
& = -\log \prod_{j=1}^{|P|} \frac{1}{N^2} \int_{-\infty}^{\infty} \sum_{l=1}^{N} \sum_{g=1}^{N} G_\sigma\left(\tilde{c}_j - c_j^l\right) G_\sigma\left(\tilde{c}_j - c_j^g\right) d\tilde{c}_j \\
& = -\log \left\{\prod_{x_{jn}\neq 0} \frac{1}{N^2} \sum_{l=1}^{N} \sum_{g=1}^{N} G_{\sigma\sqrt{2}}\left(\tilde{c}_j^g - \tilde{c}_j^l\right)\right\}. \qquad (6.14)
\end{aligned}
$$

∎

The result is easily obtained by noticing a little-realized fact that the integral of the product of two Gaussians is exactly evaluated as the value of the Gaussian computed at the difference of the arguments and whose variance is the sum of the variances of the two original Gaussian functions.

6.3 Simulations

In this section, simulations are carried out to evaluate Renyi ME criterion against W-C analysis and MV criterion. Simulation results will be presented in three separate cases. In each case, two hundred observations of each RV are drawn from an IID sampling process. Half of them is used to compute the robust shortest paths in three different robustness measures, while the remaining half is used for evaluation of the solution quality in terms of mean deviation from optimality. As such, one has $N = 100$. In all simulation cases, the estimated Renyi entropy from finite observations is used instead of the true Renyi entropy calculated from the PDFs.

6.3.1 Probability Distributions

In the first case, we shall consider the truncated normal distribution. In statistics, a truncated distribution is a conditional distribution that results from restricting the domain of some other probability distribution. Truncated distributions arise in practical statistics in cases where the values of continuous RVs are bounded. In this paper, the PDF of a normally distributed RV whose value is bounded within an interval $[0, \infty]$ is considered, as it is obvious that power consumption can not be negative. Its PDF is given by

$$f_{c_j}\left(\tilde{c}_j\right) = \begin{cases} \frac{\frac{1}{\sigma_j}\phi\left(\frac{\tilde{c}_j-\mu_j}{\sigma_j}\right)}{1-\Phi\left(-\frac{\mu_j}{\sigma_j}\right)} \text{ if } 0 \leq \tilde{c}_j \leq \infty, \\ 0 \text{ otherwise,} \end{cases} \tag{6.15}$$

where $\phi\left(\cdot\right)$ is the PDF of the standard normal distribution and $\Phi\left(\cdot\right)$ is its cumulative distribution function. μ_j and σ_j are standard terms to denote the mean and standard deviation, respectively.

Next, we shall consider two non-Gaussian distributions, namely, the continuous uniform and bimodal distributions. In statistics, the continuous uniform distribution or rectangular distribution is a family of symmetric probability distributions such that all intervals of the same length on the distribution's support are equally probable. The support of $f_{c_j}(\tilde{c}_j)$ is defined by the two parameters, u_{1_j} and u_{2_j}, which are its minimum and maximum values. As power consumption can not be negative, we set $u_{1_j} \geq 0$. As such, the PDF of the continuous uniform distribution is defined by

$$f_{c_j}(\tilde{c}_j) = \begin{cases} \frac{1}{u_{2_j} - u_{1_j}} & \text{if } \; u_{1_j} \leq \tilde{c}_j \leq u_{2_j}, \\ 0 & \text{if } \; \tilde{c}_j < u_{1_j} \; \text{ or } \; \tilde{c}_j > u_{2_j}, \end{cases} \tag{6.16}$$

On the other hand, a bimodal distribution is a continuous probability distribution with two different modes. In this chapter, a bimodal distribution is developed as a linear combination of two truncated normal distributions presented in (6.15) with the same variance but different means. Its PDF is given by

$$f_{c_j}(\tilde{c}_j) = \begin{cases} w_1 \frac{\frac{1}{\sigma_j}\phi\left(\frac{\tilde{c}_j - \mu_{1j}}{\sigma_j}\right)}{1 - \Phi\left(-\frac{\mu_{1j}}{\sigma_j}\right)} + w_2 \frac{\frac{1}{\sigma_j}\phi\left(\frac{\tilde{c}_j - \mu_{2j}}{\sigma_j}\right)}{1 - \Phi\left(-\frac{\mu_{2j}}{\sigma_j}\right)} \text{ if } 0 \leq \tilde{c}_j \leq \infty, \\ 0 \text{ otherwise,} \end{cases} \tag{6.17}$$

where w_i are weight parameters, $w_i \geq 0$ and $\sum w_i = 1$.

6.3.2 Simulation Setup and Results

All simulation cases are carried on fully FMSs. A fully FMS is defined as having all machining jobs are choice jobs and each machining resource can perform all machining jobs. It is reminded that, each choice job is associated with a routing resource, which routes parts. For simplicity, we let all routing resources be non-shared. Our purpose

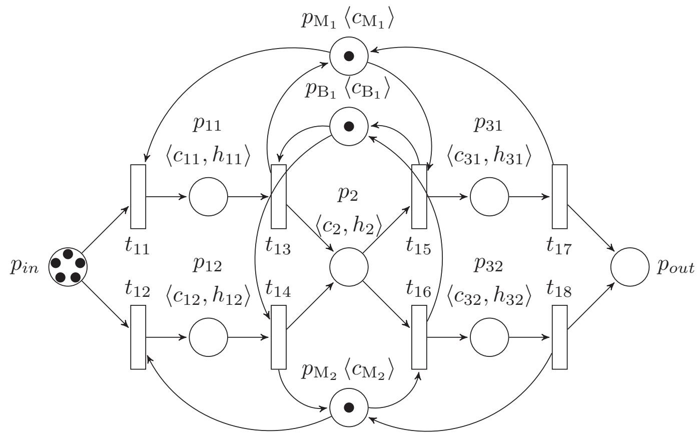


Figure 6.1: Marked WTPN models of a fully FMS.

of using fully FMSs is to maximize the number of overlapping subproblems and the solution space. The FMS sizes for all simulation cases are reported in Table 6.1.

Example 6.1 A marked WTPN models of a fully FMS with two machining jobs and two machining resources are given in Fig. 6.1. There are two machining jobs: v_1 represented by places p_{11} and p_{12}, and v_3 represented by places p_{31} and p_{32}. These machining jobs are performed by two machining resources: M_1 represented by place p_{M_1}, and M_2 represented by place p_{M_2}. Places p_{11} and p_{12} indicate that v_1 is performed by M_1 and M_2, respectively. Similarly, places p_{31} and p_{32} indicate that v_3 is performed by M_1 and M_2, respectively. There is one routing job v_3 represented by place p_2, and a routing resource B_1 represented by place p_{B_1}. Places p_{in} and p_{out} represents input and output buffers, respectively.

Table 6.1: Fully FMS Sizes for Simulation Test Cases

Test case	Machining jobs $\lvert V_z \rvert$	Machining resources $\lvert R_z \rvert$	No. of parts $\varphi(\pi)$
1	10	5	10
2	15	5	10
3	20	10	10
4	20	10	20
5	25	10	20
6	30	15	20
7	35	15	20
8	35	20	20
9	40	20	20

For each evaluation iteration i, let us denote by $J_{\text{opt}}(i)$ the cost of the shortest path, which is computed based on actual observations at iteration i, and by $J_{\text{robust}}(i)$ the cost of the path selected by a robustness measure. As such, the mean deviation from optimality of a robustness measure is given by

$$\text{mean dev (\%)} = \frac{1}{N}\sum_{i=1}^{N} \frac{J_{\text{robust}}(i) - J_{\text{opt}}(i)}{J_{\text{opt}}(i)} \times 100\%. \tag{6.18}$$

Simulation results under three distributions are reported in Figs. 6.2–6.4. The results are consistent with our prior analyses in Section 6.2. In particular, W-C analysis has the highest mean deviation for all three distributions due to its conservatism. In the case of Gaussian distribution, MV criterion slightly outperforms Renyi ME criterion. With the presence of non-Gaussian distributions, the robustness of MV criterion decreases significantly as compared to Renyi ME criterion.

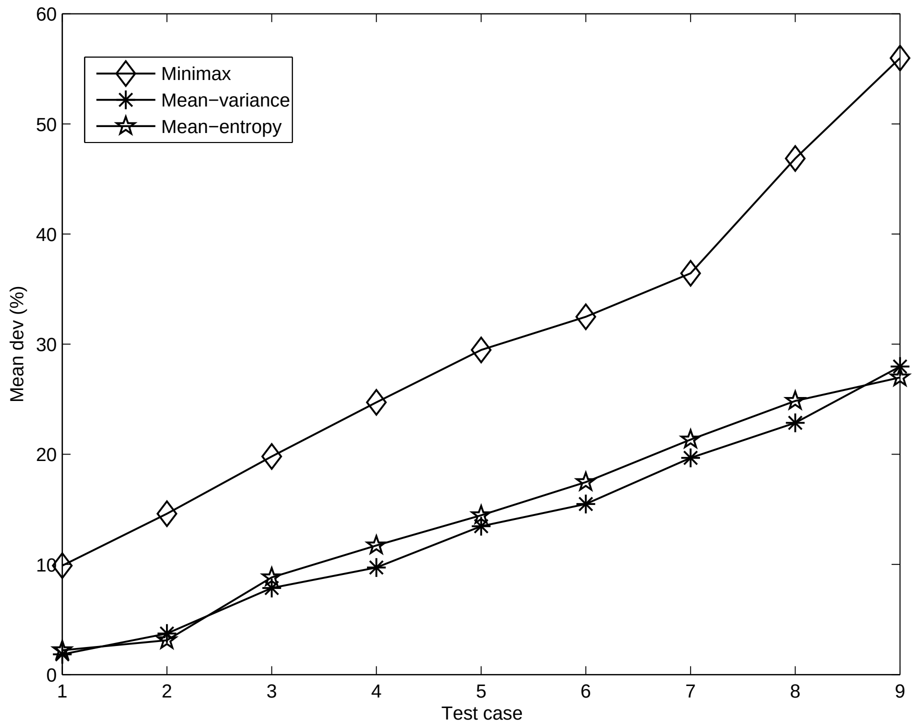


Figure 6.2: Mean deviation of three robustness measures under Gaussian distribution.

6.4 Industrial Application

The same industrial applicaiton presented in Chapters 4 and 5 is used to verify the practicality of considered robust problem and the feasibility of Renyi ME criterion. However, the productive and idle powers are sampled with finite number of observations instead of being averaged.

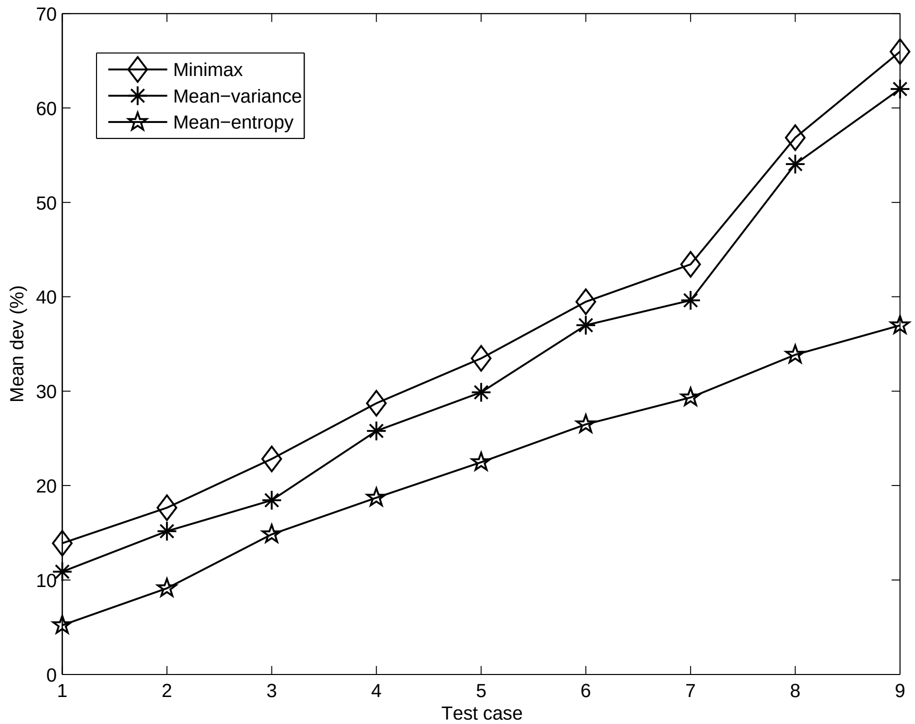


Figure 6.3: Mean deviation of three robustness measures under uniform distribution.

6.4.1 Robust Energy Analysis of Stamping Process

An energy profile recorded when a stamping machine r_i performs a specific job v_j on multiple parts is shown in Figure 4.5. The stamping machines are of different working conditions as well as energy consumption profiles. The entire stamping cycle can be divided into three main states, namely, productive, idle, and off. In idle and off states, the power data are observed to be relatively constant. As such, the idle power b_i is considered as a deterministic variable for simplicity but without loss of generality. It is then natural to set $H_2[b_i] = 0$, and $\mathrm{E}[b_i]$ is set to the average idle

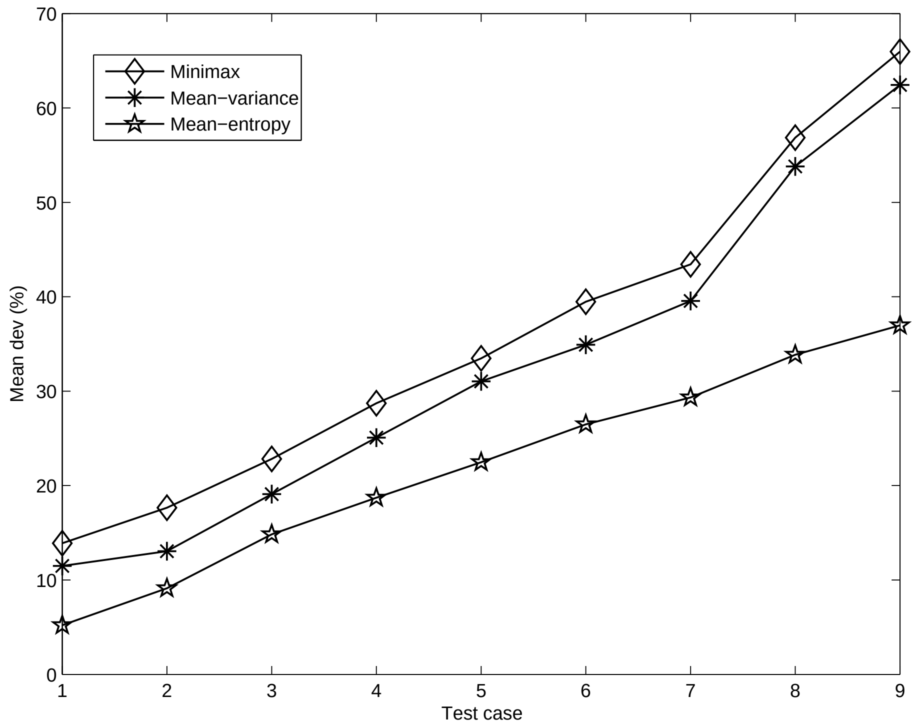


Figure 6.4: Mean deviation of three robustness measures under bimodal distribution.

power.

In productive state, many spikes are generated, and each spike is observed every time the stamping press moves down to perform stamping jobs. There is a wide range in productive powers, even for different machines of the same model. This is due to a multitude of factors, *e.g.,* tooling, machine loading, machine degradation, sensor faults, *etc.* Hence, we consider productive power a_{ij} as a RV. A stamping job generally generate a triangular spike in power consumption profile, as such each observation is calculated by the average power consumption from the rising edge to the falling edge of a triangular spike.

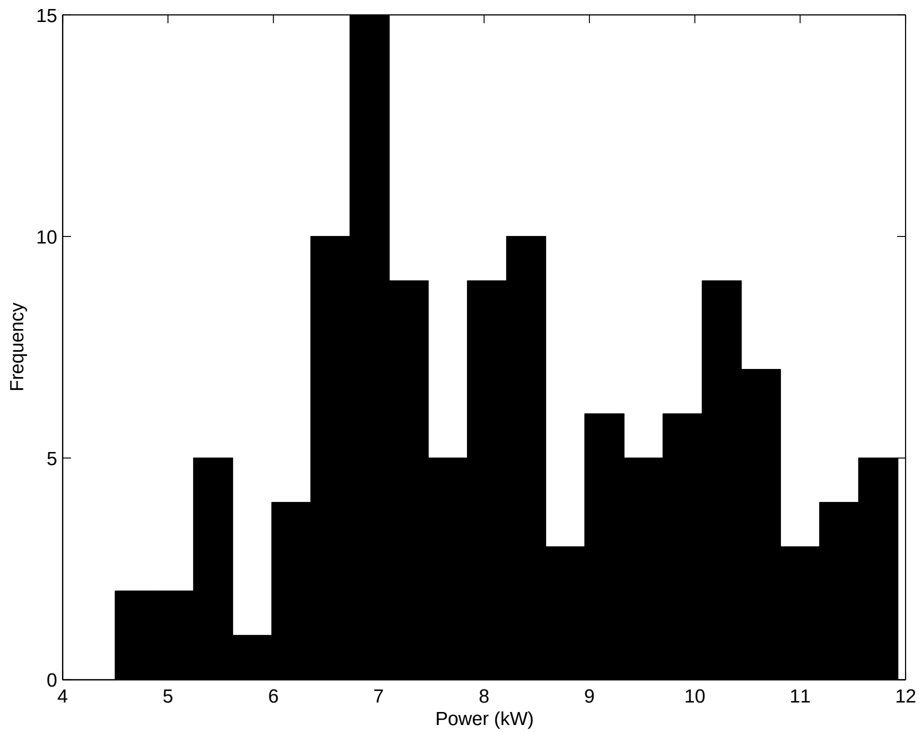


Figure 6.5: Histogram of c_{113} with 120 observations.

6.4.2 Results and Discussions

This computational test is carried on a digital computer equipped with Intel Core i7 processor and 32 gigabyte RAM. Computation of robust shortest path in W-C analysis and Renyi ME criterion are carried out using DP in MATLAB, while ILOG CPLEX v12.4 is used to compute the robust shortest path in MV criterion. CPLEX solves the robust shortest path problem using B&B procedure using Lagrangian relaxations as lower bounds to cut out dominated nodes.

Two performance metrics are considered, which are computational time and de-

Table 6.2: Performance Comparisons of W-C Analysis, MV, and Renyi ME Criteria

Parts	Mean dev (%)			Computational time (s)		
$\varphi\,(\pi)$	W-C	MV	Renyi ME	MV	Renyi ME	W-C
5	13.56	11.90	6.21	25.58	22.32	15.27
10	16.22	14.36	11.52	49.32	27.43	23.37
15	20.58	18.14	13.82	78.43	35.96	29.37
20	23.50	21.47	15.40	164.21	56.58	47.82
30	28.17	27.90	20.41	583.48	82.76	67.24
40	33.26	32.87	26.37	>1000	135.90	118.73
50	37.83	34.17	29.21	>1000	324.61	302.32
80	53.77	49.03	37.39	>1000	714.23	696.83
100	59.46	55.03	40.60	>1000	>1000	>1000

viation from optimality. In our experiment, the deviation from optimality is computed as described in (6.18). For all robustness measures, computational time is calculated as the total time required to estimate necessary quantities and to find the shortest path. In particular, computational time of W-C analysis includes searching for $\sup\{C\}$, where $\sup\{\cdot\}$ denotes supremum of a set, while computational time of MV and Renyi ME criteria includes estimation of $\mathrm{Var}\left[J\left(\mathbf{x}_0, \mathbf{u}_{[1,H]}, \mathbf{c}\right)\right]$ and $H_2\left[J\left(\mathbf{x}_{n-1}, \mathbf{u}_n, \mathbf{c}\right)\right]$, respectively. $f_{\mathrm{dp}} = 10$ Hz is used. We set the time limit for the test as 1000 s in Renyi ME run. The expected values of power costs are generally in the range $0 < \mathrm{E}\left[c_j\right] \leq 10$ (kW), while their Renyi entropy values are in the range $0 \leq H_2\left[c_j\right] \leq 1$ (bits). To weight energy cost and uncertainty equally, $\nu = 10$ is chosen in our simulation experiments. Similarly, different values of ν can be chosen, *e.g.,* $\nu = 20$ and $\nu = 30$ if the decision maker wants to weight uncertainty two and three times more importantly than energy cost, respectively.

One hundred and twenty observations are drawn for each RV from an IID sampling process. Half of data is used to compute the robust shortest paths in three different robustness measures, while the remaining is used for evaluation of the solution quality in terms of mean deviation from optimality and computational time. As such, one has $N = 60$. It is well-known by the famous central limit theorem that a sufficiently large number of IID observations of a RV will be approximately normally distributed. In practical cases, it is not always expected that a sufficiently large number of observations is available. As such, our proposed Renyi ME criterion is useful for cases where normal distributions are not observed. A histogram of RV c_{113} is shown in Fig. 6.5, where a non-Gaussian distribution can be clearly observed.

Numerical results are presented in Table 6.2, where it can be seen that our proposed Renyi ME criterion achieves least deviation from optimality as compared to W-C analysis and MV criterion for all test cases. In terms of computational time, MV criterion exceeds 1000 s from the test case of forty parts, while our proposed Renyi ME criterion is still computationally efficient up to the test case of one hundred parts. In our numerical computation, it is also observed that the Renyi ME and MV criteria provide almost equivalent maximum deviation, and their maximum deviation can be up to 27% more than the W-C analysis in all test cases.

6.5 Summary

In this chapter, a robust total energy optimization problem for a class of FMSs described in Chapter 2 was studied. Uncertainties in productive and idles powers are included using the finite observation model. Based on WTPN models of FMSs, the robust optimal schedule is determined by solving the robust shortest path problem of WTPN RG. A novel Renyi ME criterion was proposed as a new robustness measure, and the robust shortest path can be computed efficiently using DP. The effectiveness of Renyi ME criterion was rigorously verified with mathematical formulations as well as simulations and an industrial application at one stamping company. Our results showed that Renyi ME criterion achieved the least deviation from optimality (less than 41% for all experiment test cases) with significant reduction in computational time as compared to related works in current literature.

Beyond the robust energy-optimal scheduling of FMSs presented in this chapter, it is worth noting that Renyi ME criterion can be applied to solve any industrial problems, which can be cast into the robust shortest path problem. Some realistic examples include transportation network routing [156,159], portfolio management [167], and wireless sensor networks scheduling [168], *etc.*

In the next chapter, we conclude this dissertation and discuss future research directions.

Chapter 7

Conclusion and Future Work

The growing "green" trend for the next generation of high-performance manufacturing industries demands strong energy consumption reduction capabilities. This dissertation focuses on operational control and scheduling of FMSs with and without uncertainties in energy data for enhanced energy efficiencies, as well as time series analysis of energy data for real-time intelligent energy monitoring. The main findings and results presented in this dissertation are :

1. Proposing a novel approach to reduce the number of required sensors in process state tracking through identifying the operational states of MPs by extracting useful information and features in energy data. FSMs are used to model MPs, and a two-stage framework for online classification of real-time energy data in terms of MP operational states is proposed. To justify our proposed framework, comparative experiments with an existing framework are evaluated on two industrial applications, an injection moulding system and a stamping system.

Based on the obtained results, an energy data-driven decision support system (DSS) is designed to use real-time energy measurements and process operational states for effective decision-making, enabling high-performance manufacturing.

2. Formulating a total energy optimization for FMSs using WTPN and proposes a new RG-based DP scheduling algorithm. The resulted schedules are obtained with low deviation from global optimality and within adequate computational time as compared to the related works in current literature.

3. Extending the deterministic total energy optimization problem with its robust counterpart to deal with uncertainties in energy measurements. A novel robustness measure is proposed, called Renyi ME criterion, using Renyi quadratic entropy for searching the robust shortest path of WTPN RG. The effectiveness of Renyi ME criterion is compared with the related works in current literature in terms of computational complexity and deviation from global optimality.

4. Proposing an integrated control and scheduling framework, which includes two modules: the ADEC and a novel MTME, to optimize the sum of energy cost and tardiness penalty in FMSs under power uncertainties due to machine degradation. Our proposed framework is applied to an industrial stamping system with power consumption uncertainties formulated using three different probability distributions to verify it effectiveness as compared the related work in current literature in terms of deviation from Pareto optimality and mean in-

terrupted time.

Future energy-efficient technologies should be improved further to be feasible for even more dynamic and complex manufacturing environments [169]. As such, excellent capabilities to deal with sensor faults and failures, machine failures, and equipment degradation, *etc.*, will be essential. In view of the results obtained, the following works should be emphasized in future research:

1. Energy-efficient robust scheduling of FMSs under resource failures:

   Almost all existing works for FMS scheduling to date have assumed that the allocated resources do not fail. Nevertheless, it is crucial for FMSs to have the ability to tolerate or recover errors or failures automatically since all FMSs are error-prone, and they are nowadays usually complex and large-scale systems consisting of multi-cells and multi-stages of resources. To address this problem, let us denote by $\lambda(r_i)_{\tau_k}$ the failure probability of r_i at firing instance τ_k. It was shown in Chapter 5 that to drive the WTPN models from state $\mathbf{x}_k$ to state $\mathbf{x}_{k+1}$, a transition must be fired (*i.e.*, a job must be performed by some resources). Let R_{yk} be the set of resources required to drive WTPN models of FMSs from $\mathbf{x}_k$ to $\mathbf{x}_{k+1}$. As such, the objective function presented in (5.11) can be revised by

   $$J^*(\mathbf{x}_k, \tau_k) = \left\{ J(\mathbf{x}_k, \tau_k) - \nu \prod_{k=0}^{K-1} \prod_{r_i \in R_{yk}} \left(1 - \lambda(r_i)_{\tau_k}\right) \right\}, \tag{7.1}$$

   where the newly added term can be defined by the reliability of a schedule. ν is a weight parameter. Minimization of $J^*(\mathbf{x}_k, \tau_k)$ is a multi-objective optimization problem, where total energy consumption is minimized and the reliability of the schedule is maximized.

2. Energy-efficient robust scheduling of FMSs with support set of RVs:

   In many realistic FMSs, RVs c_j cannot be sampled, and each of them is only associated with an interval (support set) denoted by $\left[\underline{c_j}, \overline{c_j}\right]$, where $\underline{c_j}$ and $\overline{c_j}$ denotes the minimal and maximal values of c_j, respectively. Interval $\left[\underline{c_j}, \overline{c_j}\right]$ represents all possible values of c_j [153–155]. As such, the robust energy optimization problem considered in Chapter 6 should be extended to deal with interval data models of uncertainties. Current literature on robust optimization with interval data models of uncertainties often relies on W-C analysis, which has been criticized by many researchers for its conservatism [151, 152]. Since W-C analysis exclusively relies on the worst situation, making its solution very pessimistic, W-C value often hides the solution values on the other situations.

3. Energy-efficient condition-based maintenance (CBM) of FMS equipment:

   The machinery costs needed to operate machinery throughout its useful life can easily exceed the original equipment cost. Although the improvement in operational control and scheduling of FMSs can improve the energy efficiencies, daily operations and maintenance play even more important roles in reducing the overall environmental impact and energy cost to operate these machines. Traditional CBM in FMSs often focuses on the reliability of a resource in terms of its remaining useful life. Recently, the results of tests conducted by a

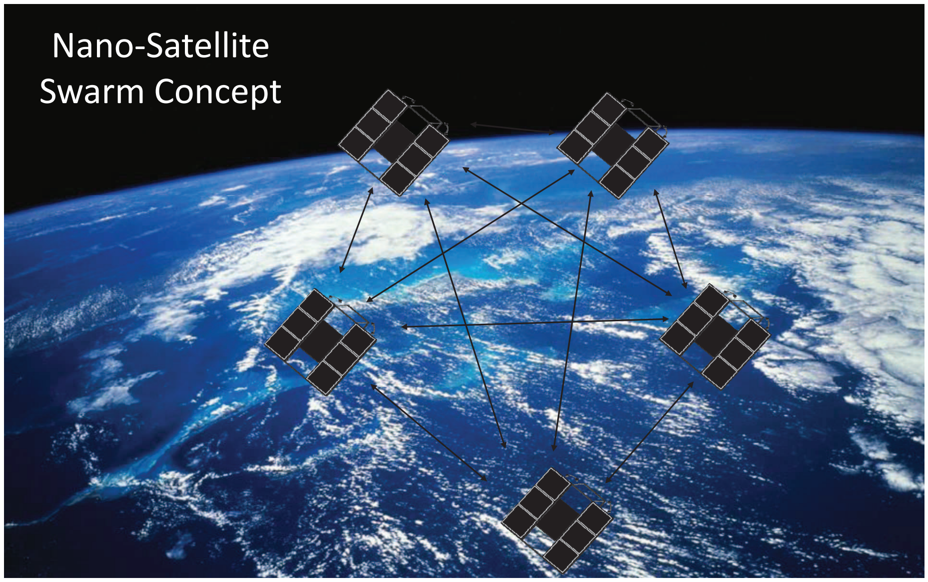


Figure 7.1: The nano-satellite swarm concept.

major process industrial user showed that the energy efficiency and machinery reliability are actually closely correlated [47].

4. Energy-efficient path-planning and scheduling of event-based nano-satellite swarms:

   Although our proposed framework was demonstrated on an FMS, it can be applied to other flexible discrete event systems as well. A potential application is energy-efficient path-planning for event-based nano-satellite swarms. Nano-satellites are now commonly being used in swarm platforms to work collaboratively to replace the more bulkier and costly satellites. An example of nano-satellite swarms is the orbital low frequency array (OLFAR) project, which aims to design a low-frequency distributed radio telescope in space as shown in Figure 7.1 [170]. OLFAR includes nano-satellites with dimension of $10\times10\times10$ cm and weight 1.3 kg. For most nano-satellite systems, reduction of energy consumption is a critical issue. A satellite with lower energy requirements requires a smaller energy source and a lighter battery pack, both of which directly translate into weight and cost savings [171, 172].

Bibliography

[1] S. H. Schneider, "What is dangerous climate change?" *Nature*, vol. 411, no. 6833, pp. 17–19, 2001.

[2] B. Meyssignac and A. Cazenave, "Sea level: A review of present-day and recent-past changes and variability," *Journal of Geodynamics*, vol. 58, pp. 96–109, 2012.

[3] J. Hansen, M. Sato, and R. Ruedy, "Perception of climate change," *Proceedings of the National Academy of Sciences*, vol. 109, no. 37, pp. 2415–2423, 2012.

[4] J. B. Smith, S. H. Schneider, M. Oppenheimer, G. W. Yohe, W. Hare, M. D. Mastrandrea, A. Patwardhan, I. Burton, J. Corfee-Morlot, C. H. Magadza *et al.*, "Assessing dangerous climate change through an update of the intergovernmental panel on climate change (IPCC), reasons for concern," *Proceedings of the National Academy of Sciences*, vol. 106, no. 11, pp. 4133–4137, 2009.

[5] T. M. Lenton, H. Held, E. Kriegler, J. W. Hall, W. Lucht, S. Rahmstorf, and H. J. Schellnhuber, "Tipping elements in the earth's climate system," *Proceedings of the National Academy of Sciences*, vol. 105, no. 6, pp. 1786–1793, 2008.

[6] J. Conti and P. Holtberg, "International energy outlook 2011," U.S. Energy Information Administration, Tech. Rep. DOE/EIA–0484, September 2011. [Online]. Available: www.eia.gov/forecasts/ieo/pdf/0484(2011).pdf

[7] J. G. Olivier, J. A. Peters, and G. Janssens-Maenhout, *Trends in global CO_2 emissions 2012 report*. The Hague, The Netherlands: PBL Netherlands Environmental Assessment Agency, 2012.

[8] M. van der Hoeven, "World energy outlook 2012," International Energy Agency, Tech. Rep. ISBN 9789264124134, November 2012. [Online]. Available: http://www.iea.org/publications/freepublications/publication/WEO2011_WEB.pdf

[9] J. Lim, "Singapore energy statistics 2011," Energy Market Authority, Tech. Rep. ISSN 2251–2624, October 2011. [Online]. Available: www.ema.gov.sg/media/files/publications/SES2011.pdf

[10] W. K. Wong, "Yearbook of statistics singapore 2012," Singapore Department of Statistics, Tech. Rep. ISSN 0583–3655, July 2012. [Online]. Available: http://www.singstat.gov.sg/publications/publications_and_papers/reference/yearbook_2012/yos2012.pdf

[11] Z. Shan, S. Qin, Q. Liu, and F. Liu, "Key manufacturing technology & equipment for energy saving and emissions reduction in mechanical equipment industry," *International Journal of Precision Engineering and Manufacturing*, vol. 13, no. 7, pp. 1095–1100, 2012.

[12] C. Martin, N. E. Worrell, M. Ruth, L. Price, R. Elliott, and A. Shipley,

"Emerging energy-efficient industrial technologies," Lawrence Berkeley National Laboratory, Tech. Rep. LBNL 46990, January 2000. [Online]. Available: http://www.escholarship.org/uc/item/5jr2m969

[13] B. Xu, T. J. Slaa, and J. Sathaye, "Characterizing costs and savings benefits from a selection of energy efficient emerging technologies in the united states," Lawrence Berkeley National Laboratory, Tech. Rep. BOA–99–205–P, December 2010. [Online]. Available: http://www.escholarship.org/uc/item/3nb0863v

[14] O. Bailey and E. Worrell, "Clean energy technologies: A preliminary inventory of the potential for electricity generation," Lawrence Berkeley National Laboratory, Tech. Rep. LBNL 57451, March 2005. [Online]. Available: http://www.escholarship.org/uc/item/3418w4kv

[15] A. Hasanbeigi and L. Price, "A review of energy use and energy efficiency technologies for the textile industry," *Renewable and Sustainable Energy Reviews*, vol. 16, no. 6, pp. 3648–3665, 2012.

[16] C. K. Pang, F. L. Lewis, T. H. Lee, and Z. Y. Dong, *Intelligent Diagnosis and Prognosis of Industrial Networked Systems*. Boca Raton, FL, USA: CRC Press, Taylor and Francis Group, 2011.

[17] E. A. Abdelaziz, R. Saidur, and S. Mekhilef, "A review on energy saving strategies in industrial sector," *Renewable and Sustainable Energy Reviews*, vol. 15, no. 1, pp. 150–168, 2011.

[18] P. Sheng, M. Srinivasan, and S. Kobayashi, "Multi-objective process planning in environmentally conscious manufacturing: A feature-based approach," *CIRP Annals-Manufacturing Technology*, vol. 44, no. 1, pp. 433–437, 1995.

[19] P. Sheng, M. Srinivasan, and G. Chryssolouris, "Hierarchical part planning strategy for environmentally conscious machining," *CIRP Annals-Manufacturing Technology*, vol. 45, no. 1, pp. 455–460, 1996.

[20] M. Srinivasan and P. Sheng, "Feature-based process planning for environmentally conscious machining–part 1: Microplanning," *Robotics and Computer-Integrated Manufacturing*, vol. 15, no. 3, pp. 257–270, 1999.

[21] Y. He, F. Liu, H. Cao, and H. Zhang, "Process planning support system for green manufacturing and its application," *Frontiers of Mechanical Engineering in China*, vol. 2, no. 1, pp. 104–109, 2007.

[22] H. Xu and D. Li, "Modeling of process parameter selection with mathematical logic for process planning," *Robotics and Computer-Integrated Manufacturing*, vol. 25, no. 3, pp. 529–535, 2009.

[23] N. Weinert, S. Chiotellis, and G. Seliger, "Methodology for planning and operating energy-efficient production systems," *CIRP Annals-Manufacturing Technology*, vol. 60, no. 1, pp. 41–44, 2011.

[24] K. Fang, N. Uhan, F. Zhao, and J. Sutherland, "A new shop scheduling approach in support of sustainable manufacturing," in *Glocalized Solutions for Sustainability in Manufacturing*, J. Hesselbach and C. Herrmann, Eds. Berlin,

Heidelberg, Germany: Springer, 2011, pp. 305–310.

[25] B. Du, H. Chen, G. Huang, and H. Yang, “Preference vector ant colony system for minimising make-span and energy consumption in a hybrid flow shop,” in *Multi-objective Evolutionary Optimisation for Product Design and Manufacturing*, L. Wang, A. H. C. Ng, and K. Deb, Eds. London, UK: Springer, 2011, pp. 279–304.

[26] G. Chen, L. Zhang, J. Arinez, and S. Biller, “Energy-efficient production systems through schedule-based operations,” *IEEE Transactions on Automation Science and Engineering*, vol. 10, no. 1, pp. 27–37, January 2013.

[27] M. Mashaei and B. Lennartson, “Energy reduction in a pallet-constrained flow shop through on-off control of idle machines,” *IEEE Transactions on Automation Science and Engineering*, vol. 10, no. 1, pp. 45–56, January 2013.

[28] O. Wigstrom, B. Lennartson, A. Vergnano, and C. Breitholtz, “High-level scheduling of energy optimal trajectories,” *IEEE Transactions on Automation Science and Engineering*, vol. 10, no. 1, pp. 57–64, January 2013.

[29] M. Mori, M. Fujishima, Y. Inamasu, and Y. Oda, “A study on energy efficiency improvement for machine tools,” *CIRP Annals-Manufacturing Technology*, vol. 60, no. 1, pp. 145–148, 2011.

[30] P. Mativenga and M. Rajemi, “Calculation of optimum cutting parameters based on minimum energy footprint,” *CIRP Annals-Manufacturing Technology*, vol. 60, no. 1, pp. 149–152, 2011.

[31] R. Schlosser, F. Klocke, and D. Lung, "Sustainabilty in manufacturing energy consumption of cutting processes," in *Advances in Sustainable Manufacturing*, G. Seliger, M. M. Khraisheh, and I. Jawahir, Eds. Berlin, Heidelberg, Germany: Springer, 2011, pp. 85–89.

[32] T. Matsumura and E. Usui, "Simulation of cutting process in peripheral milling by predictive cutting force model based on minimum cutting energy," *International Journal of Machine Tools and Manufacture*, vol. 50, no. 5, pp. 467–473, 2010.

[33] A. Dietmair, A. Verl, and P. Eberspaecher, "Model-based energy consumption optimisation in manufacturing system and machine control," *International Journal of Manufacturing Research*, vol. 6, no. 4, pp. 380–401, 2011.

[34] C. Baker and K. McKenzie, "Energy consumption of industrial spray dryers," *Drying Technology*, vol. 23, no. 1–2, pp. 365–386, 2005.

[35] W.-Q. Sun, J.-J. Cai, T. Du, and D.-W. Zhang, "Specific energy consumption analysis model and its application in typical steel manufacturing process," *International Journal of Iron and Steel Research*, vol. 17, no. 10, pp. 33–37, 2010.

[36] S. Palamutcu, "Electric energy consumption in the cotton textile processing stages," *Energy*, vol. 35, no. 7, pp. 2945–2952, 2010.

[37] J. N. Lygouras, P. Botsaris, J. Vourvoulakis, and V. Kodogiannis, "Fuzzy logic controller implementation for a solar air-conditioning system," *Applied Energy*, vol. 84, no. 12, pp. 1305–1318, 2007.

[38] J. N. Lygouras, V. Kodogiannis, T. Pachidis, K. N. Tarchanidis, and C. Koukourlis, “Variable structure TITO fuzzy-logic controller implementation for a solar air-conditioning system,” *Applied Energy*, vol. 85, no. 4, pp. 190–203, 2008.

[39] A. Al-Alili, Y. Hwang, R. Radermacher, and I. Kubo, “A high efficiency solar air conditioner using concentrating photovoltaic/thermal collectors,” *Applied Energy*, vol. 93, pp. 138–147, 2012.

[40] C. Chiou, C. Chiou, C. Chu, and S. Lin, “The application of fuzzy control on energy saving for multi-unit room air-conditioners,” *Applied Thermal Engineering*, vol. 29, no. 2, pp. 310–316, 2009.

[41] G. Heidarinejad, V. Khalajzadeh, and S. Delfani, “Performance analysis of a ground-assisted direct evaporative cooling air conditioner,” *Building and Environment*, vol. 45, no. 11, pp. 2421–2429, 2010.

[42] T. Chow, Z. Lin, and X. Yang, “Placement of condensing units of split-type air-conditioners at low-rise residences,” *Applied thermal engineering*, vol. 22, no. 13, pp. 1431–1444, 2002.

[43] H. Jiang, Y. Jiang, Y. Wang, Z. Ma, and Y. Yao, “An experimental study on a modified air conditioner with a domestic hot water supply (ACDHWS),” *Energy*, vol. 31, no. 12, pp. 1789–1803, 2006.

[44] J. Heo, J. Hong, and Y. Cho, “EARQ: Energy aware routing for real-time and reliable communication in wireless industrial sensor networks,” *IEEE Transac-*

tions on Industrial Informatics, vol. 5, no. 1, pp. 3–11, 2009.

[45] L. Palopoli, R. Passerone, and T. Rizano, "Scalable offline optimization of industrial wireless sensor networks," *IEEE Transactions on Industrial Informatics*, vol. 7, no. 2, pp. 328–339, 2011.

[46] T. M. Chiwewe and G. P. Hancke, "A distributed topology control technique for low interference and energy efficiency in wireless sensor networks," *IEEE Transactions on Industrial Informatics*, vol. 8, no. 1, pp. 11–19, 2012.

[47] B. Lu, T. Habetler, R. Harley, J. Gutierrez, and D. Durocher, "Energy evaluation goes wireless," *IEEE Industry Applications Magazine*, vol. 13, no. 2, pp. 17–23, 2007.

[48] G. Mouzon, M. B. Yildirim, and J. Twomey, "Operational methods for minimization of energy consumption of manufacturing equipment," *International Journal of Production Research*, vol. 45, no. 18–19, pp. 4247–4271, 2007.

[49] G. Mouzon and M. B. Yildirim, "A framework to minimise total energy consumption and total tardiness on a single machine," *International Journal of Sustainable Engineering*, vol. 1, no. 2, pp. 105–116, 2008.

[50] M. Yildirim and G. Mouzon, "Single-machine sustainable production planning to minimize total energy consumption and total completion time using a multiple objective genetic algorithm," *IEEE Transactions on Engineering Management*, vol. 59, no. 4, pp. 585–597, 2012.

[51] R. Neugebauer, M. Wabner, H. Rentzsch, and S. Ihlenfeldt, "Structure principles of energy efficient machine tools," *CIRP Journal of Manufacturing Science and Technology*, vol. 4, no. 2, pp. 136–147, 2011.

[52] R. Neugebauer, V. Wittstock, A. Meyer, J. Glänzel, M. Pätzold, and M. Schumann, "VR tools for the development of energy-efficient products," *CIRP Journal of Manufacturing Science and Technology*, vol. 4, no. 2, pp. 208–215, 2011.

[53] B. Huang, K. Xing, K. Abhary, and S. Spuzic, "Development of energy-saving optimization for the oval-edging oval pass design using genetic algorithm," *The International Journal of Advanced Manufacturing Technology*, vol. 61, no. 5–8, pp. 423–429, 2012.

[54] L. Kroll, P. Blau, M. Wabner, U. Frieß, J. Eulitz, and M. Klärner, "Lightweight components for energy-efficient machine tools," *CIRP Journal of Manufacturing Science and Technology*, vol. 4, no. 2, pp. 148–160, 2011.

[55] R. Schmitt, J. Bittencourt, and R. Bonefeld, "Modelling machine tools for self-optimisation of energy consumption," in *Glocalized Solutions for Sustainability in Manufacturing*, J. Hesselbach and C. Herrmann, Eds. Berlin, Heidelberg, Germany: Springer, 2011, pp. 253–257.

[56] J. P. Santos, M. Oliveira, F. G. Almeida, J. P. Pereira, and A. Reis, "Improving the environmental performance of machine-tools: Influence of technology and throughput on the electrical energy consumption of a press-brake," *Journal of Cleaner Production*, vol. 19, no. 4, pp. 356–364, 2011.

[57] M.-H. Chiang, Y. Yeh, F. Yang, and Y. Chen, "Integrated control of clamping force and energy-saving in hydraulic injection moulding machines using decoupling fuzzy sliding-mode control," *The International Journal of Advanced Manufacturing Technology*, vol. 27, no. 1–2, pp. 53–62, 2005.

[58] N. Diaz, E. Redelsheimer, and D. Dornfeld, "Energy consumption characterization and reduction strategies for milling machine tool use," in *Glocalized Solutions for Sustainability in Manufacturing*, J. Hesselbach and C. Herrmann, Eds. Berlin, Heidelberg, Germany: Springer, 2011, pp. 263–267.

[59] B. Denkena, P. Hesse, and O. Gümmer, "Energy optimized jerk-decoupling technology for translatory feed axes," *CIRP Annals-Manufacturing Technology*, vol. 58, no. 1, pp. 339–342, 2009.

[60] S. Shinnaka and T. Sagawa, "New optimal current control methods for energy-efficient and wide speed-range operation of hybrid-field synchronous motor," *IEEE Transactions on Industrial Electronics*, vol. 54, no. 5, pp. 2443–2450, 2007.

[61] X. Wang, H. Zhong, Y. Yang, and X. Mu, "Study of a novel energy efficient single-phase induction motor with three series-connected windings and two capacitors," *IEEE Transactions on Energy Conversion*, vol. 25, no. 2, pp. 433–440, 2010.

[62] A. Boglietti, A. Cavagnino, L. Feraris, and M. Lazzari, "Energy-efficient motors," *IEEE Industrial Electronics Magazine*, vol. 2, no. 4, pp. 32–37, 2008.

[63] M. Melfi, S. Evon, and R. McElveen, "Induction versus permanent magnet motors," *IEEE Industry Applications Magazine*, vol. 15, no. 6, pp. 28–35, 2009.

[64] C. V. Le, C. K. Pang, and O. P. Gan, "Energy saving and monitoring technologies in manufacturing systems with industrial case studies," in *Proceedings of the 7^{th} IEEE Conference on Industrial Electronics and Applications*, Singapore, July 18–20 2012, pp. 1916–1921.

[65] D. Ouelhadj and S. Petrovic, "A survey of dynamic scheduling in manufacturing systems," *Journal of Scheduling*, vol. 12, no. 4, pp. 417–431, 2009.

[66] S. Kwon and M.-Y. Yang, "The benefits of using instantaneous energy to monitor the transient state of the wire EDM process," *The International Journal of Advanced Manufacturing Technology*, vol. 27, no. 9–10, pp. 930–938, 2006.

[67] Y. Ren, B. Zhang, and Z. Zhou, "Specific energy in grinding of tungsten carbides of various grain sizes," *CIRP Annals-Manufacturing Technology*, vol. 58, no. 1, pp. 299–302, 2009.

[68] C. V. Le, C. K. Pang, O. P. Gan, X. M. Chee, D. H. Zhang, M. Luo, H. L. Chan, and F. L. Lewis, "Classification of energy consumption patterns for energy audit and machine scheduling in industrial manufacturing systems," *Transactions of the Institute of Measurement and Control*, vol. 35, no. 5, pp. 583–592, 2013.

[69] J. A. Buzacott and D. D. Yao, "Flexible manufacturing systems: A review of analytical models," *Management science*, vol. 32, no. 7, pp. 890–905, 1986.

[70] S. Bogdan, Z. Kovacic, F. L. Lewis, and J. Mireles, *Manufacturing systems control design: A matrix-based approach.* Berlin, Heidelberg, Germany: Springer, 2006.

[71] J. Wang, *Handbook of Finite State Based Models and Applications.* Boca Raton, FL, USA: CRC Press, Taylor and Francis Group, 2012.

[72] M. C. Zhou and M. P. Fanti, *Deadlock Resolution in Computer Integrated Systems.* New York, US: Marcel Dekker, 2005.

[73] C. Jung and T.-E. Lee, "An efficient mixed integer programming model based on timed Petri nets for diverse complex cluster tool scheduling problems," *IEEE Transactions on Semiconductor Manufacturing*, vol. 25, no. 2, pp. 186–199, 2012.

[74] C. V. Le, C. K. Pang, F. Lewis, O. P. Gan, and H. L. Chan, "Intelligent dynamic resource assignment for energy-efficiency in industrial stamping machines," in *Proceedings of 37th Annual Conference on IEEE Industrial Electronics Society*, Melbourne, Australia, November 7–10 2011, pp. 4131–4136.

[75] C. K. Pang, G. R. Hudas, D. G. Mikulski, C. V. Le, and F. L. Lewis, "Discrete event command and control of asymmetric large-scale armed forces using network centric warfare," *Unmanned Systems*, submitted.

[76] B. Huang, X.-X. Shi, and N. Xu, "Scheduling FMS with alternative routings using Petri nets and near admissible heuristic search," *The International Journal of Advanced Manufacturing Technology*, vol. 63, no. 9–12, pp. 1131–1136,

2012.

[77] T. Murata, "Petri nets: Properties, analysis and applications," *Proceedings of the IEEE*, vol. 77, no. 4, pp. 541–580, 1989.

[78] D. A. Tacconi and F. L. Lewis, "A new matrix model for discrete event systems: Application to simulation," *IEEE Control Systems Magazine*, vol. 17, no. 5, pp. 62–71, 1997.

[79] A. Bruzzone, D. Anghinolfi, M. Paolucci, and F. Tonelli, "Energy-aware scheduling for improving manufacturing process sustainability: A mathematical model for flexible flow shops," *CIRP Annals-Manufacturing Technology*, vol. 61, no. 1, pp. 459–462, 2012.

[80] Y. He, F. Liu, and J. Shi, "A framework of scheduling models in machining workshop for green manufacturing," *Journal of Advanced Manufacturing Systems*, vol. 7, no. 02, pp. 319–322, 2008.

[81] S. Kara and W. Li, "Unit process energy consumption models for material removal processes," *CIRP Annals-Manufacturing Technology*, vol. 60, no. 1, pp. 37–40, 2011.

[82] A. Vijayaraghavan and D. Dornfeld, "Automated energy monitoring of machine tools," *CIRP Annals-Manufacturing Technology*, vol. 59, no. 1, pp. 21–24, 2010.

[83] E. Endsley, E. Almeida, and D. M. Tilbury, "Modular finite state machines: Development and application to reconfigurable manufacturing cell controller

generation," *Control Engineering Practice*, vol. 14, no. 10, pp. 1127–1142, 2006.

[84] P. Solding, P. Thollander, and P. R. Moore, "Improved energy-efficient production using discrete event simulation," *Journal of Simulation*, vol. 3, no. 4, pp. 191–201, 2009.

[85] C. Burrus, R. Gopinath, and H. Guo, *Introduction to Wavelets and Wavelet Transforms, A Primer.* Upper Saddle River, NJ, USA: Prentice Hall, 1998.

[86] I. M. Johnstone and B. W. Silverman, "Empirical bayes selection of wavelet thresholds," *Annals of Statistics*, vol. 33, no. 4, pp. 1700–1752, 2005.

[87] A. M. Altaher and M. T. Ismail, "A comparison of some thresholding selection methods for wavelet regression," *World Academy of Science, Engineering and Technology*, vol. 62, no. 1, pp. 119–125, 2010.

[88] P. Fryzlewicz, "Bivariate hard thresholding in wavelet function estimation," *Statistica Sinica*, vol. 17, no. 4, pp. 1457–1481, 2007.

[89] A. Neumaier and T. Schneider, "Estimation of parameters and eigenmodes of multivariate autoregressive models," *ACM Transactions on Mathematical Software*, vol. 27, no. 1, pp. 27–57, 2001.

[90] T. Schneider and A. Neumaier, "Algorithm 808: Arfita matlab package for the estimation of parameters and eigenmodes of multivariate autoregressive models," *ACM Transactions on Mathematical Software*, vol. 27, no. 1, pp. 58–65, 2001.

[91] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," *Neural computation*, vol. 13, no. 7, pp. 1443–1471, 2001.

[92] I. Steinwart and A. Christmann, *Support Vector Machines (Information Science and Statistics).* Berlin, Heidelberg, Germany: Springer, 2008.

[93] L. M. Manevitz and M. Yousef, "One-class svms for document classification," *The Journal of Machine Learning Research*, vol. 2, pp. 139–154, 2002.

[94] J. Brydson, *Handbook for Plastics Processors.* Oxford, UK: Heinemann Newnes, 1990.

[95] A. Thiriez and T. Gutowski, "Energy saving and monitoring technologies in manufacturing systems with industrial case studies," in *Proceedings of the 2006 IEEE International Symposium on Electronics and the Environment*, Scottsdale, AZ, USA, May 8–11 2006, pp. 195–200.

[96] S. Kalpakjian and S. Schmid, *Manufacturing Engineering and Technology.* Upper Saddle River, NJ, USA: Pearson Prentice Hall, 2006.

[97] X. M. Chee, C. V. Le, D. Zhang, M. Luo, and C. K. Pang, "Intelligent identification of manufacturing operations using in-situ energy measurement in industrial injection moulding machines," in *Proceedings of 37th Annual Conference on IEEE Industrial Electronics Society*, Melbourne, Australia, November 7–10 2011, pp. 4284–4289.

[98] H. Parsaei, T. Hanley, and S. Kolli, *Manufacturing Decision Support Systems.* Berlin, Heidelberg, Germany: Springer, 1996.

[99] Z. Michalewicz, M. Schmidt, M. Michalewicz, and C. Chiriac, "Case study: An intelligent decision support system," *IEEE Intelligent Systems*, vol. 20, no. 4, pp. 44–49, 2005.

[100] S. Liu, R. I. Young, and L. Ding, "An integrated decision support system for global manufacturing co-ordination in the automotive industry," *International Journal of Computer Integrated Manufacturing*, vol. 24, no. 4, pp. 285–301, 2011.

[101] S. B. Eom, S. M. Lee, E. Kim, and C. Somarajan, "A survey of decision support system applications (1988–1994)," *Journal of the Operational Research Society*, pp. 109–120, 1998.

[102] S.-Y. Chou and Y.-H. Chang, "A decision support system for supplier selection based on a strategy-aligned fuzzy SMART approach," *Expert systems with applications*, vol. 34, no. 4, pp. 2241–2253, 2008.

[103] Y. Kristianto, A. Gunasekaran, P. Helo, and M. Sandhu, "A decision support system for integrating manufacturing and product design into the reconfiguration of the supply chain networks," *Decision Support Systems*, vol. 52, no. 4, pp. 790–801, 2012.

[104] M. T. Tabucanon, D. N. Batanov, and D. K. Verma, "Decision support system for multicriteria machine selection for flexible manufacturing systems," *Com-*

puters in industry, vol. 25, no. 2, pp. 131–143, 1994.

[105] D. Bradley, D. Dawson, , and N. Burd, *Mechatronics: Electronics in Products and Processes*. London, UK: Chapman & Hall, 1996.

[106] C. De Silva, *Control Sensors and Actuators*. Upper Saddle River, NJ, USA: Prentice Hall, 1989.

[107] S. Choi and K. Wang, "Flexible flow shop scheduling with stochastic processing times: A decomposition-based approach," *Computers & Industrial Engineering*, vol. 63, no. 2, pp. 362–373, 2012.

[108] G. E. Vieira, J. W. Herrmann, and E. Lin, "Rescheduling manufacturing systems: A framework of strategies, policies, and methods," *Journal of scheduling*, vol. 6, no. 1, pp. 39–62, 2003.

[109] H.-H. Ko, J. Kim, S.-S. Kim, and J.-G. Baek, "Dispatching rule for non-identical parallel machines with sequence-dependent setups and quality restrictions," *Computers & Industrial Engineering*, vol. 59, no. 3, pp. 448–457, 2010.

[110] W. Mouelhi-Chibani and H. Pierreval, "Training a neural network to select dispatching rules in real time," *Computers & Industrial Engineering*, vol. 58, no. 2, pp. 249–256, 2010.

[111] D. R. Sule, *Production Planning and Industrial Scheduling: Examples, Case Studies and Applications, Second Edition*. Boca Raton, FL, USA: CRC Press, Taylor and Francis Group, 2007.

[112] J. C. Tay and N. B. Ho, "Evolving dispatching rules using genetic programming for solving multi-objective flexible job-shop problems," *Computers & Industrial Engineering*, vol. 54, no. 3, pp. 453–473, 2008.

[113] S.-C. Horng, S.-S. Lin, and F.-Y. Yang, "Evolutionary algorithm for stochastic job shop scheduling with random processing time," *Expert Systems with Applications*, vol. 39, no. 3, pp. 3603–3610, 2012.

[114] K.-T. Fang and B. M. Lin, "Parallel-machine scheduling to minimize tardiness penalty and power cost," *Computers & Industrial Engineering*, vol. 64, no. 1, pp. 224–234, 2013.

[115] G. K. Bachman, L. Narici, and E. Beckenstein, *Fourier and wavelet analysis.* Berlin, Heidelberg, Germany: Springer, 2000.

[116] A. Prakash, N. Khilwani, M. Tiwari, and Y. Cohen, "Modified immune algorithm for job selection and operation allocation problem in flexible manufacturing systems," *Advances in Engineering Software*, vol. 39, no. 3, pp. 219–232, 2008.

[117] I. Christou, *Quantitative Methods in Supply Chain Managements.* Berlin, Heidelberg, Germany: Springer, 2012.

[118] C. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity.* Mineola, NY, USA: Dover Publications, 1998.

[119] M. Pinedo, *Planning and Scheduling in Manufacturing and Services.* Berlin,

Heidelberg, Germany: Springer, 2005.

[120] D. Armbruster and G. Karl, *Decision Policies for Production Networks.* Berlin, Heidelberg, Germany: Springer, 2012.

[121] I. Sindicic, S. Bogdan, and T. Petrovic, "Resource allocation in free-choice multiple reentrant manufacturing systems based on machine-job incidence matrix," *IEEE Transactions on Industrial Informatics*, vol. 7, no. 1, pp. 105–114, 2011.

[122] P. Leitao and F. J. Restivo, "Implementation of a holonic control system in a flexible manufacturing system," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 38, no. 5, pp. 699–709, 2008.

[123] H. Hu, M. Zhou, and Z. Li, "Supervisor optimization for deadlock resolution in automated manufacturing systems with Petri nets," *IEEE Transactions on Automation Science and Engineering*, vol. 8, no. 4, pp. 794–804, 2011.

[124] K. Xing, L. Han, M. Zhou, and F. Wang, "Deadlock-free genetic scheduling algorithm for automated manufacturing systems based on deadlock control policy," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 42, no. 3, pp. 603–615, 2012.

[125] H. R. Golmakani, J. K. Mills, and B. Benhabib, "Deadlock-free scheduling and control of flexible manufacturing cells using automata theory," *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, vol. 36, no. 2, pp. 327–337, 2006.

[126] Z. Wu and M. X. Weng, "Multiagent scheduling method with earliness and tardiness objectives in flexible job shops," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 35, no. 2, pp. 293–301, 2005.

[127] X. Cai, X. Wu, and X. Zhou, "Dynamically optimal policies for stochastic scheduling subject to preemptive-repeat machine breakdowns," *IEEE Transactions on Automation Science and Engineering*, vol. 2, no. 2, pp. 158–172, 2005.

[128] M. Tawarmalani and N. V. Sahinidis, *Convexification and Global Optimization in Continuous and Mixed-Integer Nonlinear Programming: Theory, Algorithms, Software, and Applications.* Berlin, Heidelberg, Germany: Springer, 2002.

[129] S. Lee and I. E. Grossmann, "A global optimization algorithm for nonconvex generalized disjunctive programming and applications to process systems," *Computers & Chemical Engineering*, vol. 25, no. 11, pp. 1675–1697, 2001.

[130] P. Belotti, J. Lee, L. Liberti, F. Margot, and A. Wächter, "Branching and bounds tightening techniques for non-convex MINLP," *Optimization Methods & Software*, vol. 24, no. 4–5, pp. 597–634, 2009.

[131] O. Exler, L. T. Antelo, J. A. Egea, A. A. Alonso, and J. R. Banga, "A tabu search-based algorithm for mixed-integer nonlinear problems and its application to integrated process and control system design," *Computers & Chemical Engineering*, vol. 32, no. 8, pp. 1877–1891, 2008.

[132] L. Yiqing, Y. Xigang, and L. Yongjian, "An improved PSO algorithm for solving non-convex NLP/MINLP problems with equality constraints," *Computers &*

chemical engineering, vol. 31, no. 3, pp. 153–162, 2007.

[133] C.-T. Young, Y. Zheng, C.-W. Yeh, and S.-S. Jang, "Information-guided genetic algorithm approach to the solution of MINLP problems," *Industrial & engineering chemistry research*, vol. 46, no. 5, pp. 1527–1537, 2007.

[134] M. Schlüter, J. A. Egea, and J. R. Banga, "Extended ant colony optimization for non-convex mixed integer nonlinear programming," *Computers & Operations Research*, vol. 36, no. 7, pp. 2217–2229, 2009.

[135] L. Liberti, N. Mladenović, and G. Nannicini, "A recipe for finding good solutions to MINLPs," *Mathematical Programming Computation*, vol. 3, no. 4, pp. 349–390, 2011.

[136] T. Berthold, "RENS: The optimal rounding," DFG Research Center Matheon, Tech. Rep., April 2012. [Online]. Available: http://nbn-resolving.de/urn/resolver.pl?urn:nbn:de:0296-matheon-11311

[137] F. L. Lewis, B. G. Horne, and C. T. Abdallah, "Computational complexity of determining resource loops in re-entrant flow lines," *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, vol. 30, no. 2, pp. 222–229, 2000.

[138] G. Halevi, *Handbook of Production Management Methods.* Oxford, UK: Butterworth-Heinemann, 2011.

[139] H. R. Yazgan, I. Beypinar, S. Boran, and C. Ocak, "A new algorithm and

multi-response taguchi method to solve line balancing problem in an automotive industry," *The International Journal of Advanced Manufacturing Technology*, vol. 57, no. 1–4, pp. 379–392, 2011.

[140] N. Pekin and M. Azizoglu, "Bi criteria flexible assembly line design problem with equipment decisions," *International Journal of Production Research*, vol. 46, no. 22, pp. 6323–6343, 2008.

[141] R. Bellman, *Dynamic Programming.* Princeton, NJ, US: Princeton University Press, 1957.

[142] P. Abdulla and A. Nylén, "Timed Petri nets and BQOs," in *Applications and Theory of Petri Nets 2001*, ser. Lecture Notes in Computer Science, J.-M. Colom and M. Koutny, Eds. Berlin, Heidelberg, Germany: Springer, 2001, vol. 2075, pp. 53–70.

[143] F. Cassez and O. H. Roux, "Structural translation from time Petri nets to timed automata," *Journal of Systems and Software*, vol. 79, no. 10, pp. 1456–1468, 2006.

[144] H. K. Alfares, "Efficient optimization of cyclic labor days-off scheduling," *OR-Spektrum*, vol. 23, no. 2, pp. 283–294, 2001.

[145] J. E. Beasley, M. Krishnamoorthy, Y. M. Sharaiha, and D. Abramson, "Scheduling aircraft landings–the static case," *Transportation science*, vol. 34, no. 2, pp. 180–197, 2000.

[146] S.-S. Leu, A.-T. Chen, and C.-H. Yang, "A GA-based fuzzy optimal model for construction time–cost trade-off," *International Journal of Project Management*, vol. 19, no. 1, pp. 47–58, 2001.

[147] S. Sharma and D. Gupta, "Minimizing rental cost under specified rental policy in two stage flow shop, the processing time associated with probabilities including break-down interval and job block criteria," *European Journal of Business and Management*, vol. 3, no. 2, pp. 85–103, 2011.

[148] P. Cowling and M. Johansson, "Using real time information for effective dynamic scheduling," *European Journal of Operational Research*, vol. 139, no. 2, pp. 230–244, 2002.

[149] Y. Nikulin and Z. Iftikhar, "A note on different modelling approaches for the robust shortest path problem," *International Journal of Mathematical Modelling and Numerical Optimisation*, vol. 3, no. 3, pp. 141–157, 2012.

[150] M. E. Bruni and F. Guerriero, "An enhanced exact procedure for the absolute robust shortest path problem," *International Transactions in Operational Research*, vol. 17, no. 2, pp. 207–220, 2010.

[151] V. Gabrel, C. Murat, and L. Wu, "New models for the robust shortest path problem: Complexity, resolution and generalization," *Annals of Operations Research*, pp. 1–24, 2011.

[152] R. Kalai, C. Lamboray, and D. Vanderpooten, "Lexicographic α-robustness: An alternative to minmax criteria," *European Journal of Operational Research*,

vol. 220, no. 3, pp. 722–728, 2012.

[153] R. Montemanni, L. M. Gambardella, and A. V. Donati, "A branch and bound algorithm for the robust shortest path problem with interval data," *Operations Research Letters*, vol. 32, no. 3, pp. 225–232, 2004.

[154] A. Kasperski and P. ZielińSki, "The robust shortest path problem in series–parallel multidigraphs with interval data," *Operations Research Letters*, vol. 34, no. 1, pp. 69–76, 2006.

[155] D. Catanzaro, M. Labbé, and M. Salazar-Neumann, "Reduction approaches for robust shortest path problems," *Computers & operations research*, vol. 38, no. 11, pp. 1610–1619, 2011.

[156] S. Sen, R. Pillai, S. Joshi, and A. K. Rathi, "A mean-variance model for route guidance in advanced traveler information systems," *Transportation Science*, vol. 35, no. 1, pp. 37–49, 2001.

[157] S. D. Boyles and S. T. Waller, "A mean-variance model for the minimum cost flow problem with stochastic arc costs," *Networks*, vol. 56, no. 3, pp. 215–227, 2010.

[158] T. Hasuike, "Robust shortest path problem based on a confidence interval in fuzzy bicriteria decision making," *Information Sciences*, vol. 221, pp. 520–533, 2013.

[159] B. Chen, W. H. Lam, A. Sumalee, Q. Li, H. Shao, and Z. Fang, "Finding reli-

able shortest paths in road networks under uncertainty," *Networks and Spatial Economics*, vol. 13, no. 2, pp. 123–148, 2013.

[160] J. C. Principe, *Information Theoretic Learning: Rényi's Entropy and Kernel Perspectives*. Berlin, Heidelberg, Germany: Springer, 2010.

[161] R. J. Bessa, V. Miranda, and J. Gama, "Entropy and correntropy against minimum square error in offline and online three-day ahead wind power forecasting," *IEEE Transactions on Power Systems*, vol. 24, no. 4, pp. 1657–1666, 2009.

[162] G. Tsaggouris and C. Zaroliagis, "Non-additive shortest paths," in *Algorithms ESA 2004*, ser. Lecture Notes in Computer Science, S. Albers and T. Radzik, Eds. Berlin, Heidelberg, Germany: Springer, 2004, vol. 3221, pp. 822–834.

[163] L. B. Reinhardt and D. Pisinger, "Multi-objective and multi-constrained non-additive shortest path problems," *Computers & Operations Research*, vol. 38, no. 3, pp. 605–616, 2011.

[164] X. Huang, "An entropy method for diversified fuzzy portfolio selection," *International Journal of Fuzzy Systems*, vol. 14, no. 1, pp. 160–165, 2012.

[165] ——, "Mean-entropy models for fuzzy portfolio selection," *Fuzzy Systems, IEEE Transactions on*, vol. 16, no. 4, pp. 1096–1101, 2008.

[166] G. C. Philippatos and N. Gressis, "Conditions of equivalence among ev, ssd, and eh portfolio selection criteria: the case for uniform, normal and lognormal distributions," *Management Science*, vol. 21, no. 6, pp. 617–625, 1975.

[167] M. C. Steinbach, "Markowitz revisited: Mean-variance models in financial portfolio analysis," *SIAM Review*, vol. 43, no. 1, pp. 31–85, 2001.

[168] C. Tian, H. Jiang, C. Wang, Z. Wu, J. Chen, and W. Liu, "Neither shortest path nor dominating set: Aggregation scheduling by greedy growing tree in multihop wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 7, pp. 3462–3472, 2011.

[169] H. Liu, Z. Jiang, and R. Y. Fung, "Modeling of large-scale complex re-entrant manufacturing systems by extended object-oriented Petri nets," *The International Journal of Advanced Manufacturing Technology*, vol. 27, no. 1–2, pp. 190–204, 2005.

[170] R. Rajan, S. Engelen, M. Bentum, and C. Verhoeven, "Orbiting low frequency array for radio astronomy," in *Proceedings of the 2011 IEEE Aerospace Conference*, Big Sky, MT, USA, March 05–12 2011, pp. 1–11.

[171] L. V. Datta and U. Guven, *Introduction to Nanosatellite Technology and Components: Applications of Cubesat Technology.* Saarbrucken, Saarland, Germany: Lambert Academic Publishing, 2011.

[172] W. Xu, B. Liang, and Y. Xu, "Survey of modeling, planning, and ground verification of space robotic systems," *Acta Astronautica*, vol. 68, no. 11, pp. 1629–1649, 2011.

List of Publications

- International refereed journal papers

    1. C. V. Le, C. K. Pang, A. Kumar, C. H. Goh, "Nano-Satellite swarm for SAR applications: Design and robust scheduling" *IEEE Transactions on Aerospace and Electronic Systems*, submitted.
    2. C. V. Le and C. K. Pang, "Mean-entropy criterion for robust energy optimization of flexible manufacturing system," *IEEE Transactions on Automation Science and Engineering*, submitted.
    3. C. K. Pang, G. R. Hudas, D. G. Mikulski, C. V. Le, F. L. Lewis, "Discrete Event Command and Control of Asymmetric Large-Scale Armed Forces Using Network Centric Warfare Unmanned Systems," *Unmanned Systems*, conditionally accepted.
    4. C. K. Pang and C. V. Le, "Optimization of total energy consumption in flexible manufacturing systems using weighted p-timed Petri nets and dynamic programming," *IEEE Transactions on Automation Science and Engineering*, in press.

5. C. V. Le and C. K. Pang, "Fast reactive scheduling to minimize tardiness penalty and energy cost under power consumption uncertainty," *Computer & Industrial Engineering*, vol. 66, no. 2, pp. 406–417, October 2013.
6. C. V. Le and C. K. Pang, "An energy data-driven decision support system for high-performance manufacturing industries," *International Journal of Automation and Logistics*, vol. 1, no. 1, pp. 61–79, 2013.
7. C. K. Pang, G. R. Hudas, M. B. Middleton, C. V. Le, and F. L. Lewis, "Discrete event command and control for networked teams with multiple military missions," *Journal of Defense Modeling and Simulation*, in press.
8. C. V. Le, C. K. Pang, O. P. Gan, X. M. Chee, D. -H. Zhang, M. Luo, H. L. Chan, and F. L. Lewis, "Classification of energy consumption patterns for energy audit and machine scheduling in industrial manufacturing systems," *Transactions of the Institute of Measurement and Control*, vol. 35, no. 5, pp. 583–592, July 2013.

- Invited session papers in international refereed conference proceedings

1. C. K. Pang and C. V. Le, "Renyi mean-entropy criterion for robust energy optimization of industrial stamping systems," to appear in *Proceedings of the 2014 IEEE International Conference on Control & Automation*, Taichung, Taiwan, June 18–20, 2014.
2. C. K. Pang and C. V. Le, "An energy data-driven decision support sys-

tem for high performance in industrial injection moulding and stamping systems," to appear in *Proceedings of the 2014 IEEE International Conference on Control & Automation*, Taichung, Taiwan, June 18–20, 2014.

3. C. K. Pang and C. V. Le, "Integrated control and reactive scheduling for FMS under power consumption uncertainty," in *Proceedings of the 2013 IEEE IECON*, SS49–1, pp. 7523–7528, Vienna, Austria, November 10–13, 2013 (invited).

4. C. K. Pang and C. V. Le, "Non-convex large-scale scheduling for energy-efficient flexible stamping systems," in *Proceedings of the 2013 IEEE ICCA*, FrB3.5, pp. 1656–1661, Hangzhou, China, June 12–14, 2013.

5. X. M. Chee, C. V. Le, D. -H. Zhang, M. Luo, and C. K. Pang, "Intelligent identification of manufacturing operations using in-situ energy measurement in industrial injection moulding machines," in *Proceedings of the 2011 IEEE IECON*, pp. 4137–4142, Melbourne, Australia, November 7–10, 2011.

6. C. V. Le, C. K. Pang, F. L. Lewis, O. P. Gan, and H. L. Chan, "Intelligent dynamic resource assignment for energy-efficiency in industrial stamping machines," in *Proceedings of the 2011 IEEE IECON*, pp. 4131–4136, Melbourne, Australia, November 7–10, 2011.

7. C. K. Pang, C. V. Le, O. P. Gan, X. M. Chee, D. -H. Zhang, M. Luo, H. L. Chan, and F. L. Lewis, "Intelligent energy audit and machine management

for energy-efficient manufacturing," in *Proceedings of the 2011 IEEE CIS*, SuD5.2, pp. 142–147, Qingdao, China, September 17–19, 2011.

- Regular session papers in international refereed conference proceedings

    1. C. K. Pang, C. V. Le, T. S. Ng, and H. L. N. Nguyen, "Systems model analysis for iterative concurrent design processes and its application to design of precision mechatronics," in *Proceedings of the International Symposium on 2013 3CA*, pp. 300–304, Singapore, December 1–2, 2013.

    2. C. K. Pang, C. V. Le, T. S. Ng, and H. L. N. Nguyen, "Systems model analysis for iterative concurrent design processes and its application to design of precision mechatronics," to appear in *Proceedings of the 2013 3CA*, Singapore, December 1–2, 2013.

    3. C. K. Pang, G. R. Hudas, C. V. Le, M. B. Middleton, O. P. Gan, and F. L. Lewis, "Discrete event command and control of multiple military missions in network centric warfare," in *Proceedings of the 2012 ICIUS*, pp. 74–79, Singapore, October 22–24, 2012.

    4. C. V. Le, O. P. Gan, and C. K. Pang, "Energy saving and monitoring technologies in manufacturing systems with industrial case studies," in *Proceedings of the 2012 IEEE ICIEA*, FrP2.2, pp. 1913–1918, Singapore, July 18–20, 2012.