

นอร์มกรณีสถิตของระบบเชิงเส้นที่มีความไม่แน่นอน  
ภายใต้สัญญาณเข้าที่มีขอบเขตของขนาดและขีดจำกัดของอัตราการเปลี่ยนแปลง



นายวาทัญญู คล้ายสงคราม

# สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรดุษฎีบัณฑิต

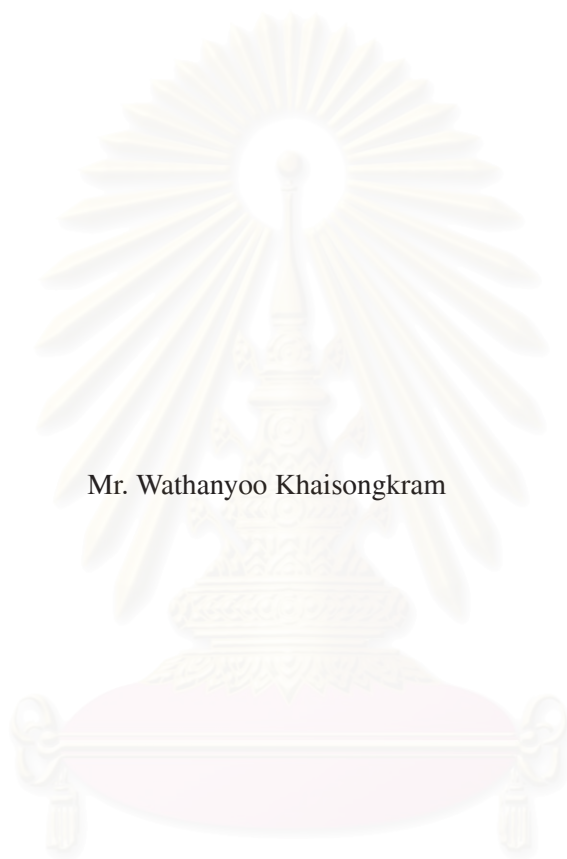
สาขาวิชาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้า

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2549

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

THE WORST-CASE NORM OF UNCERTAIN LINEAR SYSTEMS  
UNDER INPUTS WITH MAGNITUDE BOUND AND RATE LIMIT



Mr. Wathanyoo Khaisongkram

สถาบันวิทยบริการ

A Dissertation Submitted in Partial Fulfillment of the Requirements  
for the Degree of Doctor of Philosophy Program in Electrical Engineering

Department of Electrical Engineering

Faculty of Engineering

Chulalongkorn University

Academic Year 2006

Copyright of Chulalongkorn University

Thesis Title THE WORST-CASE NORM OF UNCERTAIN LINEAR SYSTEMS  
UNDER INPUTS WITH MAGNITUDE BOUND AND RATE LIMIT


By Mr. Wathanyoo Khaisongkram

Field of Study Electrical Engineering

Thesis Advisor Associate Professor David Banjerdpongchai, Ph.D.

---

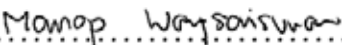
Accepted by the Faculty of Engineering, Chulalongkorn University in Partial Fulfillment  
of Requirements for the Doctoral Degree


  
..... Dean of the Faculty of Engineering  
(Professor Direk Lavansiri, Ph.D.)

THESIS COMMITTEE

  
..... Chairman  
(Associate Professor Ekachai Leelarasmee, Ph.D.)

  
..... Thesis Advisor  
(Associate Professor David Banjerdpongchai, Ph.D.)

  
..... Member  
(Assistant Professor Manop Wongsaisuwan, D.Eng.)


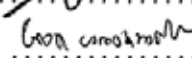
  
..... Member  
(Associate Professor Waree Kongprawechnon, Ph.D.)

  
..... Member  
(Assistant Professor Wicharn Lewkeeratiyutkul, Ph.D.)

วาทัญญู คล้ายสงคราม: นอร์มกรณีเลวสุดของระบบเชิงเส้นที่มีความไม่แน่นอนภายใต้สัญญาณเข้าที่มีขอบเขตของขนาดและขีดจำกัดของอัตราการเปลี่ยนแปลง (THE WORST-CASE NORM OF UNCERTAIN LINEAR SYSTEMS UNDER INPUTS WITH MAGNITUDE BOUND AND RATE LIMIT), อ. ที่ปรึกษา: รศ.ดร.เดวิด บรรเจิดพงศ์ชัย, 141 หน้า

หลายปีที่ผ่านมา นักวิจัยทางวิศวกรรมควบคุมใช้นอร์มกรณีเลวสุดกันอย่างกว้างขวาง ให้เป็นบรรทัดฐานเพื่อวัดผลกระทบเลวสุดที่อาจเกิดขึ้นจากการรบกวนจำเพาะ อีกทั้งยังบ่งบอกได้ว่าตัวควบคุมที่เลือกนั้นสามารถจัดการผลที่ไม่พึงปรารถนาได้ดีเพียงใด ในวิทยานิพนธ์นี้เราพิจารณานอร์มกรณีเลวสุดของระบบเชิงเส้นไม่แปรตามเวลาที่มีความไม่แน่นอน เมื่อสัญญาณรบกวนมีรูปแบบเป็นสัญญาณเข้าที่มีขอบเขตของขนาดและขีดจำกัดของอัตราการเปลี่ยนแปลง ส่วนแรกของวิทยานิพนธ์นี้ครอบคลุมการวิเคราะห์และคำนวณนอร์มกรณีเลวสุดของระบบเชิงเส้น ในกรณีที่ไม่มีค่าความไม่แน่นอน การคำนวณนอร์มกรณีเลวสุดนี้มีรูปแบบเป็นปัญหาการควบคุมที่เหมาะสมที่สุดซึ่งให้ผลเฉลยที่อธิบายลักษณะของสัญญาณเข้าเลวสุด เราพัฒนาขั้นตอนวิธีใหม่ที่มีชื่อว่า Successive Pang Interval Search เพื่อสร้างสัญญาณเข้าเลวสุดและคำนวณนอร์มกรณีเลวสุดตามลำดับ ส่วนที่สองของวิทยานิพนธ์นี้กล่าวถึงการคำนวณนอร์มกรณีเลวสุดเมื่อระบบมีความไม่แน่นอน การวิเคราะห์และการคำนวณในกรณีที่ระบบมีความไม่แน่นอนเป็นพื้นฐานสำคัญสำหรับการวิเคราะห์และการคำนวณในกรณีที่ระบบมีความไม่แน่นอน เมื่อประยุกต์แนวทางการทำให้เป็นเวลาวิยุต การคำนวณนอร์มกรณีเลวสุดนี้มีรูปแบบเป็นปัญหาการหาค่ามากที่สุดของฟังก์ชันคอนเวกซ์ซึ่งนับเป็นปัญหาเอ็นพีแบบยาก เราจึงแนะนำค่าขอบเขตบนและค่าขอบเขตล่างใหม่ของนอร์มกรณีเลวสุด ด้วยการหาผลเฉลยของปัญหาโปรแกรมเชิงเส้นแบบมากเลขศูนย์ ในการคำนวณนอร์มดังกล่าวอย่างแม่นยำขั้นตอนวิธีการแตกกิ่งและวางขอบเขตเชิงลำดับชั้นได้รับการพัฒนาขึ้นมาใหม่ โดยใช้เทคนิคการแตกกิ่งและวางขอบเขตแบบมาตรฐานร่วมกับขั้นตอนที่เรียกว่า Reduction of Ambiguity Magnitude Threshold ขั้นตอนวิธีใหม่นี้ผ่านการตรวจสอบความสมเหตุสมผล และเปรียบเทียบผลเชิงตัวเลข กับการค้นหาทั่วทั้งหมดและขั้นตอนวิธีแตกกิ่งและวางขอบเขตแบบมาตรฐาน ปรากฏว่าวิธีการแตกกิ่งและวางขอบเขตเชิงลำดับชั้นให้ผลลัพธ์ที่ถูกต้อง อีกทั้งมีความเร็วในการคำนวณสูงมากเมื่อเทียบกับวิธีที่มีอยู่ ฉะนั้นวิธีที่นำเสนอสามารถใช้งานได้จริงในกรณีที่ปัญหามีมิติสูงมากๆ นอกจากนี้เราแสดงประโยชน์ของขั้นตอนวิธีใหม่โดยประยุกต์กับการปรับจูนตัวควบคุมพีไอดีแบบหลายวัตถุประสงค์ สำหรับระบบรองรับแบบไวงานภายใต้ผู้นำหนักบรรทุกที่แปรผันได้และมีการรบกวนจากพื้นถนน การออกแบบตัวควบคุมพีไอดีนี้ใช้เวลาตามสมควร และการจำลองระบบยังแสดงให้เห็นอย่างชัดเจนว่าผลที่ได้สอดคล้องกับเงื่อนไขการออกแบบทุกประการ

ภาควิชา ..... วิศวกรรมไฟฟ้า .....  
สาขาวิชา ..... วิศวกรรมไฟฟ้า .....  
ปีการศึกษา ..... 2549 .....

ลายมือชื่อนิสิต .....  .....  
ลายมือชื่ออาจารย์ที่ปรึกษา .....  .....

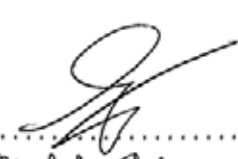
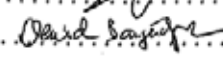
##4671822921: MAJOR ELECTRICAL ENGINEERING

KEY WORD: PERFORMANCE ANALYSIS / DISTURBANCE REJECTION / WORST-CASE NORM / UNCERTAINTY / LINEAR TIME-INVARIANT SYSTEMS / MAGNITUDE BOUND / RATE LIMIT / OPTIMAL CONTROL / LINEAR PROGRAMMING / BRANCH-AND-BOUND ALGORITHM / SUSPENSION SYSTEM / PID TUNING

WATHANYOO KHAISONGKRAM: THE WORST-CASE NORM OF UNCERTAIN LINEAR SYSTEMS UNDER INPUTS WITH MAGNITUDE BOUND AND RATE LIMIT, THESIS ADVISOR: ASSOC. PROF. DAVID BANJERDPONGCHAI, Ph.D., 141 pp.

For many years, researchers in control systems have widely employed the worst-case norms (WCNs) as performance indices measuring how worst control systems can be affected by specific disturbances, which in turn indicates how well a selected controller can reject undesired effects. In this dissertation, we consider the WCN of uncertain linear time-invariant systems when the disturbance inputs are modelled to have bounded magnitude and limited rate. The first part of this work is devoted to analysis and computation of the WCN in the absence of uncertainty. The WCN computation is formulated as an optimal control problem, whose solution yields characterization of the worst-case input. A novel algorithm called *Successive Pang Interval Search* (SPIS) is developed to construct the worst-case input and compute the WCN. The second part of this work involves analysis and computation of the WCN in the presence of uncertainty. The WCN analysis and computation of linear systems without uncertainty has now become the solid groundwork for the case of uncertain linear systems. The WCN computation in this case is formulated via discretization approach, which leads to an  $\mathcal{NP}$ -hard convex maximization problem. Novel upper and lower bounds of the WCN are introduced and can be obtained by solving sparse linear programming problems. To compute the exact WCN, we develop an algorithm called *Hierarchical Branch-and-Bound* (HBB) algorithm. This algorithm employs a standard Branch-and-Bound (BB) technique with a procedure called *Reduction of Ambiguity Magnitude Threshold* (RAMT). We validate the algorithm and compare numerical results with that obtained by an exhaustive search and a standard BB algorithm. HBB algorithm yields correct results with excellent computational speed and outperforms existing standard algorithms; hence, it is deemed to provide a viable means to attain the WCN computation of high dimensional problems. In addition, we demonstrate a practical use of HBB algorithm by applying to multi-objective PID tuning for an active suspension system subjected to variable load mass and road disturbances. The PID design is achieved with reasonable time and the simulation results clearly show that all design specifications are satisfied.

Department ..... Electrical Engineering  
 Field of study ..... Electrical Engineering  
 Academic year ..... 2006

Student's signature .....   
 Advisor's signature ..... 

## Acknowledgments

I am deeply indebted to my supervisor, Associate Professor David Banjerdpongchai, for his persistent support and helpful guidance since the time when I was an undergraduate. He kindly introduced me to this challenging arena of control engineering, sincerely equipped me with necessary knowledge which helped strengthen me academically. He also encouraged me to grow spiritually in the path of Jesus Christ.

I owed a debt of gratitude to my co-advisor, Professor Stephen Boyd, for his fruitful comments during my visit to Information Systems Laboratory, Stanford University. In particular, he suggested an idea of the uncertainty model and associated problem formulation of the WCN. Many other Professors in the Department of Electrical Engineering, Faculty of Engineering, Chulalongkorn University also deserve special recognition in this dissertation. I am deeply thankful to Associate Professor Suvalai Pratishthananda, Associate Professor Varaporn Jaovisidha, Associate Professor Watcharapong Khovidhungij, and Assistant Professor Suchin Arunsawatwong for providing me the clear background on control system engineering and inspiring my interests to important problems in this field. In addition, I would like to express my gratitude to the members of my committees, Associate Professor Ekachai Leelarasmee, Assistant Professor Manop Wongsaisuwon, Associate Professor Waree Kongprawechnon, and Associate Professor Wicharn Lewkeeratiyutkul, for their enthusiasm in examining the dissertation and their productive advices.

Most of the research is carried out at Control Systems Research Laboratory (CSRL), Department of Electrical Engineering, Chulalongkorn University. I greatly appreciate the financial from the Thailand Research Fund through the Royal Golden Jubilee Ph.D. Program under Grant No. PHD/0255/2545. I am also thankful to many current and past CSRL members, namely, Mr. Thapana Nampradit, Miss Jitkomut Songsiri, Mr. Pished Bunnun, Mr. Ekkaluk Piamboriboon, Miss Kamonwan Thiptawonnukoon, and many other generous and cheerful colleagues and friends. Finally, I would like to thank my parents, Mr. Kamthorn Khaisongkram and Mrs. Nittaya Khaisongkram; my sister, Miss Nutharikar Khaisongkram; and my best friend Mrs. Khanidtha Meevasana. For their many years of extensive support, overwhelming love, and unceasing encouragement, I will remain eternally grateful.

# Contents

<b>Abstract (Thai)</b> .....	<b>iv</b>
<b>Abstract (English)</b> .....	<b>v</b>
<b>Acknowledgments</b> .....	<b>vi</b>
<b>Contents</b> .....	<b>vii</b>
<b>List of Tables</b> .....	<b>x</b>
<b>List of Figures</b> .....	<b>xi</b>
<b>CHAPTER I INTRODUCTION</b> .....	<b>1</b>
1.1 Motivation .....	1
1.2 Literature Review .....	2
1.2.1 The WCN of Finite-Dimensional Convolution Systems .....	2
1.2.2 Maximization of Convex Functions .....	4
1.3 Objectives .....	6
1.4 Scope .....	6
1.5 Methodology .....	6
1.6 Contributions .....	7
1.7 Dissertation Outline .....	7
<b>CHAPTER II ANALYSIS OF THE WCN OF LINEAR SYSTEMS</b> .....	<b>9</b>
2.1 Analytical Preliminaries .....	9
2.1.1 Monotonicity and the WCN Approximation .....	10
2.1.2 Finiteness .....	11
2.2 Problem Formulation .....	12
2.3 Characterization of the Worst-Case Input .....	13
2.3.1 Necessary Conditions of the Worst-Case Input .....	15
2.3.2 Sufficiency of the Necessary Conditions .....	18
2.3.3 Uniqueness of the Worst-Case Input .....	19
2.3.4 Pang-bang Characteristics .....	20
2.4 Summary .....	25
<b>CHAPTER III COMPUTATION OF THE WCN OF LINEAR SYSTEMS</b> .....	<b>26</b>
3.1 Construction of the Worst-Case Input .....	26
3.1.1 Successive Pang Interval Search .....	26

3.1.2	Suitable Terminal Time	31
3.1.3	Detection of Crossing the Initial Time of a Plausible Pang Interval	33
3.1.4	Precise Location of Pang Intervals and Restarting Instants	41
3.1.5	The First and the Last Intervals	42
3.2	Computational Accuracy	45
3.2.1	Discrete-Time Formulation	45
3.2.2	Truncation Error	46
3.2.3	Discretization Error	47
3.3	Numerical Examples	48
3.4	Summary	50
<b>CHAPTER IV ANALYSIS OF THE WCN OF UNCERTAIN LINEAR SYSTEMS</b>		<b>53</b>
4.1	Analytical Preliminaries	53
4.1.1	Monotonicity and the WCN Approximation	54
4.1.2	Finiteness	56
4.2	Problem Formulation	57
4.3	Upper and Lower Bounds of the WCN	60
4.3.1	Upper Bound	60
4.3.2	Lower Bound	62
4.4	Bounds of the WCN Given Elements of $y$	64
4.4.1	Upper Bound	65
4.4.2	Lower Bound	66
4.5	LP Solvers via the Interior-Point Method	67
4.6	Numerical Examples	69
4.7	Summary	70
<b>CHAPTER V COMPUTATION OF THE WCN OF UNCERTAIN LINEAR SYSTEMS</b>		<b>72</b>
5.1	The Standard BB Algorithm	72
5.1.1	Branching Strategies	74
5.2	Numerical Examples	77
5.3	Solution of Discretized Problems with Increasing Sampling Rates	83
5.4	Upper Bound on $\{\alpha_j\}$	88
5.5	Hierarchical Branch-and-Bound Algorithm	92
5.5.1	Initiation of HBB	94
5.6	Numerical Examples	96
5.7	Summary	98
<b>CHAPTER VI APPLICATION TO ACTIVE SUSPENSION SYSTEM</b>		<b>102</b>
6.1	Dynamic Model of Active Suspension Systems	103
6.2	Control Design Specifications	104



6.3	PID Controller Design Procedure . . . . .	105
6.4	Summary . . . . .	108
<b>CHAPTER VII CONCLUSIONS.....</b>		<b>111</b>
7.1	Summary . . . . .	111
7.2	Further Improvements . . . . .	111
7.3	Possible Extensions . . . . .	113
<b>REFERENCES . . . . .</b>		<b>119</b>
<b>APPENDICES . . . . .</b>		<b>124</b>
	APPENDIX A . . . . .	125
	APPENDIX B . . . . .	127
	APPENDIX C . . . . .	131
	<b>Biography . . . . .</b>	<b>139</b>



สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย

## List of Tables

3.1	Comparison between the exact WCNs, the WCNs computed via continuous-time and discrete-time approaches, the actual computational errors, and their bounds. . . . .	50
5.1	Comparison between the computation times of the BB algorithm and the exhaustive search for the computations of the WCNs of three uncertain systems: system 1, 2, and 3, when $N$ varies from 10 to 20. The abbreviations s, m, and h signify the time units second, minute, and hour, respectively. . . . .	78
5.2	Comparison between the computation times of the BB algorithm and HBB algorithm used in computing the WCN of the three uncertain systems when $N = 48$ . . . . .	97
6.1	The WCNs of the PID-control active suspension systems when the PID parameters are obtained from the Z-N tuning method and the MBP. . . . .	107

สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย

## List of Figures

1.1	Disturbances whose magnitudes are bounded by $M$ and rates are limited by $D$ . . . . .	1
2.1	The cascaded system $h_1(t)$ containing a fictitious time lag and $h(t)$ , as a subsystem. The input $w(t)$ is shown to have its magnitude bounded by $M$ and its rate bounded by $D$ when the fictitious input $v(t)$ is bounded by $\min\{M, D\}$ . The output signal of both systems is $z(t)$ , which is bounded when $v(t)$ is bounded and $h_1(t)$ is BIBO stable. . . . .	12
2.2	An input having a pang-bang profile. The input comprises only pang and bang intervals over the time horizon. In the bang intervals, the input rest at its boundary $\pm M$ , whereas in the pang intervals the input has its rate at the limit $\pm D$ . . . . .	14
2.3	The boundary condition of the worst-case input. . . . .	16
2.4	The derivative condition of the worst-case input. . . . .	17
2.5	The magnitude condition of the worst-case input. . . . .	18
2.6	The spurious pang interval $\bar{\mathcal{I}}_{P,k}$ precedes the actual pang interval $\mathcal{I}_{P,k}$ . If SPIS proceeds to identify $\bar{\mathcal{I}}_{P,k}$ to be a pang interval, SPIS will be stuck at $t_4$ because the succeeding bang interval that has $t_4$ as an initial time cannot be constructed. In this case, SPIS will note down this spurious pang interval and move on to discover $\mathcal{I}_{P,k}$ instead. . . . .	25
3.1	The segments $\pi_1, \dots, \pi_6$ of $\tilde{s}(T-t)$ partitioned by the corresponding peak times $\tau_{p_1}, \dots, \tau_{p_7}$ . . . . .	28
3.2	The flow chart of the successive pang interval search (SPIS) algorithm. . . . .	30
3.3	The cutting instants $t'_0, \dots, t'_3$ and the cutting segments $\pi'_0, \dots, \pi'_3$ of $\tilde{s}(T-t)$ obtained at $t_c$ . . . . .	34
3.4	The cutting instants $t'_0, \dots, t'_6$ and the cutting segments $\pi'_0, \dots, \pi'_6$ of $\tilde{s}(T-t)$ obtained at $t_c (= t'_0)$ , and the cutting instants $t''_0, \dots, t''_6$ and the cutting segments $\pi''_0, \dots, \pi''_6$ of $\tilde{s}(T-t)$ obtained at $t_{c-} (= t''_0)$ . . . . .	35
3.5	The cutting instants $t'_0, \dots, t'_4$ and the cutting segments $\pi'_0, \dots, \pi'_4$ of $\tilde{s}(T-t)$ obtained when $p_{n+1}(0) = \beta_c$ . . . . .	43
3.6	The worst-case inputs associated with the WCN computational problem of the second order systems; (—) Case 1, (— —) Case 2, and ( $\cdot \cdot \cdot$ ) Case 3. . . . .	51
3.7	The worst-case outputs of the second order systems; (—) Case 1, (— —) Case 2, and ( $\cdot \cdot \cdot$ ) Case 3. . . . .	51
3.8	Comparison of the computation times used by SPIS and the discrete-time approach to solve the WCN computations for the second order systems: (a) Case 1, (b) Case 2, and (c) Case 3. . . . .	52
4.1	An uncertain system represented by a set $\mathbb{H}$ shown as a band of impulse response associated with the impulse envelope $(h_u(t), h_l(t))$ . . . . .	54
4.2	Geometric interpretation of the linear relaxation in the tetrahedral method. . . . .	63
4.3	The impulse response envelopes $(h_u(t), h_l(t))$ , which bound from above and below, all admissible impulse responses of the uncertain systems: (a) system 1, (b) system 2, and (c) system 3. . . . .	70

4.4	The upper bounds and lower bounds of the WCNs when the problem dimension varies from 10 to 100: (a) system 1, (b) system 2, and (c) system 3. . . . .	71
5.1	Branching a three-dimensional cube $\mathcal{B}_{\text{init}}$ into two-dimensional sub-cubes $\mathcal{B}'_0$ and $\mathcal{B}''_0$ whose vertices make up those of $\mathcal{B}_{\text{init}}$ . . . . .	73
5.2	Flow chart of the BB algorithm. . . . .	76
5.3	For WCN computation of system 1 using standard BB algorithm, the number of active nodes at each iteration: (a) $N = 10$ , (c) $N = 15$ , and (e) $N = 20$ , and the convergence of upper and lower bounds: (b) $N = 10$ , (d) $N = 15$ , (f) $N = 20$ . . . . .	80
5.4	For WCN computation of system 2 using standard BB algorithm, the number of active nodes at each iteration: (a) $N = 10$ , (c) $N = 15$ , and (e) $N = 20$ , and the convergence of upper and lower bounds: (b) $N = 10$ , (d) $N = 15$ , (f) $N = 20$ . . . . .	81
5.5	For WCN computation of system 3 using standard BB algorithm, the number of active nodes at each iteration: (a) $N = 10$ , (c) $N = 15$ , and (e) $N = 20$ , and the convergence of upper and lower bounds: (b) $N = 10$ , (d) $N = 15$ , (f) $N = 20$ . . . . .	82
5.6	The WCN of three uncertain linear convolution systems when the problem dimension varies from 10 to 20. . . . .	83
5.7	The input, impulse response, and output in the worst-case scenario of the WCN computation of system 1 with the dimension of 50. The estimated WCN should equal the magnitude of the worst-case output at $t = 10$ second. . . . .	84
5.8	The top subplot displays locations of $\hat{x}_i^{(j)}$ (marked with circle) along the graph of $\hat{w}^{(j)}(t)$ , while the middle subplot displays the locations of $\hat{\mathbf{x}}_i^{(j)}$ where the circle markers represent the points with $ \hat{\mathbf{x}}_i^{(j)}  > \alpha_j$ and the plus markers represent those with $ \hat{\mathbf{x}}_i^{(j)}  < \alpha_j$ . The bottom subplot displays $\hat{y}_i^{(j+1)}$ at the same instants where $ \hat{\mathbf{x}}_i^{(j)}  > \alpha_j$ . Some points lie along $h_u(t)$ and others along $h_l(t)$ , depending on the sign of $\hat{\mathbf{x}}_i^{(j)}$ . . . . .	88
5.9	Comparison between the worst-case inputs $\hat{w}^{(j)}(t)$ and $\hat{w}^{(j+1)}(t)$ . The cross marks represent points of $\hat{x}^{(j)}$ , while the circle marks represent those of $\hat{x}^{(j+1)}$ . . . . .	90
5.10	Distribution of $\hat{\beta}$ , which is computed as $2\ \hat{w}^{(2)}(t) - \hat{w}^{(1)}(t)\ _\infty / (D\tau_{s,1})$ , among totally 1,000 test problems. . . . .	92
5.11	Flowchart of HBB algorithm. . . . .	94
5.12	For WCN computation with $N = 48$ , the number of active nodes at each iteration: (a) system 1, (c) system 2, and (e) system 3, and the convergence of upper and lower bounds: (b) system 1, (d) system 2, (f) system 3. For the case of system 1, the number iteration is excessively high; the relevant computation time is about 5 days while the other two cases take only about 7 minutes. . . . .	99
5.13	The convergence of the upper and the lower bounds of the WCN computation for system 1 when $N = 48$ , redrawn within the range from the first iteration through the 1,000th iteration. .	100

5.14	Computation of the WCN of system 1 via HBB algorithm when $N = 1280$ : (a) the values of the WCN converges to certain limit, (b) the value of $ (p_{j+1}^* - p_j^*)/p_j^* $ at $j$ th iteration, (c) the cumulative number of inner iterations (BB iterations) at each outer iteration (HBB iteration), and (d) the input, impulse response, and output in the worst-case scenario. . . . .	101
6.1	The schematic diagram of a vehicle active suspension system. . . . .	103
6.2	The block diagram of classical PID control for an active suspension system. . . . .	107
6.3	Disturbance representing road roughness profile used in simulation of suspension systems. . .	108
6.4	Comparison of vehicle chassis accelerations and suspension travels between passive systems (dotted line), and PID-control active systems (solid line): (a) $m_1 = 10$ kg, (b) $m_1 = 100$ kg, (c) $m_1 = 150$ kg, (d) $m_1 = 200$ kg. . . . .	109
6.5	Control forces, provided by the PID controller with $K = 3794$ , $\tau_I = 46$ and $\tau_D = 1.62$ , acting on the active suspension control systems: (a) $m_1 = 10$ kg, (b) $m_1 = 100$ kg, (c) $m_1 = 150$ kg, (d) $m_1 = 200$ kg. The force magnitudes are always maintained within the allowable limit of $\pm 5$ kN. . . . .	110
1	The worst-case input of a lightly-damped second-order system, <i>i.e.</i> , $\omega_d \leq \pi D/2M$ . . .	134
2	The worst-case input of an underdamped second-order system, <i>i.e.</i> , $\omega_d > \pi D/2M$ . . .	136

# CHAPTER I

## INTRODUCTION

### 1.1 Motivation

One fundamental purpose of automatic control design is to regulate system outputs within the vicinity of their corresponding set points under the presence of plausible disturbances. There are a variety of disturbance sources in control systems including varying operating environments, measurement noises, inaccuracy in mathematical modelling, *etc.* Thus, practical controller design methods are usually required to compensate for the effect of these disturbances. However, disturbance characteristics with which a controller can effectively handle depend critically on a disturbance model used in a controller design process. Many available control design methods characterize disturbances as step signal or random noise. Nevertheless, these two disturbance models are somewhat unrealistic. For example, the rate of change of the step signal at the step time is infinite, and the magnitude of a random noise, at some points of time, can be extremely large even if its variance is finite. These unrealistic characteristics give rise to some conservatism in controller design paradigms.

To measure the performance of a control system in suppressing the effect of disturbances, it is desirable to indicate the worst-case effect that disturbances can induce. Specifically, the magnitude of this effect is usually represented by the size of output. This leads to the very definition of the *worst-case norm*, or in short, the *WCN*; this is the worst-case magnitude of outputs that can be generated when inputs are subjected to certain conditions by which an admissible input collection is characterized. Generally, the WCN may entail several measures of output magnitude and several types of input collection. In this work we consider only the supremum norm of the output, and only the collection that consists of all inputs with *bounded magnitudes and limited rates of change*. Figure 1.1 illustrates an example in the input collection where the magnitude bound is  $M$ , while the rate limit is  $D$ .

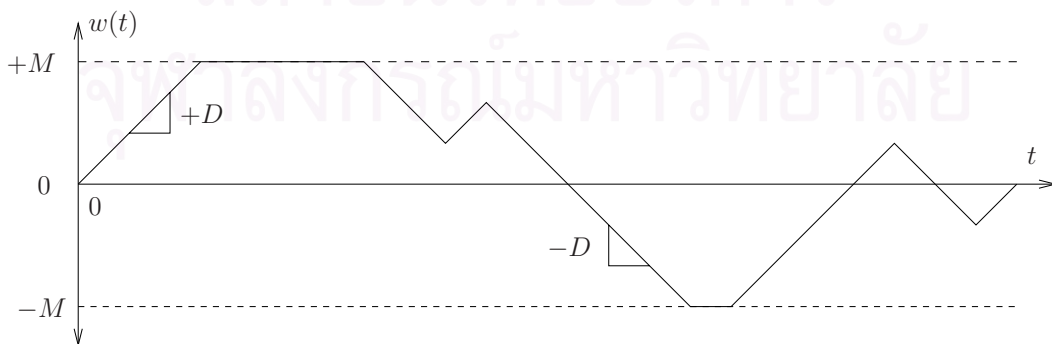


Figure 1.1: Disturbances whose magnitudes are bounded by  $M$  and rates are limited by  $D$ .

In many industrial processes, it is fairly practical and realistic to model disturbances as signals with bounded magnitudes and limited rate of change since these disturbances usually involves masses or inertias that cannot change instantly. An example of systems having this type of disturbances is a distillation column. One of the disturbances is the deviation of the input feed rate which varies over the time and is limited by the pipe dimension causing a certain bounding condition on its magnitude. In addition, its rate of change is confined by the mass of raw material fed into the column, and by the power of the feed pump. Another disturbance is the deviation of the feed composition which relies on the foregoing chemical reactor, introducing a bounding condition on its magnitude. Similarly, its rate of change is not arbitrary, but is restricted by the rate of change of the raw-material flow rate. Hence, it usually takes some time for the feed composition to change its value. In addition to the disturbance input, actual mathematical models usually incorporate *model uncertainty* as they can never model physical systems exactly. Uncertainty in the distillation column arises when we use a linear model to approximate the highly nonlinear dynamics of the distillation column. Since the linear model is usually obtained at a specific equilibrium point, slight shifts in operating conditions during the distillation process can cause a significant difference between actual nonlinear dynamics and its linear model. Generally, this error in modelling, though not known precisely, can be bound in some manner.

Another example of uncertain mechanical systems subject disturbances with bounded magnitudes and rates of change is a vehicle suspension system. The source of disturbance is the roughness of the road surface in contact with the wheels. The deviation of the road surface from the level can be reasonably modelled to have bounded magnitude and limited rate of change. This is because excessive or sharp deviations of the surface may not be encountered in regular road conditions. The suspension system also involves uncertainty as masses of passengers and loads vary from one situation to another. In this case, the passenger and load masses cannot exceed the maximum allowable load that the vehicle can carry, which suggests the bound on this uncertainty. To comprehensively analyze the closed-loop regulatory performance of a control system, the WCN should be obtained over the set of all possible plant uncertainty. Specifically, an effort should be made to compute the WCN of uncertain systems instead of just nominal systems.

## 1.2 Literature Review

### 1.2.1 The WCN of Finite-Dimensional Convolution Systems

The concept of using system norms as performance measures indicating *gain* of control systems was pioneered by Zames [1], Narendra and Goldwyn [2], and Sandberg [3], for example. It is a well-known fact that a relaxed version of the WCN, by omitting the rate limit  $D$ , is a product of  $M$  and  $\mathcal{L}_1$ -norm. Dahleh and Diaz-Bobillo [4] provide comprehensive exposition on  $\mathcal{L}_1$  theory. The development of the WCN itself can be traced back to Birch and Jackson [5] who studied the problem of computing the worst-case peak magnitude of a convolution system by constructing the worst-case input. Nevertheless, the conditions that Birch and Jackson derived and claimed to be necessary and

sufficient appear to be only necessary. It can be shown, for some cases, that there is an input other than the worst-case input which satisfies Birch and Jackson's conditions. It is as far as their main result for second-order systems that Birch and Jackson's conditions assume both necessity and sufficiency. Furthermore, their procedure to construct the worst-case input is based on graphical analysis without the mathematical justification.

Thereafter, Chang [6, 7] have investigated the equivalent problem of transferring the output to the desired target in minimum time. He has derived necessary conditions for the worst-case input of finite-dimensional convolution systems by applying the optimal-control framework for the first time. He also remarked that although the methods to construct the worst-case input for second-order or third-order systems could be easily determined but that for higher-order systems would require a tedious calculation.

Shortly afterward, Horowitz [8] has proposed a set of rules that can be used in constructing the worst-case input of certain systems. He has justified the existence and uniqueness of a solution; hence, this implicitly implies sufficiency of his rules. However, he has not extended the results to general systems. Later, Bongiorno Jr. [9] has considered the problem in MIMO case and first given sufficient conditions for the worst-case input in general. Nevertheless, an approach to construct such input has not been clearly mentioned.

Apart from these research works, Boyd and Barratt [10] pointed out that the norm-computation problem can be simply formulated as a free-terminal-time optimal control problem with control and state variable constraints. They also suggested that the problem would be solved by conventional numerical methods in optimal control [11–13]. Nevertheless, the solution of the formulated optimal control problem is not trivial. In fact, it is still questionable whether these methods are practical and effective for computation. More specifically, optimal control problems with constraints on control and state variables usually yield a set of necessary conditions constituting nonlinear two-point boundary value problems (TPBVPs). The nonlinearity of the problems depend upon the techniques that are adopted not only to convert the inequality into equality constraints, but also to augment Hamiltonian functions with these constraints.

One of such traditional techniques incorporates the penalty functions of fictitious state variables in terms of Heaviside step functions that are active whenever the constraints are violated [14–16]. However, this technique appears to be a conceptual tool for deriving necessary conditions of the problem rather than an effective computational tool. In fact, possible methods [17] to solve for the worst-case input are seemingly impractical in that they suffer from intractable computational errors due to numerical integration of Heaviside functions. Also, they require some ad hoc initial guesses on optimal trajectories which cause certain difficulties in applying the method to general cases.

Another relevant literature presented by Saridis and Rekasius [18] incorporated the same input collection with a performance measure called the worst-case error. They exploited an optimal control formulation which resulted in a nonlinear TPBVP, yet without any use of a Heaviside function. This yielded smoother responses for integration. They also developed the combined numerical-analytical method to construct the worst-case input. Nevertheless, the convergence of such method is not guar-



anteed. Furthermore, Lane [19] obtained a set of necessary and sufficient conditions for the worst-case input of convolution systems and the rules to construct it. The proofs of necessity and sufficiency, in some extent, resemble [7] and [9] with some additional argument of dynamic programming. He has also implemented a FORTRAN code to compute the WCN, yet his method is rather tedious and needs to determine an involved auxiliary function. Another recent work is by Reinelt [20]. His necessary conditions descended from Birch and Jackson's conditions [5]. Furthermore, Reinelt's necessary conditions are limited to the case that the rate limit of an input must be somewhat faster than the natural frequency of a convolution system. Besides, his numerical computation is only a discrete approximation in which none of his necessary conditions were exercised.

The most recent results by Khaisongkram and Banjerdpongchai [21, 22] employed an optimal control formulation. Necessary conditions acquired by the Pontryagin's Maximum Principle were analyzed in a straightforward manner to determine the practical characterization of the worst-case input. Such characterization serves as a means to derive the systematic method that constructs the worst-case input. The optimal control formulation and solution in [22] is actually the refinement of that in [21] with additional knowledge of computable upper bounds on errors in computing the WCN.

Although there have been many significant attempts so far in computing the WCN of convolution systems subject to inputs with bounded magnitude and rate, a few research studies consider the same norm on uncertain systems. Reinelt [23] considered the WCN of uncertain linear systems, and formulated the computational problem as a general quadratic programming. Nevertheless, there are certain drawbacks in Reinelt's formulation. First, the formulation was based on his former work [20], and incomplete in some aspects mentioned previously. Second, he did not focus on how to solve the formulated problem. In fact, the general quadratic programming is a significant problem in mathematical programming, and is still under extensive ongoing investigations. Combining with his discrete-approximation approach, his proposition may not work for general convolution systems. However, with further substantial improvement, his discretizing approach presented in [20, 23] can be promising. In fact, formulating WCN computational problem via this approach results in a bilinear programming, which can be converted to a convex maximization problem with polytopic constraints. This means some advanced knowledge in convex maximization should be taken into account. The following section is devoted to review the literature in such area.

### 1.2.2 Maximization of Convex Functions

Due to the standard nature of linearly-constrained convex maximization problem that its solution lies in one of the vertices (extreme points) of a polyhedron generated by the constraints, the most obvious algorithm successively searches along adjacent vertices and ranks the objective value to obtain the optimal solution. Murty [24] seems to first present this extreme-point ranking approach in 1968.

A large portion of research studies on minimizing a concave function have been involved with branch-and-bound (BB) algorithm and some cutting plane methods. The BB algorithm is first proposed by Falk [25] in 1973, to solve for a linear max-min problem, which is equivalent to a bilinear programming. This algorithm has been continually modified afterward. One major application of BB

algorithm is to utilize it with a cutting plane method. A popular approach of cutting plane method with BB technique is the conical algorithm presented by Tuy [26] in 1964. He first introduced the concept of convexity cuts that can be used to eliminate parts of feasible region from consideration. However, his method still suffers from no guarantee on finite convergence [27], which later led to a number of modified versions since then [28–32].

In 1976, Konno [33] studied the problem of maximization of a convex quadratic function over a polyhedron, and in 1980, his research was restricted to the case of maximizing over a hypercube [34]. He is the first person who proposed the equivalence between convex quadratic programming and bilinear programming, so that a cutting plane algorithm for bilinear programming can be exploited to find a solution. This cutting plane algorithm is equipped with enumerated elements to guarantee finite convergence.

As a number of engineering problems (including robust control analysis and design) have fallen into a category of concave minimization (or equivalently, convex maximization), this type of optimization gains increasing attention. Similar to Konno's direction, some works confine concave objective functions to concave quadratic objective functions; others extend alternatively to indefinite quadratic objective functions. General polytopic constraints are also simplified as box constraints. Examples of box-constrained indefinite quadratic programming can be seen in [35–39]. Note that most recent works attempt to tackle general quadratic programming because of its challenging intractability. In fact, Pardalos and Vavasis demonstrated that this problem is  $\mathcal{NP}$ -hard even with solely one negative eigenvalue [40].

Several of previous works rely on BB mechanism as a local search, while some means were proposed to choose a good initial searching point which tends to yield a global optimal solution. Particularly, in 1998, An and Tao [36] have employed a BB scheme with relaxed box constraints using ellipsoidal estimation. The approach successfully deleted much of feasible region from further consideration. Furthermore, in 2004, Angelis *et al.* [38] have proposed a novel algorithm to minimize a general quadratic function subject to simple box constraints. Their method consists of two phases. In the first phase, the algorithm finds a promising starting point by approximating the hypercube, induced by the box constraints, with a Euclidean ball. In the second phase, the algorithm employs (as usual) a local BB search to locate the exact solution of the problem on one of the hypercube vertex.

When the WCN computational problem is cast as a convex maximization via a discretizing approach, one would expect that high discretizing rate would yield high computational accuracy. Apparently, this high rate implies a large-scale optimization problem. Nevertheless, despite an extensive studies on convex maximization, none of this has focus on a large-scale problem, *e.g.*, a problem with dimension up to thousands or greater. In fact, some research works deal with problem dimension of around ten with the main attempts to generalize their methods [32, 41–43]; some others intend to tackle more specialized problem [44–46]. However, they can still handle a problem with dimension up to only fifty.

### 1.3 Objectives

The objective of this dissertation are to analyze and compute the WCN of finite-dimensional convolution systems in the presence of dynamic uncertainty and subject to disturbances. In particular, the uncertainty is bounded and the disturbance magnitude and rate are also bounded. It is also aimed to establish an effective algorithm (in terms of computational time and accuracy) to compute such WCN, and to develop a ready-to-use computer program for the algorithm. It should be emphasized that *there has been no algorithm to compute this WCN so far*.

### 1.4 Scope

1. Develop an algorithm for computing the WCN of finite-dimensional convolution systems with time-varying uncertainty and subject to inputs with magnitude bound and rate limit.
2. Implement a computer program based on the aforementioned algorithm.
3. Test the computer program with selected numerical examples.

### 1.5 Methodology

1. Collect and study literature on the WCN of linear systems under inputs with magnitude bound and rate limit as well as other related performance indices. In addition, some literature was reviewed on basic optimal control, optimal control with singularity, optimal control with state-variable constraint, numerical algorithm in optimal control, the Generalized Karush-Kuhn-Tucker Theorem, and the application of Pontryagin's Maximum Principle.
2. Refine the theoretical results in [47], and consequently modify the computational algorithm therein, which is used for computing the WCN of finite-dimensional convolution systems. This are carried out to improve its computational efficiency and accuracy. Specifically, the sufficiency of the necessary conditions confirms that the computed input is the worst-case input; the computational error can be predicted; the truncation error can be specified as a criterion to select suitable truncating terminal time; the satisfactory computational speed and accuracy were exhibited via numerical examples.
3. Formulate the problem of computing the WCN of linear systems under inputs with magnitude bound and rate limit when the systems contain dynamic uncertainty. To do so, the impulse response of the linear system is sampled and the problem is cast as a finite-dimensional convex maximization problem. Analyze the upper bound and lower bound of the objective function of such maximization. Consider how these analytical knowledge can be applied to solve for the exact optimum of the objective function, and design convex maximization solvers which are efficient in terms of computation time.

4. Study literature on bilinear programming, general convex maximization, convex maximization (or concave minimization) with box constraints, convex quadratic maximization, BB techniques and cutting-plane method for concave minimization.
5. Develop computer programs. Validate such programs via some testing such as obtaining the errors between the computed values and the corresponding exact values. Observe the computational time as well as accuracy.
6. Select an appropriate system with uncertainty and disturbances subject to bounded magnitude and limited rate of change. Design a controller that minimizes the WCN of this process and apply the controller to the system simulator using MATLAB/Simulink. Analyze and evaluate the control performance.

## 1.6 Contributions

1. Mathematical foundation of the WCN of uncertain finite-dimensional convolution systems under inputs with bounded magnitude and limited rate.
2. Novel computational tools with the corresponding computer programs for computing such WCN.
3. An example of real application of the WCN as a performance index in control system design.

## 1.7 Dissertation Outline

In this dissertation, we consider finite-dimensional linear time-invariant systems or simply convolution systems. The systems are causal and of single input and single output. The WCN is defined in some particular manner based on specific classes of input and impulse response. The dissertation is organized as follows:

In the next chapter, the WCN of linear systems without uncertainty is considered. The WCN is mathematically defined and simplified. Some properties of the WCN are analyzed. The WCN computational problem is cast as a fixed-terminal time optimal control problem with only the terminal cost. Then, the Pontryagin's Maximum Principle and the Generalized Karush-Kuhn-Tucker Theorem are applied to derive the necessary conditions which characterize the worst-case input.

In Chapter 3, a computational algorithm is developed based on the derived conditions in the previous chapter. This algorithm, named as Successive Pang Interval Search (SPIS), attempts to construct the worst-case input by determining where is it the input has its rate at the limit  $D$ . The construction proceed along the time axis throughout a finite time horizon, and the WCN is simply the convolution integral of the worst-case input and the impulse response of the system. The computational errors are analyzed, and finally the WCN obtained from SPIS is verified with second-order systems whose analytical solutions of the WCN are available.

In Chapter 4, we turn our attention to the WCN when the linear system is subjected to specific form of uncertainty. Some properties of the WCN are presented in parallel to the case of no uncertainty. The problem is formulated via a discrete-time approach, resulting in an  $\mathcal{NP}$ -hard bilinear programming problem such that the efficient algorithm does not exist to solve for a global solution. A convex maximization amounts to the originally formulated problem has also been formulated. A new practical upper bound and tight lower bound are introduced for later use.

In Chapter 5, an overview of the BB algorithm is described. Exploiting the known upper/lower bounds, we use this algorithm to solve for the exact WCN. A promising branching technique is proposed. The algorithm is later validated with the exhaustive search. Nevertheless, the BB algorithm is not as effective because it consumes significant computation time, so we invent a new algorithm called Hierarchical Branch-and-Bound (HBB) algorithm to compute the WCN. This algorithm is based on the BB computation. HBB can solve for WCN of some problems that satisfying certain assumptions. A heuristic selection of algorithm parameters is given, and the algorithm is respectively validated with the BB algorithm.

In Chapter 6, a numerical example is presented to illustrate the application of the WCN as a control design criterion. An example is an active vehicle suspension control system. The standard design criteria are considered. A search algorithm called moving boundary process (MBP) is employed to seek for an appropriate controller parameters, and a simulation is carried out in comparison with passive suspension control.

In the last chapter, the main contribution of each chapter is summarized and the important achievement of the dissertation is given. The suggestions for possible improvements, which required further studies, are summarized as well.

สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย

## CHAPTER II

### ANALYSIS OF THE WCN OF LINEAR SYSTEMS

#### 2.1 Analytical Preliminaries

For all  $t$ , let  $w(t)$  be a real-valued continuous function, and its derivative  $\dot{w}(t)$  be piecewise continuous. The input set  $\mathcal{W}$  is characterized by a magnitude bound and a rate limit as

$$\mathcal{W} \triangleq \{w(t) : w(t) = 0, \forall t \leq 0; |w(t)| \leq M, |\dot{w}(t)| \leq D, \forall t > 0\} \quad (2.1)$$

where  $0 < M < \infty$  and  $0 < D < \infty$ . This input set is illustrated in Figure 1.1. To deal with definiteness, we assume that the derivative  $\dot{w}(t)$  at each point of discontinuity takes the value of the left-sided neighborhood. In other words, this means

$$\dot{w}(t) = \lim_{\tau \rightarrow t^-} \dot{w}(\tau),$$

for all  $t$  at which  $\dot{w}(t)$  is discontinuous. This assumption seems not to be numerically necessary for computing the WCN in that the differences of two inputs at finite instants will surely have no effect on the output of finite-dimensional convolution systems. However, it will benefit mathematical rigor in developing definitions and theorems in later sections.

Note that the definition of the input set in (2.1), along with the continuity of  $w(t)$ , implies the causal continuity associated with the zero initial condition,  $w(0) = 0$ , for all admissible inputs. Although this condition is not commonly seen in previous works, it does make sense because the concept of imposing a rate limit, in addition to a magnitude bound, originates from the physical requirement of input continuity at  $t > 0$ . Thus, assuming  $w(t)$  to be continuous at  $t = 0$  is reasonable. Later on, we will show that this assumption is not only reasonable, but also beneficial.

Let  $\mathbb{H}_0$  be a shorthand notation for a set of single-input single-output (SISO), strictly proper, finite-dimensional, causal, linear time-invariant systems. It is interesting to note that most SISO linear time-invariant systems are in  $\mathbb{H}_0$ . Also remark that  $\mathbb{H}_0$  is actually a vector space, on which a norm can be defined. Let  $h(t)$  be an impulse response of a convolution system in  $\mathbb{H}_0$ . Let the system output be denoted by  $z(t)$ . For the convolution system, we have

$$z(t) = h(t) * w(t) = \int_0^t h(t - \tau)w(\tau)d\tau.$$

Accordingly, the worst-case magnitude of  $z(t)$  at each  $t$ , denoted by  $\xi(t) : [0, \infty) \mapsto [0, \infty)$ , is defined as

$$\xi(t) \triangleq \max_{w \in \mathcal{W}} |z(t)| = \max_{w \in \mathcal{W}} |h(t) * w(t)|. \quad (2.2)$$

Following the maximum value theorem [48],  $\mathcal{W}$  is compact, hence,  $z(t)$  attains its maximum on  $\mathcal{W}$  at each  $t$ . Let the element in  $\mathcal{W}$  that causes the worst-case magnitude  $\xi(t)$  be referred to as

the worst-case input with a notation  $\hat{w}(t)$ . With respect to the underlying input set  $\mathcal{W}$ , the WCN  $\|\cdot\|_{\text{wc}} : \mathbb{H}_0 \mapsto [0, \infty]$  is defined as the worst-case peak magnitude expressing in terms of  $\xi(t)$  as follows:

$$\|h\|_{\text{wc}} \triangleq \sup_{t \geq 0} \xi(t). \quad (2.3)$$

It is easy to show that this norm is well defined on  $\mathbb{H}_0$  and that it satisfies all properties of norm except that we allow the norm to assume the value  $+\infty$ . This relaxation permits us to extend the definition of the WCN to unstable systems.

Be reminded that the term  $h(t) * w(t)$  is linear in  $w(t)$  and the bounding conditions on  $w(t)$  are symmetrically defined, *i.e.*, for all  $t > 0$

$$\begin{aligned} -M &\leq w(t) \leq M, \\ -D &\leq \dot{w}(t) \leq D. \end{aligned}$$

Hence,  $-\hat{w}(t)$  is also the worst-case input apart from  $\hat{w}(t)$ . Based on this fact, the definition of the worst-case output magnitude in (2.2) can be simplified by omitting the absolute-value operator. We obtain an equivalent definition as

$$\xi(t) = \max_{w \in \mathcal{W}} z(t) = \max_{w \in \mathcal{W}} [h(t) * w(t)]. \quad (2.4)$$

The WCN in (2.3) can be further simplified by the standard fact that the worst-case magnitude is a monotonic function of time. This was first established by Lane [19] and also reported by Reinelt [20]. However, to make the dissertation self-contained, we provide a concise presentation here.

### 2.1.1 Monotonicity and the WCN Approximation

To show that  $\xi(t)$  is actually a nondecreasing function of time, let  $t_1 < t_2$ , and assume that  $\hat{w}_1(t)$  yields  $\xi(t_1)$ , *i.e.*,  $\xi(t_1) = h(t_1) * \hat{w}_1(t_1)$ . Define  $w_2(t)$  by shifting  $w_1(t)$  with  $\Delta t = t_2 - t_1$ . That is

$$w_2(t) \triangleq \begin{cases} 0, & 0 \leq t \leq t_1, \\ \hat{w}_1(t - \Delta t), & \Delta t \leq t \leq t_2. \end{cases}$$

By a simple change of an integration variable, it follows that

$$h(t_2) * w_2(t_2) = h(t_1) * \hat{w}_1(t_1) = \xi(t_1).$$

Now, assume that  $\hat{w}_2(t)$  yields  $\xi(t_2)$ . Evidently, by definition (2.4), we have

$$\xi(t_2) = h(t_2) * \hat{w}_2(t_2) \geq h(t_2) * w_2(t_2) = \xi(t_1).$$

As mentioned earlier, the causal continuity plays an important role in the previous argument. If there is no such a condition,  $\hat{w}_1(0)$  may not equal zero, resulting in a discontinuity of  $w_2(t)$  at  $t = \Delta t$ . Consequently,  $w_2(t)$  is excluded from  $\mathcal{W}$ , and then  $\xi(t_2)$  may not have any explicit relation with  $\xi(t_1)$ .

The assumption on causal continuity is beneficial in making the definition (2.3) equivalent to

$$\|h\|_{\text{wc}} = \lim_{t \rightarrow \infty} \xi(t). \quad (2.5)$$

This implies that we can approximate  $\|h\|_{\text{wc}}$  by  $\xi(T)$  with arbitrary degree of accuracy by taking  $T$  sufficiently large. How we can determine a suitable choice of  $T$  will be explained in details in Section 3.1.2.

### 2.1.2 Finiteness

It is worth noting that in order for the WCN to be useful in performance analysis, it must first be finite. Lane [19] fully verified that the necessary and sufficient condition on finiteness of its WCN is that a convolution system is BIBO stable.

The sufficiency follows from the well-known fact that an upper bound of the WCN is  $M\|h(t)\|_1$  where  $\|h(t)\|_1$  denotes the  $\mathcal{L}_1$ -norm of a convolution system [19, 21]. Hence, if the system is BIBO stable, its  $\mathcal{L}_1$ -norm is finite and so is the WCN.

To make this dissertation self-contained, we establish an alternative proof of necessity as follows. Here, the idea used in establishing the finiteness of the WCN is motivated by [10]. Suppose that the WCN of  $h(t)$  is finite. Let us consider a simple lag  $1/(2s + 1)$ , and implicitly define  $h_1(t)$  as follows:

$$H_1(s) \triangleq H(s) \left( \frac{1}{2s + 1} \right)$$

where  $H_1(s)$  and  $H(s)$  stand for the Laplace transforms of  $h_1(t)$  and  $h(t)$ , respectively. It is obvious that the system with an impulse response  $h_1(t)$  is in  $\mathbb{H}_0$ . Since the lag is stable and has no unstable zero,  $h(t)$  is BIBO stable if and only if  $h_1(t)$  is. Hence, we have to show that  $h_1(t)$  is BIBO stable.

Let  $v(t)$  be an input of  $h_1(t)$ . As illustrated in Figure 2.1, the signal  $w(t)$ , which is an input of  $h(t)$ , can now be regarded as an output of the lag, and is induced by  $v(t)$ , i.e.,  $w(t) = (1/2)e^{-t/2} * v(t)$  where  $(1/2)e^{-t/2}$  is an impulse response of  $1/(2s + 1)$ . Consider an input set

$$\mathcal{V} \triangleq \{v(t) : |v(t)| \leq \min\{M, D\}\}, \quad (2.6)$$

which includes all *bounded* input  $v(t)$ . It is easy to verify that  $\|(1/2)e^{-t/2}\|_1 = 1$ . Thus, we have

$$|w(t)| \leq \|(1/2)e^{-t/2}\|_1 |v(t)| = \min\{M, D\} \leq M. \quad (2.7)$$

Furthermore, since an ordinary differential equation that relates  $w(t)$  to  $v(t)$  is  $2\dot{w}(t) + w(t) = v(t)$ , we can show that

$$|\dot{w}(t)| = \frac{1}{2}|v(t) - w(t)| \leq \frac{1}{2}(|v(t)| + |w(t)|) \leq \frac{1}{2}(2 \min\{M, D\}) \leq D. \quad (2.8)$$

Note that we have made use of (2.7), and the definition (2.6).

The inequalities (2.7) and (2.8) conclude that any  $v(t) \in \mathcal{V}$  yields a signal  $w(t)$  which lies in  $\mathcal{W}$ . Moreover, the output  $z(t)$  must be bounded since we have assumed the finiteness of the WCN of  $h(t)$ . Shortly speaking, any  $v(t) \in \mathcal{V}$  will result in a bounded output  $z(t)$ , which implies that  $\|h_1(t)\|_1$  is finite, or equivalently,  $h_1(t)$  is BIBO stable.



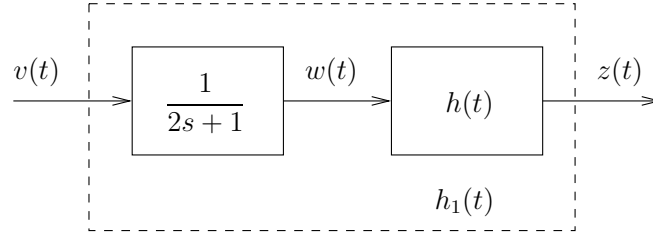


Figure 2.1: The cascaded system  $h_1(t)$  containing a fictitious time lag and  $h(t)$ , as a subsystem. The input  $w(t)$  is shown to have its magnitude bounded by  $M$  and its rate bounded by  $D$  when the fictitious input  $v(t)$  is bounded by  $\min\{M, D\}$ . The output signal of both systems is  $z(t)$ , which is bounded when  $v(t)$  is bounded and  $h_1(t)$  is BIBO stable.

## 2.2 Problem Formulation

Recall that  $(A, B, C)$  is a minimal realization of a convolution system  $h(t)$ . Assume further that  $h(t)$  is strictly proper, and  $x(t) \in \mathbb{R}^n$  is a state vector of its state space representation. To compute the worst-case magnitude, we define an auxiliary state variable  $x_{n+1}(t) \in \mathbb{R}$  and a control signal  $u(t)$  as follows.

$$\begin{aligned} x_{n+1}(t) &\triangleq w(t), \\ u(t) &\triangleq \dot{w}(t). \end{aligned}$$

The fixed-terminal-time optimal control problem can be posed as

$$\begin{aligned} \max_u \quad & Cx(T) \\ \text{s.t.} \quad & \dot{x}(t) = Ax(t) + Bx_{n+1}(t), \quad x(0) = 0, \\ & \dot{x}_{n+1}(t) = u(t), \quad x_{n+1}(0) = 0, \\ & -M \leq x_{n+1}(t) \leq M, \quad 0 \leq t \leq T, \\ & -D \leq u(t) \leq D, \quad 0 \leq t \leq T. \end{aligned}$$

Notice that the initial time is 0, the terminal time is  $T$ , the objective cost is  $z(T) = Cx(T)$ , and the initial condition of  $x_{n+1}(0)$  is 0. To define the Hamiltonian function, the inequality constraint is rewritten as

$$x_{n+1}^2(t) \leq M^2. \quad (2.9)$$

The Hamiltonian function is defined in terms of system dynamics and the inequality constraint on  $x_{n+1}(t)$  as

$$\mathcal{H}(x, x_{n+1}, u, p, p_{n+1}, \mu) \triangleq p^T(Ax + Bx_{n+1}) + p_{n+1}u + \mu(M^2 - x_{n+1}^2) \quad (2.10)$$

where  $p(t) \in \mathbb{R}^n$ ,  $p_{n+1}(t) \in \mathbb{R}$ , and  $\mu(t) \in \mathbb{R}$  are Lagrange multipliers corresponding to  $\dot{x}(t)$ ,  $\dot{x}_{n+1}(t)$ , and the constraint (2.9), respectively. For simplicity, the argument  $t$  is omitted in (2.10). By the Generalized Karush-Kuhn-Tucker Theorem [49], the variable  $\mu(t)$  is nonnegative and vanishes whenever the constraint (2.9) is inactive, i.e.,  $|x_{n+1}^2(t)| < M^2$ . The adjoint equations are obtained as

$$\dot{p}(t) = -A^T p(t), \quad (2.11)$$

$$\dot{p}_{n+1}(t) = -B^T p(t) + 2\mu(t)x_{n+1}(t) \quad (2.12)$$

with the following transversality conditions:

$$p(T) = C^T, \quad (2.13)$$

$$p_{n+1}(T) = 0. \quad (2.14)$$

The response of  $p(t)$  can be readily obtained from the differential equation (2.11) and the final condition (2.13):

$$p(t) = e^{A^T(T-t)}C^T. \quad (2.15)$$

Then, substituting  $p(t)$  into (2.12) gives

$$\begin{aligned} \dot{p}_{n+1}(t) &= -B^T e^{A^T(T-t)}C^T + 2\mu(t)x_{n+1}(t) \\ &= -h(T-t) + 2\mu(t)x_{n+1}(t). \end{aligned} \quad (2.16)$$

Note here that we consider the scalar case only, *i.e.*,  $h^T(T-t) = h(T-t)$ . Integrating both sides of (2.16) from  $t_1$  to  $t_2$ , we obtain

$$p_{n+1}(t_2) - p_{n+1}(t_1) = [s(T-t_2) - s(T-t_1)] + 2 \int_{t_1}^{t_2} \mu(t)x_{n+1}(t)dt \quad (2.17)$$

where  $s(t)$  is the step response corresponding to  $h(t)$ . For later reference, it is noted here that the Weierstrass-Erdmann corner condition [16, 17], which requires the continuity of  $\mathcal{H}$  and all Lagrange multipliers, implies that  $p_{n+1}(t)$  must be continuous everywhere.

By the Pontryagin's Maximum Principle [13], the optimal control signal  $u(t)$  is chosen to maximize the Hamiltonian function in (2.10), *i.e.*, the control signal should have the same sign as  $p_{n+1}(t)$ , with the greatest attainable magnitude. This gives the following optimal control law

$$\hat{u}(t) = D\text{sgn}[p_{n+1}(t)]. \quad (2.18)$$

By the definition (2.10), the singular control may occur when  $p_{n+1}(t) = 0$ . This, however, does not violate the law (2.18). To verify this claim, suppose  $p_{n+1}(t) = 0$  everywhere in finite time intervals referred to as  $\mathcal{I}$ . Hence, by (2.16), we have

$$h(T-t) = 2\mu(t)x_{n+1}(t). \quad (2.19)$$

Since  $h(T-t)$  is a smooth function,  $\mu(t)$  must be nonzero almost everywhere in  $\mathcal{I}$ . This suggests that the constraint (2.9) should be active, *i.e.*,  $|x_{n+1}(t)| = M$ , everywhere in  $\mathcal{I}$ . Hence,  $u(t) = 0 = \text{sgn}[p_{n+1}(t)]$  everywhere in  $\mathcal{I}$ . It can be seen that the optimal control profile is simply bang-off-bang. As  $\hat{u}(t)$  becomes zero in the *off* time interval, we have  $|x_{n+1}(t)| = M$ .

### 2.3 Characterization of the Worst-Case Input

In this section, we will give a few theorems representing necessary conditions of the worst-case input characteristics. As discussed previously, the rate of change of the worst-case input is either zero or  $\pm D$ . While the rate of change is zero, we have shown that the worst-case-input magnitude is  $\pm M$ .

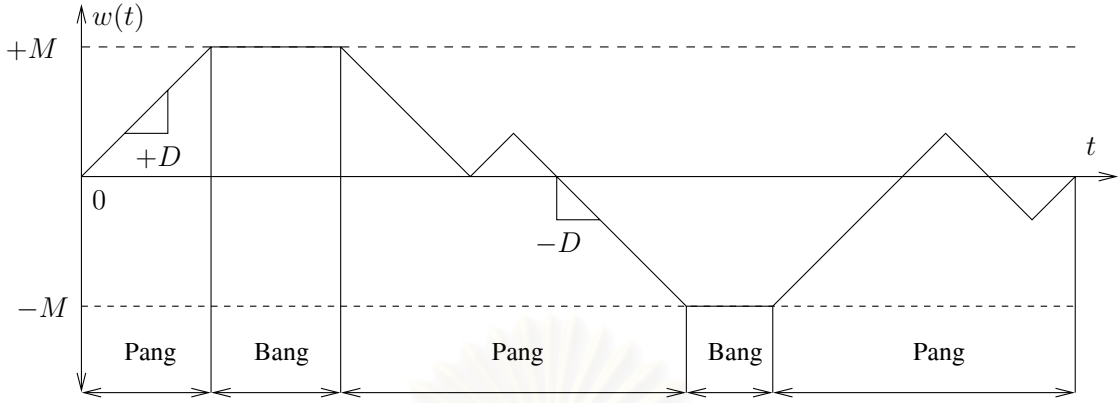


Figure 2.2: An input having a pang-bang profile. The input comprises only pang and bang intervals over the time horizon. In the bang intervals, the input rest at its boundary  $\pm M$ , whereas in the pang intervals the input has its rate at the limit  $\pm D$ .

Thus, we can summarize that, at any instant of time, the worst-case input has either the magnitude or the rate of change equal to its limit. This characteristic of input was named by Chang [6] as *pang-bang*. He classified time intervals of the worst-case input into two types alternating each other along the time axis. The *bang* interval is a time interval in which the magnitude of the worst-case input is at its limit, while the *pang* interval is a time interval in which the rate of change of the worst-case input is at its limit. An example of a pang-bang signal is illustrated in Figure 2.2. The precise definitions of such intervals are given as follows.

Let  $\hat{x}_{n+1}(t)$  be the optimal trajectory generated by the optimal control  $\hat{u}(t)$  in (2.18). The notations of the worst-case input are restored as  $\hat{w}(t) = \hat{x}_{n+1}(t)$  and  $\hat{w}'(t) = \hat{u}(t)$ .

**Definition 2.1** The pang interval of  $\hat{w}(t)$  denoted by  $\mathcal{I}_P$  is the time interval  $(t_0, t_f]$  in which  $|\hat{w}'(t)| = D$  everywhere, and there is no other time interval  $(\hat{t}_0, \hat{t}_f]$  with the same property and  $(t_0, t_f] \subsetneq (\hat{t}_0, \hat{t}_f]$ <sup>1</sup>.

**Definition 2.2** The bang interval of  $\hat{w}(t)$  denoted by  $\mathcal{I}_B$  is the time interval  $(t_0, t_f]$  in which  $|\hat{w}(t)| = M$  everywhere, and there is no other time interval  $(\hat{t}_0, \hat{t}_f]$  with the same property and  $(t_0, t_f] \subsetneq (\hat{t}_0, \hat{t}_f]$ .

Pang intervals and bang intervals are enumerated as  $\mathcal{I}_{P,k}$  and  $\mathcal{I}_{B,k}$ , respectively, using subscript  $k$  starting from zero. In addition, let us denote the initial time and the terminal time of the  $k$ th interval by  $t_{0,k}$  and  $t_{f,k}$ . Note that since the worst-case input is assumed to start from zero, the first interval is a pang interval which is the only *closed* pang interval, i.e.,  $\mathcal{I}_{P,1} = [t_{0,1}, t_{f,1}]$ .

We define pang and bang intervals to be half-open half-closed so as to conform with the assumption on definiteness mentioned in Section 2.1. Specifically, since the derivative of  $w(t) \in \mathcal{W}$  at an instant of discontinuity takes the same value as its left limit, this instant should belong to the left-

<sup>1</sup> $B \subsetneq A \iff B \subseteq A$  but  $B \neq A$

sided interval, *i.e.*, the foregoing interval. Hence, a pang or bang interval should include the terminal time instant but exclude the initial time instant.

### 2.3.1 Necessary Conditions of the Worst-Case Input

By the results in the preceding section, we may state the following theorems which characterize the necessary conditions of the worst-case input in pang and bang intervals. The early version of the necessary conditions in Theorems 2.1–2.3 first appeared in [5]. Similar theorems with some differences in notation and presentation style can be found in [6, 7]. Even so, the performance objective considered by Chang is the minimum terminal time not the maximum output magnitude. The similar necessary conditions but for the problem of maximizing the output magnitude of MIMO time-varying systems are given in [9]. Note that although these conditions were already obtained, we decided to give alternative proofs to make the presentation self-contained, and in particular, to reveal how these theorems can be immediately derived from the implications of the optimal control formulation. The alternative proofs are rather concise compared to the former versions. To begin with, a boundary condition for pang interval is examined. Then, the conditions on the signs of the worst-case-input magnitude and rate of change are derived.

**Theorem 2.1 (the boundary condition)** *In  $\mathcal{I}_{P,k}$  for  $k \neq 1$ , the backward step response at  $t_{0,k}$  and  $t_{f,k}$  are equal, that is,*

$$s(T - t_{f,k}) = s(T - t_{0,k}).$$

However, if  $k = 1$ , we have

$$s(T - t_{f,1}) = s(T) - p_{n+1}(0).$$

*Proof:* Recall that a pang interval is an interval where the constraint (2.9) is inactive, and hence  $\mu(t) = 0$ . From (2.17), we have, at  $t_2 = t_{f,k}$  and  $t_1 = t_{0,k}$ , the relation

$$p_{n+1}(t_{f,k}) - p_{n+1}(t_{0,k}) = s(T - t_{f,k}) - s(T - t_{0,k}). \quad (2.20)$$

Furthermore, recall also that a pang interval is connected to bang intervals where  $p_{n+1}(t) = 0$ , and although this pang interval is the last interval, we still have  $p_{n+1}(T) = 0$  as stated in (2.14). Hence, with the fact that  $p_{n+1}(t)$  must be continuous everywhere, it can be deduced that  $p_{n+1}(t_{f,k}) = p_{n+1}(t_{0,k}) = 0$  in  $\mathcal{I}_{P,k}$ . By substituting this into (2.20), the proof is complete.

For the case that  $k = 1$ , the proof can be established similarly except that  $p_{n+1}(t_{0,1})$  is not necessarily zero. Recalling that  $t_{0,1} = 0$ , from (2.17), we have, at  $t_2 = t_{f,1}$  and  $t_1 = 0$ ,

$$0 - p_{n+1}(0) = s(T - t_{f,1}) - s(T - 0),$$

which leads to the desired result.  $\square$

Figure 2.3 displays an example of the worst-case input satisfying the boundary condition in each pang interval. In particular, there are three pang intervals in this figure:  $(t_{0,1}, t_{f,1}]$ ,  $(t_{0,2}, t_{f,2}]$ , and  $(t_{0,3}, t_{f,3}]$ . The boundary conditions are fulfilled since  $s(T - t_{0,1}) = s(T - t_{f,1})$ ,  $s(T - t_{0,2}) = s(T - t_{f,2})$ , and  $s(T - t_{0,3}) = s(T - t_{f,3})$ .

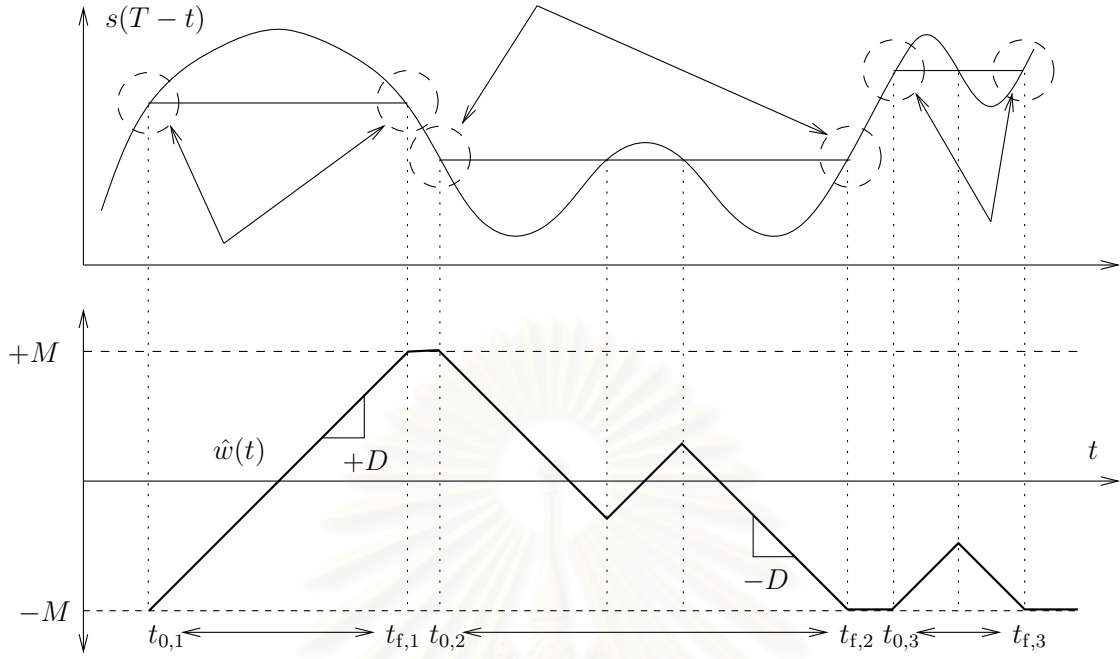


Figure 2.3: The boundary condition of the worst-case input.

**Theorem 2.2 (the derivative condition)** In  $\mathcal{I}_{P,k}$ , the rate of change of  $\hat{w}(t)$  is determined by

$$\dot{\hat{w}}(t) = \begin{cases} D \operatorname{sgn}[s(T-t) - s(T-t_{0,k})], & k \neq 1, \\ D \operatorname{sgn}[s(T-t) - s(T) + p_{n+1}(0)], & k = 1. \end{cases}$$

*Proof:* Consider the  $k$ th pang interval. With the same reasoning as in Theorem 2.1, the relation (2.17) at  $t_1 = t_{0,k}$  and  $t_2 = t \in (t_{0,k}, t_{f,k}]$  is reduced to

$$p_{n+1}(t) - p_{n+1}(t_{0,k}) = s(T-t) - s(T-t_{0,k}). \quad (2.21)$$

For  $k \neq 1$ , we have  $p_{n+1}(t_{0,k}) = 0$ ; hence, from (2.21), it follows that

$$p_{n+1}(t) = s(T-t) - s(T-t_{0,k}).$$

On the other hand, if  $k = 1$ , we have  $t_{0,1} = 0$  and that

$$p_{n+1}(t) = s(T-t) - s(T) + p_{n+1}(0).$$

Substituting the value of  $p_{n+1}(t)$  into (2.18) and restoring the notation  $u(t) = \dot{\hat{w}}(t)$  will finish the proof.  $\square$

The graphical meaning of this theorem is illustrated in Figure 2.4 for the same example of the worst-case input as in the previous theorem. Notice the theoretical relation between the input slope and the step response  $s(T-t)$ . For example, between  $t_{0,1}$  and  $t_{f,1}$ , we can see that  $s(T-t) > s(T-t_{0,1})$ ; hence,  $\dot{\hat{w}}(t) = +D$ . More complicated situation occurs during  $t_{0,2}$  and  $t_{f,2}$ . We have  $\dot{\hat{w}}(t) = +D$  when  $s(T-t) > s(T-t_{0,2})$ , while  $\dot{\hat{w}}(t) = -D$  when  $s(T-t) < s(T-t_{0,2})$ .

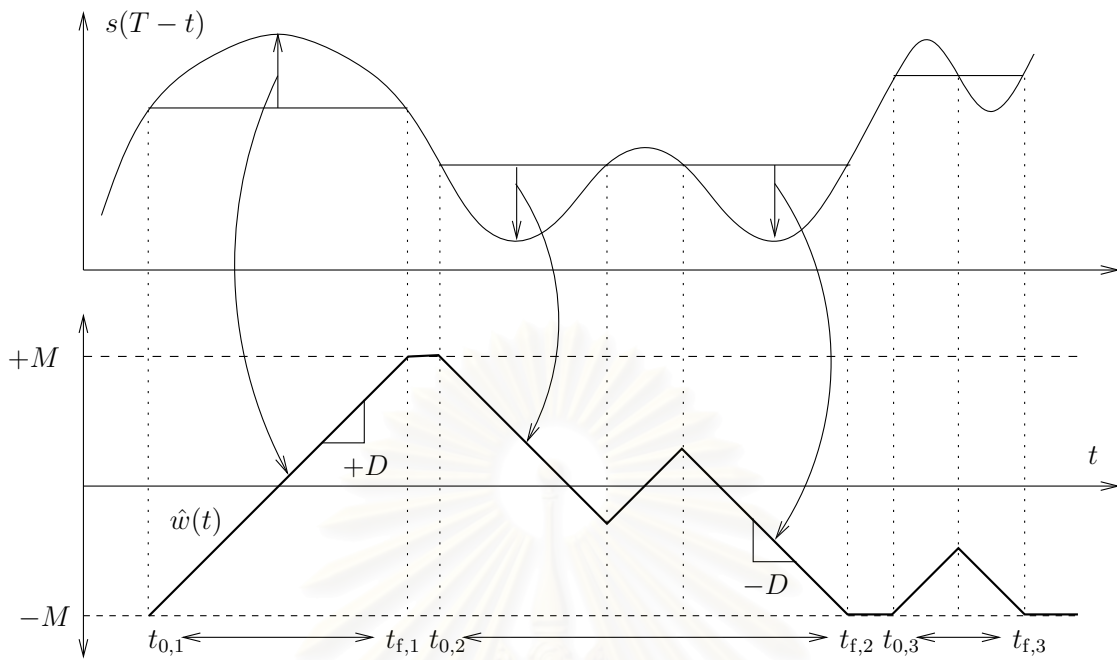


Figure 2.4: The derivative condition of the worst-case input.

**Remark 2.1** By means of Theorem 2.1, the rate of change of  $\hat{w}(t)$  in  $\mathcal{I}_{P,k}$  can also be written as

$$\dot{\hat{w}}(t) = D \operatorname{sgn}[s(T-t) - s(T-t_{f,k})]$$

for all  $k$ .

**Theorem 2.3 (the magnitude condition)** In  $\mathcal{I}_B$ , the magnitude of  $\hat{w}(t)$  is determined by

$$\hat{w}(t) = M \operatorname{sgn}[h(T-t)].$$

*Proof:* As mentioned in Section 2.2, the Lagrange multiplier  $\mu(t)$  in this interval cannot be zero, and thus, must take only positive value. Recalling that  $\hat{w}(t) = x_{n+1}(t)$ , the relation (2.19) then implies that

$$\operatorname{sgn}[\hat{w}(t)] = \operatorname{sgn}\left[\frac{h(T-t)}{2\mu(t)}\right] = \operatorname{sgn}[h(T-t)].$$

Since the magnitude of  $\hat{w}(t)$  in this interval should be at its limit  $M$ , we can see that  $\hat{w}(t) = M \operatorname{sgn}[h(T-t)]$ .  $\square$

The same example of the worst-case input is shown in Figure 2.5 to demonstrate the magnitude condition. For the sake of consistency, we still employ the step response in this figure, instead of the impulse response. Note that since  $h(T-t) = -ds(T-t)/dt$ , the sign of  $h(T-t)$  is opposite to that of the slope of  $s(T-t)$ .

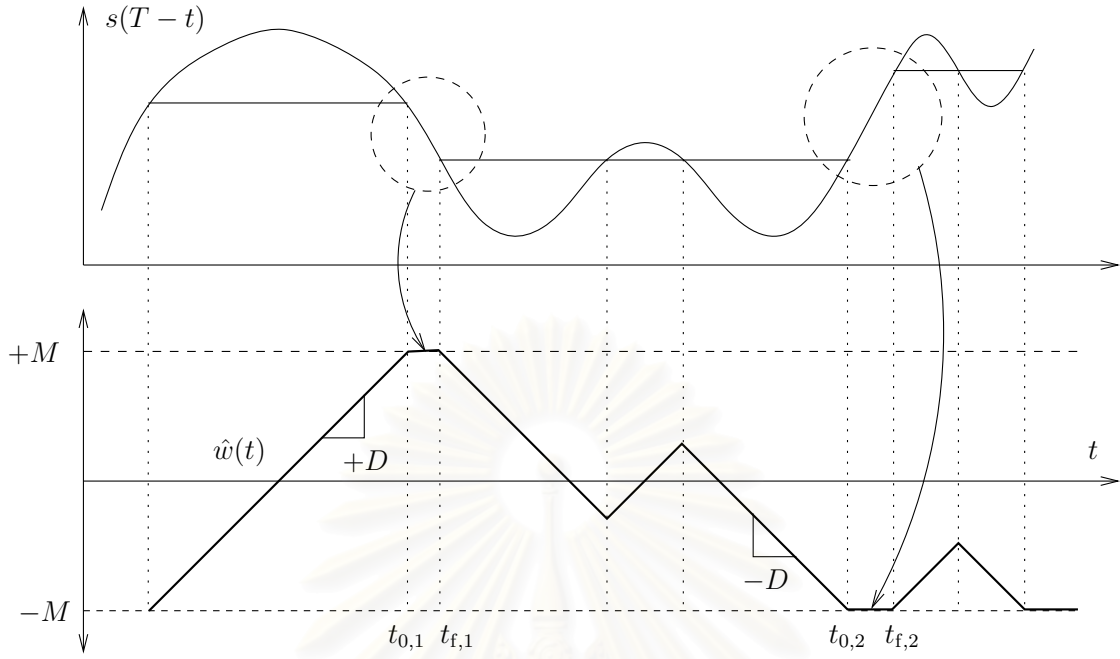


Figure 2.5: The magnitude condition of the worst-case input.

### 2.3.2 Sufficiency of the Necessary Conditions

Bongiorno Jr. has shown in [9] that the necessary conditions proposed by Chang in [7] are also sufficient conditions for an admissible input to be the worst-case input. The class of systems considered by Bongiorno consists of MIMO time-varying systems represented by state transfer matrices. Here, we will modify Bongiorno's proof in our particular time-invariant version by showing that any admissible input satisfying Theorems 2.1–2.3 is the worst-case input, *i.e.*, maximizes the objective cost  $z(T) = Cx(T)$ . Suppose  $\hat{w}(t)$  is an admissible input satisfying such theorems, and let  $\hat{x}(t)$  be its corresponding state trajectory. We have to show that

$$C[\hat{x}(T) - x(T)] \geq 0,$$

for all  $x(t)$  corresponding to  $w \in \mathcal{W}$ . This is equivalent to verifying that

$$\int_0^T h(T-t)[\hat{w}(t) - w(t)]dt \geq 0. \quad (2.22)$$

Therefore, we will decompose the left-handed side of (2.22) in each  $\mathcal{I}_{P,k}$  and  $\mathcal{I}_{B,k}$  of  $\hat{w}(t)$ .

#### Bang-Interval Analysis

In the bang interval  $\mathcal{I}_{B,k}$ , the magnitude of  $\hat{w}(t)$  is determined by Theorem 2.3. Thus, we have the integrand

$$h(T-t)[\hat{w}(t) - w(t)] = M|h(T-t)| - h(T-t)w(t) \geq 0, \quad \forall t \in \mathcal{I}_{B,k}.$$

This simply yields the relation

$$\int_{t_{0,k}}^{t_{f,k}} h(T-t)[\hat{w}(t) - w(t)]dt \geq 0.$$

### Pang-Interval Analysis

In the pang interval  $\mathcal{I}_{P,k}$ , the integral

$$\int_{t_{0,k}}^{t_{f,k}} h(T-t)[\hat{w}(t) - w(t)]dt$$

can be integrated by part as

$$- [s(T-t) - s(T-t_{f,k})][\hat{w}(t) - w(t)] \Big|_{t_{0,k}}^{t_{f,k}} + \int_{t_{0,k}}^{t_{f,k}} [s(T-t) - s(T-t_{f,k})][\dot{\hat{w}}(t) - \dot{w}(t)]dt. \quad (2.23)$$

Note that  $\int h(T-t)dt = -s(T-t) + c$  where  $c$  can be any constant. Here, we choose  $c = s(T-t_{f,k})$ .

The integration (2.23) can be rewritten as

$$[s(T-t_{0,k}) - s(T-t_{f,k})][\hat{w}(t_{0,k}) - w(t_{0,k})] + \int_{t_{0,k}}^{t_{f,k}} [s(T-t) - s(T-t_{f,k})][\dot{\hat{w}}(t) - \dot{w}(t)]dt. \quad (2.24)$$

Since the boundary condition in each pang interval is given by Theorem 2.1. The first term of (2.24) vanishes since  $s(T-t_{0,k}) = s(T-t_{f,k})$  for all  $k \neq 1$ . For the case  $k = 1$ , we have  $\hat{w}(t_{0,1}) = \hat{w}(0) = w(0) = w(t_{0,1}) = 0$ ; hence the first term of (2.24) also vanishes. Thus, the integration (2.24) is reduced to

$$\int_{t_{0,k}}^{t_{f,k}} [s(T-t) - s(T-t_{f,k})][\dot{\hat{w}}(t) - \dot{w}(t)]dt.$$

Since the rate of change of  $\hat{w}(t)$  is determined by Theorem 2.2, we have the integrand:

$$\begin{aligned} [s(T-t) - s(T-t_{f,k})][\dot{\hat{w}}(t) - \dot{w}(t)] &= D|s(T-t) - s(T-t_{f,k})| \\ &\quad - [s(T-t) - s(T-t_{f,k})]\dot{w}(t) \\ &\geq 0. \end{aligned}$$

Hence, it is easy to see that

$$\int_{t_{0,k}}^{t_{f,k}} h(T-t)[\hat{w}(t) - w(t)]dt = \int_{t_{0,k}}^{t_{f,k}} [s(T-t) - s(T-t_{f,k})][\dot{\hat{w}}(t) - \dot{w}(t)]dt \geq 0.$$

### 2.3.3 Uniqueness of the Worst-Case Input

The proof that the worst-case input satisfying the necessary and sufficient conditions is unique, is first presented in [9]. Actually, there is a condition that must be satisfied in order for the worst-case input to be unique. Our formulated problem satisfies such condition since there is a constraint on initial condition of admissible inputs, that is,  $w(0) = 0$ . This fact will be brought up later in the proof.

So as to verify the uniqueness, we assume that there is another worst-case input  $\tilde{w}(t)$  which satisfies Theorems 2.1–2.3, and its corresponding state trajectory is  $\tilde{x}(T)$ . We have already shown



that a combination of these theorems is equivalent to saying that  $\tilde{w}(t)$  induces the worst-case output. Thus, it can be easily seen that

$$C\hat{x}(T) = C\tilde{x}(T),$$

or similarly,

$$\int_0^T h(T-t)[\hat{w}(t) - \tilde{w}(t)]dt = 0.$$

By analyzing this integral separately in each interval:  $\mathcal{I}_{B,k}$  and  $\mathcal{I}_{P,k}$  of  $\hat{w}(t)$ , and using the same argument as in Section 2.3.2, we can say that

$$\begin{aligned} \hat{w}(t) &= \tilde{w}(t), & \text{in } \mathcal{I}_{B,k}, \\ \dot{\hat{w}}(t) &= \dot{\tilde{w}}(t), & \text{in } \mathcal{I}_{P,k}. \end{aligned}$$

Equivalently, it can be shown that

$$\hat{w}(t) = \tilde{w}(t), \quad \text{in } \mathcal{I}_{B,k}, \quad (2.25)$$

$$\hat{w}(t) = \tilde{w}(t) + c_k, \quad \text{in } \mathcal{I}_{P,k} \quad (2.26)$$

where  $c_k$  is a constant. Condition (2.25) suggests that the magnitude of  $\tilde{w}(t)$  is the same as that of  $\hat{w}(t)$  in the bang interval. In addition, from (2.26) and owing to the continuity of admissible inputs, and to the fact that a pang interval is connected to at least one bang interval, we have  $c_k = 0$ . Thus, the magnitude of  $\tilde{w}(t)$  is also the same as that of  $\hat{w}(t)$  in the pang interval. This concludes that  $\tilde{w} = \hat{w}$ .

The above argument is invalid if there is no bang interval of  $\hat{w}(t)$ , that is,  $\hat{w}(t)$  consists of only one pang interval. Therefore, in this case, we recall the fact that every admissible input starts from zero; hence,

$$0 = \hat{w}(0) = \tilde{w}(0) + c_k = 0 + c_k.$$

This means  $c_k = 0$ , and we still have  $\hat{w} = \tilde{w}$ .

### 2.3.4 Pang-bang Characteristics

While deriving the necessary and sufficient conditions for the worst-case input, we have assumed that the input considered is a member of the admissible input set  $\mathcal{W}$ . Nevertheless, the facts that an input be worst-case, and that it be admissible at the first place are totally different. To construct the worst-case input in practice, additional constraints which represent  $\mathcal{W}$  should be posed. We will refer to these constraints as the *admissibility constraints*.

Recall that the worst-case input  $\hat{w}(t)$  satisfies pang-bang characteristic, namely, its magnitude in bang intervals is determined through Theorem 2.3, and its derivative in pang intervals is obtained through Theorem 2.2. Nevertheless, there has been no guarantee so far, during the construction process, that the magnitude of  $\hat{w}(t)$  in *pang intervals* would also satisfy the magnitude bound. This leads to the requirement of the admissibility constraints. In the later section, it will be shown that this constraint plays an important role in searching for positions of pang intervals of the worst-case

input. In order to state such constraint, it is convenient to define certain shorthand notations which will facilitate further discussion.

As mentioned previously, it can be seen that a pang interval is composed of one or several segment(s) with the slope of either  $+D$  or  $-D$ . We will denote each of these segments as a *pang subinterval*. As a consequence, a pang interval can then be classified into two types according to the number of pang subintervals inside it. The detailed definitions are as follows.

**Definition 2.3** *A pang subinterval inside a pang interval  $\mathcal{I}_P$  is the time interval  $\mathcal{I}$  in which  $\dot{w}(t) = +D$  everywhere or  $\dot{w}(t) = -D$  everywhere, and there is no other time interval  $\tilde{\mathcal{I}}$  with the same property and  $\mathcal{I} \subsetneq \tilde{\mathcal{I}}$ .*

Recall that the assumption regarding the values of input derivative at the instants of discontinuity (see Section 2.1), suggests that these values should equal the left-sided limit. Hence, a pang subinterval itself (which does not start at zero) is half-open half-closed, akin to a pang interval.

**Definition 2.4** *An odd pang interval of  $\hat{w}(t)$  is a pang interval inside which the number of pang subinterval(s) is odd, while an even pang interval of  $\hat{w}(t)$  is a pang interval inside which the number of pang subintervals is even.*

**Definition 2.5** *Let the time instants  $t_{1,k}, \dots, t_{m,k}$  be all possible instants within  $\mathcal{I}_{P,k}$  such that  $t_{0,k} < t_{1,k} < \dots < t_{m,k} < t_{f,k}$ , and*

$$p_{n+1}(t_{1,k}) = p_{n+1}(t_{2,k}) = \dots = p_{n+1}(t_{m,k}) = 0.$$

*We will refer to these instants as switching instants.*

Between adjacent switching instants, the multiplier  $p_{n+1}$  has different sign, hence the name switching instants. For consistency, sometimes we will refer to the subscript  $f$  as  $m + 1$ . Note that  $m + 1$  is odd in an odd pang interval and it is even in an even pang interval. Immediately, a brief lemma analogous to Theorem 2.1 is obtained as follows.

**Lemma 2.1** *At the switching instants  $t_{1,k}, \dots, t_{m,k}$  in  $\mathcal{I}_{P,k}$ ,*

$$s(T - t_{1,k}) = \dots = s(T - t_{m,k}) = s(T - t_{f,k}).$$

*Moreover, if  $k \neq 1$ , the backward step responses at these instants are equal to  $s(T - t_{0,k})$ .*

*Proof:* Consider the condition (2.17) in pang interval when  $t_1 = t_{i,k}$ ,  $t_2 = t_{j,k}$  where  $i, j = 1, \dots, m$ . Since  $p_{n+1}(t_{i,k}) = p_{n+1}(t_{j,k}) = 0$ , it is straightforward that

$$s(T - t_{i,k}) = s(T - t_{j,k}).$$

Next, consider again the condition (2.17) in  $\mathcal{I}_{P,k}$  when  $t_1 = t_{f,k}$  and  $t_2 = t_{i,k}$ , where  $i = 1, \dots, m$ . As discussed in Theorem 2.1,  $p_{n+1}(t_f)$  is 0 for every  $\mathcal{I}_{P,k}$  even though it is the last interval. Since  $p_{n+1}(t_{i,k}) = 0$  for  $i = 1, \dots, m$  by its definition, we have  $s(T - t_{i,k}) = s(T - t_{f,k})$ . In addition, if

$k \neq 1$ , through Theorem 2.1, we also have  $s(T - t_{i,k}) = s(T - t_{0,k})$   $\square$

For each  $\mathcal{I}_{P,k}$ , let the length of each pang subinterval, *i.e.*,  $t_{i,k} - t_{i-1,k}$ , be denoted by  $\Delta t_{i,k}$  for  $i = 1, \dots, m+1$ . Another useful terms can be defined as follows.

**Definition 2.6** In  $\mathcal{I}_{P,k}$ , the cumulative polar-summation of pang subinterval's length, or in short, the cumulative summation  $\theta_{1,k}, \dots, \theta_{m+1,k}$  is defined as

$$\theta_{i,k} \triangleq \sum_{j=1}^i (-1)^{j+1} \Delta t_{j,k}, \quad i = 1, \dots, m+1.$$

Recall that  $\theta_{m+1,k}$  can also be regarded as  $\theta_{f,k}$ . The term *polar-summation* signifies that the sign (pole) of  $\Delta t_{j,k}$  in the definition of  $\theta_{i,k}$  alternates at each  $j$ .

Now, we are ready to state the conditions on the admissibility constraints. Although the admissibility constraints are developed for construction of the worst-case input, the nature of the constraints themselves can be applied to the class of all pang-bang signals in  $\mathcal{W}$ .

**Theorem 2.4 (admissibility constraints)** For  $k \neq 1$ , consider  $\mathcal{I}_{P,k}$ . Suppose that  $|\hat{w}(t_{f,k})| = M$ , *i.e.*,  $\mathcal{I}_{P,k}$  terminates when the worst-case-input magnitude is at its limit<sup>2</sup>. The following conditions must hold:

(i) At each switching instant,

$$0 \leq \theta_{i,k} \leq \frac{2M}{D}, \quad i = 1, \dots, m. \quad (2.27)$$

(ii) At the terminal time  $t_{f,k}$ ,

$$\theta_{f,k} = \begin{cases} \frac{2M}{D}, & \text{if } \mathcal{I}_{P,k} \text{ is odd,} \\ 0, & \text{if } \mathcal{I}_{P,k} \text{ is even.} \end{cases} \quad (2.28)$$

Otherwise, in  $\mathcal{I}_{P,1}$ , the following conditions must hold:

(iii) At each switching instant,

$$-\frac{M}{D} \leq \theta_{i,1} \leq \frac{M}{D}, \quad i = 1, \dots, m. \quad (2.29)$$

(iv) At the terminal time  $t_{f,1}$ ,

$$\theta_{f,1} = \pm \frac{M}{D}. \quad (2.30)$$

---

<sup>2</sup>This assumption holds for every pang interval (except when  $\mathcal{I}_{P,k}$  is the last interval of the worst-case input) because a pang interval, which is not the last interval, must be followed by a bang interval.

*Proof:* (i) From Theorem 2.2 when  $k \neq 1$ , we have  $\dot{\hat{w}}(t) = D\text{sgn}[s(T-t) - s(T-t_{0,k})]$ . Thus, the worst-case input at each switching instant  $t_{i,k}$  can be straightforwardly computed as

$$\begin{aligned}
\hat{w}(t_{i,k}) &= \int_{t_{0,k}}^{t_{i,k}} \dot{\hat{w}}(t) dt + \hat{w}(t_{0,k}) \\
&= \int_{t_{0,k}}^{t_{i,k}} D\text{sgn}[s(T-t) - s(T-t_{0,k})] dt + \hat{w}(t_{0,k}) \\
&= D \int_{t_{0,k}}^{t_{i,k}} \text{sgn}[s(T-t) - s(T-t_{0,k})] dt + \hat{w}(t_{0,k}) \\
&= D \sum_{j=1}^i \alpha_{j,k} \Delta t_{j,k} + \hat{w}(t_{0,k})
\end{aligned} \tag{2.31}$$

where  $\alpha_{j,k}$  is the sign of  $s(T-t) - s(T-t_{0,k})$  in the pang subinterval  $(t_{j,k}, t_{j-1,k}]$ .

If  $\hat{w}(t_{0,k}) = -M$ , then  $\dot{\hat{w}}(t)$  should equal  $D$  in the first pang subinterval of  $\mathcal{I}_{P,k}$ ; otherwise, the input  $\hat{w}(t)$  would have gone below  $-M$  in this pang subinterval, excluded itself from the collection  $\mathcal{W}$ . Hence, from Theorem 2.2, the sign of  $s(T-t) - s(T-t_{0,k})$  should be positive, in the first pang subinterval, to match that of  $\dot{\hat{w}}(t)$ . This implies that  $\alpha_{1,k} = 1$ . Furthermore, in later pang subintervals,  $s(T-t)$  crosses over the level of  $s(T-t_{0,k})$  at each switching instant  $t_{j,k}$ , causing  $\alpha_{j,k}$  to alternate along  $\mathcal{I}_{P,k}$ . For this reason, the relation (2.31) becomes

$$\hat{w}(t_{i,k}) = D\theta_{i,k} - M. \tag{2.32}$$

Since  $|\hat{w}(t_{i,k})|$  is bounded by  $M$ , we have

$$-M \leq D\theta_{i,k} - M \leq M,$$

which simply yields the condition (2.27).

If  $\hat{w}(t_{0,k}) = +M$ , it can be shown in similar manner that  $\alpha_{1,k} = -1$ , and the sequels will alternate their signs. In this particular case, the relation (2.31) can be expressed as

$$\hat{w}(t_{i,k}) = -D\theta_{i,k} + M. \tag{2.33}$$

Then, the bounding condition on  $\hat{w}(t_{i,k})$  is imparted to  $\theta_{i,k}$ , which results in (2.27) as well.

(ii) The terminal condition (2.28) in a pang interval  $\mathcal{I}_{P,k}$  can be derived in a straightforward fashion. Suppose  $\mathcal{I}_{P,k}$  is odd and  $\hat{w}(t_{0,k}) = -M$ . This means  $\hat{w}(t_{f,k}) = -\hat{w}(t_{0,k}) = M$  because the number of pang subintervals is odd. By replacing the subscript  $i$  by  $f$  in (2.32), we have

$$\hat{w}(t_{f,k}) = D\theta_{f,k} - M = M,$$

and the condition (2.28) can be immediately obtained. If  $\hat{w}(t_{0,k}) = M$ , we alternatively applied (2.33) in this case, and get

$$\hat{w}(t_{f,k}) = -D\theta_{f,k} + M = -M,$$

which yields the same result.

The terminal condition of  $\theta_{f,k}$  where  $\mathcal{I}_{P,k}$  is even can be deduced simply with the equivalent manner, *i.e.*, with the fact that  $\hat{w}(t_{f,k}) = \hat{w}(t_{0,k})$ , so the proof is left to the reader.

(iii) From Theorem 2.2 when  $k = 1$ , the derivative  $\dot{\hat{w}}(t)$  is equal to  $D\text{sgn}[s(T-t) - s(T) + p_{n+1}(0)]$ . Recall that the initial condition  $\hat{w}(t_{0,k}) = \hat{w}(0) = 0$  holds for any input in  $\mathcal{W}$ . Thus, the worst-case input at each switching instant  $t_{i,1}$  is as follows:

$$\begin{aligned}
 \hat{w}(t_{i,1}) &= \int_0^{t_{i,1}} \dot{\hat{w}}(t) dt \\
 &= \int_0^{t_{i,1}} D\text{sgn}[s(T-t) - s(T) + p_{n+1}(0)] dt \\
 &= D \int_0^{t_{i,1}} \text{sgn}[s(T-t) - s(T) + p_{n+1}(0)] dt \\
 &= D \sum_{j=1}^i \alpha_{j,1} \Delta t_{j,1}
 \end{aligned} \tag{2.34}$$

where  $\alpha_{j,1}$  is the sign of  $s(T-t) - s(T) + p_{n+1}(0)$  in the pang subinterval  $(t_{1,1}, t_{0,1}]$ . Here,  $\alpha_{1,1}$  can be either plus or minus, depending on the starting direction of  $\hat{w}(t)$ . Hence, from (2.34), the linear relation between  $\hat{w}(t_{i,1})$  and  $\theta_{i,1}$  is of the form

$$\hat{w}(t_{i,1}) = \pm D\theta_{i,1}. \tag{2.35}$$

Therefore, with a little algebras, the magnitude bound on  $\hat{w}(t)$  is conveyed to  $\theta_i$  as in (2.29).

(iv) The terminal condition (2.30) in the first pang interval immediately follows from the previous result. From (2.35), replacing the subscript  $i$  with  $f$ , we have

$$\pm D\theta_{f,1} = \hat{w}(t_{f,1}) = \pm M,$$

which gives (2.30). □

It is important to note that the admissibility constraints together with Theorem 2.1–2.3 provide only necessary conditions for an interval to be a pang interval of the worst-case input. There may be some interval which is not a pang interval of the worst-case input yet satisfies Theorem 2.1–2.3 and Theorem 2.4. Henceforth, we refer to any time interval fulfilling these theorems as a *plausible pang interval* of the worst-case input. In addition, if a plausible pang interval is not actually a pang interval of the worst-case input, then it is called a *spurious pang interval*. In accordance with these nomenclatures, a pang interval of the worst-case input may sometimes be mentioned as an *actual pang interval* of the worst-case input.

For better insight, an example when a spurious pang interval coincides with an actual pang intervals are given. Consider Figure 2.6 where  $s(T-t)$  is assumed to be nonincreasing before  $t_1$  and nondecreasing after  $t_7$ . Given that  $t_4 - t_1 = 2M/D$  and  $t_7 - t_3 = t_3 - t_2$ , it can be readily verified that both  $\bar{\mathcal{I}}_{P,k}$  and  $\mathcal{I}_{P,k}$  satisfy Theorem 2.1–2.3 and Theorem 2.4. Nonetheless, we will show that  $\bar{\mathcal{I}}_{P,k}$  is a spurious pang interval of the worst-case input. First, recall that since the terminal time of  $\bar{\mathcal{I}}_{P,k}$  is  $t_4$ , the connected bang interval, says  $\bar{\mathcal{I}}_{B,k}$ , has the initial time equal  $t_4$ . Second, wherever the

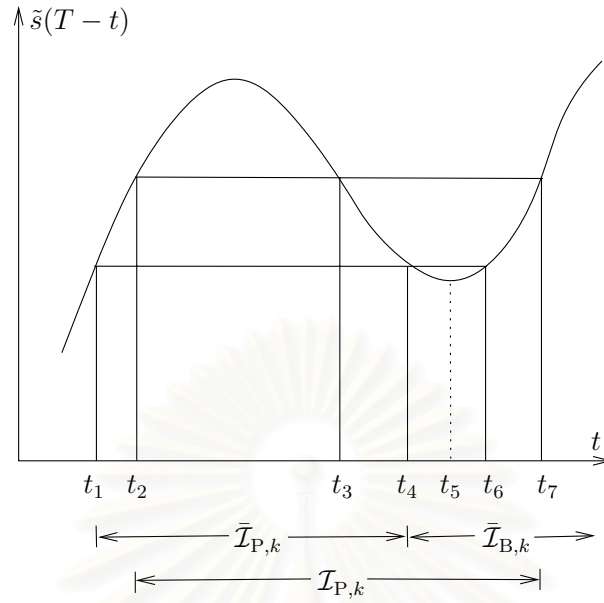


Figure 2.6: The spurious pang interval  $\bar{\mathcal{I}}_{P,k}$  precedes the actual pang interval  $\mathcal{I}_{P,k}$ . If SPIS proceeds to identify  $\bar{\mathcal{I}}_{P,k}$  to be a pang interval, SPIS will be stuck at  $t_4$  because the succeeding bang interval that has  $t_4$  as an initial time cannot be constructed. In this case, SPIS will note down this spurious pang interval and move on to discover  $\mathcal{I}_{P,k}$  instead.

next pang interval  $\mathcal{I}_{P,k+1}$  is, its initial time, which is the terminal time of  $\bar{\mathcal{I}}_{B,k}$ , must be located after  $t_5$ . This is because  $t_6 - t_4$  is too narrow to contain a pang interval.

In short, we are pointing out that the bang interval  $\bar{\mathcal{I}}_{B,k}$  starts before  $t_5$  and ends after  $t_5$ . However, Theorem 2.3 specifies that the sign of  $h(T - t)$  must be the same along the entire bang interval. In contrast, at  $t_5$  the slope of  $s(T - t)$  changes its sign, and  $h(T - t)$  does either. Thus,  $\bar{\mathcal{I}}_{B,k}$  is invalid, which consequently concludes that  $\bar{\mathcal{I}}_{P,k}$  is a spurious pang interval. In this example,  $\mathcal{I}_{P,k}$  is the actual pang interval. As mentioned before, the admissibility constraints will be used in the process of constructing the worst-case input. Accordingly, the possible existence of a spurious pang interval must be carefully taken into account.

## 2.4 Summary

We have proposed analysis of the WCN of finite-dimensional convolution systems where the WCN is defined as the maximum of output magnitude produced by any admissible inputs with magnitude bound and rate limit. Once the formulation of the WCN is simplified, we employ the optimal control scheme to derive the necessary and sufficient conditions of the worst-case input. Then, the admissibility constraints are developed. The worst-case input characterization together with this admissibility constraints will become handy in the next chapter to construct the worst-case input and compute the WCN, respectively.

## CHAPTER III

### COMPUTATION OF THE WCN OF LINEAR SYSTEMS

#### 3.1 Construction of the Worst-Case Input

We start with the main result of this Chapter—the practical method for obtaining the worst-case input, and respectively, the WCN. Although a few methods have already been proposed, yet only the result by Lane [19] can be considered as workable. He listed the necessary and sufficient conditions, and stipulated a few rules that lead to the construction of the worst-case input. Still, his method includes the complicate use of an auxiliary graphical element called a *switching function*, which makes the procedure rather involved. Our work is different from the algorithm in [19] in that we have made no use of this switching function.

The classical method developed by Birch and Jackson [5] is a graphical procedure that starts from the first trial of a candidate for the worst-case input. Then, some adjustments on this candidate are made by means of graphical inspection of the considered step response. A series of trials followed by adjustments may arise until the worst-case input is obtained. However, the flaw in their necessary and sufficient conditions, and the deficiency in establishing the systematic procedure (as described in Section 1.2.1) prevent possible extension of the work to general cases.

##### 3.1.1 Successive Pang Interval Search

The algorithm presented in this dissertation is partially based on that of Birch and Jackson's. The existence and uniqueness of the solution to the problem of determining the worst-case input (discussed in Section 2.1) guarantees that our algorithm in pursuit of the worst-case input will eventually succeed. In other words, the convergence of the algorithm is guaranteed. In order to determine the shape of the worst-case input for a finite dimensional convolution system we need to examine graphically the step response of such system via Theorems 2.1–2.4. Since the computation is implemented on a digital computer, to treat this problem numerically, the step response and also the impulse response must be sampled at an adequate rate  $1/\tau_s$ . It will be discussed later how this sampling rate affects the computational accuracy of the WCN. Given the appropriate terminal time  $T$ , we choose  $\tau_s$  so that  $T$  is a multiple of which. The standard fact is that  $n_s = (T/\tau_s) + 1$  is the amount of sampling instants. Let  $\tau_1, \tau_2, \dots, \tau_{n_s}$  stand for these instants for which  $\tau_1 < \tau_2 < \dots < \tau_{n_s}$ . It is clear that we have  $\tau_1 = 0$  and  $\tau_{n_s} = T$ .

For fixed  $i$  where  $1 \leq i < n_s$ , consider  $t \in [\tau_i, \tau_{i+1}]$ . Let  $\tilde{s}(T-t)$  be a piecewise-linear estimate of the backward step response  $s(T-t)$  of a finite-dimensional convolution system. Precisely, it is

defined as follows:

$$\begin{aligned}\tilde{s}(T) &\triangleq s(T), \\ \tilde{s}(T-t) &\triangleq s(T-\tau_i) + \left(\frac{t-\tau_i}{\tau_s}\right) \Delta s(T-\tau_i), \quad \tau_i < t \leq \tau_{i+1}, \quad i = 1, \dots, n_s - 1.\end{aligned}\quad (3.1)$$

where  $\Delta s(T-\tau_i) = s(T-\tau_{i+1}) - s(T-\tau_i)$  for  $i = 1, \dots, n_s - 1$ . Roughly speaking, this definition simply says that  $\tilde{s}(T-\tau_i) = s(T-\tau_i)$  for  $i = 1, \dots, n_s$ , while  $\tilde{s}(T-t)$  between contiguous sampling instants is obtained via interpolation. It should be noted that the scope of this dissertation is confined to solely finite-dimensional convolution systems, which involve smooth impulse responses, and hence, step responses. According to this fact,  $s(T-t)$  cannot be static over a finite period, which means that it is numerically improbable for  $\tilde{s}(T-t)$  to be equal at two adjacent sampling instants. Therefore, we will assume that  $\tilde{s}(T-\tau_i) \neq \tilde{s}(T-\tau_{i+1})$  for all  $i = 1, \dots, n_s - 1$ . We now consider the extrema of  $\tilde{s}(T-t)$ .

**Definition 3.1** Let the peak times of  $\tilde{s}(T-t)$  be denoted by  $\tau_{p_1}, \tau_{p_2}, \dots, \tau_{p_{n_p}}$  where  $n_p$  is the total number of peak times and  $p_i \in \{1, 2, \dots, n_s\}$ . The peak times are defined as all possible time instants with the property such that  $\tau_{p_1} < \tau_{p_2} < \dots < \tau_{p_{n_p}}$ , and

$$\begin{aligned}\tau_{p_1} &= 0, \quad \tau_{p_{n_p}} = T, \\ \tilde{s}(T-\tau_{p_i}) &> \max\{\tilde{s}(T-\tau_{p_{i-1}}), \tilde{s}(T-\tau_{p_{i+1}})\}, \quad \text{or} \\ \tilde{s}(T-\tau_{p_i}) &< \min\{\tilde{s}(T-\tau_{p_{i-1}}), \tilde{s}(T-\tau_{p_{i+1}})\}, \quad i = 2, \dots, n_p - 1.\end{aligned}$$

In other words, the peak time are the time instants where  $\tilde{s}(T-t)$  is at its zenith or nadir, including zero and  $T$ . Notice that although the above inequalities are strict, this definition is still valid. This is because the aforementioned assumption suggests that there is no adjacent pair of sampling instants that achieve the same value of  $\tilde{s}(T-t)$ . Assume as well that  $\tau_s$  is sufficiently small so that there is more than one sampling instant between adjacent peak times, *i.e.*,  $p_i - p_{i-1} > 2$ . In terms of the peak times, we can divide the whole time interval  $[0, T]$  into  $n_p - 1$  disjoint segments as follows:

$$\begin{aligned}\pi_1 &\triangleq \{t : 0 \leq t \leq \tau_{p_2}\}, \\ \pi_i &\triangleq \{t : \tau_{p_i} < t \leq \tau_{p_{i+1}}\}, \quad i = 2, \dots, n_p - 1.\end{aligned}$$

An example of  $\tilde{s}(T-t)$  partitioned into segments with respect to its peak times is delineated in figure 3.1. Let the *current time*  $t_c$  be the time instant of the present consideration, and similarly let  $\pi_c$  stand for the *current segment* which is the segment containing  $t_c$ . Moreover, let the former time be denoted by  $t_{c-}$ , and be defined as the past time instant before we move to  $t_c$ . Note in advance that most of the time, but not at all times,  $t_c$  (or  $t_{c-}$ ) equals one of the sampling instants  $\tau_i$ . Up to this point, we are now ready to outline our algorithm. The reader may, at the same time, consider a flow chart in figure 3.2 for better understanding. Some substantial steps will later be explained in details.

**Step 0 (data structuring)** Compute the suitable terminal time  $T$  via simple *bisection algorithm*.

Sample the step response  $s(T-t)$  to get  $\tilde{s}(T-t)$  with the sampling period  $\tau_s$ . Obtain all peak times of  $\tilde{s}(T-t)$  by direct comparison among  $\tilde{s}(T-\tau_i)$  for  $i = 1, \dots, n_s$ , along the time axis.



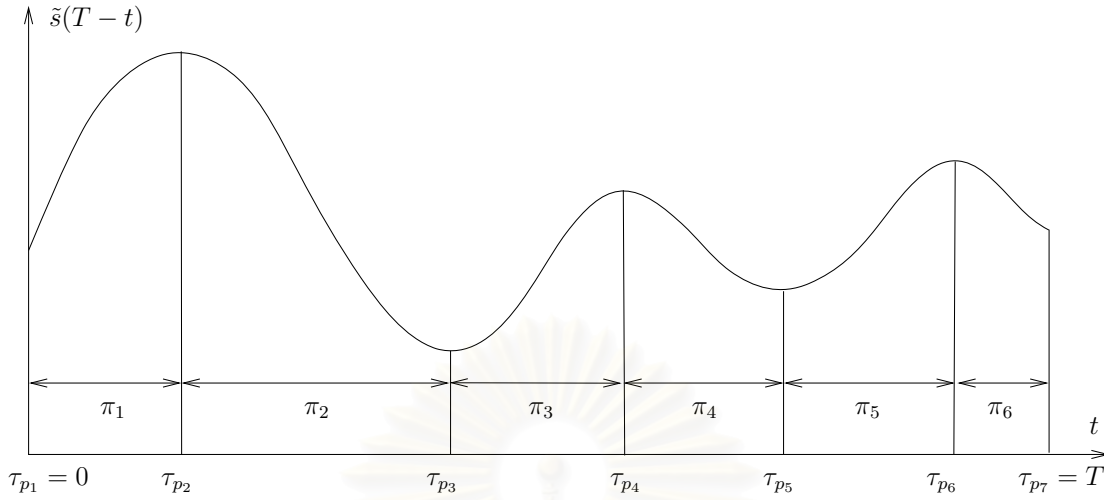


Figure 3.1: The segments  $\pi_1, \dots, \pi_6$  of  $\tilde{s}(T-t)$  partitioned by the corresponding peak times  $\tau_{p_1}, \dots, \tau_{p_7}$ .

**Step 1 (initialization)** Obtain all plausible first pang intervals  $\mathcal{I}_1$  of the worst-case input<sup>1</sup>. Then, obtain all plausible last pang intervals  $\mathcal{I}_L$  of the worst-case input. These are carried out with respect to the admissibility constraints in Theorem 2.4. Let  $\mathfrak{J}_1$  denote the collection of the former intervals, and  $\mathfrak{J}_L$  denote the collection of the latter.

Enumerate the members of  $\mathfrak{J}_1$ , and begin with the first element of  $\mathfrak{J}_1$  (denoted by  $\mathcal{I}_1^1$  in figure 3.2). Suppose that the terminal time  $t_{f,1}$  of  $\mathcal{I}_1^1$  falls into a segment  $\pi_f$ . Set  $t_c$  to be  $t_{f,1}$ , and set  $\pi_c$  to be  $\pi_f$ . Now we can start the search for the next pang interval.

**Step 2 (trial experiment)** If  $\pi_c$  is the last segment  $\pi_{n_p-1}$ , we have proceeded to the last interval of the worst-case input which is, in this situation, a bang interval occupying  $(t_c, T]$ . Go to Step 5 to determine the shape of the worst-case input in each interval, and to compute the worst-case output eventually. Otherwise, if  $\pi_c$  is the segment to which any initial time of  $\mathcal{I}_L \in \mathfrak{J}_L$  (denoted by  $t_{0,L}$  in figure 3.2) belongs, then check whether or not  $t_c$  is less than  $t_{0,L}$ . When this condition is fulfilled, the interval  $\mathcal{I}_L$ , among other members of  $\mathfrak{J}_L$ , is the actual pang interval of the worst-case input. Also, go to Step 5 to finalize the worst-case input construction.

If the above cases do not apply, check whether  $t_c$  is a peak time. If so, go to Step 4, otherwise, proceed with the search by collecting a piece of information at  $t_c$ . This information is needed in examining a plausible location of the next pang interval. In so doing, we conduct a quick experiment: presuming that  $t_c$  is an initial time of a pang interval, and computing the cumulative summation at this position. It is not unusual that this trial cumulative summation does not meet the admissibility constraints (conditions (i) and (ii) in Theorem 2.4). In fact, to meet the exact time instant which is the initial time of the next pang interval is nearly impossible.

After getting the experiment result, compare it with the former result at  $t_{c-}$ , if there is any.

<sup>1</sup>Recall that the first interval must be a pang interval since admissible inputs should start from  $w(0) = 0$ . In addition,  $\mathcal{I}_1$  and  $\mathcal{I}_{P,1}$  are of the same meaning, but we use  $\mathcal{I}_1$  here for consistency with  $\mathcal{I}_L$  that denotes a plausible last pang interval.

This tells us whether or not we have crossed an initial time  $t_{0,k}$  of a plausible pang interval  $\mathcal{I}_{P,k}$ , while moving from  $t_{c-}$  to  $t_c$ . If so, get to Step 3 to determine the precise location of this recently-discovered plausible pang interval. If there is no such crossing, store the current experiment result for future comparison; update  $t_c$  to be the next sampling instant<sup>2</sup>. Also, update  $t_{c-}$  and repeat this step over again.

**Step 3 (precise location)** If we have crossed an initial time  $t_{0,k}$  of a plausible pang interval  $\mathcal{I}_{P,k}$ , the algorithm enters this step to precisely locate  $\mathcal{I}_{P,k}$ . In particular, we have to determine the initial time  $t_{0,k}$ , the terminal time  $t_{f,k}$ , and all the switching instants  $t_{i,k}$  of  $\mathcal{I}_{P,k}$ . Recall that the current time  $t_c$  is situated in the current segment  $\pi_c$ . As a result of the experiment in Step 2, additional useful data that we gain are as follows: the segment  $\pi_f$  into which  $t_{f,k}$  falls, the time instants closely around  $t_{0,k}$ , and those around  $t_{f,k}$ . With these data, we can adopt the standard bisection algorithm to find a more precise location of  $\mathcal{I}_{P,k}$ .

Before moving back to Step 2, store the current time for future purpose. This stored time instant is called the *restarting instant* which is represented by  $t_{rst}$  in figure 3.2. If one or more restarting instants already exist then we append the new restarting instant to the set of the old ones. Then, similar to Step 1, update the current segment by setting  $\pi_c = \pi_f$ . Set  $t_c$  to be the terminal time  $t_{f,k}$ .

**Step 4 (search correction)** When the algorithm enters this step, it implies that one or more previously-discovered plausible pang intervals are spurious. Thus, delete the latest location of the plausible pang interval, return to the latest restarting instant, and go back over to Step 2.

However, if the latest plausible pang interval is in  $\mathcal{J}_1$ , nothing is left when this interval is deleted, and there is no more restarting instant. In this case, we move to the next element of  $\mathcal{J}_1$ , and update  $\pi_c$  and  $t_c$  in accordance with the routine in the Step 1. Then, go over to Step 2.

**Step 5 (finalization)** Entering this step, the algorithm has successfully determined the pattern of pang and bang intervals. The final task of constructing the worst-case input is to determine its magnitudes in bang intervals and its rates of change in pang intervals. This is readily done by following Theorems 2.2–2.3, and hence the construction of the worst-case input  $\hat{w}(t)$  is finished. The WCN (the worst-case output) can be simply calculated by numerical integration of the convolution  $h(t) * \hat{w}(t)$ , and the algorithm terminates. Note that such numerical integration is performed at  $\tau_1, \dots, \tau_{n_s}$  plus the corner of  $\hat{w}(t)$ , *i.e.*, the points of discontinuity of  $\hat{w}(t)$ .

In brief, the first step initializes the algorithm, and then the search for locations of pang interval is accomplished in Step 2. The locations of pang intervals are accurately determined in Step 3. The algorithm may terminate after Step 1 or Step 3. The correction for any spurious pang interval is carried out in Step 4. As we finish locating all the actual pang intervals, the bang intervals are just the other intervals in between them. The algorithm then finalizes the computation in Step 5.

---

<sup>2</sup>Perhaps, this should be called the nearest sampling instant ahead since  $t_c$  itself may not be the sampling instant.

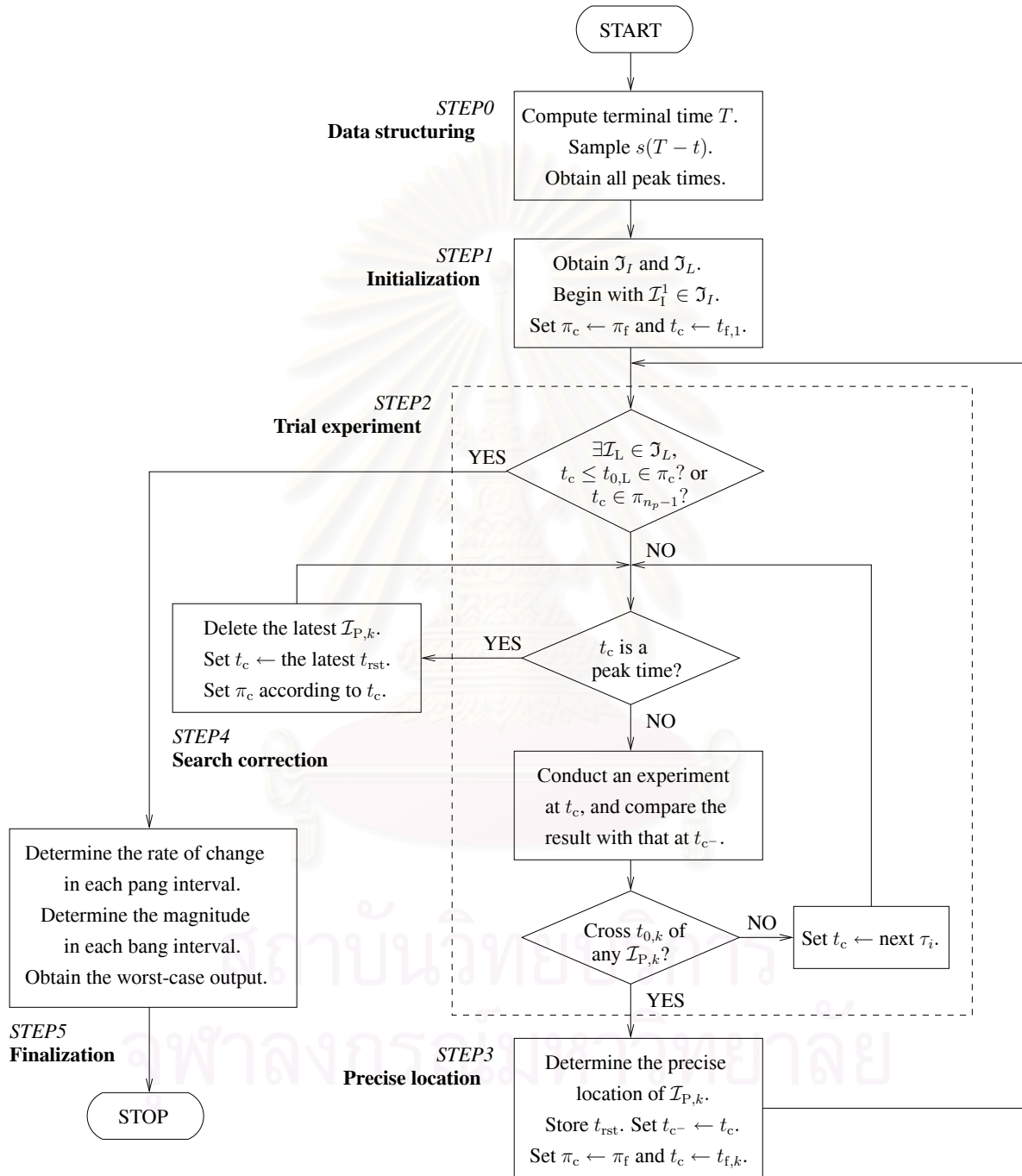


Figure 3.2: The flow chart of the successive pang interval search (SPIS) algorithm.

Apparently, this algorithm moves from one sampling instant to another to locate the positions of pang intervals of the worst-case input. Owing to this distinct feature, we will refer to this algorithm as the *Successive Pang-Interval Search* algorithm or briefly as SPIS. The following subsections furnish the detailed descriptions in some steps. We attempt to arrange these details not in chronological order but according to their priority. This would assist the reader to gradually form a clear insight into the main idea of our algorithm.

### 3.1.2 Suitable Terminal Time

The choice of the terminal time  $T$  can be selected in terms of the truncation difference defined as the difference between  $\|h\|_{\text{wc}}$  and  $\xi(T)$ . However, this difference cannot exactly be determined because, in most cases, we cannot obtain the exact value of  $\|h\|_{\text{wc}}$ . Thus, it needs to utilize the upper bound of  $\|h\|_{\text{wc}} - \xi(T)$  instead. Note that we need not to write  $|\|h\|_{\text{wc}} - \xi(T)|$  since  $\|h\|_{\text{wc}} - \xi(T)$  is positive. From (2.4) and (2.5), it is easy to see that  $\|h\|_{\text{wc}} - \xi(T)$  can be bounded by  $M \int_T^\infty |h(t)| dt$  (see Appendix A). Still, this quantity cannot be efficiently computed in practice due to the need for improper numerical integration.

To determine a computable bound of  $\|h\|_{\text{wc}} - \xi(T)$ , assume that the convolution system  $h(t)$  has finite dimension equal  $n$  with the minimal realization  $(A, B, C)$ . Since the left-shifted impulse response  $h(t + T) = Ce^{At}e^{AT}B$  can be viewed as an auxiliary impulse response  $h_T(t)$  with a new realization  $(A, e^{AT}B, C)$ , a bound of  $\|h\|_{\text{wc}} - \xi(T)$  is derived as follows [50].

$$\|h\|_{\text{wc}} - \xi(T) \leq M \int_T^\infty |h(t)| dt = M \int_0^\infty |h_T(t)| dt \leq 2M \sum_{i=1}^n \sigma_i(T) \quad (3.2)$$

where  $n$  is the system order, and  $\sigma_i(T)$  is the Hankel singular value of  $h_T(t)$ . Since the Hankel singular values of  $h_T(t)$  are the square roots of the eigenvalues of the product of the observability and controllability gramians of  $(A, e^{AT}B, C)$ , this bound can be readily calculated by solving two Lyapunov equations together with eigenvalue computation. For convenience, let us denote the bound on the truncation difference  $\|h\|_{\text{wc}} - \xi(T)$  as

$$e(T) \triangleq 2M \sum_{i=1}^n \sigma_i(T). \quad (3.3)$$

Since  $e(T)$  is computed in terms of  $\sigma(T)$ , the bound  $e(T)$  itself depends upon the truncating time instant  $T$ , which gives rise to the need for the argument  $T$ .

In real application, we want to specify  $e(T)$  that yields an adequately large terminal time  $T$ . This task can be eased off by using the standard fact that, for each  $i$ , the Hankel singular value  $\sigma_i(T)$  of  $h_T(t)$  does not increase as  $T$  increases. Equivalently, the bound (3.2) is the nonincreasing function of  $T$ . Thus, general exact line search methods, such as bisection algorithm, suffice to accomplish this task.

Here, our bisection algorithm on continuous-time systems is based on the scheme, first proposed by Balakrishnan and Boyd [51], that finds the smallest discrete instant which yields the spec-

ified value of the bound for  $l_1$ -norm of the *tail* of the impulse response of the discrete-time finite-dimensional convolution system with the desired degree of accuracy.

Suppose that we need to keep  $\|h\|_{wc} - \xi(T)$  below the value  $\epsilon_1$ . It suffices to find  $T$  that gives

$$e(T) \leq \epsilon_1.$$

To start our bisection algorithm, we need to specify valid upper and lower limits of  $T$ . Let these limits be denoted by  $T_{up}$  and  $T_{low}$ , respectively. In other words, we must have  $e(T_{up}) < \epsilon_1$  and  $e(T_{low}) > \epsilon_1$ . Since  $T$  may take any positive value,  $T_{low}$  must be set to zero. For  $T_{up}$ , we start from any specific time instant and increase it twice until  $e(T_{up}) < \epsilon_1$ . The detailed algorithm is as follows:

**begin**

$T_{up} := 1;$

**while**  $e(T_{up}) \geq \epsilon_1$  **do**

$T_{up} \leftarrow 2T_{up};$

**end;**

**end.**

One may use this  $T_{up}$  as the terminal time since it holds already that

$$\|h\|_{wc} - \xi(T_{up}) < e(T_{up}) < \epsilon_1$$

However, it is preferable to find a good estimate of the minimum  $T$  such that  $e(T) \leq \epsilon_1$ . This is because unnecessarily large  $T$  requires more sampling instants so as to retain the same sampling precision, which may consequently increase the overall computation time. This leads to the bisection algorithm which iterates to find  $T_{up}$  and  $T_{low}$  satisfying the following termination criteria:

$$e(T_{low}) - e(T_{up}) < \epsilon_2$$

where  $\epsilon_2$  is sufficiently less than  $\epsilon_1$ . In this work, we use  $\epsilon_2 = 10^{-5}\epsilon_1$ . Using  $T_{up}$  and  $T_{low}$  obtained previously, we now describe our bisection algorithm:

**begin**

**while**  $e(T_{low}) - e(T_{up}) \geq \epsilon_2$  **do**

$T \leftarrow (T_{up} + T_{low})/2;$

**if**  $e(T) < \epsilon_1$  **then**

$T_{up} \leftarrow T;$

**else**

$T_{low} \leftarrow T;$

**end;**

**end;**

$T \leftarrow T_{up};$

**end.**

Before terminating the algorithm, despite insignificant difference, we set  $T$  to be  $T_{up}$  instead of the usual average of  $T_{up}$  and  $T_{low}$  since  $e(T_{up}) < \epsilon_1$  while  $e((T_{up} + T_{low})/2)$  may not.

### 3.1.3 Detection of Crossing the Initial Time of a Plausible Pang Interval

The major idea of SPIS is contained in Step 2: moving from one point to another while gathering some information through an experiment, and using this information to roughly determine the location of plausible pang intervals. In this section, we consider any plausible pang interval except the first and the last one. It is noted that this step shares some similar concepts as the *adjustment* step of Birch and Jackson's procedure when it is facing the *difficulty* [5]. The trial experiment can be described as follows.

Suppose that the current segment and the current time are  $\pi_c$  and  $t_c \neq 0$ , respectively. The experiment attacks a simple question: is the current time  $t_c$  likely to be an initial time of a plausible pang interval? To test this presumption, we project a reference level, with the magnitude of  $\tilde{s}(T - t_c)$ , outward from zero along the time axis. Then, all segments in which  $\tilde{s}(T - t)$  crosses over this reference level are determined. Let these segments be referred to as the *cutting segments* and be denoted by  $\pi'_1, \dots, \pi'_{n_c}$  where  $n_c$  is the total number of crossings. Let  $\Pi'$  be the set containing all cutting segments at the current time. Then, the following time instants at which  $\tilde{s}(T - t)$  passes over the reference line are defined.

**Definition 3.2** *The cutting instants, with respect to  $t_c$ , are all time instants  $t'_i$  such that  $t_c < t'_1 < \dots < t'_{n_c}$ , and*

$$\tilde{s}(T - t'_i) = \tilde{s}(T - t_c), \quad i = 1, \dots, n_c$$

Since  $\tilde{s}(T - t)$  is monotonic in each segment, it is obvious that there is only one cutting instant in each segment. For consistency, we may refer to  $t_c$  as the zeroth cutting instant  $t'_0$  and  $\pi_c$  as the zeroth cutting segment  $\pi'_0$ . Furthermore, it is numerically reasonable to assume that all cutting instants do not exactly coincide with any peak times. Figure 3.3 shows how the cutting instants and the corresponding cutting segments at  $t_c$  are determined by the reference level  $\tilde{s}(T - t_c)$ . To obtain the cutting instants in practice, we compare  $\tilde{s}(T - t)$  with  $\tilde{s}(T - t_c)$  at each sampling instant and pick out only  $\tau_i$  such that  $\tau_i > t_c$  and

$$\tilde{s}(T - \tau_i) < \tilde{s}(T - t_c) < \tilde{s}(T - \tau_{i+1}).$$

Then, the corresponding cutting instant  $t'_i$  can be acquired via simple interpolation:

$$t'_i = \tau_i + \tau_s \left[ \frac{\tilde{s}(T - t_c) - \tilde{s}(T - \tau_i)}{\tilde{s}(T - \tau_{i+1}) - \tilde{s}(T - \tau_i)} \right].$$

Before we state the main theorem, let investigate some attributes of cutting segments and cutting instants while moving from one point to another. For ease of subsequent exposition, the cutting segments are divided into two classes by the following definition.

**Definition 3.3** *With respect to the current segment  $\pi_c$ , an odd segment is any cutting segment  $\pi'_i$  of which  $i$  is an odd number. Similarly, an even segment is any cutting segment  $\pi'_i$  of which  $i$  is an even number.*

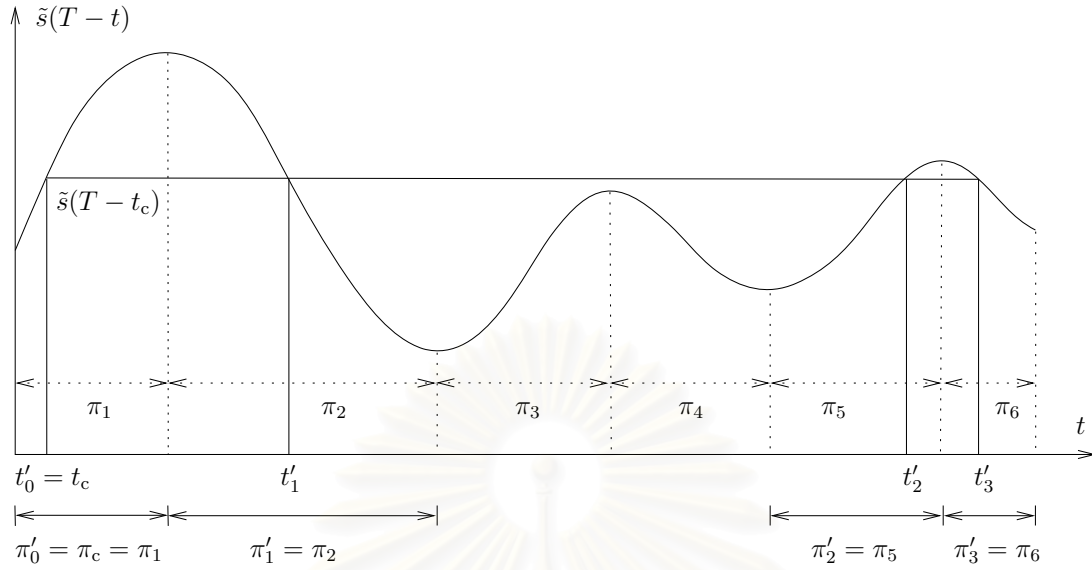


Figure 3.3: The cutting instants  $t'_0, \dots, t'_3$  and the cutting segments  $\pi'_0, \dots, \pi'_3$  of  $\tilde{s}(T-t)$  obtained at  $t_c$ .

Apparently, an odd segment alternates with an even one. In addition, it is observed that  $\tilde{s}(T-t)$  is monotonic in each segment. Thus, we prefer to distinguish between segments with upward direction and downward direction.

**Definition 3.4** A segment  $\pi_i$  is called an up segment if  $\tilde{s}(T-t)$  is monotonically increasing on it, that is,

$$\tilde{s}(T-t_2) > \tilde{s}(T-t_1), \quad \forall t_1, t_2 \in \pi_i, t_1 < t_2.$$

A segment  $\pi_i$  is called a down segment if  $\tilde{s}(T-t)$  is monotonically decreasing on it, that is,

$$\tilde{s}(T-t_2) < \tilde{s}(T-t_1), \quad \forall t_1, t_2 \in \pi_i, t_1 < t_2.$$

Recall that  $\tilde{s}(T-t)$  is not equal at any contiguous sampling instants. Thus, we can use the strict inequalities in this definition because there is no such case that  $\tilde{s}(T-t)$  takes the same value in the same segment. Notice that an up segment also alternates with a down segment. Hence, we arrive at this straightforward proposition.

**Proposition 3.1** The direction of an even segment is the same as that of the current segment, but the direction of an odd segment is opposite.

Consider the former time instant  $t_{c-}$ . Let the cutting segments at  $t_{c-}$  be represented by  $\pi''_j$ , and the cutting instants by  $t''_j$  for  $j = 0, \dots, n_{c-}$ . Likewise, let  $\Pi''$  signify the set of all cutting segments obtained at  $t_{c-}$ . If a common element between  $\Pi'$  and  $\Pi''$  exists, then there is a simple relationship between the cutting instants, obtained at  $t_c$  and at  $t_{c-}$ , on this element. However, the condition under which such relationship can take place may be relaxed from the sole equality of  $\pi'_i$  and  $\pi''_j$ . This relaxed condition is a kind of equivalence relation defined as follows.

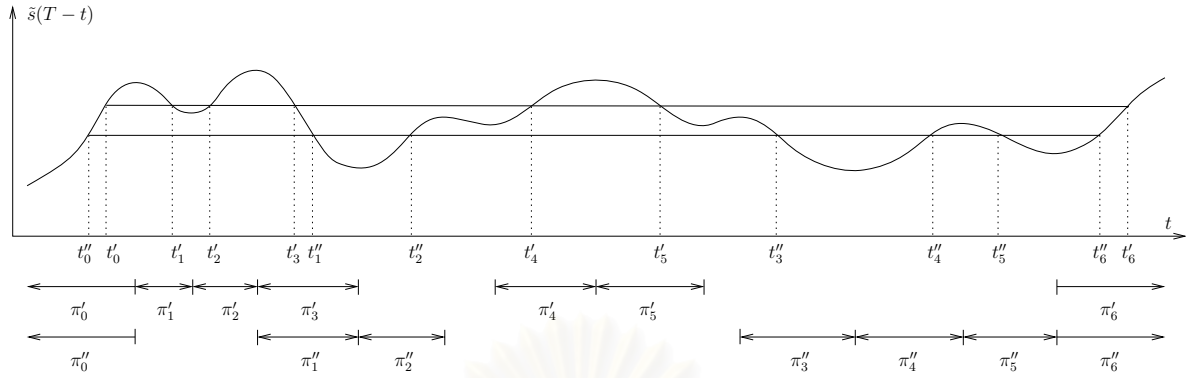


Figure 3.4: The cutting instants  $t'_0, \dots, t'_6$  and the cutting segments  $\pi'_0, \dots, \pi'_6$  of  $\tilde{s}(T-t)$  obtained at  $t_c (= t'_0)$ , and the cutting instants  $t''_0, \dots, t''_6$  and the cutting segments  $\pi''_0, \dots, \pi''_6$  of  $\tilde{s}(T-t)$  obtained at  $t_{c-} (= t''_0)$ .

**Definition 3.5** Segments  $\pi'_i \in \Pi'$  and  $\pi''_j \in \Pi''$  are said to be equivalent, or  $\pi'_i \equiv \pi''_j$ , if there is no element of  $\Pi'$  or  $\Pi''$  situated between  $\pi'_i$  and  $\pi''_j$ , or situated at  $\pi'_i$  or  $\pi''_j$  (except  $\pi'_i$  and  $\pi''_j$  themselves). In addition, we will refer to a duplet  $(\pi'_i, \pi''_j) \in \Pi' \times \Pi''$  as an equivalent pair.

Apparently, the equality  $\pi'_i = \pi''_j$  is a special case of  $\pi'_i \equiv \pi''_j$ . The meaning of this equivalence is that if there is any segment between  $\pi'_i$  and  $\pi''_j$  on which  $\tilde{s}(T-t)$  does not cross the levels  $\tilde{s}(T-t'_0)$  and  $\tilde{s}(T-t''_0)$ , then we may ignore it from our consideration. For more insight, let us consider figure 3.4 which shows the cutting instants and cutting segments obtained at  $t_c (= t'_0)$  and  $t_{c-} (= t''_0)$ . In this figure,  $\pi'_0 = \pi''_0 = \pi_c$ , and we have four equivalent pairs:

$$\begin{aligned} (\pi'_3, \pi''_1) & \text{ where } \pi'_3 = \pi''_1, \\ (\pi'_4, \pi''_2) & \text{ where } \pi'_4 \neq \pi''_2, \\ (\pi'_5, \pi''_3) & \text{ where } \pi'_5 \neq \pi''_3, \\ (\pi'_6, \pi''_6) & \text{ where } \pi'_6 = \pi''_6. \end{aligned}$$

Notice that, in this example, there are some  $\pi'_i$  that are not equivalent to any element in  $\Pi''$ , namely,  $\pi'_1, \pi'_2$ , and conversely, there are also some  $\pi''_j$  that are not equivalent to any element in  $\Pi'$ , namely,  $\pi''_4, \pi''_5$ . We found that an equivalent pair  $(\pi'_i, \pi''_j)$  possesses certain properties the same as which two equal cutting segments possess. Before we proceed to see this, let assume here that  $\pi_c$  is an up segment. This causes no loss of generality since if  $\pi_c$  is a down segment of  $\tilde{s}(T-t)$ , then it is an up segment of  $-\tilde{s}(T-t)$ . Thus, since all cutting instants obtained with  $-\tilde{s}(T-t)$  are the same as those with  $\tilde{s}(T-t)$ , the structure of the proof when  $\pi_c$  is a down segment is analogous. The following lemma exhibits a few properties of an equivalent pair.

**Lemma 3.1** Suppose that  $\pi'_0 = \pi''_0 = \pi_c$ <sup>3</sup>, and  $\pi'_i \equiv \pi''_j$  but  $\pi'_i \neq \pi''_j$ . Then the following are equivalent:

<sup>3</sup>To see the nature of this assumption, refer to the main algorithm, SPIS. If we are directed to Step 2 after finishing from Step 2 itself (to repeat the step), in this case,  $t_c$  and  $t_{c-}$  are still in the same segment, i.e.,  $\pi'_0 = \pi''_0 = \pi_c$ . However, if we enter Step 2 after finishing from Steps 1, 3, or 4, such time instants may not fall into the same segment. Nevertheless,



- (i)  $\pi'_i$  precedes  $\pi''_j$ , i.e.,  $\forall t_1 \in \pi'_i, \forall t_2 \in \pi''_j, t_1 < t_2$ .
- (ii)  $\pi'_i$  and  $\pi''_j$  are odd segments.

For the case that  $\pi'_i = \pi''_j$ ,  $\pi'_i$  is an odd segment if and only if  $\pi''_j$  is. Moreover, the following conditions hold:

- (iii) If  $i, j$  are odd, then  $t'_i < t''_j$ .
- (iv) If  $i, j$  are even, then  $t'_i > t''_j$ .

*Proof:* See the Appendix B.

As an example, consider again the equivalent pairs in figure 3.4. It is seen that, as 2 and 4 are even numbers, we have  $\pi''_2$  precedes  $\pi'_4$ , and hence,  $t''_2 < t'_4$ . In addition, as 3 and 5 are odd numbers, we have  $\pi'_5$  precedes  $\pi''_3$ , and hence  $t'_5 < t''_3$ . The immediate consequence of Lemma 3.1 is that a cutting segment in  $\Pi'$  can be equivalent to at most only one cutting segment in  $\Pi''$  and vice versa. This is presented as follows.

**Corollary 3.1 (uniqueness of equivalence)** *Suppose that  $\pi'_0 = \pi''_0 = \pi_c$ . The equivalent pair  $(\pi'_{i_1}, \pi''_{j_1})$  is unique. That is, if  $\exists \pi'_{i_1} \in \Pi', \exists \pi''_{j_1} \in \Pi''$  such that  $\pi'_{i_1} \equiv \pi''_{j_1}$ , then  $\forall \pi'_{i_2} \in \Pi', \pi'_{i_2} \not\equiv \pi''_{j_1}$  if  $\pi'_{i_2} \neq \pi'_{i_1}$ , and similarly,  $\forall \pi''_{j_2} \in \Pi'', \pi''_{j_2} \not\equiv \pi'_{i_1}$  if  $\pi''_{j_2} \neq \pi''_{j_1}$ .*

*Proof:* See the Appendix B.

Informally, Corollary 3.1 tells that the equivalence relation is one-to-one. Thus, let us match element of  $\Pi'$  with element of  $\Pi''$ , only those possible, in terms of this relation. Assume also that this matching starts from  $(\pi'_0, \pi''_0)$  and proceeds until it reaches a specific equivalent pair  $(\pi'_a, \pi''_b)$ . Let a set  $\Phi_{ab} \subset \Pi' \times \Pi''$  contain those matched pairs. Arrange elements of  $\Phi_{ab}$  with the direction outwards from zero. Again, we make use of an example in figure 3.4. We will include the equivalent pair  $(\pi'_6, \pi''_6)$  and all equivalent pairs before it, that is,  $a = b = 6$ . In this case,

$$\Phi_{66} = \{(\pi'_0, \pi''_0), (\pi'_3, \pi''_1), (\pi'_4, \pi''_2), (\pi'_5, \pi''_3), (\pi'_6, \pi''_6)\}. \quad (3.4)$$

We have already remarked that some cutting segments in  $\Pi'$  may not match with those in  $\Pi''$ , and vice versa. For instance, such segments are  $\pi'_1$  and  $\pi'_2$  in  $\Pi'$  and  $\pi''_3$  and  $\pi''_4$  in  $\Pi''$ . Intuitively, if there is any cutting segment left between two contiguous pairs  $(\pi'_i, \pi''_j)$  and  $(\pi'_k, \pi''_l)$ , then it should belong to  $\Pi'$  only, or to  $\Pi''$  only, but not both. As in figure 3.4,  $\pi'_1$  and  $\pi'_2$  both are in  $\Pi'$  and are the only segments between the equivalent pairs  $(\pi'_0, \pi''_0)$  and  $(\pi'_3, \pi''_1)$ . This claim is verified in the following lemma.

---

it does not matter that  $t_c$  and  $t_{c-}$  be in the same segment in these cases because if we have just moved from Steps 1, 3, or 4, it means we are about to draw the first trial of the next plausible pang interval, and neither cutting instants nor cutting segments at  $t_{c-}$  play any role.

**Lemma 3.2** Suppose that  $\pi'_0 = \pi''_0 = \pi_c$ , and  $\exists \pi'_a \in \Pi'$ ,  $\exists \pi''_b \in \Pi''$  such that  $\pi'_a \equiv \pi''_b$ . Assume that  $(\pi'_i, \pi''_j)$  and  $(\pi'_k, \pi''_l)$  are two contiguous equivalent pairs in  $\Phi_{ab}$  such that  $(\pi'_k, \pi''_l)$  precedes  $(\pi'_i, \pi''_j)$ , that is,  $\forall t_1 \in \pi'_k \cup \pi''_l, \forall t_2 \in \pi'_i \cup \pi''_j, t_1 < t_2$ . Then,

- (i)  $i, j$  are odd if and only if  $k, l$  are even.
- (ii) if  $i, j$  are even ( $k, l$  are odd), then  $i = k + 1$ .
- (iii) if  $i, j$  are odd ( $k, l$  are even), then  $j = l + 1$ .

*Proof:* See the Appendix B.

A specific implication of this lemma appears explicitly in (3.4). It is easy to show that the members of  $\Phi_{66}$  satisfy Lemma 3.2. Refer back to the beginning of this subsection that  $t_c$  is presumed to be an initial time of a plausible pang interval. As a result, the cutting instants  $t'_i$  are consistently presumed to be switching instants. By this idea, it needs to examine whether each cutting instant satisfies condition (i) in Theorem 2.4. Hence, by imitating  $\theta_{i,k}$  in the previous section, we will define another decisive quantity as follows. At each  $t'_i$  for  $i = 1, \dots, n_c$ , the *trial cumulative summation*  $\theta'_i$  is

$$\theta'_i \triangleq \sum_{m=1}^i (-1)^{m+1} \Delta t'_m, \quad i = 1, \dots, n_c$$

where  $\Delta t'_m = t'_m - t'_{m-1}$ . By means of condition (i) in Theorem 2.4, we can rule out some implausible cutting instants which yield unacceptable  $\theta'_i$ . In particular, if at any  $t'_i$ , we have  $\theta'_i > 2M/D$  or  $\theta'_i < 0$ , then we can drop the cutting instants from  $i + 1$  onwards<sup>4</sup>. Let  $\theta''_j$  stand for the trial cumulative summation acquired at cutting instants  $t''_j$  corresponding to the former time instant  $t_{c-}$ . We are now in the position to state the main results of this section.

**Proposition 3.2 (monotonicity)** Suppose that  $\pi'_0 = \pi''_0 = \pi_c$ , and  $\pi'_a \equiv \pi''_b$ . Then,

$$\theta'_a < \theta''_b. \quad (3.5)$$

*Proof:* See the Appendix B.

Roughly speaking, this proposition says that the trial cumulative summation is a decreasing function of searching steps. In other words, as we progress forwards from one time instant to another, the cumulative summation acquired at each cutting segment is decreasing. The particular example in figure 3.4 will deepen the reader's understanding. Consider the case that  $a = b = 6$ . By examining the position of each cutting instants, it is easy to see that  $\theta'_6 < \theta''_6$ . Proposition 3.2 also yields another beneficial result. Consider  $\theta'_i$  or  $\theta''_j$  which  $\pi'_i \in \Pi'$  only or  $\pi''_j \in \Pi''$  only. These trial cumulative summations can be bounded by either of those corresponding to the near by  $\pi'_i, \pi''_j$  where  $(\pi'_i, \pi''_j) \in \Phi_{ab}$ .

<sup>4</sup>We cannot drop out  $t'_i$  because there may be the case that  $\theta'_i \approx 2M/D$  but  $\theta'_i > 2M/D$ , or the case that  $\theta'_i \approx 0$  but  $\theta'_i < 0$ .

**Corollary 3.2** Suppose that  $\pi'_0 = \pi''_0 = \pi_c$ . Let  $(\pi'_i, \pi''_j)$  and  $(\pi'_k, \pi''_l)$  be two contiguous equivalent pairs in  $\Phi_{ab}$  such that  $(\pi'_k, \pi''_l)$  precedes  $(\pi'_i, \pi''_j)$ . Then,

(i) if  $i, j$  are even ( $k, l$  are odd), then

$$\theta''_j > \theta'_i, \quad \hat{j} = l + 1, \dots, j. \quad (3.6)$$

(ii) if  $i, j$  are odd ( $k, l$  are even), then

$$\theta'_i < \theta''_j, \quad \hat{i} = k + 1, \dots, i. \quad (3.7)$$

*Proof:* See the Appendix B.

As we have formed a hypothesis that  $t_c$  be an initial time of a plausible pang interval, it follows that we must find a suitable terminal time which satisfies the admissibility constraints. Here, we will employ the trial cumulative summation in testing whether any cutting instant is eligible to be a terminal time of such pang interval. The following main theorem furnishes conditions for the experiment to examine whether  $t_c$  can actually be associated with the initial time of a plausible pang interval. If so, it follows where it is roughly the terminal time.

**Theorem 3.1** There exists a plausible  $k$ th pang interval,  $(t_{0,k}, t_{f,k}]$ , for which  $t_{c-} \leq t_{0,k} \leq t_c$ , if the following conditions hold:

(i) There exists the instants  $t'_a, t''_b$  ( $a, b \neq 0$ ) falling in the same segment:

$$\pi'_a = \pi''_b, \quad (3.8)$$

(ii) The cutting instants through  $t'_a$  and through  $t''_b$  must yield plausible trial cumulative summations:

$$\begin{aligned} 0 \leq \theta'_i &\leq \frac{2M}{D}, \quad i = 1, \dots, a - 1, \\ 0 \leq \theta''_j &\leq \frac{2M}{D}, \quad j = 1, \dots, b - 1. \end{aligned} \quad (3.9)$$

(iii) There is a sign change in terms of the trial cumulative summations while moving from  $t_{c-}$  to  $t_c$ :

$$\begin{aligned} \text{sgn} \left[ \theta'_a - \frac{2M}{D} \right] &= -\text{sgn} \left[ \theta''_b - \frac{2M}{D} \right], \quad \text{if } a, b \text{ are odd,} \\ \text{sgn} \theta'_a &= -\text{sgn} \theta''_b, \quad \text{if } a, b \text{ are even.} \end{aligned} \quad (3.10)$$

Furthermore, we have

$$\begin{aligned} t'_a \leq t_{f,k} \leq t''_b &\quad \text{if } a, b \text{ are odd,} \\ t''_b \leq t_{f,k} \leq t'_a &\quad \text{if } a, b \text{ are even.} \end{aligned}$$

*Proof:* We have to show that there is a time interval  $(t_{0,k}, t_{f,k}]$  for which  $t_{c-} \leq t_{0,k} \leq t_c$  that satisfies Theorem 2.4. Consider any  $t_0^*$  in  $[t_0'', t_0'] (= [t_{c-}, t_c])$ . Let  $t_\nu^*$  for  $\nu = 1, \dots, d$  be cutting instants obtained when the initial time is at  $t_0^*$ . Note that  $t_d^*$  is the last most cutting instant, which belongs to  $\pi_a' (= \pi_b'')$ . Let the cutting segments  $\pi_\nu^* \in \Pi^*$  and the trial cumulative summation  $\theta_\nu^*$  be corresponding to  $t_\nu^*$  for  $\nu = 1, \dots, d$ .

First we have to determine a suitable choice of  $t_0^*$  for the cases that  $a, b$  are odd and that  $a, b$  are even. Suppose that  $a, b$  are odd numbers. Since  $t_0'' \leq t_0^* \leq t_0'$ , and  $\pi_0'' = \pi_0' = \pi_0^*$ , by Proposition 3.2, we have

$$\theta_a' \leq \theta_d^* \leq \theta_b''. \quad (3.11)$$

Then, define a function  $\theta(t) : [t_0'', t_0'] \rightarrow [\theta_b'', \theta_a']^5$  to be  $\theta_d^*$  when treating  $t$  as  $t_0^*$ . Since  $\tilde{s}(T - t)$  is continuous on  $[t_0'', t_0']$ , the function  $\theta(t)$  is also continuous on  $[t_0'', t_0']$ . In addition, it is obvious that  $\theta(t_0'') = \theta_b''$  and  $\theta(t_0') = \theta_a'$ . Hence, the function  $\theta(t)$  maps  $[t_0'', t_0']$  onto  $[\theta_b'', \theta_a']$ .

Furthermore, by condition (iii) when  $a, b$  are odd, and by Proposition 3.2, it is readily seen that

$$\theta_a' < \frac{2M}{D} < \theta_b''. \quad (3.12)$$

According to (3.11), (3.12), and that  $\theta(t)$  is onto, there must exist  $t \in [t_0'', t_0']$  where  $\theta(t) = 2M/D$ . Let this  $t$  be our choice of  $t_0^*$ . For the case that  $a, b$  are even, define the same function  $\theta(t)$  and follow similar reasoning. By condition (iii) when  $a, b$  are even, and by Proposition 3.2, the inequalities (3.12) is changed to

$$\theta_a' < 0 < \theta_b''. \quad (3.13)$$

This also yields a good candidate of  $t_0^*$ .

Now we claim that this  $t_0^*$  is an initial time of a plausible pang interval  $\mathcal{I}_{P,k}$  where  $(t_{\nu-1}^*, t_\nu^*]$  can be compared to a pang subinterval of  $\mathcal{I}_{P,k}$ . By Lemma 3.1, if  $a, b$  are odd, then so is  $d$ . This suggests that  $\mathcal{I}_{P,k}$  is odd. As mentioned previously,  $t_0^*$  is chosen so that  $\theta_d^* = 2M/D$ . Letting  $m = d - 1$ ,  $t_{f,k} = t_d^*$ , and  $\theta_{f,k} = \theta_d^*$ , condition (ii) of Theorem 2.4 (condition on the cumulative summation at the terminal time of  $\mathcal{I}_{P,k}$ ; see Section 2.3) is partially fulfilled. Another part is that when  $a, b$  are even. Then,  $d$  and  $\mathcal{I}_{P,k}$  are even. Since  $\theta_d^* = 0$  in this case, by letting  $\theta_{f,k} = \theta_d^*$ , condition (ii) of Theorem 2.4 is now fulfilled.

Recall that  $t_d^*$  is attained in  $\pi_d' (= \pi_a'' = \pi_b'')$ . Since  $t_0'' \leq t_0^* \leq t_0'$ , we have  $\tilde{s}(T - t_0'') \leq \tilde{s}(T - t_0^*) \leq \tilde{s}(T - t_0')$ , or equivalently  $\tilde{s}(T - t_b'') \leq \tilde{s}(T - t_d^*) \leq \tilde{s}(T - t_a')$ . Because  $\tilde{s}(T - t)$  is monotonic in each segment,  $t_d^*$  should lie between  $t_a'$  and  $t_b''$ . If  $a, b, d$  are odd, then, by Lemma 3.1,  $t_a' \leq t_{f,k} \leq t_b''$ . In contrast, if  $a, b, d$  are even, then  $t_b'' \leq t_{f,k} \leq t_a'$ . These verify (3.11).

The rest is to establish the satisfaction of condition (i) of Theorem 2.4, which is the cumulative summation at each switching instant. Let define each switching instants  $t_{\nu,k}$  to be the cutting instant  $t_\nu^*$ , and let  $\theta_{\nu,k} = \theta_\nu^*$  for  $\nu = 1, \dots, d$ . To show that such condition is satisfied, we need to verify that

$$0 \leq \theta_\nu^* \leq \frac{2M}{D}, \quad \nu = 1, \dots, d. \quad (3.14)$$

<sup>5</sup>By Proposition 3.2,  $\theta(t)$  is a decreasing function on  $[t_0'', t_0']$ .

This is primarily concerned with the previous proposition and corollary. Within this proof, the left inequality of (3.14) will be mentioned as the *first* inequality, while the right inequality of (3.14) will be mentioned as the *second* inequality. For each  $\pi_\nu^*$ , there are four possible cases:

1.  $\exists \pi'_i \in \Pi', \exists \pi''_j \in \Pi'', \pi_\nu^* \equiv \pi'_i$  and  $\pi_\nu^* \equiv \pi''_j$ .
2.  $\exists \pi'_i \in \Pi', \pi_\nu^* \equiv \pi'_i$  but  $\forall \pi''_j \in \Pi'', \pi_\nu^* \not\equiv \pi''_j$ .
3.  $\exists \pi''_j \in \Pi'', \pi_\nu^* \equiv \pi''_j$  but  $\forall \pi'_i \in \Pi', \pi_\nu^* \not\equiv \pi'_i$ .
4.  $\forall \pi'_i \in \Pi', \forall \pi''_j \in \Pi'', \pi_\nu^* \not\equiv \pi'_i$  and  $\pi_\nu^* \not\equiv \pi''_j$ .

For the first case, referring to (3.14), the first inequality can be simply derived via Proposition 3.2 by considering  $t_0^*$  as the former time and  $t'_0$  as the current time, given that  $\theta'_i$  satisfies (3.9):

$$\theta_\nu^* > \theta'_i > 0.$$

The second inequality is derived via the same proposition by considering  $t''_0$  as the former time and  $t_0^*$  as the current time, given that  $\theta''_j$  satisfies (3.9):

$$\theta_\nu^* < \theta''_j < \frac{2M}{D}.$$

For the second case, the first inequality is obtained with the same fashion as in the first case. However, the second inequality is obtained via Corollary 3.2. Treating  $t''_0$  as the former time and  $t_0^*$  as the current time, let us consider the nearest equivalent pair that follows  $\pi_\nu^*$ , says  $(\pi_{\hat{\nu}}^*, \pi''_j)$ . Recall that Lemma 3.2 states that, between two contiguous pairs, there can exist either members of  $\Pi^*$  only or members of  $\Pi''$  only. In this case  $\pi_\nu^*$  is the member of  $\Pi^*$ , hence statement (iii) of Lemma 3.2 applies here, and we have  $\hat{\nu}$  and  $j$  being odd numbers. Therefore, by statement (ii) in Corollary 3.2, we have

$$\theta_\nu^* < \theta''_j < \frac{2M}{D},$$

given that  $\theta''_j$  satisfies (3.9).

The third case is rather similar to the second case. We can derive the second inequality by means of Proposition 3.2, considering  $t''_0$  as the former time and  $t_0^*$  as the current time. The first inequality is derived via Lemma 3.2 and statement (i) in Corollary 3.2. Note that in this case,  $\hat{\nu}$  must be an even number.

For the last case, both first and second inequalities must be obtained through Lemma 3.2 and statement (i) in Corollary 3.2. Hence, for any  $t_0^* \in [t''_0, t'_0]$ , the corresponding  $\theta_\nu^*$  satisfies (3.14), and as a consequence, condition (i) of Theorem 2.4 is fulfilled.  $\square$

As we have developed a means to detect roughly a plausible location of a pang interval, the subsequent task is to determine the accurate location of such plausible pang interval, which is discussed in the following section.

### 3.1.4 Precise Location of Pang Intervals and Restarting Instants

Proposition 3.2 in the previous section states that as we move along each sampling instant in  $\pi_c$ , the trial cumulative summation attained in certain cutting segment is decreasing. This monotonicity allows us to apply the basic bisection method to gain the precise initial time  $t_{0,k}$  of the discovered plausible pang interval  $\mathcal{I}_{P,k}$ . Suppose that  $t''_0$  and  $t'_0$  are given which yield the cutting instants  $t'_a \in \pi'_a$  and  $t''_b \in \pi''_b$  satisfying Theorem 3.1. Following the idea of the proof of Theorem 3.1, we define a decreasing continuous function  $\theta(t) : [t''_0, t'_0] \rightarrow [\theta''_b, \theta'_a]$  to be the trial cumulative summation obtained in  $\pi'_a$  ( $= \pi''_b$ ) when treating  $t$  as the current time. In particular,  $\theta(t)$  decreases from  $\theta(t''_0)$  ( $= \theta''_b$ ) down to  $\theta(t'_0)$  ( $= \theta'_a$ ).

Let  $t_{\text{up}}$  be the time for which  $\theta(t_{\text{up}})$  gives an upper bound of  $\theta(t_{0,k})$ , whereas  $t_{\text{low}}$  be the time for which  $\theta(t_{\text{low}})$  gives a lower bound. Specifically, this implies

$$\theta(t_{\text{low}}) < \theta(t_{0,k}) < \theta(t_{\text{up}}).$$

Consider condition (iii) in Theorem 3.1. For simplicity let a constant  $\rho$  be defined as

$$\rho \triangleq \begin{cases} \frac{2M}{D}, & \text{if } a, b \text{ are odd,} \\ 0, & \text{if } a, b \text{ are even.} \end{cases}$$

In addition, let  $\epsilon$  be some sufficiently small positive number representing the tolerance of the bisection method. The termination criterion is then

$$\theta(t_{\text{up}}) - \theta(t_{\text{low}}) < 2\epsilon.$$

The bisection method to compute  $t_{0,k}$  is described as follows:

**begin**

$t_{\text{up}} := t''_0;$

$t_{\text{low}} := t'_0;$

**while**  $\theta(t_{\text{up}}) - \theta(t_{\text{low}}) \geq 2\epsilon$  **do**

$t_{\text{mid}} \leftarrow (t_{\text{up}} + t_{\text{low}})/2;$

**if**  $\theta(t_{\text{mid}}) > \rho$  **then**

$t_{\text{up}} \leftarrow t_{\text{mid}};$

**else**

$t_{\text{low}} \leftarrow t_{\text{mid}};$

**end;**

**end;**

$t_{0,k} \leftarrow (t_{\text{up}} + t_{\text{low}})/2;$

**end.**

This method yields  $t_{0,k}$  with an error less than  $\epsilon$ . The terminal time  $t_{f,k}$  and the switching instants  $t_{i,k}$  of  $\mathcal{I}_{P,k}$  are obtained automatically as the cutting instant in  $\pi'_a$  and all other cutting instants before this, respectively, when  $t_{0,k}$  is treated as the current time.

Another topic to discuss in this section is the restarting instant,  $t_{\text{rst}}$ . Generally, when the plausible pang interval  $\mathcal{I}_{P,k}$  is found, the search proceeds to determine the next plausible pang interval

$\mathcal{I}_{P,k+1}$  by beginning at  $t_{f,k}$ . However, there exists the case which  $\mathcal{I}_{P,k}$  is a spurious pang interval of the worst-case input. In this case,  $\mathcal{I}_{P,k}$  must be deleted from our consideration. Thereafter, we have to go back over to the Step 2 of SPIS, make use of the time instant very close to  $t_{0,k}$ , and move on again with the search to locate the new  $\mathcal{I}_{P,k}$ . Thus, after finishing the bisection method, it is necessary to store the time instant situated very close to  $t_{0,k}$ . In addition, this time instant should be situated after  $t_{0,k}$  so as to exclude  $t_{0,k}$  itself from the new search. Hence, we choose  $t_{\text{rst}}$  to be  $t_{\text{low}}$  obtained just before the bisection method terminates.

As mentioned in Section 2.3.4, the admissibility constraints in Theorem 2.4 yield only the necessary characteristics of the pang interval. Since the experiment makes use of Theorem 2.4, any pang interval determined through the experiment is only a plausible pang interval. In fact, we should expect that the experiment sometimes returns spurious pang intervals. It is not until we completely construct the worst-case input along the entire interval  $[0, T)$  that all plausible pang intervals which are located and still remain in the final pattern of the worst-case input become the actual pang intervals.

### 3.1.5 The First and the Last Intervals

In Theorem 3.1, some conditions have been proposed which determine roughly where plausible pang intervals are located. Nevertheless, we have made two assumptions:  $t_{c^-} \neq 0$  and that the pang interval being located terminates when the worst-case input is at its boundary<sup>6</sup>. In other words, the first and the second assumptions exclude, from our consideration, the case of locating the first and the last pang intervals, respectively. Therefore, we need to address the special experiments for these two cases as well.

#### The First Interval

Since the first interval of the worst-case input is a pang interval, the terms *first interval* and *first pang interval* will be used interchangeably. The concept of the experiment in this case is similar to that stated in Section 3.1.3 with a few changes of variables. While the normal version of the successive search is done along time axis, the special version for  $\mathcal{I}_{P,1}$  is carried out along the transverse axis. Particularly, a problem of determining the first pang interval arises because we do not know in advance the initial value  $p_{n+1}(0)$  (see Theorem 2.2 for the case that  $k = 1$ ), or equivalently, we do not know the terminal time  $t_{f,1}$  used in computing the step response  $s(T - t_{f,1})$  (see Remark 2.1).

In stead of moving along the time axis to seek for  $t_{f,1}$ , it appears to be less time consuming to seek for  $s(T - t_{f,1})$  directly, and then  $t_{f,1}$  may be determined as a consequence. This is due to the fact that  $s(T - t)$  is not necessarily a one-to-one correspondence. Recall that we have approximated  $s(T - t)$  by  $\tilde{s}(T - t)$ . Let the variable of the transverse axis be denoted by  $\beta$  representing the

---

<sup>6</sup>This assumption is made while stating the admissibility constraints in Theorem 2.4. It allows us to test the trial cumulative summation at the terminal instant of a plausible pang interval (see condition (iii) of Theorem 3.1).

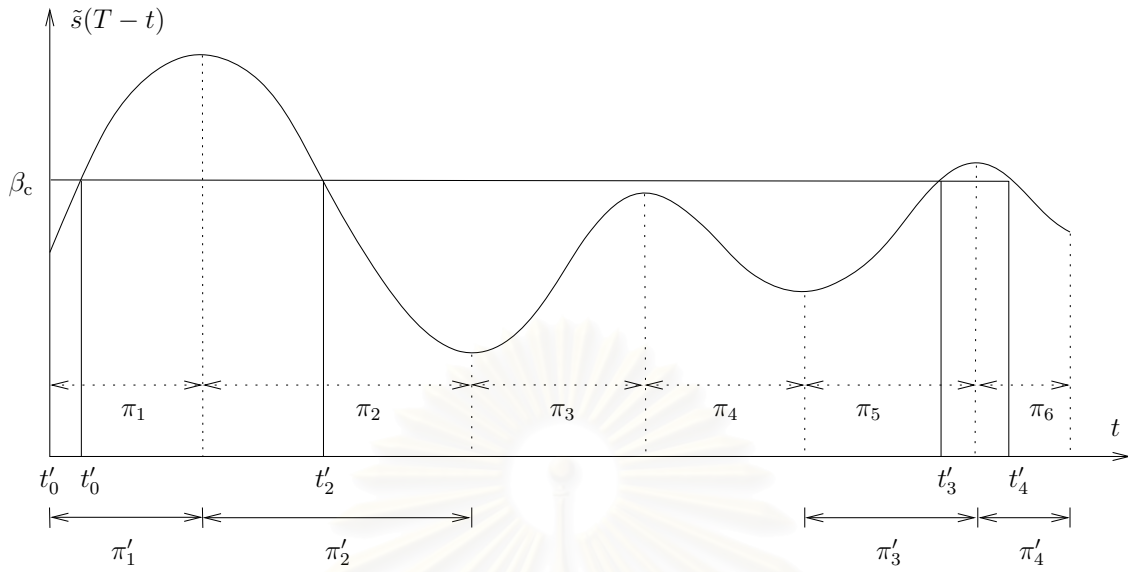


Figure 3.5: The cutting instants  $t'_0, \dots, t'_4$  and the cutting segments  $\pi'_0, \dots, \pi'_4$  of  $\tilde{s}(T-t)$  obtained when  $p_{n+1}(0) = \beta_c$ .

magnitude of  $\tilde{s}(T-t)$ . The range of this search is the set  $\mathcal{B} \triangleq \{\beta : \beta_{\min} \leq \beta \leq \beta_{\max}\}$  where

$$\begin{aligned}\beta_{\min} &= \min_{0 \leq t \leq T} \tilde{s}(T-t), \\ \beta_{\max} &= \max_{0 \leq t \leq T} \tilde{s}(T-t).\end{aligned}$$

The searching steps are obtained by uniformly sampling  $\mathcal{B}$  using sufficiently small distance between two samples. Let all the sampled points be  $\beta_1, \dots, \beta_{n_b}$  arranged in order of magnitude. Here,  $n_b$  is the total number of the searching steps.

We start the search from  $\beta_{\min}$ . For consistency, let  $\beta_c$  be the current searching point and  $\beta_{c-}$  be the former searching point. In the same fashion as the normal experiment, the reference level with the magnitude of  $\beta_c$  is projected and the cutting instants and the corresponding cutting segments are obtained. Figure 3.5 depicts how we can obtain these instants and intervals. Imitating some notations from Section 3.1.3, the theorem for detecting the crossing of a plausible first pang interval is as follows.

**Theorem 3.2** *There exists the plausible first pang interval,  $[0, t_{f,1}]$ , for which  $\beta_{c-} \leq s(T-t_{f,1}) \leq \beta_c$ , if the following conditions hold:*

- (i) *There exists the instants  $t'_a, t''_b$  ( $a, b \neq 0$ ) falling in the same segment:*

$$\pi'_a = \pi''_b,$$

- (ii) *The cutting instants through  $t'_a$  and through  $t''_b$  must yield plausible trial cumulative summations:*

$$\begin{aligned}-\frac{M}{D} &\leq \theta'_i \leq \frac{M}{D}, \quad i = 1, \dots, a-1, \\ -\frac{M}{D} &\leq \theta''_j \leq \frac{M}{D}, \quad j = 1, \dots, b-1.\end{aligned}$$



(iii) *There is a sign change in terms of the trial cumulative summations while moving from  $\beta_{c-}$  to  $\beta_c$ :*

$$\begin{aligned}\operatorname{sgn} \left[ \theta'_a - \frac{M}{D} \right] &= -\operatorname{sgn} \left[ \theta''_b - \frac{M}{D} \right], \quad \text{if } a, b \text{ are odd,} \\ \operatorname{sgn} \left[ \theta'_a + \frac{M}{D} \right] &= -\operatorname{sgn} \left[ \theta''_b + \frac{M}{D} \right], \quad \text{if } a, b \text{ are even.}\end{aligned}$$

Furthermore, we have

$$\begin{aligned}t'_a \leq t_{f,1} \leq t''_b &\quad \text{if } a, b \text{ are odd,} \\ t''_b \leq t_{f,1} \leq t'_a &\quad \text{if } a, b \text{ are even.}\end{aligned}$$

The proof of this theorem is omitted since it follows that in Section 3.1.3 analogously, except that conditions (iii) and (iv) of Theorem 2.4 must be adopted instead of condition (i) and (ii). Note that there may be several plausible first pang intervals of  $\mathcal{I}_{P,1}$ . We use each of these ones after another to start the search for subsequent plausible pang intervals until SPIS finishes.

### The Last Interval

The special experiment for the last interval,  $\mathcal{I}_L$ , is rather different. The first important fact to be noted is that  $\mathcal{I}_L$  can be either bang or pang interval. The case is trivial when  $\mathcal{I}_L$  is a bang interval. If SPIS proceeds to determine a pang interval  $\mathcal{I}_{f,k}$  whose terminal time  $t_{f,k}$  falls into the last segment  $\pi_{n_p-1}$ , then the last interval  $(t_{f,k}, T] \subset \pi_{n_p-1}$  is simply the bang interval. This is because  $\tilde{s}(T-t)$  is monotonic in every segment, hence the sign of  $h(T-t)$  is the same on this last interval, satisfying the magnitude condition in Theorem 2.3.

Because the case of the last interval being a bang interval is obvious and automatic, we focus our attention on determining all plausible last pang intervals. Let extend the use of the subscript L to anything connected with the last interval. Accordingly, let denote the last pang interval by  $\mathcal{I}_{P,L}$ . Since there is no terminal condition of the worst-case input, *i.e.*,  $\hat{w}(T)$  is not constrained,  $\mathcal{I}_{P,L}$  may not terminate when the worst-case input is at its boundary. This invalidates the condition (ii) in Theorem 2.4, and thus, leaves only the extension of condition (i) to the terminal time:

$$\begin{aligned}0 \leq \theta_{i,L} &\leq \frac{2M}{D}, \quad i = 1, \dots, n_c, \\ 0 \leq \theta_{f,L} &\leq \frac{2M}{D}\end{aligned}\tag{3.15}$$

where  $n_c$  is the total number of the switching instants in  $\mathcal{I}_{P,L}$ . From Theorem 2.1 and Lemma 2.1, we have

$$\tilde{s}(T - t_{0,L}) = \tilde{s}(T - t_{1,L}) = \dots = \tilde{s}(T - t_{f,L}) = \tilde{s}(0).$$

Note that  $\tilde{s}(0)$  and  $s(0)$  are equal. Let us draw a reference level with the magnitude of  $\tilde{s}(0)$  along  $[0, T]$ , and denote the cutting instants similarly as the normal experiment. However, the definition of the trial cumulative summation is modified so that the summation starts from any particular cutting instant, says  $t'_{\hat{i}}$ , as follows:

$$\theta'_{\hat{i},i} \triangleq \sum_{m=\hat{i}}^i (-1)^{m+1} \Delta t'_m, \quad i = \hat{i} + 1, \dots, n_c, \quad \hat{i} = 1, \dots, n_c - 1.$$

From (3.15), we easily arrive at the following theorem on the experiment for a plausible last pang interval.

**Theorem 3.3** *The time interval  $(t_{0,L}, T]$  is the plausible last pang interval, if  $t_{0,L}$  is one of the cutting instants, and the following condition holds:*

$$0 \leq \theta'_{\hat{i},i} \leq \frac{2M}{D}, \quad i = \hat{i} + 1, \dots, n_c$$

where  $\hat{i}$  is the index of the cutting instant  $t'_i$  with which  $t_{0,L}$  coincides.

Suppose we have recently determined the precise location of a plausible pang interval  $\mathcal{I}_{P,k}$ . If the terminal time  $t_{f,k}$  and the initial time  $t_{0,L}$  of a plausible last pang interval  $\mathcal{I}_L$  coexist in the same segment, and if  $t_{f,k} < t_{0,L}$ , then  $h(T - t)$  has the same sign on  $(t_{f,k}, t_{0,L}]$ , which suggests that  $(t_{f,k}, t_{0,L}]$  is the bang interval  $\mathcal{I}_{B,k}$ , and  $\mathcal{I}_{P,k+1} = \mathcal{I}_L$  is the last pang interval.

## 3.2 Computational Accuracy

To present the computational accuracy of SPIS, it is necessary to consider another computational method of the WCN for comparison. We choose a natural discretization approach that transforms the worst-case-norm computation into a large-scale finite-dimensional linear programming (LP). Although this approach for the worst-case-norm problem is not new, there are only a few related literature [20, 22], and no original reference contains a complete and explicit exposition on this topic. Perhaps, this is because the formulation technique based on a classical Euler finite-difference is obvious. In addition, the number of the design variables of the LP problem grows exponentially as better accuracy is imposed [22], causing an adverse effect for the computation. We sometimes refer to this approach as a *discrete-time approach* and refer to our proposed approach as a *continuous-time approach*. In the following subsection, we explain the discrete-time approach

### 3.2.1 Discrete-Time Formulation

We observe that even for the continuous-time approach, SPIS still requires discretization of time responses. This fact motivates us to consider the formulation in discrete-time domain. Let  $h[k]$  be the discrete equivalent of  $h(t)$  obtained by passing  $h(t)$  into a sampler with a sampling period of  $\tau_s$ , i.e.,  $t = k\tau_s$ . Here, we use the same sampling period as that of the continuous-time approach for consistency. The worst-case magnitude of output in (2.4) is modified as

$$\xi[k] \triangleq \tau_s \max_{w \in \mathcal{W}} \{h[k] * w[k]\}, \quad (3.16)$$

Recall that the discrete convolution is given by  $\sum_{i=0}^k h[k-i]w[i]$ . The presence of  $\tau_s$  in (3.16) results from approximation of the integral by the corresponding Riemann sum. It is noted that, for  $t = k\tau_s$ , an estimate of  $\xi(t)$  (the Riemann sum of convolution integral over  $[0, t]$ ) is not equal to  $\xi[k]$ , but  $\xi[k-1]$  instead. The WCN in (2.5) is approximated as

$$\|h\|_{wc} \approx \lim_{k \rightarrow \infty} \xi[k]. \quad (3.17)$$

The bounding conditions (2.1) are inherited into the discrete-time domain as follows. For all  $k \geq 0$ ,

$$\begin{aligned} -M &\leq w[k] \leq M, \\ -\tau_s D &\leq w[k+1] - w[k] \leq \tau_s D \end{aligned}$$

where  $w[k] = 0, \forall k \leq 0$ . Notice that the initial condition  $w[0] = 0$  is still assumed for the validity of definition (3.17).

Choose the terminal time  $T$  to equal that of the continuous-time approach. Let  $N + 1$  be the number of sampling points for which  $N$  is computed as  $N = T/\tau_s$  or  $N = n_s - 1$ . To estimate  $\xi(T)$ , we must compute  $\xi[N - 1]$ . Let  $c$  and  $q$  be vectors in  $\mathbb{R}^N$  representing the time series of  $h[N - 1 - k]$  and  $w[k]$ , respectively. That is, for  $k = 0, 1, \dots, N - 1$

$$\begin{aligned} c_{k+1} &= h[N - 1 - k], \\ q_{k+1} &= w[k]. \end{aligned}$$

The computation of the WCN can be cast as a linear programming problem.

$$\begin{aligned} \max_{q \in \mathbb{R}^N} \quad & \tau_s c^T q \\ \text{s.t.} \quad & q_1 = 0 \\ & -M \leq q_k \leq M, \quad k = 2, \dots, N, \\ & q_{k+1} - q_k \leq \tau_s D, \quad k = 1, \dots, N - 1, \\ & q_{k+1} - q_k \geq -\tau_s D, \quad k = 1, \dots, N - 1. \end{aligned} \tag{3.18}$$

It is easy to see that the constraint matrix of this problem is highly sparse with a banded structure of  $4 \times 1$  blocks, consisting of two diagonal blocks and the other two bidiagonal blocks. Thus, the number of nonzero elements in such matrix increases only linearly with  $N$ . Intuitively, the solution of  $\xi[N - 1]$  should become closer to the WCN as  $N$  becomes larger. This increases the size of linear programming (3.18). However, the known sparsity and structure of (3.18) will potentially help alleviate the difficulty of this large-scale problem.

In computing the WCN with digital computers, some approximation errors arise inevitably in both continuous-time and discrete-time approaches. Lane [19] discussed some types of computational errors, namely, a truncation error, an integration error, and an input error. In this dissertation, we used the truncation-error bound which is more practical and easily computable than that given in [19]. Furthermore, the integration error and the input error in [19] are combined together as a discretization error. Next, we present computable bounds of such errors in the continuous-time and discrete-time approaches.

### 3.2.2 Truncation Error

With reference to Section 2.1.1 and 3.1.2, a terminal time  $T$  in the process to compute  $\xi(T)$  or  $\xi[N - 1]$  is selected by taking into account the difference between  $\|h\|_{\text{wc}}$  and  $\xi(T)$ . This difference called *truncation error* is an error that arises when truncating the *tail* of the impulse response  $h(t)$ . Remark that in Section 3.1.2, we have shown that truncation error can be bounded by the efficiently computable quantity  $e(T) = 2M \sum_{i=1}^n \sigma_i(T)$  where  $\sigma_i(T)$  is the Hankel singular value of  $h_T(t)$ . This will be later exploited to obtain a suitable terminal time  $T$  which yields a truncation error less than the prescribed value.

### 3.2.3 Discretization Error

Besides the truncation error, discretizing  $s(T - t)$  or  $h(T - t)$  causes an additional approximation error, called *discretization error*. For the continuous-time approach, the experiment in Step 2 of SPIS needs to discretize the step response  $s(T - t)$ . Furthermore, to calculate the worst-case output magnitude in Step 5, it is required to perform a numerical integration between the worst-case input and  $h(T - t)$ , which gives rise to the discretization error.

The impulse response  $h(T - t)$  sampled in Step 5 of SPIS yields the sample-and-hold impulse response  $h_d(T - t)$  where  $h_d(T - t) = h[N - k]$ , for  $k\tau_s \leq t < (k + 1)\tau_s$ , and  $T = N\tau_s$ . Since pang intervals can be precisely determined in Step 3 of SPIS, we assume that the error between the exact worst-case input and the computed input is insignificant. For notation simplicity, let  $\hat{w}(t)$  denote the worst-case input corresponding to  $\xi(T)$ , and  $\xi_d(T)$  represent  $h_d(t) * \hat{w}(t)$ . The discretization error for the continuous-time case is defined as  $|\xi(T) - \xi_d(T)|$ .

**Theorem 3.4** *The discretization error  $|\xi(T) - \xi_d(T)|$  is bounded by*

$$|\xi(T) - \xi_d(T)| \leq \tau_s M \|h[k] - h[k - 1]\|_1. \quad (3.19)$$

Due to space limit, see the proof of Theorem 3.4 in [22]. Due to a demand of good precision,  $N$  becomes larger, and  $\tau_s$  becomes smaller. This causes  $h[k] - h[k - 1]$  to approach  $\tau_s \dot{h}(k\tau_s)$ . Hence, according to the bound (3.19), the discretization error diminishes at least quadratically with  $\tau_s$ .

For the discrete-time approach, the discretization error is defined as  $|\xi(T) - \xi[N - 1]|$ .

**Theorem 3.5** *The discretization error  $|\xi(T) - \xi[N - 1]|$  is bounded by*

$$|\xi(T) - \xi[N - 1]| \leq \tau_s M \|h[k] - h[k - 1]\|_1 + \tau_s D \|h(t)\|_1. \quad (3.20)$$

See the proof of Theorem 3.5 in [22]. It is observed that, the first term of (3.20) decreases by the rate of  $\tau_s^2$ . However, the second term of (3.20) decreases by the rate of  $\tau_s$ . This suggests that the discretization error in the discrete-time case diminishes at least linearly with  $\tau_s$ .

Remark here that error bounds (3.19) and (3.20) are meaningful for certain ranges of  $D$ . For instance, if  $D$  becomes extremely large so that  $\tau_s D > M$ , the second term of (3.20) exceeds  $M \|h(t)\|_1$ , which is an upper bound of the WCN. Furthermore, to make all the bounds on the discretization errors computable, we employ the same technique as that in (3.2). Let  $\sigma_{d_k}$  be the Hankel singular value of the discrete impulse response  $h[k] - h[k - 1]$ , and let  $\sigma_{c_k}$  be the Hankel singular value of  $h(t)$ . The computable bounds on discretization errors for the continuous-time and the discrete-time approaches become

$$2\tau_s \left( M \sum_{k=1}^n \sigma_{d_k} \right), \quad \text{and} \quad (3.21)$$

$$2\tau_s \left( M \sum_{k=1}^n \sigma_{d_k} + D \sum_{k=1}^n \sigma_{c_k} \right), \quad (3.22)$$

respectively. Note that the realization of  $h[k] - h[k - 1]$  can be readily obtained as follows. Let  $H(z)$  be a transfer function corresponding to  $h[k]$  with a minimal realization  $(A_d, B_d, C_d)$ . Immediately, the transfer function of  $h[k] - h[k - 1]$  is  $(1 - z^{-1})H(z)$  with the following realization:

$$\left( \begin{array}{cc|c} A_d & 0 & B_d \\ C_d & 0 & 0 \\ \hline C_d & -1 & 0 \end{array} \right).$$

Comparing the bounds (3.21) and (3.22), it is obvious that the discretization-error bound for the continuous-time approach is less than that of the discrete-time approach provided that  $h(t)$  is not identically zero. This suggests that our proposed algorithm (SPIS) is likely to yield better accuracy than the discretization-based method.

### 3.3 Numerical Examples

To further elaborate the idea of how much computational errors arise in the continuous-time and the discrete-time computational approach, a few numerical examples are given. We consider a second-order convolution system  $h(t)$  whose Laplace transform is

$$H(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}. \quad (3.23)$$

A closed-form solution of the WCN of this class of linear systems appears in Appendix C. The natural frequency is fixed at  $\omega_n = 10$  rad/sec and the value damping ratio  $\zeta$  is divided into three intervals, namely,

*Case 1* Overdamped system:  $\zeta > 1$ ,

*Case 2* Underdamped system:  $\zeta_c < \zeta < 1$ ,

*Case 3* Lightly-damped system:  $0 < \zeta < \zeta_c$

where  $\zeta_c = \sqrt{1 - (\pi D/2M\omega_n)^2}$ . We choose  $\zeta = 2$  in the first case. Then, we choose  $M$  and  $D$  to be 1 and 5, respectively, to make  $\zeta_c$  a real number. In this example,  $\zeta_c = 0.6190$ . Consequently, we can choose  $\zeta = 0.8$  in the second case and  $\zeta = 0.2$  in the third case.

Let us now consider the WCN of the system (3.23) in three distinct cases. For the first case,  $\|h\|_{wc} = M$ . This is actually the product of the  $\mathcal{L}_1$ -norm of  $h(t)$ , which is equal to 1, and  $M$ . For the second and the third cases, the WCNs are

$$\|h\|_{wc} = M + \frac{D \coth\left(\frac{\pi}{2} \cot \phi\right)}{\omega_n e^{(\pi-\phi) \cot \phi}},$$

$$\|h\|_{wc} = M + \frac{D [\sin(\psi + \phi) - e^{-\sigma T_c} \sin(\omega_d T_c + \psi + \phi)]}{\omega_d e^{(\psi-\phi) \cot \phi} (e^{\pi \cot \phi} - 1)},$$

respectively. Here,  $\cot \phi = \zeta / \sqrt{1 - \zeta^2}$ ,  $T_c = 2M/D$ ,  $\sigma = \zeta\omega_n$ ,  $\omega_d = \omega_n \sqrt{1 - \zeta^2}$ , and

$$\psi = \tan^{-1} \left( \frac{\sin(\omega_d T_c)}{e^{\sigma T_c} - \cos(\omega_d T_c)} \right).$$

The WCNs of all cases are computed and then compared to those obtained via the continuous-time and discrete-time approaches. In particular, two main computer programs for computing the WCNs via the continuous-time and the discrete-time approaches are developed.

We implement all computations using MATLAB 6.5 on a 2.8 GHz Pentium IV PC with 512 MB of RAM.

The discrete-time approach employs `linprog.m` as linear programming solver and makes use of matrix sparsity. In each case, the terminal time  $T$  is chosen so that the truncation-error bound  $e(T)$  equals 0.1%. The number of sampling intervals  $N$  (or  $n_s - 1$ ) is selected to be 1,000 in all cases. The sampling period  $\tau_s$  for each case is computed accordingly.

The worst-case inputs for all cases are exhibited in Figure 3.6, and the corresponding worst-case outputs are demonstrated in Figure 3.7. We note that, for the simplicity of illustrations, it is desirable to display all input responses (or all output responses) in the same axis with the same terminal time. Hence, we set  $T = 2.5$  in all cases and obtain these inputs and outputs just for the plotting purpose. For the first case, the impulse response of the overdamped system is nonnegative, so the worst-case input is equal to the positive boundary. However, having the requirement that the worst-case input starts from zero, it takes 0.2 second (equal to  $M/D$ ) to change from zero to the upper limit with the rate  $D$ .

For Cases 2 and 3, the patterns of the worst-case inputs look more complicated. The worst-case input characteristics attempt to match the step responses with respect to Theorems 2.1–2.3. In Case 2, the worst-case input fluctuates between upper and lower limits. However, in Case 3, in order to follow the rapid response of the system, the worst-case input cannot completely change from one boundary to another. Instead, it oscillates and adheres to the upper limit. This is because the impulse response lies mostly on the positive side. For both cases, we observe that the oscillation period of the worst-case input, around the middle of the time interval  $[0, T]$ , is equal to  $2\pi/\omega_d$ , where  $\omega_d = \omega_n\sqrt{1 - \zeta^2}$  is the damped natural frequency. In particular, these oscillation periods are 1.0472 and 0.6413 for Cases 2 and 3, respectively. The worst-case outputs reach the maximum at the terminal instant. The magnitude of the worst-case output at  $T$  is the WCN with some errors as described in the previous section.

The resulting WCNs and errors are summarized in Table 3.1. To see the differences between the results obtained by the continuous-time and the discrete-time approaches, the WCNs are displayed to the eighth digit. The actual errors between the computed norms and the exact values, and the corresponding error bounds are obtained by means of Theorems 3.4–3.5. Note that the error bounds are composed of the truncation-error bound and the discretization-error bound.

We can see that, in all cases, the WCNs obtained via continuous-time and discrete-time approach are virtually equal. The actual errors are less than the derived error bounds. However, in Case 3, the error bounds of both continuous-time and discrete-time approaches are significantly conservative. Since the truncation errors are kept within 0.1%, the conservatism seems to emerge from the discretization-error bounds. The reason is that the  $\mathcal{L}_1$ -norm of the linear system in Case 3 is relatively large, and hence, increases the bound in (3.20).

Table 3.1: Comparison between the exact WCNs, the WCNs computed via continuous-time and discrete-time approaches, the actual computational errors, and their bounds.

Case	Exact norm	Continuous-time approach			Discrete-time approach		
		norm	actual error (%)	error bound (%)	norm	actual error (%)	error bound (%)
1	1.0000	0.99862042	0.1380	1.4368	0.99862043	0.1380	2.8935
2	1.0180	1.01598373	0.2035	1.2043	1.01598381	0.2035	1.9089
3	2.1230	2.12221375	0.0383	9.3009	2.12227713	0.0353	13.975

As far as the effectiveness is concerned, it can be seen that, in Cases 1 and 2, the actual errors of the continuous-time and discrete-time approaches are identically equal up to the fourth digit. In Case 3, the actual error of the continuous-time approach is slightly greater than that of the discrete-time approach. This implies that the accuracy of the continuous-time approach is almost indistinguishable to that of the discrete-time approach. However, the average computation times of the computer program for SPIS are rather shorter than that of the discrete-time approach. To show this, we vary  $N$  from  $10^3$  to  $10^4$ , compute the WCN via the two approaches, and measure the computation time for each  $N$ . The results on computation time are shown in Figure 3.8(a)–3.8(c).

It should be remarked that computer programs of both approaches can be further optimized to obtain better performance, that is, decreasing time in computing the WCN. In fact, certain programming techniques can be employed to improve performance of computer programs in exchange with complexity of their codes. For examples, certain `for` loops can be vectorized; or some commands which are flexible but time-consuming can be changed to MEX-files; the banded structure of the linear programming in (3.18) can be fully exploited.

### 3.4 Summary

The main contribution in this chapter is the computational algorithm, SPIS, which is a direct result of the characterization of the worst-case input. Owing to the existence and uniqueness of the worst-case input, the SPIS algorithm is guaranteed to converge and is valid for any finite-dimensional convolution systems. Moreover, we have provided computable bounds on solution errors generated by SPIS. The effectiveness of SPIS is verified by comparing the algorithm with the straightforward discrete-time approach. Numerical examples reveal that the precisions of two approaches are comparable, but the computation time of SPIS is significantly shorter than that of the discrete-time approach.

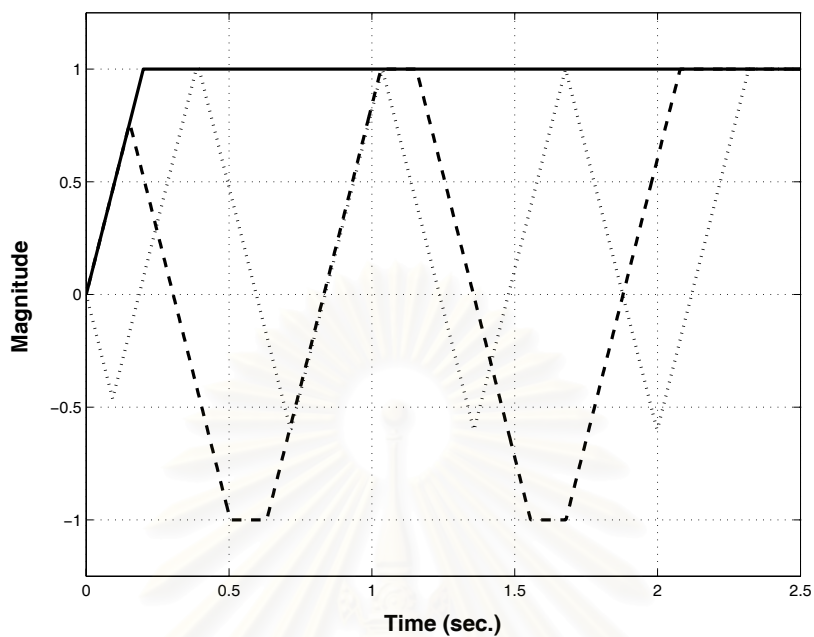


Figure 3.6: The worst-case inputs associated with the WCN computational problem of the second order systems; (—) Case 1, (---) Case 2, and (···) Case 3.

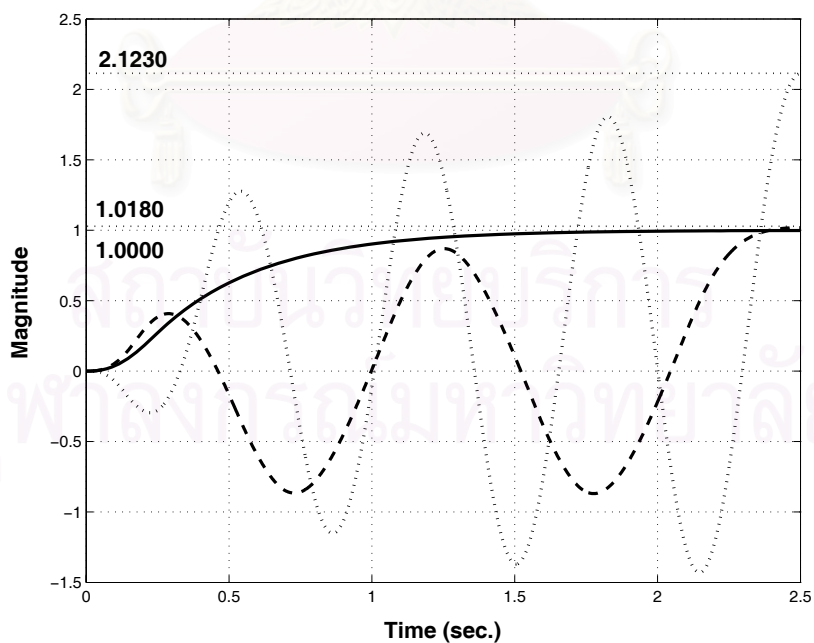


Figure 3.7: The worst-case outputs of the second order systems; (—) Case 1, (---) Case 2, and (···) Case 3.



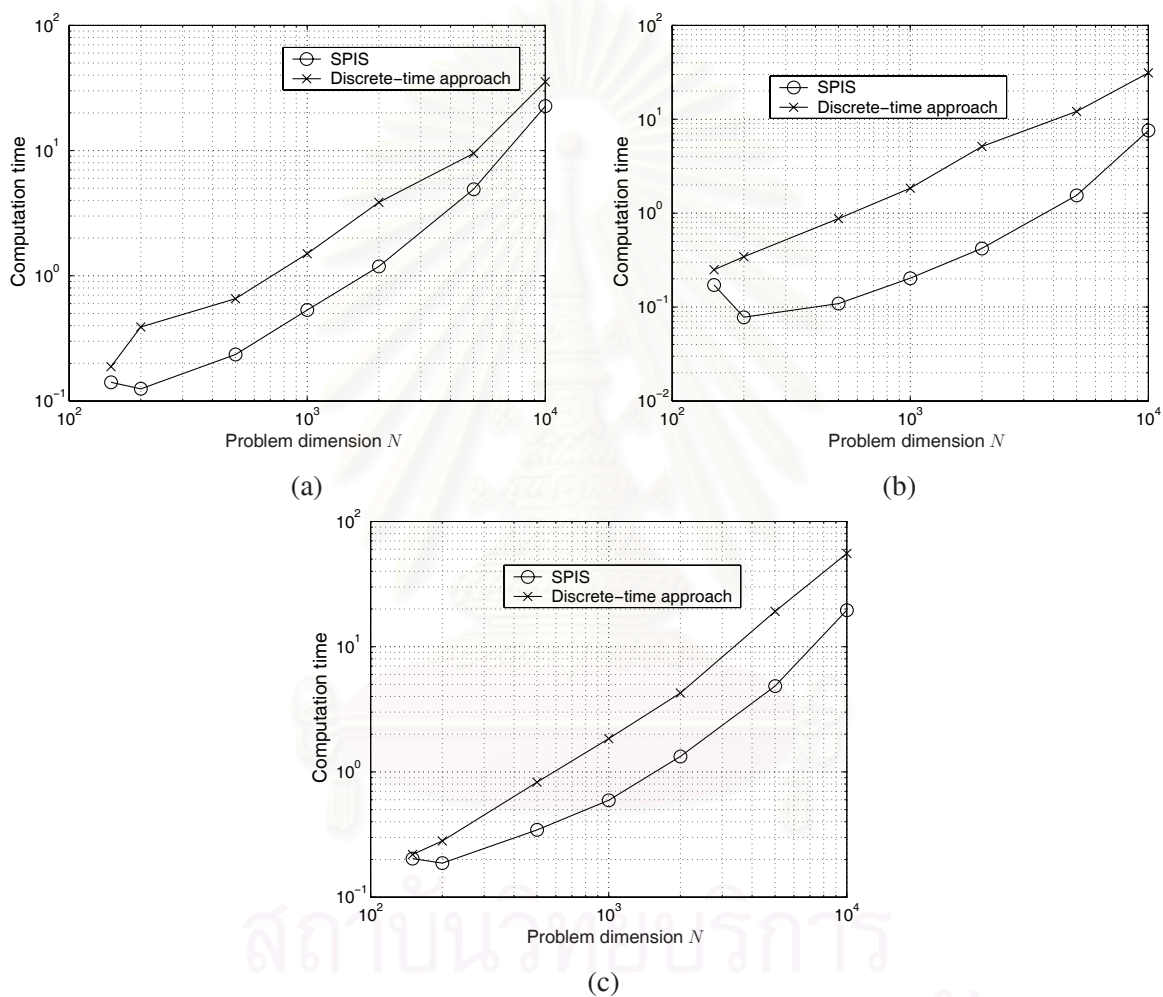


Figure 3.8: Comparison of the computation times used by SPIS and the discrete-time approach to solve the WCN computations for the second order systems: (a) Case 1, (b) Case 2, and (c) Case 3.

## CHAPTER IV

### ANALYSIS OF THE WCN OF UNCERTAIN LINEAR SYSTEMS

#### 4.1 Analytical Preliminaries

In the past chapters, disturbances are characterized with bounding constraints not only on magnitudes but also rates of change to reduce conservatism by furnish with more realism in modelling such disturbances. In addition to disturbances, mathematical descriptions of real systems usually involve approximation. The differences between the actual system and the approximate model are referred to as *model uncertainty*. In many practical situations, it is beneficial to incorporate model uncertainty in control system analysis and design. In this chapter, we analyze performance of uncertain linear systems subject to disturbances with magnitude and rate bounds. The definition of the WCN is still the maximum magnitude of an output of a linear system, which is now subjected to specific form of uncertainty.

The assumptions on input signal  $w(t)$  and the definition of  $\mathcal{W}$  hold the same as in Chapter 2. Let  $h_u(t)$  and  $h_l(t)$  be the impulse responses of some convolution systems in  $\mathbb{H}_0$  such that  $h_l(t) \leq h_u(t)$  for all  $t$ . Uncertain systems are characterized by a set  $\mathbb{H}$  containing systems with impulse responses  $h(t) \in \mathbb{H}_0$  satisfying

$$h_l(t) \leq h(t) \leq h_u(t), \quad \forall t \geq 0. \quad (4.1)$$

For convenience, we will refer to a pair  $(h_u(t), h_l(t))$  as an *impulse envelope* that defines the uncertain systems. We say that uncertain systems are stable if all impulse responses  $h(t) \in \mathbb{H}$  are stable. Figure 4.1 depicts an example of the set  $\mathbb{H}$ .

Then, the worst-case magnitude of the output attained at a given  $t$  over  $\mathcal{W}$  and  $\in \mathbb{H}$ , is redefined as

$$\xi(t) \triangleq \sup_{w \in \mathcal{W}} \sup_{h \in \mathbb{H}} |z(t)| = \sup_{w \in \mathcal{W}} \sup_{h \in \mathbb{H}} |h(t) * w(t)|. \quad (4.2)$$

Let the worst-case input and the worst-case impulse response associated with  $\xi(t)$  are represented by  $\hat{w}(t)$  and  $\hat{h}(t)$ , respectively. The WCN  $\|\cdot\|_{\text{wc}} : \mathbb{H} \mapsto [0, \infty]$  is the worst-case peak magnitude redefined in terms of  $\xi(t)$  as

$$\|h\|_{\text{wc}} \triangleq \sup_{t \geq 0} \xi(t). \quad (4.3)$$

This norm is well defined on  $\mathbb{H}$  and the norm can take the value  $+\infty$ <sup>1</sup>. As the bounds of the input are symmetric (2.1), we can remove an absolute-value operator in the definition (4.2) as

$$\xi(t) = \max_{w \in \mathcal{W}} \sup_{h \in \mathbb{H}} z(t) = \max_{w \in \mathcal{W}} \sup_{h \in \mathbb{H}} [h(t) * w(t)]. \quad (4.4)$$

---

<sup>1</sup>Like the case of linear systems with no uncertainty, this occurs when  $\mathbb{H}$  contains unstable impulse response(s). This is shown in the later section.

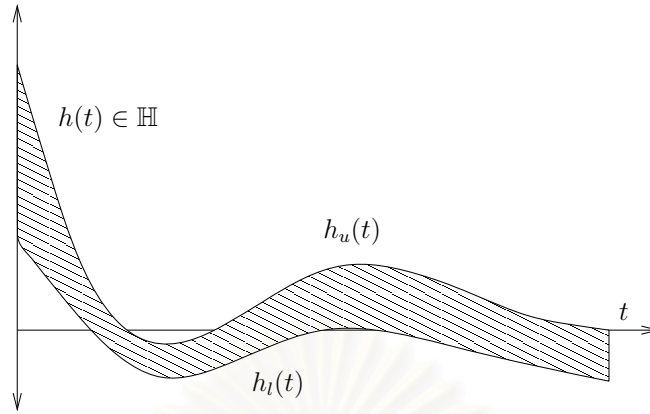


Figure 4.1: An uncertain system represented by a set  $\mathbb{H}$  shown as a band of impulse response associated with the impulse envelope  $(h_u(t), h_l(t))$ .

In the same fashion as in Section 2.1.1, this worst-case peak magnitude can be shown to be monotonic in time, which later simplify the expression of the WCN in (4.3).

#### 4.1.1 Monotonicity and the WCN Approximation

Suppose  $t_1 < t_2$ . We will show that  $\xi(t_1) \leq \xi(t_2)$ . Let  $\xi(t_1) = \hat{h}_1(t_1) * \hat{w}_1(t_1)$ , and let  $w_2(t)$  be a time shift of  $w_1(t)$  with an amount of  $\Delta t = t_2 - t_1$ . Specifically, we have

$$w_2(t) \triangleq \begin{cases} 0, & 0 \leq t \leq \Delta t, \\ \hat{w}_1(t - \Delta t), & \Delta t \leq t \leq t_2. \end{cases}$$

Due to the causal continuity,  $w_2(t)$  is well-defined so that it is contained in  $\mathcal{W}$ . In addition, define  $h_2(t)$  to equal  $\hat{h}_1(t)$  as

$$h_2(t) \triangleq \begin{cases} \hat{h}_1(t), & 0 \leq t \leq t_1, \\ h_u(t), & t_1 \leq t \leq t_2. \end{cases}$$

Obviously,  $h_2(t) \in \mathbb{H}$ . Note that for each  $t \in [t_1, t_2]$ , the response  $h_2(t)$  can take any value in between  $h_l(t)$  and  $h_u(t)$ . Here, we choose  $h_u(t)$ ,  $\forall t \in [t_1, t_2]$ . It is straightforward that

$$h_2(t_2) * w_2(t_2) = \hat{h}_1(t_1) * \hat{w}_1(t_1) = \xi(t_1).$$

Next, suppose that  $\hat{w}_2(t)$  is associated with  $\xi(t_2)$ . Through definition (4.4), it can be seen that

$$\xi(t_2) = \hat{h}(t_2) * \hat{w}_2(t_2) \geq h_2(t_2) * w_2(t_2) = \xi(t_1).$$

This infers that  $\xi(t)$  is a nondecreasing function of time, and it attains the maximum when  $t$  approaches infinity, that is,

$$\|h\|_{\text{wc}} = \lim_{t \rightarrow \infty} \xi(t). \quad (4.5)$$

Similar to the case with no uncertainty, we can then approximate  $\|h\|_{\text{wc}}$  with  $\xi(T)$  for proper choice of  $T$  terminal time. The bisection algorithm given in Section 3.1.2 can readily be employed here with some modification of the computable bound on the truncation difference defined in (3.3). To obtain a

computable bound on the truncation error, *i.e.*, the difference  $\|h\|_{\text{wc}} - \xi(T)$ , we first note that a bound (which is not computable) on this difference takes the form (see Appendix A)

$$M \max_{h \in \mathbb{H}} \int_T^\infty |h(t)| dt \quad (4.6)$$

when computing the WCN of linear uncertain systems. To find the computable error bound, we first relax (4.6) to eliminate the maximization over  $\mathbb{H}$ . This is done as follows:

$$\begin{aligned} M \max_{h \in \mathbb{H}} \int_T^\infty |h(t)| dt &\leq M \int_T^\infty \max\{|h_u(t)|, |h_l(t)|\} dt \\ &\leq M \int_T^\infty |h_u(t)| + |h_l(t)| dt \\ &= M \left( \int_T^\infty |h_u(t)| dt + \int_T^\infty |h_l(t)| dt \right) \\ &= M \left( \int_0^\infty |h_u(t+T)| dt + \int_0^\infty |h_l(t+T)| dt \right). \end{aligned} \quad (4.7)$$

Suppose the realizations of  $h_u(t)$  and  $h_l(t)$  are  $(A_u, B_u, C_u)$  and  $(A_l, B_l, C_l)$ , respectively. Then, let  $h_{u,T}(t)$  and  $h_{l,T}(t)$  be the auxiliary impulse responses with the realizations  $(A_u, e^{A_u T} B_u, C_u)$  and  $(A_l, e^{A_l T} B_l, C_l)$ , respectively. It is easy to see that

$$\begin{aligned} h_{u,T}(t) &= h_u(t+T) \\ h_{l,T}(t) &= h_l(t+T). \end{aligned}$$

From (4.7), we have

$$\begin{aligned} M \max_{h \in \mathbb{H}} \int_T^\infty |h(t)| dt &\leq M \left( \int_0^\infty |h_{u,T}(t)| dt + \int_0^\infty |h_{l,T}(t)| dt \right) \\ &\leq 2M \left( \sum_{i=1}^{n_u} \sigma_{u,i}(T) + \sum_{i=1}^{n_l} \sigma_{l,i}(T) \right). \end{aligned} \quad (4.8)$$

where  $n_u, n_l$  are the orders of the systems  $h_u(t), h_l(t)$ , respectively, and  $\sigma_{u,i}(T), \sigma_{l,i}(T)$  are the  $i$ th Hankel singular values of  $h_{u,T}(t), h_{l,T}(t)$ , respectively. The reader is referred to [50] for the derivation of the last inequality of (4.8). With the inequality (4.8), we redefine the computable bound on truncation error of the WCN computation for uncertain linear systems associated with the impulse envelope  $(h_u(t), h_l(t))$  as

$$e(T) \triangleq 2M \left( \sum_{i=1}^{n_u} \sigma_{u,i}(T) + \sum_{i=1}^{n_l} \sigma_{l,i}(T) \right). \quad (4.9)$$

The bisection algorithm can now be used to find  $T$  that yields an acceptable amount of the truncation error,  $\|h\|_{\text{wc}} - \xi(T)$ . For the ease of later discussion, when the term WCN is used, it may refer to the worst-case magnitude  $\xi(T)$  with a sufficiently large  $T$ , which makes  $\xi(T)$  the precise approximation the WCN.

### 4.1.2 Finiteness

The condition for the WCN to be finite, in the case when the linear systems are subjected to uncertainty, can be derived through that given in Section 2.1.2. The finiteness condition for that case is that the linear system is BIBO stable. For this case, one can speculate that the WCN is finite when both  $h_u(t)$  and  $h_l(t)$  are BIBO stable.

For the necessity, if either  $h_u(t)$  or  $h_l(t)$  is not BIBO stable, then selecting the unstable one to construct a system  $\tilde{h}(t)$  whose uncertain set,  $\tilde{\mathbb{H}}$ , consists of such a system alone. From 2.1.2, we have  $\|\tilde{h}(t)\|_{\text{wc}} = \infty$  because  $\tilde{h}(t)$  is unstable. Since  $\tilde{\mathbb{H}} \subset \mathbb{H}$ , from (4.3), we can deduce that

$$\|h\|_{\text{wc}} \geq \|\tilde{h}(t)\|_{\text{wc}} = \infty,$$

saying that  $\|h\|_{\text{wc}}$  is not finite.

To show the sufficiency, we introduce a convenient measure here. Let  $\|h(t)\|_{\text{wc},1}$  stands for the worst-case  $\mathcal{L}_1$ -norm defined as

$$\|h(t)\|_{\text{wc},1} = \sup_{h \in \mathbb{H}} \int_0^{\infty} |h(t)| dt. \quad (4.10)$$

It is easy to see that  $M\|h(t)\|_{\text{wc},1}$  is an upper bound of the WCN. This readily follows from the fact the the WCN of a nominal system is bounded from above by  $M\|h(t)\|_1$ .

Suppose now that both  $h_u(t)$  and  $h_l(t)$  are BIBO stable, that is, their  $\mathcal{L}_1$ -norms are finite. Note that, in this case,  $\mathbb{H}$  can be shown to be a compact set. Thus, from (4.10) and the maximum value theorem [48], we have

$$\|h(t)\|_{\text{wc},1} = \sup_{h \in \mathbb{H}} \int_0^{\infty} |h(t)| dt = \max_{h \in \mathbb{H}} \int_0^{\infty} |h(t)| dt. \quad (4.11)$$

For each element  $h(t) \in \mathbb{H}$ , we have

$$\int_0^{\infty} |h(t)| dt \leq \int_0^{\infty} \max\{|h_u(t)|, |h_l(t)|\} dt \quad (4.12)$$

$$\leq \int_0^{\infty} |h_u(t)| + |h_l(t)| dt \quad (4.13)$$

$$= \int_0^{\infty} |h_u(t)| dt + \int_0^{\infty} |h_l(t)| dt \quad (4.14)$$

$$< \infty. \quad (4.15)$$

This suggests that

$$\max_{h \in \mathbb{H}} \int_0^{\infty} |h(t)| dt < \infty$$

From (4.11), we can conclude that

$$\|h\|_{\text{wc}} \leq \|h(t)\|_{\text{wc},1} < \infty.$$

Therefore, the necessary and sufficient condition for the finiteness of the WCN of an uncertain system is that the corresponding impulse envelope is BIBO stable.

## 4.2 Problem Formulation

From the previous section, we know that when uncertain systems are not BIBO stable, the WCN will reach  $\infty$ . Hence, in order to compute the WCN, we should check first whether or not the system is BIBO stable. If not, the WCN is said to be equal to  $\infty$ , thereby leaving us to compute the WCN of stable systems only. Henceforth, uncertain systems are always assumed to be stable. Generally,  $\mathcal{W}$  is a compact set, but  $\mathbb{H}$  is compact only for stable systems, which, in this case, are so; thus, from the maximum value theorem [48], we can see that (4.4) gives

$$\xi(t) = \max_{w \in \mathcal{W}} \max_{h \in \mathbb{H}} [h(t) * w(t)].$$

Our problem of obtaining the WCN turns to be the problem of computing  $\xi(T)$  for sufficiently large  $T$ , that is to solve for

$$\xi(T) = \max_{w \in \mathcal{W}} \max_{h \in \mathbb{H}} [h(T) * w(T)]. \quad (4.16)$$

In this section, the problem of computing the WCN of uncertain linear systems under disturbances with magnitude and rate bounds is cast as a bilinear programming via a basic discretization of signals. Let  $T$  be a given terminal time where the impulse response is truncated. Let us discretize  $w(t)$  and  $h(t)$  with a sampling period  $\tau_s$  to obtain  $w[i]$  and  $h[i]$ , which means  $t = i\tau_s$ . Let the total number of sampling interval be given and denoted by  $N$ . This means the number of sampling instants is  $N + 1$ , and we have  $i = 0$  when  $t = 0$  and  $i = N$  when  $t = T$ . Furthermore, the sampling period can be calculated as  $\tau_s = T/N$ .

The magnitude and rate constraints of disturbance inputs are converted to discrete signals as follows. For all  $0 \leq i \leq N$ ,

$$-M \leq w[i] \leq M, \quad (4.17)$$

$$-\tau_s D \leq w[i + 1] - w[i] \leq \tau_s D \quad (4.18)$$

where  $w[i] = 0, \forall i \leq 0$ . Note that the forward difference has been applied to the constraint on rate limit. Next, let us discretize  $h_u(t)$  and  $h_l(t)$  with the same sampling period as above to obtain  $u[i]$  and  $l[i]$ . The bounding condition on the impulse response in (4.1) is obtained as

$$h_l[i] \leq h[i] \leq h_u[i], \quad i \geq 0.$$

The convolution integral  $\int_0^T h(T-t)w(t)dt$  in (4.16) can be approximated using trapezoidal rule as

$$\tau_s \left( \frac{1}{2} h[N]w[0] + \frac{1}{2} h[0]w[N] + \sum_{i=1}^{N-1} h[N-i]w[i] \right). \quad (4.19)$$

Since  $w[0] = 0$ , the convolution approximation (4.19) is reduced to

$$\tau_s \left( \frac{1}{2} h[0]w[N] + \sum_{i=1}^{N-1} h[N-i]w[i] \right) \quad (4.20)$$

Furthermore, applying the condition  $w[0] = 0$  to  $-\tau_s D \leq w[1] - w[0] \leq \tau_s D$  results in

$$-\tau_s D \leq w[1] \leq \tau_s D. \quad (4.21)$$

From (4.20) and (4.21), the term  $\xi(T)$  in (4.16) can be approximated tightly, as  $N$  increases, with the solution of

$$\begin{aligned} \max \quad & \tau_s \left( \frac{1}{2} h[0] w[N] + \sum_{i=1}^{N-1} h[N-i] w[i] \right) \\ \text{s.t.} \quad & -M \leq w[i] \leq M, & i = 1, \dots, N, \\ & -\tau_s D \leq w[1] \leq \tau_s D, \\ & -\tau_s D \leq w[i+1] - w[i] \leq \tau_s D, & i = 1, \dots, N-1, \\ & h_l[i] \leq h[i] \leq h_u[i], & i = 0, \dots, N-1. \end{aligned} \quad (4.22)$$

For simplicity, the following vectors are introduced in the subsequent discussion.

$$\begin{aligned} x_i &= w[i], & i = 1, \dots, N, \\ y_i &= h[N-i], & i = 1, \dots, N-1, & y_N &= \frac{1}{2} h[0], \\ u_i &= h_u[N-i], & i = 1, \dots, N-1, & u_N &= \frac{1}{2} h_u[0], \\ l_i &= h_l[N-i], & i = 1, \dots, N-1, & l_N &= \frac{1}{2} h_l[0]. \end{aligned}$$

Note that the respective transformations of  $h[i]$ ,  $h_u[i]$ ,  $h_l[i]$  to  $y_i$ ,  $u_i$ ,  $l_i$  may be different according to the technique in estimating the convolution integral  $\int_0^T h(T-t)w(t)dt$ . The problem (4.22) can then be rewritten as

$$\begin{aligned} \max \quad & \tau_s x^T y \\ \text{s.t.} \quad & -M \leq x_i \leq M, & i = 1, \dots, N, \\ & -\tau_s D \leq x_1 \leq \tau_s D, \\ & -\tau_s D \leq x_{i+1} - x_i \leq \tau_s D, & i = 1, \dots, N-1, \\ & l_i \leq y_i \leq u_i, & i = 1, \dots, N. \end{aligned} \quad (4.23)$$

The optimization (4.23) is the inner product maximization over a polytope, which is a type of bilinear programming problem [52]. For the bilinear programming, the optimizing objective is linear on the space of either  $x$  or  $y$ , but is not linear in the space of  $(x, y)$ . For later use, let us denote the optimal value of (4.23) by  $p^*$ , and let the set  $\mathcal{S}$  contains all the points  $(x, y) \in \mathbb{R}^{2N}$  feasible to the constraints of the problem. Also, let the optimal solution of (4.23) be denoted by  $(\hat{x}, \hat{y})$ .

Next, we will show that (4.23) is equivalent to a convex maximization. First, we notice that the constraints of (4.23) are disjoint, *i.e.*, there exists no constraint that relates  $x$  and  $y$ . This is due to the fact that we initially deal with the constraints on disturbance input and uncertainty separately. Consequently, we can decompose the constraints of (4.23) into two parts: the part that is associated with  $x$  only, and the other part that is associated with  $y$  only. To see this, let us define  $\mathcal{X}$  to be a set that contains every point  $x \in \mathbb{R}^N$  that satisfies the constraints

$$\begin{aligned} -M &\leq x_i \leq M, & i = 1, \dots, N, \\ -\tau_s D &\leq x_1 \leq \tau_s D, \\ -\tau_s D &\leq x_{i+1} - x_i \leq \tau_s D, & i = 1, \dots, N-1. \end{aligned} \quad (4.24)$$

Similarly, define  $\mathcal{Y}$  to be a set containing every point  $y \in \mathbb{R}^N$  that satisfies the constraint

$$l_i \leq y_i \leq u_i, \quad i = 1, \dots, N. \quad (4.25)$$

By representing the constraints of (4.23) in terms of  $\mathcal{X}$  and  $\mathcal{Y}$ , this maximization becomes

$$\max_{x \in \mathcal{X}, y \in \mathcal{Y}} \tau_s x^T y. \quad (4.26)$$

As the constraints of (4.26) are disjoint, they can be handled separately in the sense that we can maximize over the space of  $x$  first, for each  $y$ , then later maximize over the space of  $y$ . This means that (4.26) can be expressed as

$$\max_{x \in \mathcal{X}, y \in \mathcal{Y}} \tau_s x^T y = \max_{y \in \mathcal{Y}} \max_{x \in \mathcal{X}} \tau_s x^T y. \quad (4.27)$$

For convenience, we define a function  $g(y) : \mathcal{Y} \rightarrow \mathbb{R}$  as follows

$$g(y) \triangleq \max_{x \in \mathcal{X}} \tau_s x^T y. \quad (4.28)$$

This allows us to rewrite (4.27) as

$$\max_{y \in \mathcal{Y}} g(y). \quad (4.29)$$

Recall that  $(\hat{x}, \hat{y})$  solves (4.23), and hence, solves (4.26) as well. Then, (4.29) is equivalent to (4.23) in the sense that  $\hat{y}$  is a solution of (4.29), and

$$\hat{x} = \operatorname{argmax}_{x \in \mathcal{X}} \tau_s x^T \hat{y}.$$

We restore the representation of the constraint of (4.29) to the inequality form to obtain

$$\begin{aligned} \max \quad & g(y) \\ \text{s.t.} \quad & l_i \leq y_i \leq u_i, \quad i = 1, \dots, N. \end{aligned} \quad (4.30)$$

We can say that (4.30) is a convex maximization problem if we can verify that  $g(y)$  is a convex function. The following proposition gives such verification.

**Proposition 4.1** *The function  $g(y) : \mathcal{Y} \rightarrow \mathbb{R}$  defined as in (4.28) is a convex function.*

*Proof.* To show that  $g(y)$  is convex, we first establish the following inequality. Let  $y_1, y_2 \in \mathcal{Y}$ . For any  $\lambda \in [0, 1]$ , and any  $x \in \mathcal{X}$ , we have

$$\begin{aligned} \tau_s x^T (\lambda y_1 + (1 - \lambda) y_2) &= \tau_s (\lambda x^T y_1 + (1 - \lambda) x^T y_2), \\ &\leq \max_{x \in \mathcal{X}} \tau_s \lambda x^T y_1 + \max_{x \in \mathcal{X}} \tau_s (1 - \lambda) x^T y_2, \\ &= \lambda \max_{x \in \mathcal{X}} \tau_s x^T y_1 + (1 - \lambda) \max_{x \in \mathcal{X}} \tau_s x^T y_2, \\ &= \lambda g(y_1) + (1 - \lambda) g(y_2). \end{aligned}$$

Since the above inequality holds for any  $x \in \mathcal{X}$ , from (4.28), we must have

$$\begin{aligned} g(\lambda y_1 + (1 - \lambda) y_2) &= \max_{x \in \mathcal{X}} \tau_s x^T (\lambda y_1 + (1 - \lambda) y_2), \\ &\leq \lambda g(y_1) + (1 - \lambda) g(y_2), \end{aligned}$$



which concludes the proof.  $\square$

Specifically, the problem (4.30) is referred to as a convex maximization problem over a polytope, or more precisely, a hyperrectangle. This problem needs to be solved in order to determine the WCN of uncertain systems. It is well-known that local solutions of this type of problem are at the vertices of the hyperrectangle. In other words, this means the optimal solution of (4.30), *i.e.*,  $\hat{y}$ , will have its elements at their limits. This fact can reduced (4.30) to an equivalent problem

$$\begin{aligned} \max \quad & g(y) \\ \text{s.t.} \quad & y_i \in \{l_i, u_i\}, \quad i = 1, \dots, N. \end{aligned} \quad (4.31)$$

This falls into the class of nonconvex  $\mathcal{NP}$ -hard problems, which can be shown to be equivalent to  $\mathcal{NP}$ -hard combinatorial optimization or  $\mathcal{NP}$ -hard integer programming [52]. For a problem like (4.31), although the local solutions are known to be at the hyperrectangle vertices, the number of these solutions grows exponentially with problem dimension. In addition, a condition to examine whether or not a local solution of the problem is globally optimal does not even exist. Thus, computing the WCN of uncertain systems by solving (4.31) is no longer a trivial task like when there is no uncertainty. To say this clearer, we do not claim that the computation of the WCN is complicate. In fact, computational scheme to obtain the WCN is *straightforward* as we only need to go through all possible combinations of  $y_i$ 's. That is not simple is to obtain this WCN *efficiently* as the exhaustive scheme is extremely time-consuming. Before we proceed with the WCN computation, it is preferable to first establish the means to acquire the upper and lower bounds of the WCN. This is discussed in the following section.

### 4.3 Upper and Lower Bounds of the WCN

We now turn to describe the upper and lower bounds of  $p^*$ . In general, there are several upper and lower bounds of the WCN, but our proposed upper and lower bounds are tight in some sense. In particular, the bounds possess certain property that will be proved useful in solving for the exact WCN. Such a property is that the two bounds converge to each other as more elements of  $y$  are given. In fact, these bounds equal if the whole vector  $y$  are given. We will call the upper and lower bounds that satisfy this property as the *valid bounds*.

#### 4.3.1 Upper Bound

To derive an upper bound of (4.31), we start by going back to the original and equivalent problem (4.23). The idea to establish a valid upper bound is to somehow relax the bilinear programming. The relaxation should *not* be done with the objective function, *i.e.*, finding another function that bounds the objective. This is because even if  $y$  are given, the bounding function can still differ from the actual objective function. In stead, the relaxation should be introduced to the constraints of (4.23) in suitable manner. Particularly, we want to find a set described in terms of  $(x, y)$  that tightly encloses

$\mathcal{S}$  in such a way that this set resembles  $\mathcal{S}$  better when more elements of  $y$  are known. We find that such a set can be a tight convex hull of  $\mathcal{S}$ .

To realize this idea, the problem (4.23) is approximated to a sparse LP problem via a convex-hull based technique. This approximation, given in [53], is motivated by the polyhedral method<sup>2</sup> proposed by Yajima and Fujie [55]. Their method was designed for nonconvex quadratic programming with box constraints. First,  $x$  and  $y$  are shifted so as to have  $\mathcal{S}$  contained in the positive orthant of the new coordinate. That is,

$$\begin{aligned}\tilde{x} &= x + M\mathbf{1}, \\ \tilde{y} &= y - l\end{aligned}\tag{4.32}$$

where  $\mathbf{1}$  is a column vector in  $\mathbb{R}^N$  with all elements equal one. The reason for this translation is that all the variables are made positive which will be useful in the subsequent formulation. Subsequently, the problem (4.23) can be expressed as

$$\begin{aligned}\max \quad & \tau_s(\tilde{x}^T \tilde{y} + l^T \tilde{x} - M\mathbf{1}^T \tilde{y} - M\mathbf{1}^T l) \\ \text{s.t.} \quad & 0 \leq \tilde{x}_i \leq 2M, & i = 1, \dots, N, \\ & M - \tau_s D \leq \tilde{x}_1 \leq M + \tau_s D, \\ & -\tau_s D \leq \tilde{x}_{i+1} - \tilde{x}_i \leq \tau_s D, & i = 1, \dots, N-1, \\ & 0 \leq \tilde{y}_i \leq u_i - l_i, & i = 1, \dots, N.\end{aligned}\tag{4.33}$$

Let us denote the feasible set of (4.23) in the new coordinate  $(\tilde{x}, \tilde{y})$  as  $\tilde{\mathcal{S}}$ , which is a translation of  $\mathcal{S}$ . By defining a new vector  $\tilde{z}$  as

$$\tilde{z}_i \triangleq \tilde{x}_i \tilde{y}_i, \quad i = 1, \dots, N,\tag{4.34}$$

the problem (4.23) can be equivalently rewritten as

$$\begin{aligned}\max \quad & \tau_s(\mathbf{1}^T \tilde{z} + l^T \tilde{x} - M\mathbf{1}^T \tilde{y} - M\mathbf{1}^T l) \\ \text{s.t.} \quad & 0 \leq \tilde{x}_i \leq 2M, & i = 1, \dots, N, \\ & M - \tau_s D \leq \tilde{x}_1 \leq M + \tau_s D, \\ & -\tau_s D \leq \tilde{x}_{i+1} - \tilde{x}_i \leq \tau_s D, & i = 1, \dots, N-1, \\ & 0 \leq \tilde{y}_i \leq u_i - l_i, & i = 1, \dots, N, \\ & \tilde{z}_i = \tilde{x}_i \tilde{y}_i, & i = 1, \dots, N.\end{aligned}\tag{4.35}$$

Let the set  $\tilde{\mathcal{S}}$  contain every feasible point  $(\tilde{x}, \tilde{y}, \tilde{z}) \in \mathbb{R}^{3N}$ . Note that the projection of this set onto the space of  $(\tilde{x}, \tilde{y})$  is  $\tilde{\mathcal{S}}$ . Although the objective of (4.35) is linear, the last constraint in (4.35) is not. A theorem on the upper bound of  $p^*$  can be stated as follows.

**Theorem 4.1** *Let  $\bar{p}^*$  denote the optimal value of the following sparse LP:*

$$\begin{aligned}\max \quad & \tau_s(\mathbf{1}^T \tilde{z} + l^T \tilde{x} - M\mathbf{1}^T \tilde{y} - M\mathbf{1}^T l) \\ \text{s.t.} \quad & 0 \leq \tilde{x}_i \leq 2M, & i = 1, \dots, N, \\ & M - \tau_s D \leq \tilde{x}_1 \leq M + \tau_s D, \\ & -\tau_s D \leq \tilde{x}_{i+1} - \tilde{x}_i \leq \tau_s D, & i = 1, \dots, N-1, \\ & 0 \leq \tilde{y}_i \leq u_i - l_i, & i = 1, \dots, N, \\ & \tilde{z}_i \leq (u_i - l_i)\tilde{x}_i, & i = 1, \dots, N, \\ & \tilde{z}_i \leq 2M\tilde{y}_i, & i = 1, \dots, N.\end{aligned}\tag{4.36}$$

<sup>2</sup>This method is closely related to the original linearization method introduced by Padberg [54] to solve the concave quadratic 0-1 programming.

Then, we have  $\bar{p}^* \geq p^*$ .

*Proof:* Let  $\tilde{\mathbf{S}}_r$  denote the set consists of all feasible points  $(\tilde{x}, \tilde{y}, \tilde{z}) \in \mathbb{R}^{3N}$  of (4.36). We simply claim that  $\tilde{\mathbf{S}} \subseteq \tilde{\mathbf{S}}_r$ . To show this, we notice that the only difference between the constraints of (4.35) and (4.36) is the one(s) associated with variable  $\tilde{z}$ . Thus, we will focus solely on these constraints. Considering the augmented triplet  $(\tilde{x}, \tilde{y}, \tilde{z}) \in \tilde{\mathbf{S}}$ , which means that  $(\tilde{x}, \tilde{y}, \tilde{z})$  satisfies the last constraint of (4.35), or specifically, the relation (4.34). Due to the magnitude constraints of  $\tilde{x}_i$  and  $\tilde{y}_i$ , it is easy to derive the linear necessary conditions:

$$\begin{aligned} \tilde{z}_i &\leq (u_i - l_i)\tilde{x}_i, & i = 1, \dots, N, \\ \tilde{z}_i &\leq 2M\tilde{y}_i, & i = 1, \dots, N, \end{aligned} \quad (4.37)$$

which can be viewed as the relaxation of (4.34). By this argument, we must have  $(\tilde{x}, \tilde{y}, \tilde{z}) \in \tilde{\mathbf{S}}_r$ , which consequently implies our claim.

As  $\tilde{\mathbf{S}}$  is contained in  $\tilde{\mathbf{S}}_r$  and the problems (4.35), (4.36) share the same objective function, we conclude that  $\bar{p}^*$  bounds the optimal value of (4.35) from above. Since (4.35) and (4.23) are equivalent, their optimal values are identical. Hence, it follows that  $\bar{p}^* \geq p^*$ .  $\square$

Geometrically, the relaxation of (4.34) in this theorem can be view as constructing a convex hull<sup>3</sup> of a surface  $\tilde{z}_i = \tilde{x}_i\tilde{y}_i$ . This convex hull is characterized by the following linear inequalities:

$$\begin{aligned} \tilde{z}_i &\leq (u_i - l_i)\tilde{x}_i, & i = 1, \dots, N, \\ \tilde{z}_i &\leq 2M\tilde{y}_i, & i = 1, \dots, N, \\ \tilde{z}_i &\geq 0, & i = 1, \dots, N, \\ \tilde{z}_i &\geq (u_i - l_i)(\tilde{x}_i - 2M) + 2M\tilde{y}_i, & i = 1, \dots, N. \end{aligned} \quad (4.38)$$

However, because the coefficient of  $\tilde{z}_i$  in the objective function of (4.35) is positive, we only need to bound  $\tilde{z}_i$  from above, and hence, the last two constraints in (4.38) can be dropped, leaving only those in (4.37). Figure 4.2 depicts the geometric interpretation of the relaxation. In this figure, we use  $2M = u_i - l_i = 1$ . It can be seen that the relaxed constraints (4.38) outline the tightest tetrahedron enclosing the surface  $\tilde{z}_i = \tilde{x}_i\tilde{y}_i$ .

We are now ready to explain the reason for imposing positiveness for  $\tilde{x}$  and  $\tilde{y}$  by the translation (4.32). Intuitively, if they were allowed to assume negative values, the surface  $\tilde{z}_i = \tilde{x}_i\tilde{y}_i$  would extend to the negative side of  $\tilde{z}_i$  which subsequently amplifies the size and shape of the tetrahedron, and therefore, introduce more conservatism to the relaxation.

Computing the upper bound of the WCN in practice by solving an LP (4.36) can be really efficient if its sparsity pattern is exploited in the LP solver. At the present time, there are several LP solvers, but they do not take full advantage of the special structure for our problem. Thus, it is necessary to develop the specialized LP solver which will substantially expedite the computation. This LP solver is introduced in later section.

### 4.3.2 Lower Bound

The choice of the lower bound of  $p^*$  is obvious, as one may pick any pair  $(x, y) \in \mathcal{S}$  and compute  $\tau_s x^T y$ . Our procedure to obtain a relatively good lower bound is as follows. Let  $(\tilde{x}^*, \tilde{y}^*, \tilde{z}^*) \in \tilde{\mathbf{S}}_r$  be

<sup>3</sup>The convex hull of a set  $\mathcal{A}$  is the intersection of all convex sets containing  $\mathcal{A}$ , i.e., the smallest convex set enclosing  $\mathcal{A}$ .

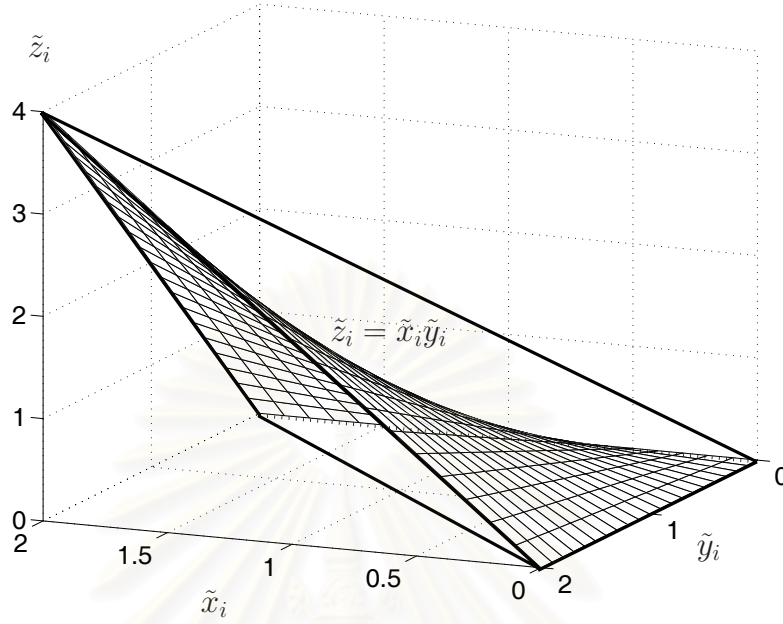


Figure 4.2: Geometric interpretation of the linear relaxation in the tetrahedral method.

the optimal solution of (4.36).

1. Restore  $\tilde{x}^*$ ,  $\tilde{y}^*$  to the original coordinate so as to get  $x^*$ ,  $y^*$ . Since  $(\tilde{x}^*, \tilde{y}^*)$  satisfies the constraints of (4.36), it is obvious that  $(x^*, y^*) \in \mathcal{S}$ .
2. The product  $\tau_s x^{*T} y^*$  may be immediately used as a lower bound of the maximization (4.23). Nevertheless, we can improve the candidate  $(x^*, y^*)$  so that it yield a tighter lower bound. This is done by applying a simple algorithm suggested in [52] with a starting point equal to  $(x^*, y^*)$ . We briefly describe the algorithm as follows:

**Algorithm 4.1**

*begin*

*round  $y_i^*$  to either  $u_i$  or  $l_i$ ;*

*denote the result as  $\hat{y}^*$ ;*

*solve the problem:*

$$\begin{aligned} \max \quad & \tau_s \hat{y}^{*T} x \\ \text{s.t.} \quad & -M \leq x_i \leq M, \quad i = 1, \dots, N, \\ & -\tau_s D \leq x_1 \leq \tau_s D, \\ & -\tau_s D \leq x_{i+1} - x_i \leq \tau_s D, \quad i = 1, \dots, N-1. \end{aligned}$$

*denote the solution as  $\hat{x}^*$ ;*

*solve the problem:*

$$\begin{aligned} \max \quad & \tau_s \hat{x}^{*T} y \\ \text{s.t.} \quad & l_i \leq y_i \leq u_i, \quad i = 1, \dots, N; \end{aligned}$$

denote the solution as  $\hat{y}^*$ ;

**while**  $|(\hat{x}^{*T}\hat{y}^*)/(x^{*T}y^*) - 1| > \epsilon_1$  **do**

$y^* \leftarrow \hat{y}^*$ ;  $x^* \leftarrow \hat{x}^*$ ;

solve the problem:

$$\begin{aligned} \max \quad & \tau_s \hat{y}^{*T} x \\ \text{s.t.} \quad & -M \leq x_i \leq M, \quad i = 1, \dots, N, \\ & -\tau_s D \leq x_1 \leq \tau_s D, \\ & -\tau_s D \leq x_{i+1} - x_i \leq \tau_s D, \quad i = 1, \dots, N-1. \end{aligned}$$

denote the solution as  $\hat{x}^*$ ;

solve the problem:

$$\begin{aligned} \max \quad & \tau_s \hat{x}^{*T} y \\ \text{s.t.} \quad & l_i \leq y_i \leq u_i, \quad i = 1, \dots, N; \end{aligned}$$

denote the solution as  $\hat{y}^*$ ;

**end**;

**end**.

This procedure iteratively solves two LP problems reduced from (4.23). In particular, the bilinear programming with either  $x$  or  $y$  is fixed results in LP problems. As the LPs are alternately solved, the objective will be increased until it reaches a local optimal solution of (4.23). This local optimum from Algorithm 4.1 can be shown to satisfy the *KKT condition* [52]. Theoretically, Algorithm 4.1 stops when  $(\hat{x}^*, \hat{y}^*)$  reaches a *KKT point*, which is actually a stationary point. Thus,  $\epsilon_1$  should represent the bound on computational error arising in solving the two LPs. Next, we give a theorem on the lower bound of  $p^*$ .

**Theorem 4.2** *The lower bound of  $p^*$  is  $\underline{p}^* = \tau_s \hat{x}^{*T} \hat{y}^*$  where  $\hat{x}^*$  and  $\hat{y}^*$  are obtained from Algorithm 4.1.*

*Proof:* It already mentioned that a pair  $(x^*, y^*)$  obtained by translating the solution of (4.36) is in  $\mathcal{S}$ . The rounding of  $y^*$  to get  $\hat{y}^*$  does also keep  $(x^*, \hat{y}^*)$  inside  $\mathcal{S}$  as it does not violate the constraints of (4.23). In addition, it is easy to see that Algorithm 4.1 itself restrains  $(\hat{x}^*, \hat{y}^*) \in \mathcal{S}$  every time a relevant LP problem is solved. Hence, as  $(x^*, y^*)$  is input into Algorithm 4.1, the resulting  $(\hat{x}^*, \hat{y}^*)$  should still be in  $\mathcal{S}$ . This suggests that  $\tau_s \hat{x}^{*T} \hat{y}^* \leq p^*$ .  $\square$

#### 4.4 Bounds of the WCN Given Elements of $y$

When some elements of  $y$  are given, the problem dimensions of (4.23), and thus, (4.31) are reduced. This information on  $y$  assists in sharpening the upper and lower bounds. Let  $i_1, i_2, \dots, i_{N_f}$  be indices of elements of  $y$  which are given, and let these indices be in ascending order:

$$i_1 < i_2 < \dots < i_{N_f}.$$

In addition, let  $i_{N_f+1}, i_{N_f+2}, \dots, i_N$  be indices of elements of  $y$  which are *not* given, and also let these indices be in ascending order:

$$i_{N_f+1} < i_{N_f+2} < \dots < i_N.$$

Furthermore, define index sets  $\Omega_f$  and  $\Omega'_f$  in parallel as

$$\begin{aligned}\Omega_f &= \{i_1, \dots, i_{N_f}\}, \\ \Omega'_f &= \{i_{N_f+1}, \dots, i_N\}.\end{aligned}$$

The  $f$  subscript here signifies the word *fixed* where it is used with quantities pertaining to the fixed or given elements of  $y$ . Next, we define a constant vector  $a$  to contain the known elements of  $y$  as follows:

$$a_j \triangleq y_{i_j}, \quad j = 1, \dots, N_f.$$

Consider a vector space generated by the standard basis  $\{e_i\}$  where  $i$ 's are indices of elements of  $y$  that are not given, *i.e.*,  $i \in \Omega'_f$ ; then, let  $y_r \in \mathcal{R}^{(N-N_f)}$  be the projection of  $y$  onto this space, *i.e.*,

$$y_{r,j} \triangleq y_{i_j}, \quad j = N_f + 1, \dots, N.$$

Here, the subscript  $r$  signifies the word *reduced*, meaning that it is a new optimization variable reduced from  $y$ . To make later mathematical representations convenient, we define in addition matrices  $D_f \in \mathcal{R}^{N_f \times N}$  and  $D'_f \in \mathcal{R}^{(N-N_f) \times N}$  as follows. Let  $D_f$  be a sparse matrix where its  $j$ th row containing only zero elements except the  $i_j$ th element that equals 1 (where  $i_j \in \Omega_f$ ). Similarly, let  $D'_f$  be a sparse matrix where its  $j$ th row containing only zero elements except the  $i_{N_f+j}$ th element that equals 1 (where  $i_{N_f+j} \in \Omega'_f$ ). The upper and lower bounds for this particular case can be now described.

#### 4.4.1 Upper Bound

We change  $x$  and  $y$  to a new coordinate as in (4.32), obtaining  $\tilde{x}$  and  $\tilde{y}$ . In the same manner, we define the translation of  $a$  and  $y_r$  as follows:

$$\begin{aligned}\tilde{a}_j &\triangleq \tilde{y}_{i_j}, \quad j = 1, \dots, N_f, \\ \tilde{y}_{r,j} &\triangleq \tilde{y}_{i_j}, \quad j = N_f + 1, \dots, N.\end{aligned}$$

Consequently, the maximization (4.33) can be expressed as

$$\begin{aligned}\max \quad & \tau_s (\tilde{y}_r^T D'_f \tilde{x} + (\tilde{a}^T D_f + l^T) \tilde{x} - M \mathbf{1}^T \tilde{y}_r - M \mathbf{1}^T (\tilde{a} + l)) \\ \text{s.t.} \quad & 0 \leq \tilde{x}_i \leq 2M, \quad i = 1, \dots, N, \\ & M - \tau_s D \leq \tilde{x}_1 \leq M + \tau_s D, \\ & -\tau_s D \leq \tilde{x}_{i+1} - \tilde{x}_i \leq \tau_s D, \quad i = 1, \dots, N-1, \\ & 0 \leq \tilde{y}_{r,j} \leq u_{i_j} - l_{i_j}, \quad j = N_f + 1, \dots, N.\end{aligned} \tag{4.39}$$

Then, define  $\tilde{z} \in \mathcal{R}^{(N-N_f)}$  in parallel with (4.34) as

$$\begin{aligned}\tilde{z}_j &\triangleq \tilde{x}_{i_j} \tilde{y}_{i_j}, \quad j = N_f + 1, \dots, N, \\ &= \tilde{x}_{i_j} \tilde{y}_{r,j}, \quad j = N_f + 1, \dots, N.\end{aligned}$$

With this newly defined variable, it can be shown that (4.39) is equivalent to

$$\begin{aligned}
\max \quad & \tau_s(\mathbf{1}^T \tilde{z} + (\tilde{a}^T D_f + l^T) \tilde{x} - M \mathbf{1}^T \tilde{y}_r - M \mathbf{1}^T (\tilde{a} + l)) \\
\text{s.t.} \quad & 0 \leq \tilde{x}_i \leq 2M, & i = 1, \dots, N, \\
& M - \tau_s D \leq \tilde{x}_1 \leq M + \tau_s D, \\
& -\tau_s D \leq \tilde{x}_{i+1} - \tilde{x}_i \leq \tau_s D, & i = 1, \dots, N-1, \\
& 0 \leq \tilde{y}_{r,j} \leq u_{i_j} - l_{i_j}, & j = N_f + 1, \dots, N, \\
& \tilde{z}_j = \tilde{x}_{i_j} \tilde{y}_{r,j}, & j = N_f + 1, \dots, N.
\end{aligned} \tag{4.40}$$

Let the set  $\tilde{\mathbf{S}}_f$  consist of every feasible point  $(\tilde{x}, \tilde{y}_r, \tilde{z}) \in \mathbb{R}^{3N-2N_f}$ . Suppose also that (4.40) has its optimal value represented by  $p_f^*$ . Then, an upper bound on  $p_f^*$  can be given as below.

**Theorem 4.3** Let  $\bar{p}_f^*$  denote the optimal value of the following sparse LP:

$$\begin{aligned}
\max \quad & \tau_s(\mathbf{1}^T \tilde{z} + (\tilde{a}^T D_f + l^T) \tilde{x} - M \mathbf{1}^T \tilde{y}_r - M \mathbf{1}^T (\tilde{a} + l)) \\
\text{s.t.} \quad & 0 \leq \tilde{x}_i \leq 2M, & i = 1, \dots, N, \\
& M - \tau_s D \leq \tilde{x}_1 \leq M + \tau_s D, \\
& -\tau_s D \leq \tilde{x}_{i+1} - \tilde{x}_i \leq \tau_s D, & i = 1, \dots, N-1, \\
& 0 \leq \tilde{y}_{r,j} \leq u_{i_j} - l_{i_j}, & j = N_f + 1, \dots, N, \\
& \tilde{z}_i \leq (u_{i_j} - l_{i_j}) \tilde{x}_{i_j}, & j = N_f + 1, \dots, N \\
& \tilde{z}_i \leq 2M \tilde{y}_{r,j}, & j = N_f + 1, \dots, N.
\end{aligned} \tag{4.41}$$

Then, we have  $\bar{p}_f^* \geq p_f^*$ .

*Proof:* Let  $\tilde{\mathbf{S}}_{fr}$  denote the feasible set of (4.41) containing all feasible points  $(\tilde{x}, \tilde{y}_r, \tilde{z}) \in \mathbb{R}^{3N-2N_f}$ . The problem (4.41) is obtained from (4.40) by convex-hull relaxation. With similar arguments as in Theorem 4.1, we then have  $\tilde{\mathbf{S}}_f \subseteq \tilde{\mathbf{S}}_{fr}$ . Thus, it can be concluded that  $\bar{p}_f^* \geq p_f^*$ .  $\square$

Note that the specialized LP solver for (4.41) can be introduced based on that for (4.36). The knowledge on the optimal solution of (4.41) can yield a smart starting point to obtain the lower bounds on  $p_f^*$ , *i.e.*, the optimal value of (4.23) and (4.31) when some  $y_i$  are given.

#### 4.4.2 Lower Bound

The choice of the good lower bound of  $p_f^*$  can be described as follows. Let  $(\tilde{x}^*, \tilde{y}_r^*, \tilde{z}^*) \in \tilde{\mathbf{S}}_{fr}$  be the optimal solution of (4.41).

1. Restore  $\tilde{x}^*$ ,  $\tilde{y}_r^*$  to the original coordinate so as to get  $x^*$ ,  $y_r^*$ .
2. The product  $\tau_s x^{*T} y_r^*$  can be improved to give a tighter lower bound. This is done by applying an algorithm akin to Algorithm 4.1 as below:

#### Algorithm 4.2

*begin*

*round*  $y_{r,j}^*$  *to either*  $u_{i_j}$  *or*  $l_{i_j}$ ;

*denote the result as*  $\hat{y}_r^*$ ;

*solve the problem:*

$$\begin{aligned}
& \max \quad \tau_s (\hat{y}_r^{*T} D'_f x + a^T D_f x) \\
& \text{s.t.} \quad -M \leq x_i \leq M, \quad i = 1, \dots, N, \\
& \quad \quad -\tau_s D \leq x_1 \leq \tau_s D, \\
& \quad \quad -\tau_s D \leq x_{i+1} - x_i \leq \tau_s D, \quad i = 1, \dots, N-1.
\end{aligned}$$

denote the solution as  $\hat{x}^*$ ;

solve the problem:

$$\begin{aligned}
& \max \quad \tau_s ((\hat{x}^* D'_f)^T y_r + a^T D_f \hat{x}^*) \\
& \text{s.t.} \quad l_{i_j} \leq y_{r,j} \leq u_{i_j}, \quad j = N_f + 1, \dots, N;
\end{aligned}$$

denote the solution as  $\hat{y}_r^*$ ;

**while**  $|(\hat{x}^{*T} \hat{y}^*) / (x^{*T} y^*) - 1| > \epsilon_1$  **do**

$y^* \leftarrow \hat{y}^*$ ;  $x^* \leftarrow \hat{x}^*$ ;

solve the problem:

$$\begin{aligned}
& \max \quad \tau_s (\hat{y}_r^{*T} D'_f x + a^T D_f x) \\
& \text{s.t.} \quad -M \leq x_i \leq M, \quad i = 1, \dots, N, \\
& \quad \quad -\tau_s D \leq x_1 \leq \tau_s D, \\
& \quad \quad -\tau_s D \leq x_{i+1} - x_i \leq \tau_s D, \quad i = 1, \dots, N-1.
\end{aligned}$$

denote the solution as  $\hat{x}^*$ ;

solve the problem:

$$\begin{aligned}
& \max \quad \tau_s ((\hat{x}^* D'_f)^T y_r + a^T D_f \hat{x}^*) \\
& \text{s.t.} \quad l_{i_j} \leq y_{r,j} \leq u_{i_j}, \quad j = N_f + 1, \dots, N;
\end{aligned}$$

denote the solution as  $\hat{y}_r^*$ ;

**end**;

**end**.

This procedure iteratively solves two LP problems reduced from (4.23) when  $y_{i_j}$  are fixed for  $j = 1, \dots, N_f$ . This algorithm terminates when it finds  $(\hat{x}^*, \hat{y}_r^*)$  which is a *KKT* point. A theorem on the lower bound of  $p_f^*$  can be given in accordance with Theorem 4.2 as follows.

**Theorem 4.4** *The lower bound of  $p_f^*$  is  $\underline{p}_f^* = \tau_s \hat{x}^{*T} \hat{y}_r^*$  where  $\hat{x}^*$  and  $\hat{y}_r^*$  are obtained from Algorithm 4.2.*

#### 4.5 LP Solvers via the Interior-Point Method

We employ the basic primal interior-point method with centering steps solved by the Newton's method, which is a powerful yet simple method for unconstrained optimization. Here, we discuss how to implement the LP solvers focusing on the procedures to determine the Hessian and to solve for the Newton system. These procedures yield the Newton step in each Newton iteration and take up most operations of the overall computation. Note that a procedure to determine the gradient for the Newton system is straightforward and uses fewer flops, so we do not present it here.



Recall that the upper bounds of the WCN via the convex-hull based technique is obtained by solving the LP problem (4.36). To begin with the interior-point method, we rearrange this LP problem into the inequality form:

$$\begin{aligned} \max \quad & c^T \zeta \\ \text{s.t.} \quad & A\zeta \preceq b \end{aligned} \quad (4.42)$$

where  $\zeta$  is the optimization variable, and  $c^T \zeta$  is the linear objective. The number of rows of  $A$  is equal to  $m$ , which is the number of the inequality constraints.

For the interior-point method, the optimal solution of (4.42) is approximated by solving an unconstrained (concave) logarithmic-barrier maximization with an objective function:

$$\phi(\zeta) = \gamma c^T \zeta + \sum_{i=1}^m \log(b_i - a_i^T \zeta)$$

where  $b_i$  is the  $i$ th element of  $b$ ,  $a_i$  is the  $i$ th row of  $A$ , and  $\gamma$  is the parameter which adjusts the approximation accuracy. The Hessian of this objective function can be expressed as

$$\nabla^2 \phi(\zeta) = A^T \Lambda A. \quad (4.43)$$

The diagonal matrix  $\Lambda$  has each of its diagonal entry specified by  $1/d_i^2$  for  $i = 1, \dots, m$  where  $d = A\zeta - b$ , and  $\zeta$  is updated in each Newton iteration. The reader is referred to [56] for comprehensive details.

Let the optimization variable of the LP problem (4.36) be arranged as

$$\zeta = \begin{bmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{z} \end{bmatrix}.$$

where  $\tilde{x}$ ,  $\tilde{y}$  and  $\tilde{z}$  are all in  $\mathbb{R}^N$ . The LP parameters  $A$  and  $b$  are

$$A = \begin{bmatrix} I & 0 & 0 \\ -I & 0 & 0 \\ \Theta & 0 & 0 \\ -\Theta & 0 & 0 \\ 0 & I & 0 \\ 0 & -I & 0 \\ -U & 0 & I \\ 0 & -2MI & I \end{bmatrix}, \quad b = \begin{bmatrix} M\mathbf{1} \\ M\mathbf{1} \\ v_1 \\ v_2 \\ u - l \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

where  $U$  is a diagonal matrix whose diagonal elements are from  $u - l$ , and  $v_1, v_2$  are vectors with all elements equal  $\tau_s D \mathbf{1}$  except the first element, which is equal to  $\tau_s D + M$  for  $v_1$  and equal to  $\tau_s D - M$  for  $v_2$ , respectively. Here,  $A$  contains  $12N - 2$  nonzero elements.

To evaluate (4.43), we decompose  $\Lambda$  according to the partitions of  $A$  as

$$\Lambda = \text{diag} \{ \Lambda_1, \Lambda_2, \dots, \Lambda_8 \}$$

so that the Hessian can be written as

$$\nabla^2 \phi(\zeta) = \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix}.$$

where

$$\begin{aligned} Q_{11} &= \Lambda_1 + \Lambda_2 + \Theta^T(\Lambda_3 + \Lambda_4)\Theta + U\Lambda_7U, \\ Q_{12} &= Q_{21}^T = [0 \quad -U\Lambda_7], \\ Q_{22} &= \begin{bmatrix} \Lambda_5 + \Lambda_6 + 4M^2\Lambda_8 & -2M\Lambda_8 \\ -2M\Lambda_8 & \Lambda_7 + \Lambda_8 \end{bmatrix}. \end{aligned}$$

Since the product  $U\Lambda_7U$  gives a diagonal matrix, the efficient way to form  $Q_{11}$  is introduced as follows. Let  $\lambda_i \in \mathbb{R}^N$  stand for the vector that forms the diagonal elements of  $\Lambda_i$  for  $i = 1, \dots, 4$  and  $i = 7$ . With some algebra, we can show that  $Q_{11}$  is a tridiagonal matrix with the diagonal elements given by

$$U^2\lambda_7 + \sum_{i=1}^4 \lambda_i + \begin{bmatrix} \hat{\lambda}_3 \\ 0 \end{bmatrix} + \begin{bmatrix} \hat{\lambda}_4 \\ 0 \end{bmatrix},$$

and both super- and subdiagonal elements given by  $-(\hat{\lambda}_3 + \hat{\lambda}_4)$ . Here, the vector  $\hat{\lambda}_i \in \mathbb{R}^{N-1}$  is obtained by removing the first element of  $\lambda_i$ , that is,  $\hat{\lambda}_{i,k} = \lambda_{i,k+1}$  for  $k = 1, \dots, N-1$ . Computations of  $Q_{12}$ ,  $Q_{21}$  and  $Q_{22}$  are straightforward, so the details are omitted. The cost of forming  $\nabla^2\phi(\zeta)$  is equal to  $32N$  flops.

To solve for the Newton step, we introduce the most efficient way in terms of computational cost as follows. First, denote the Newton system to be solved as  $\nabla^2\phi(\zeta)\Delta\zeta = -g$  where  $\Delta\zeta$  is the Newton step, and  $g = \nabla\phi(\zeta)$  is the gradient. Then, partition this Newton system according to the partitions of the Hessian as

$$\begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix} \begin{bmatrix} \Delta\zeta_1 \\ \Delta\zeta_2 \end{bmatrix} = - \begin{bmatrix} g_1 \\ g_2 \end{bmatrix}.$$

To find  $\Delta\zeta_1$ ,  $\Delta\zeta_2$ , the following procedure is executed:

1. Find the inverse of  $Q_{22}$ ;
2. Compute  $Q_{12}Q_{22}^{-1}Q_{21}$  and  $Q_{12}Q_{22}^{-1}g_2$ ;
3. Obtain  $\hat{Q} = Q_{11} - Q_{12}Q_{22}^{-1}Q_{21}$  and  $\hat{g} = g_1 - Q_{12}Q_{22}^{-1}g_2$ ;
4. Solve for  $\Delta\zeta_1$  from  $\hat{Q}\Delta\zeta_1 = -\hat{g}$ ;
5. Compute  $\Delta\zeta_2 = Q_{22}^{-1}(Q_{21}\Delta\zeta_1 - g_2)$ .

Note that the inverse of  $Q_{22}$  can be computed very efficiently and note also that the linear equation in the fourth step is of a tridiagonal form since  $\hat{Q}$  is tridiagonal (because  $Q_{11}$  is). For these particular structures of  $Q_{ij}$ , this procedure costs  $31N$  flops. In summary, the cost for determining the Hessian and solving for the Newton step for the LP (4.36) is  $63N$  flops.

## 4.6 Numerical Examples

For some insight, brief numerical examples of upper and lower bounds of the WCN are given. We make use of three uncertain linear systems. Their impulse envelopes are illustrated in Figures 4.3(a)–4.3(c). For each WCN computation, we set  $M = 1$ ,  $D = 1.2$  and  $T = 10$ . Note that the same

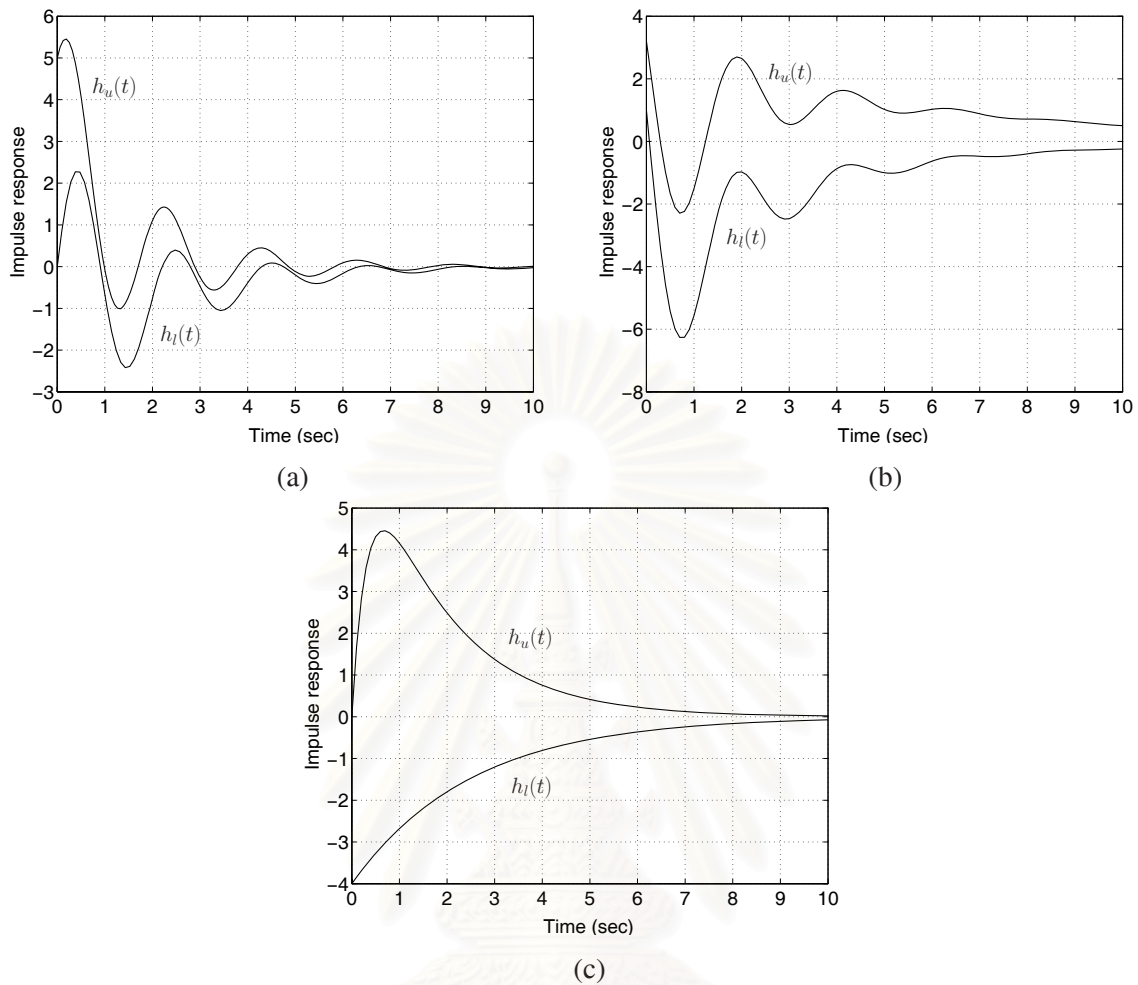


Figure 4.3: The impulse response envelopes  $(h_u(t), h_l(t))$ , which bound from above and below, all admissible impulse responses of the uncertain systems: (a) system 1, (b) system 2, and (c) system 3.

terminal time is used in all computations for convenience. We compute the WCN of each system when the problem dimension  $N$  varies from 10 to 100. The upper bounds of the WCNs are computed by solving the related LP problems obtained from (4.36), while the lower bounds are computed via Algorithm 4.1. As  $N$  increases, the sampling rate used in formulating the WCN computation increases, so we can expect the bounds to get more precise. The results are plotted in Figure 4.4(a)–4.4(c). The upper and lower bounds in three cases converge to certain limits except the lower bound in the case of system 1 that exhibits some fluctuation. With  $N = 100$ , the differences between the two bounds of system 1–3 are 17.75%, 9.44% and 5.82%, respectively. Note that all percentages are computed relative to the upper bounds.

## 4.7 Summary

In this chapter, the computational problem of the WCN of uncertain linear systems is analyzed. Some properties and problem simplifications are first presented. Then, the problem is cast as a convex max-

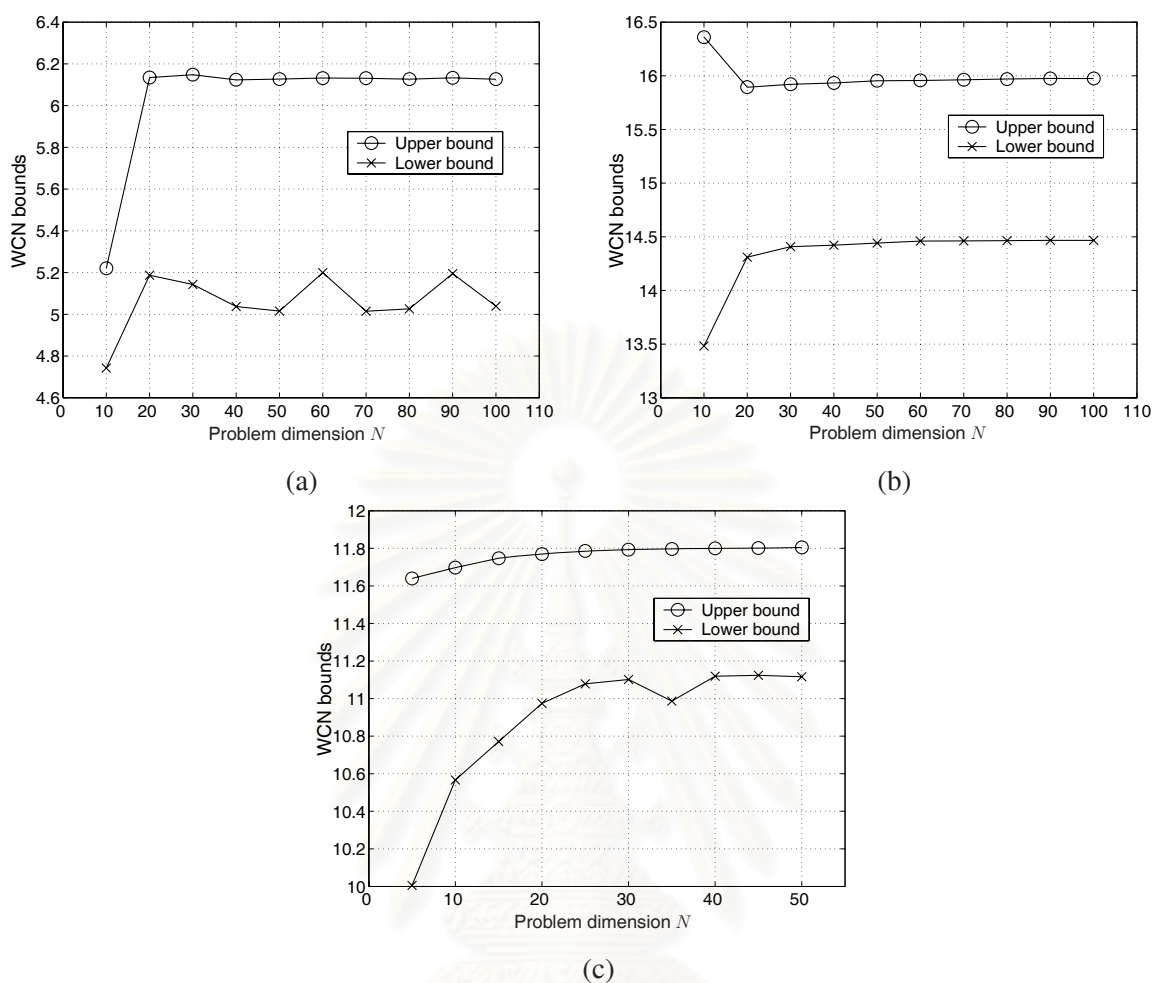


Figure 4.4: The upper bounds and lower bounds of the WCNs when the problem dimension varies from 10 to 100: (a) system 1, (b) system 2, and (c) system 3.

imization problem over a hyperrectangle. The upper bounds are computed via the linear relaxation of the original problem, while lower bounds are obtained by alternately solving two simple LP problems. The upper and lower bounds with the knowledge of some elements of  $y$  are presented afterward. The chapter ends with the specialized LP solver which is used to compute the upper bound of the WCN.

## CHAPTER V

### COMPUTATION OF THE WCN OF UNCERTAIN LINEAR SYSTEMS

In the preceding chapter, the WCN computation for uncertain linear system appears to be a convex maximization (4.31). This class of maximization is actually known to be nonconvex. In fact, a maximization of convex (quadratic) function with box-constraints in finite-dimensional space is shown to be an  $\mathcal{NP}$ -hard [57] problem. In addition, the number of local solutions, which lie on vertices of the polytope, increases with problem dimension at exponential rate. For clarity, we need to elaborate that a large number of local solutions would not have caused any difficulty if there was a condition to examine whether or not a local solution is globally optimal as in the case of linear programming (LP). Unfortunately, this kind of condition does not exist for the polytopic-constrained convex maximization, which causes the major difficulty. Thus, a global optimization for this problem must employ a scheme of comparing objective functions at local solutions, *i.e.*, vertices of the polytope. Due to the nature of  $\mathcal{NP}$ -hard problems, a thorough comparison of all local solutions at the vertices is impractical. Hence, a partial comparison using known analytical knowledge to reduce the enumeration.

A common optimization method to tackle this difficult problem is a branch-and-bound (BB) algorithm. It is well-known that effective bounds of the exact WCN critically imply the algorithm efficiency. In the preceding Chapter, we already give the novel upper and lower bounds of the WCN, and in this chapter an effective branching technique is designed. A computer program for this BB algorithm is developed to compute the WCN. Note that the computation time consumed by this method can be very long in some circumstances due to comparison of local solutions [58]. Hence, later in this chapter, we construct a new optimization scheme based on BB algorithm to accelerate the WCN computation.

#### 5.1 The Standard BB Algorithm

BB algorithms have been proposed for solving a wide range of global optimizations including those emerging from control engineering problems [58–60]. BB algorithm are generally used to estimate the global optimum by iteratively tightening the upper and lower bounds of the optimal value. Specifically, the algorithms proceed until the gap between two bounds is reducing to an acceptable magnitude. However, it is interesting to note that, in our problem (4.31), this gap can be made exactly zero. This is because all considered points in (4.31) constitute a discrete feasible set, and thus, we only need to carry out the BB algorithm over just a finite number of candidates. For this reason, the algorithm can proceed through these finite candidate until it finds the optimizer, and at that instance, the upper and lower bounds are identical and equal to the exact solution, which is here the WCN.

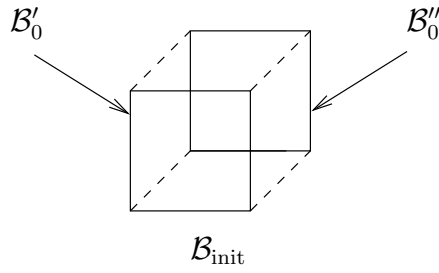


Figure 5.1: Branching a three-dimensional cube  $\mathcal{B}_{\text{init}}$  into two-dimensional sub-cubes  $\mathcal{B}'_0$  and  $\mathcal{B}''_0$  whose vertices make up those of  $\mathcal{B}_{\text{init}}$ .

The BB scheme for (4.31) can be described as follows. Let  $\mathcal{B}_{\text{init}}$  be the hyperrectangle generated by the constraints of the problem (4.31). Let  $\Gamma_0$  be an initial set of a hyperrectangle in consideration, that is,  $\Gamma_0$  lists only  $\mathcal{B}_{\text{init}}$ . First, upper and lower bounds,  $U_0$  and  $L_0$ , of  $p^*$  are computed over  $\mathcal{B}_{\text{init}}$ . It is noted that the difference of these bounds, over a hyperrectangle  $\mathcal{B}$ , must approach zero as  $\mathcal{B}$  shrinks to a point. After that, we choose the appropriate  $i$ th element of  $y$  to be fixed, and the branching process is carried out by fixing  $y_i = l_i$  and then  $y_i = u_i$ . This branching can be viewed as splitting  $\mathcal{B}_{\text{init}}$  into two sub-hyperrectangles with lower dimension. The sub-hyperrectangles have no common vertices, and their vertices constitute those of  $\mathcal{B}_{\text{init}}$ . Let these sub-hyperrectangles be denoted by  $\mathcal{B}'_0$  and  $\mathcal{B}''_0$ , respectively. Figure 5.1 displays this branching step. The current list  $\Gamma_1$  is then updated to be  $\{\mathcal{B}'_0, \mathcal{B}''_0\}$  as  $\mathcal{B}_{\text{init}}$  is already removed.

Then, the bounding process of the first iteration is started by computing upper and lower bounds  $U_{\mathcal{B}'_0}$ ,  $U_{\mathcal{B}''_0}$ ,  $L_{\mathcal{B}'_0}$ , and  $L_{\mathcal{B}''_0}$  of  $g(y)$  over these sub-hyperrectangles. The bounds  $U_1$  and  $L_1$  for the first step are subsequently obtained as

$$\begin{aligned} U_1 &= \max\{U_{\mathcal{B}'_0}, U_{\mathcal{B}''_0}\}, \\ L_1 &= \max\{L_{\mathcal{B}'_0}, L_{\mathcal{B}''_0}\}. \end{aligned}$$

Since  $p^*$  is in either  $\mathcal{B}'_0$  or  $\mathcal{B}''_0$ , we can say that either  $U_{\mathcal{B}'_0}$  or  $U_{\mathcal{B}''_0}$  is greater than  $p^*$ , and hence,  $U_1 \geq p^*$ . In addition, since both  $L_{\mathcal{B}'_0}$  and  $L_{\mathcal{B}''_0}$  are lower than  $p^*$ , we have  $L_1 \leq p^*$ . Subsequently, we pick the hyperrectangle  $\hat{\mathcal{B}}_1$  associated with the upper bound that gives  $U_1$ . The reason behind this choice is that it seems to be the *most promising* hyperrectangle. The process then continues by branching  $\hat{\mathcal{B}}_1$  with respect to another proper element of  $y$ .

In general, for the  $k$ th iteration, let  $\Gamma_k$  be the list of all hyperrectangles  $\mathcal{B}_i$ ,  $U_k$  be the upper bound, and  $L_k$  be the lower bound. We select the hyperrectangle  $\hat{\mathcal{B}}_k$  associated with  $U_k$  and branch it to get  $\mathcal{B}'_k$  and  $\mathcal{B}''_k$ . Update the list  $\Gamma_k$  by deleting  $\hat{\mathcal{B}}_k$ , adding its two children, and obtain  $\Gamma_{k+1}$ . Then, compute the upper and lower bounds  $U_{\mathcal{B}_i}$ ,  $L_{\mathcal{B}_i}$  of each hyperrectangle, and obtain  $U_{k+1}$  and  $L_{k+1}$  from

$$\begin{aligned} U_{k+1} &= \max_{\mathcal{B}_i \in \Gamma_{k+1}} U_{\mathcal{B}_i}, \\ L_{k+1} &= \max_{\mathcal{B}_i \in \Gamma_{k+1}} L_{\mathcal{B}_i}. \end{aligned}$$

Next, we will show that  $U_{k+1}$  and  $L_{k+1}$  are still the upper and the lower bounds of  $p^*$ , respectively. Suppose  $p^*$  lies in some vertices of hyperrectangle  $\mathcal{B}_i$  in the list. Then, we have

$$U_{k+1} \geq U_{\mathcal{B}_i} \geq p^*.$$

Furthermore, let  $\mathcal{B}_j$  be the hyperrectangle associated with  $L_{k+1}$ , *i.e.*,  $L_{\mathcal{B}_j} = L_{k+1}$ . Note that  $\mathcal{B}_j$  may or may not be the same hyperrectangle as  $\mathcal{B}_i$ . Considering  $\hat{y} \in \mathcal{B}_j$ , it now follows that

$$L_{k+1} = L_{\mathcal{B}_j} \leq g(\hat{y}) \leq p^*.$$

Afterward, the bounds are compared and if they satisfy the stopping criterion,

$$U_{k+1} - L_{k+1} < \epsilon, \quad (5.1)$$

the algorithm terminates. It is noted that, theoretically,  $U_k$  and  $L_k$  approach each other at the optimal value, that is, the algorithm can proceed until  $U_k = L_k = p^*$ . Nevertheless, to save the computation time, it may be preferable to stop earlier when we reach the acceptable accuracy, *i.e.*, when  $U_k$  and  $L_k$  are apart at an acceptable distance of  $\epsilon > 0$ .

The process of the BB algorithm can be plotted as a binary tree diagram. We may regard a hyperrectangle  $\mathcal{B}_i \in \Gamma_k$  in the diagram as a *node* described by its bounds,  $U_{\mathcal{B}_i}$  and  $L_{\mathcal{B}_i}$ . Starting from the initial node associated with  $\mathcal{B}_{\text{init}}$ , the branching process splits and removes it. Then, the children nodes are bounded, one of them is selected to be branched, and the algorithm moves on from one iteration to another. As two children nodes are introduced when its parent node is deleted, we simply see that the total number of nodes increases by one at each iteration; in other words, the number of nodes at the  $k$ th iteration should equal  $k + 1$  (as at the zeroth iteration, there is one initial node). However, at the end of the  $k$ th iteration, after the bounding step and before starting the  $(k + 1)$ th branching step, we can eliminate nodes (hyperrectangles in the list  $\Gamma_{k+1}$ ) if their upper bounds  $U_{\mathcal{B}_i}$  are less than  $L_{k+1}$ . This is because the values of  $g(y)$ ,  $\forall y \in \mathcal{B}_i$  are lower than  $L_{k+1}$ , and hence, they are even lower than  $p^*$ . This general practice is usually referred to as *pruning*. The total number of nodes can be reduced from  $k + 1$  this way, and we shall call the nodes that are left unpruned as the *active nodes*. The flow chart describing BB algorithm is given in Figure 5.2. It is important to note that the convergence of this method is trivial since the worst case is guaranteed to be an exhaustive search as the number of iterations cannot exceed  $2^N$ .

### 5.1.1 Branching Strategies

This section is devoted to describe branching techniques. As a result of solving (4.36), we then have a means to choose which component of  $y$  should be branched, *i.e.*, fixed at either  $u_i$  or  $l_i$ , and the bounds in each case are calculated. Let us define a useful term to ease up the following discussion.

**Definition 5.1** A component  $y_i$  of  $y$  is more plausible than  $y_j$  if it is relatively closer to its bounds than  $y_j$ . In other words,  $y_i$  is more plausible if it is relatively far from the midpoint of  $[l_i, u_i]$  than  $y_j$  from the midpoint of  $[l_j, u_j]$ . Mathematically, this means

$$\frac{1}{(u_i - l_i)} \left( y_i - \frac{(l_i + u_i)}{2} \right) \geq \frac{1}{(u_j - l_j)} \left( y_j - \frac{(l_j + u_j)}{2} \right).$$

Conversely, a component  $y_i$  of  $y$  is more implausible if it is less plausible. A component  $y_i$  of  $y$  is called the most plausible component if it is more plausible than any others.

These definitions are given according to their natures, *i.e.*, if  $y_i$  is relatively closer to either  $u_i$  or  $l_i$  then it is likely to believe that the  $i$ th component of the optimal  $y$  should lie on such boundary. In our work, we consider two main branching strategies:

1. branch along the most plausible component,
2. branch along the most implausible component.

Presumably, one may speculate that the first approach should yield better convergence for our problem as it choose to branch the *best* component first. In contrast to our perception, we have investigated empirically that this choice of branching strategy gives slower convergent rate than branching along the most implausible component first.

An explanation for this phenomenon is based on our observation that if we choose to branch along the most plausible component, the implausible components would be left to branch later. Since they are implausible components, they have a tendency to take many iterations to conclude if the path we branch is correct or not. Hence, if it is spurious, we might have to start over and many iterations would require again and again. On the other hand, if we choose the second strategy, the plausible components would be left to branch later. These components require only a few iterations to check whether or not the path we branch is correct. If we make wrong decision, then it would require only some additional iterations to get to the optimal solution. Even though the probability for making wrong decisions in the beginning branching in the second strategy is higher than that of the first one, the number of iterations required to start over is greater. Therefore, it is wise to *clarify the ambiguous components first*. Similarly, in later iterations, when the dimension is reduced, we pick the component  $\bar{y}_i$  that is the farthest from its bounds among the other components. Note that in this case, we have to retrace to determine which component of  $y$  is associated with  $\bar{y}_i$ .



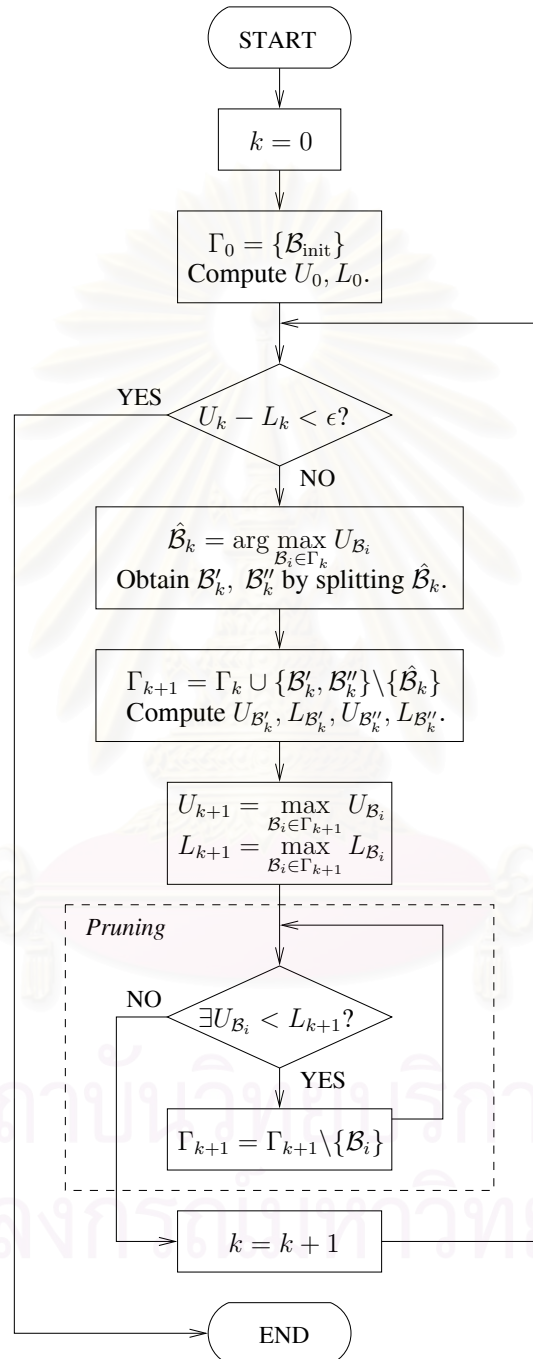


Figure 5.2: Flow chart of the BB algorithm.

## 5.2 Numerical Examples

Three examples of uncertain linear systems in the former chapter are used. Recall that Figures 4.3(a)–4.3(c) display envelopes of the impulse responses for these systems. To validate the algorithm, we use the standard exhaustive search to seek for the benchmark solution. Recall that the global optimal solution of (4.31) is already known to be at one of the vertices of the feasible polytope; hence, the exhaustive search is carried out in a straightforward way as follows:

1. Enumerate all the vertices of the feasible hyperrectangle in  $y$  space. This is to obtain all possible combinations of  $y$  when  $y_i$  is fixed at either  $u_i$  or  $l_i$ .
2. Consider these combinations one by one and evaluate  $g(y)$  at each combination, *i.e.*, each vertex, and store the values.
3. Compare these values to obtain the maximum which is the WCN.

Since this search entirely examines all possible local solutions, the global solution is assured to be correct. Suppose that the BB method identifies a vertex as the one corresponding to the global solution. If such vertex is the same as that identified by the exhaustive search, this simply judges that the BB algorithm functions properly.

The input bounds  $M$  and  $D$  are set to 1 and 1.2, respectively. The terminal time is fixed at  $T = 10$ . The problem dimension is the number of the sampling instants minus one. During the validation, the problem dimension varies from 10 to 20 as the exhaustive search is not effective beyond this range<sup>1</sup>. The executions were performed on 2.8 GHz Pentium 4 PC with 512 MB of RAM. After the validation, the BB algorithm yields identical solutions as those by the exhaustive search. To see the computational efficiency, Table 5.1 displays the times consumed by the standard BB algorithm against the exhaustive search. For all three systems, the computation time used by the exhaustive search increases approximately by twice as much when  $N$  increases by one. This is evidently because when the problem dimension is increased by one, the number of possible combinations of  $y_i$ 's doubles<sup>2</sup>, and hence, the number of elements of the feasible set of (4.31) increases accordingly. This means we have twice more local solutions to search through, which should double the computation time.

For the BB computations of system 1, the trend of computation times exhibit some fluctuation. However, the overall trend appears to gradually incline. There is a big contrast of the computation time when  $N = 18$  which equal 13.9 seconds where it drops significantly down below those of the other neighboring cases (39.1 and 40.7 seconds). It is found that the number of iterations taken to finish the BB computation for  $N = 18$  is 28 while it takes 77 and 78 iterations for the case of  $N = 17$  and 19, respectively. This explains the relatively short computation time. This drop in computation time when problem dimension increases, however, is possible since the rate of convergence of the BB algorithm cannot be theoretically determined in general. For the case of system 2 and 3, the trends

<sup>1</sup>The exhaustive search can take a week for the problem dimensions greater than 24, on current machines.

<sup>2</sup>The number of all combinations of  $y_i$ 's is  $2^N$  when  $N$  is the problem dimension. It is obvious that  $2^{N+1}$  is twice as much as  $2^N$ .

of the BB computation times seem to rise in a step change manner with a bit variation. Remark the steep rise in BB computation times of system 3 when  $N$  change from 17 to 18 (the computation times increase from 8.4 to 21.6 seconds). Similar to the case of system 1, the significant increase in the number of iterations accounts for this sharp change in this case, but there is no nice specific reason behind this.

Table 5.1: Comparison between the computation times of the BB algorithm and the exhaustive search for the computations of the WCNs of three uncertain systems: system 1, 2, and 3, when  $N$  varies from 10 to 20. The abbreviations s, m, and h signify the time units second, minute, and hour, respectively.

$N$	System 1		System 2		System 3	
	BB algo.	Exh. search	BB algo.	Exh. search	BB algo.	Exh. search
<b>10</b>	2.7 s	53.4 s	6.1 s	51.5 s	8.6 s	57.9 s
<b>11</b>	13.5 s	1.8 m	9.5 s	1.7 m	8.7 s	1.9 m
<b>12</b>	14.2 s	3.7 m	5.5 s	3.4 m	7.3 s	3.7 m
<b>13</b>	17.0 s	7.7 m	9.9 s	7.1 m	9.1 s	7.8 m
<b>14</b>	35.5 s	15.7 m	9.3 s	14.4 m	7.9 s	15.7 m
<b>15</b>	21.1 s	31.7 m	9.7 s	29.0 m	7.8 s	31.7 m
<b>16</b>	53.5 s	1.1 h	8.9 s	59.5 m	9.1 s	1.1 h
<b>17</b>	39.1 s	2.2 h	12.7 s	2.1 h	8.4 s	2.2 h
<b>18</b>	13.9 s	4.5 h	17.7 s	4.1 h	21.6 s	4.0 h
<b>19</b>	40.7 s	9.2 h	17.2 s	8.6 h	26.2 s	8.4 h
<b>20</b>	1.5 m	17.2 h	19.7 s	16.3 h	28.4 s	17.0 h

At each iteration, the number of active nodes and the convergence of the upper and lower bounds in WCN computation of system 1 via BB algorithm is depicted in Figure 5.3(a)–5.3(f), for  $N = 10, 15,$  and  $20$ . For systems 2 and 3 these are shown in Figure 5.4 and 5.5, respectively. From each of the figures related to the number of active nodes, it can be seen that this number rises up, reaching its zenith at around half way of the total number of iterations used. Without pruning, the number of nodes branched by the algorithm should increase by one at each iteration. It can be seen in these figures that the BB algorithm also starts with this rate of node branching. Later, however, the pruning process begins to take effect and decelerates the rate. After the algorithm processes through the peak, the number of nodes drops down because the computed upper and lower bounds,  $U_k$  and  $L_k$ , are getting more and more accurate, and then, more nodes are being pruned as  $k$  increases. In these particular examples, only one node is left when the algorithm terminates. It should be noted that the BB algorithm can terminate even though more than one nodes are left behind, provided that (5.1) is satisfied. From Figure 5.3–5.5, we see that the lower bound is relatively tighter than the upper bound because the lower bound reaches the optimal value (the WCN) much earlier, waiting for the upper bound and confirm the optimality. Another point to note is that it is still reasonable

to work with for  $N = 20$  on current processing units, and that, with this choice of  $N$ , the error of discretizing the original problem (4.16) is still acceptable [22]. Even though the computations with selected dimensions are reasonably in order of seconds, the computation time will take a few hours for the WCN problem of system 1 with dimension over 40 because the number of local solutions grows exponentially.

To show the effect of discretization, the WCNs of three linear uncertain systems are plotted in Figure 5.6. The WCN of all systems tend to converge to certain limits as the sampling instants are increased. Furthermore, to add more insight into the solutions of the BB computation, we execute the WCN computation of system 1 with dimension of 50 and plot the all signals belonging to the worst-case scenario with the terminal time  $T = 10$ . These signals are displayed in Figure 5.7, which are the worst-case input, the worst-case impulse response, and the worst-case output, respectively. The magnitude of the worst-case output at  $t = T$  should represent the WCN, which is here equal 5.194. Notice also that the worst-case impulse response stays at  $h_u(t)$  for 1.2 second and moves down to  $h_l(t)$  at the next sampling instant. This agrees with the zero-crossing instant of the worst-case input at  $t = 8.8$  as this equals  $T - 1.2$ .

We see that the BB algorithm consumed less computation time as compared to an exhaustive search, and hence, make the WCN computation realizable (as the exhaustive will take a lifetime to compute a problem with dimension greater than just 50, using present processing units). However, the computation speed of the BB algorithm is still considered as impractical if we have to apply it to the controller design phase which needs to solve for WCN repeatedly. Thus, a specialized technique is required to accelerate the BB algorithm for faster WCN computation.

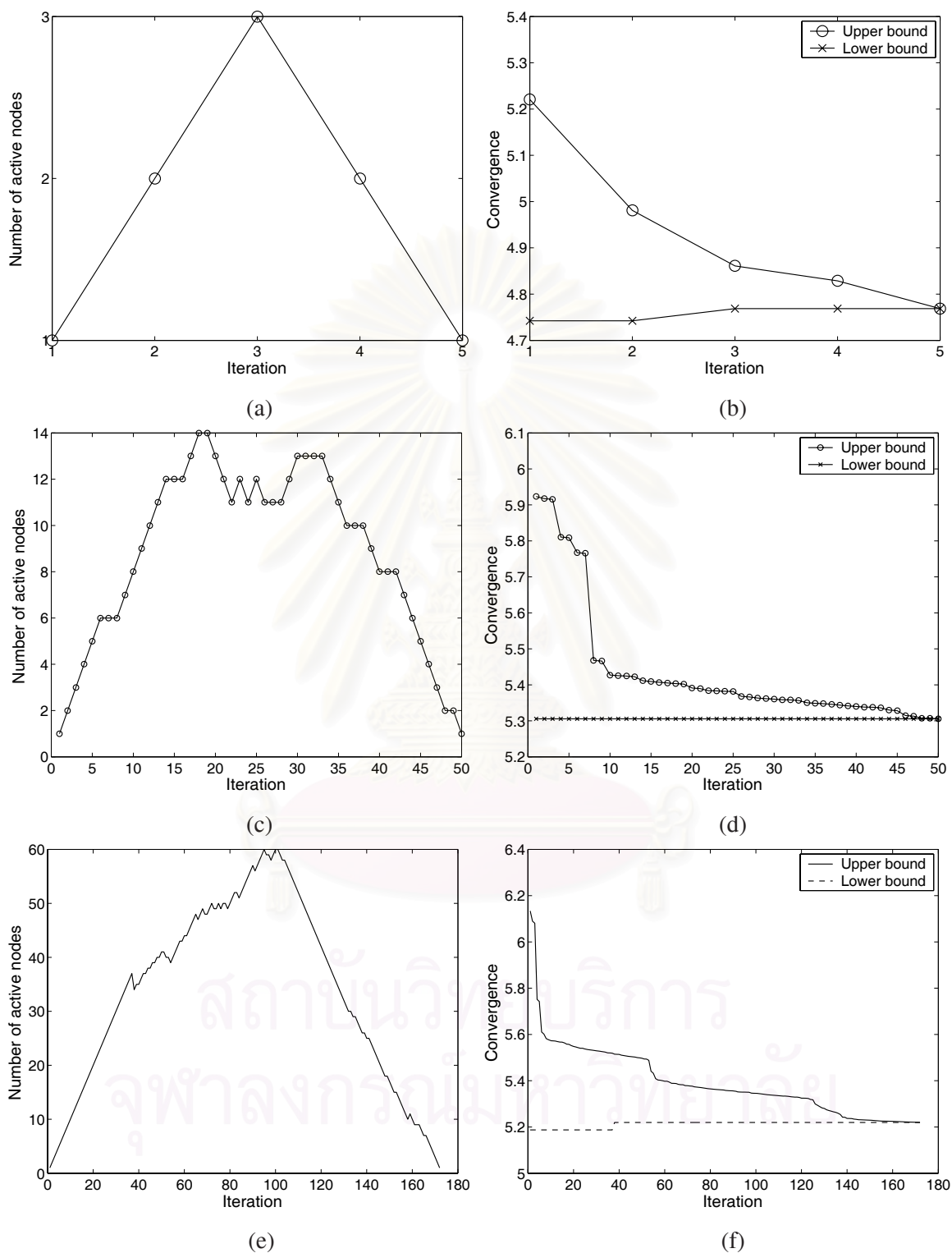


Figure 5.3: For WCN computation of system 1 using standard BB algorithm, the number of active nodes at each iteration: (a)  $N = 10$ , (c)  $N = 15$ , and (e)  $N = 20$ , and the convergence of upper and lower bounds: (b)  $N = 10$ , (d)  $N = 15$ , (f)  $N = 20$ .

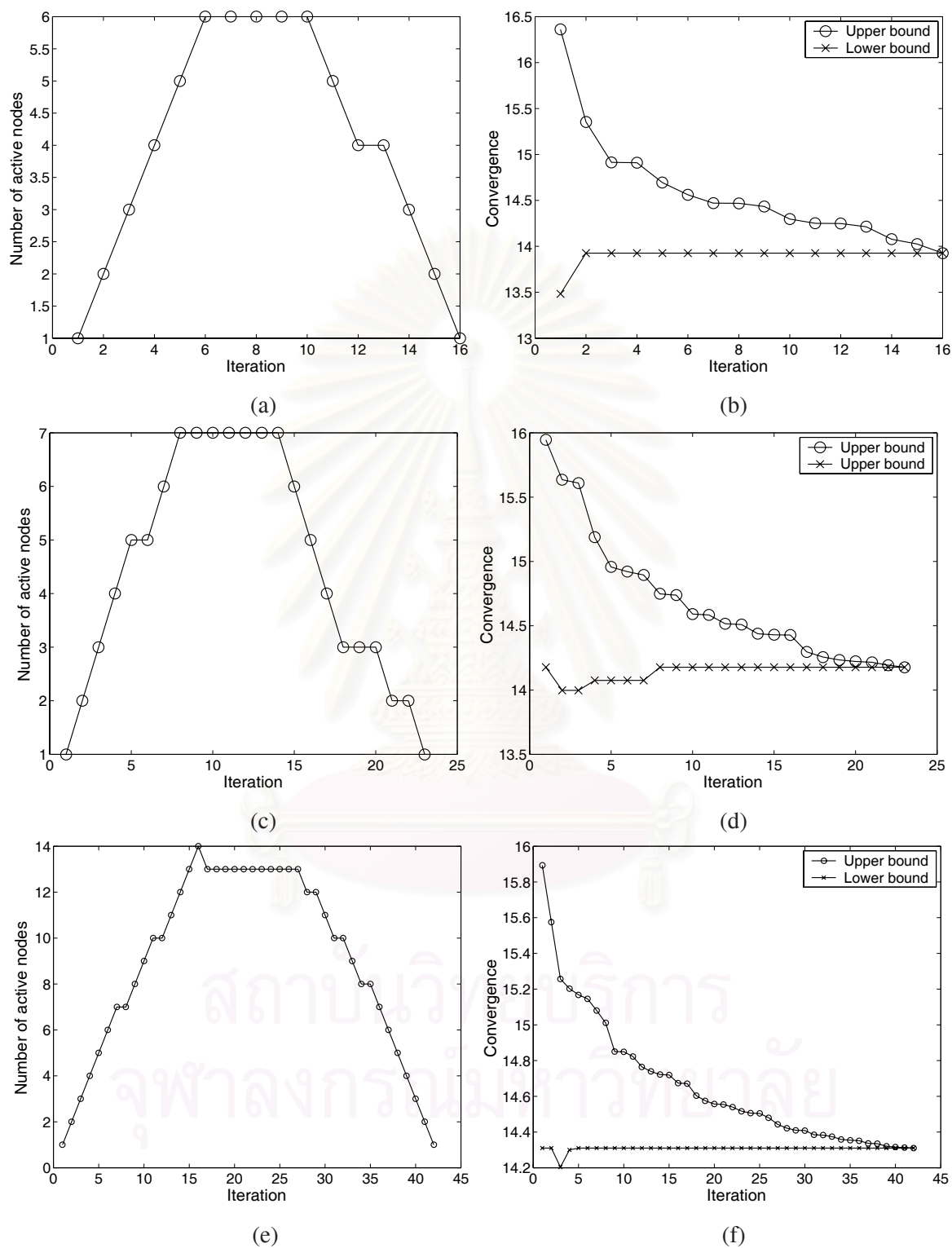


Figure 5.4: For WCN computation of system 2 using standard BB algorithm, the number of active nodes at each iteration: (a)  $N = 10$ , (c)  $N = 15$ , and (e)  $N = 20$ , and the convergence of upper and lower bounds: (b)  $N = 10$ , (d)  $N = 15$ , (f)  $N = 20$ .

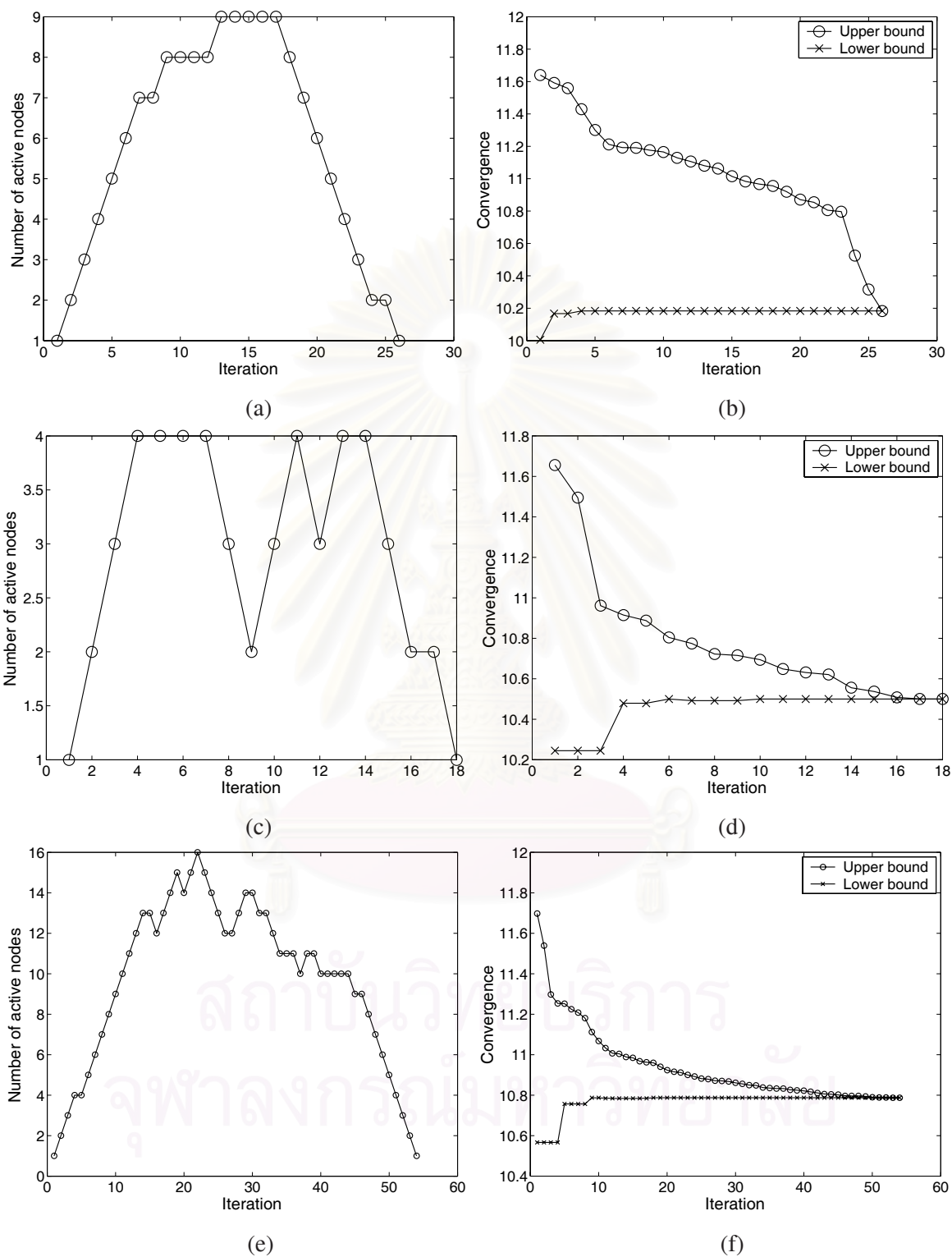


Figure 5.5: For WCN computation of system 3 using standard BB algorithm, the number of active nodes at each iteration: (a)  $N = 10$ , (c)  $N = 15$ , and (e)  $N = 20$ , and the convergence of upper and lower bounds: (b)  $N = 10$ , (d)  $N = 15$ , (f)  $N = 20$ .

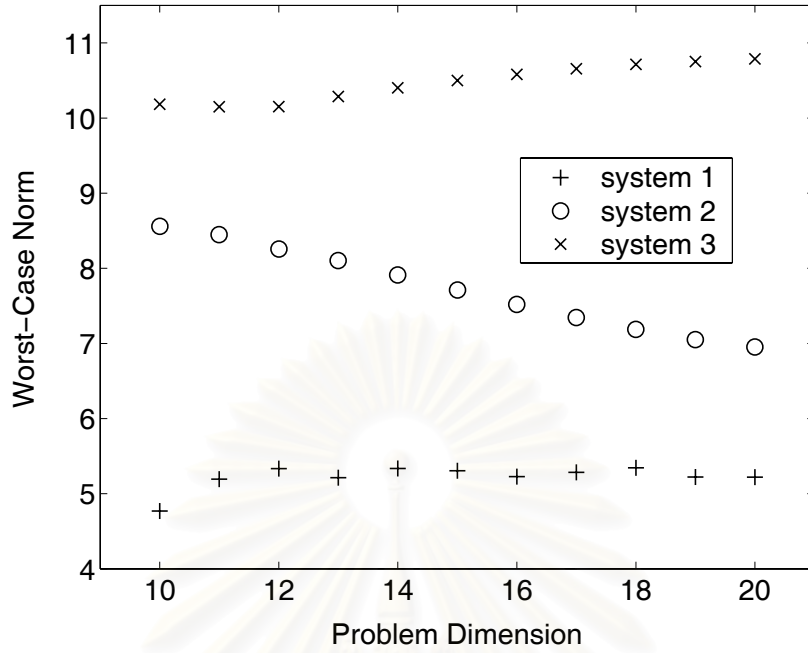


Figure 5.6: The WCN of three uncertain linear convolution systems when the problem dimension varies from 10 to 20.

### 5.3 Solution of Discretized Problems with Increasing Sampling Rates

Our idea to speed up the BB algorithm originates from the fact that the formulation of WCN computation involves discretization of signals, and with growing sampling rates the solutions become more accurate. Therefore, to avoid solving a high dimension problem in one shot, it is desirable to solve a problem with low dimension first, and then exploit its solution in solving a problem with higher dimension. This section is devoted to develop a groundwork for this idea by first establishing a relationship between solutions of two problems which the discretizing rate of one problem is twice that of the other.

Recall that, in setting up (4.31), the original problem (4.16) is discretized with the sampling period  $\tau$ . Henceforth, let us assume that the number of sampling intervals takes the form  $N_j = N_1 2^{j-1}$ , whereby the considered sampling period is of the form

$$\tau_{s,j} \triangleq T/(N_1 2^{j-1}) \quad (5.2)$$

for natural number  $j$ . Let  $P_j$  stands for the problem (4.23) associated with  $\tau_{s,j}$ , i.e.,

$$P_j \begin{cases} \max & \tau_{s,j} x^T y \\ \text{s.t.} & -M \leq x_i \leq M, & i = 1, \dots, N, \\ & -\tau_{s,j} D \leq x_1 \leq \tau_{s,j} D, \\ & -\tau_{s,j} D \leq x_{i+1} - x_i \leq \tau_{s,j} D, & i = 1, \dots, N-1, \\ & l_i \leq y_i \leq u_i, & i = 1, \dots, N. \end{cases}$$

Let the discretized bounds on impulse response corresponding to  $P_j$  be denoted by  $u^{(j)}$  and  $l^{(j)}$ . Also, let us denote the optimal solution of  $P_j$  as  $\hat{x}^{(j)}$  and  $\hat{y}^{(j)}$ , and the optimal value of  $P_j$  as  $p_j^*$ .



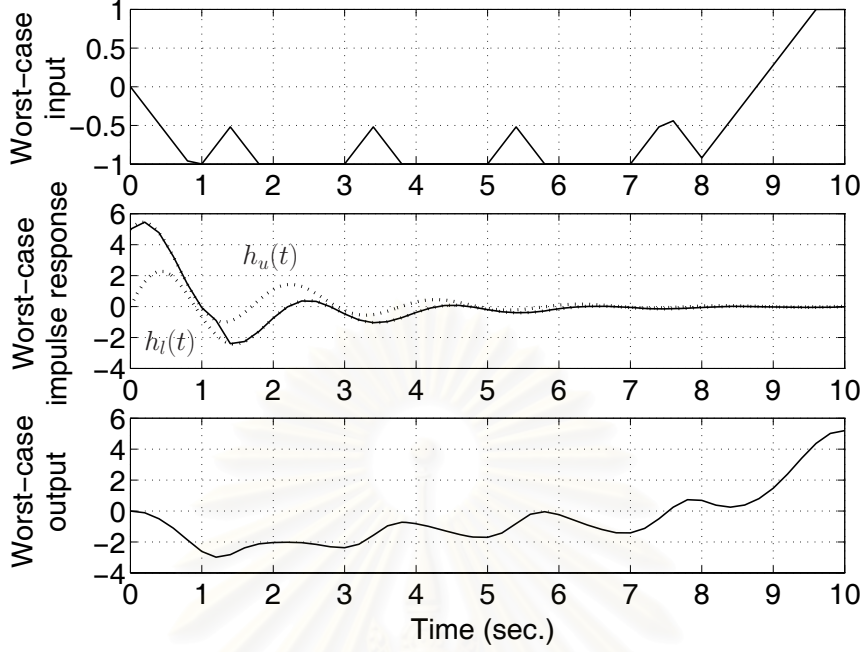


Figure 5.7: The input, impulse response, and output in the worst-case scenario of the WCN computation of system 1 with the dimension of 50. The estimated WCN should equal the magnitude of the worst-case output at  $t = 10$  second.

In order to discuss the solutions of a series of  $P_j$  we need to make  $\hat{x}^{(j)}$  (and also  $\hat{y}^{(j)}$ ) comparable for any two values of  $j$ . In so doing, let us resolve the worst-case discretized input  $\hat{w}^{(j)}[i]$ , and the worst-case discretized impulse response  $\hat{h}^{(j)}[i]$  from  $\hat{x}^{(j)}$  and  $\hat{y}^{(j)}$ , respectively using the 1<sup>st</sup> order hold. Recall that the discretization is conducted with the sampling period of  $\tau_{s,j}$ . Furthermore, let us perform linear interpolations of these time series to obtain continuous-time signals  $\hat{w}^{(j)}(t)$  and  $\hat{h}^{(j)}(t)$ , respectively. These signals are defined on  $(-\infty, T]$ . Intuitively, when  $j$  increases,  $\tau_{s,j}$  becomes smaller and the optimal value of  $P_j$  should approach the WCN of the system. The  $\hat{w}^{(j)}(t)$  should also converge to certain limit as  $j$  increases unless there are multiple accumulation points. In this dissertation, we assume that the selected WCN computational problems conform such behavior. In particular, let us pose the following assumption.

**Assumption 5.1** *Let us consider the WCN computational problem of which the sequence  $\{\hat{w}^{(j)}(t)\}$  is a Cauchy sequence in  $L_\infty$ . This is equivalent to saying that there exist a convergent positive-valued sequence  $\{\alpha_j\}$  such that*

$$\left\| \hat{w}^{(j+1)}(t) - \hat{w}^{(j)}(t) \right\|_\infty \leq \alpha_j, \quad j = 1, \dots$$

where  $\|\cdot\|_\infty$  signifies the supremum norm of function.

If this assumption holds, we can say that  $\hat{w}^{(j+1)}(t)$  is getting closer pointwise to  $\hat{w}^{(j)}(t)$  as  $j$  rises. Actually, for all the WCN computational problems we have tried so far, we observed that this assumption holds. However, a rigorous proof has never been established so far to show that it holds in

general. Assumption 5.1 leads to a beneficial idea of how we may predetermine the solution of  $P_{j+1}$  with the solution of  $P_j$ . This is because  $\hat{w}^{(j+1)}(t)$  should locate in the vicinity of  $\hat{w}^{(j)}(t)$ . Specifically, if the sequence  $\{\alpha_j\}$  exists and is known, then we can deduce a lemma.

**Lemma 5.1** For any  $t$ , if  $\hat{w}^{(j)}(t) > \alpha_j$ , then we have

$$\hat{w}^{(j+1)}(t) > 0.$$

In addition, if  $\hat{w}^{(j)}(t) < -\alpha_j$ , then we have

$$\hat{w}^{(j+1)}(t) < 0.$$

*Proof.* For any  $t$ , it is obvious that

$$\left| \hat{w}^{(j+1)}(t) - \hat{w}^{(j)}(t) \right| \leq \alpha_j,$$

which means

$$\hat{w}^{(j)}(t) - \alpha_j \leq \hat{w}^{(j+1)}(t) \leq \hat{w}^{(j)}(t) + \alpha_j.$$

The results immediately follow.  $\square$

To see how this helps accelerate the BB computation in solving  $P_{j+1}$ , we have to state the following lemma and corollary.

**Lemma 5.2** Consider problem (4.23) when  $x$  is given. An optimal solution  $\hat{y}$  has its elements obtained as

$$\hat{y}_i = \begin{cases} u_i, & x_i > 0, \\ 0, & x_i = 0, \\ l_i, & x_i < 0. \end{cases} \quad (5.3)$$

*Proof.* Let us consider any feasible  $y$ , i.e.,

$$l_i \leq y_i \leq u_i.$$

The proof is straightforward as follows. If  $x_i > 0$ , we have

$$x_i(\hat{y}_i - y_i) = x_i(u_i - y_i) \geq 0.$$

Similarly, if  $x_i < 0$ , we infer

$$x_i(\hat{y}_i - y_i) = x_i(l_i - y_i) \geq 0.$$

This leads to

$$\sum_{i=1}^N x_i(\hat{y}_i - y_i) \geq 0$$

as the terms  $x_i(\hat{y}_i - y_i)$  vanishes if  $x_i = 0$ . Consequently, this suggests that

$$\sum_{i=1}^N x_i(\hat{y}_i - y_i) = x^T(\hat{y} - y) = x^T\hat{y} - x^T y \geq 0$$

As  $\tau_s$  is positive, we have  $\tau_s x^T \hat{y} \geq \tau_s x^T y$ , which completes the proof.  $\square$

An immediate remark should be mentioned here, as we can see from Lemma 5.2 that  $\hat{y}_i$  can be inferred directly from  $x_i$  only (involving no other elements of  $x$ ). Hence, Lemma 5.2 can be restated regardless of temporal variable.

**Corollary 5.1** *Consider the problem of seeking an optimal solution  $\hat{y}$  of (4.23) when some element of  $x$  is given, says only  $i$  in some index set  $\Omega$ . Then,  $y_i$  for  $i \in \Omega$  can be readily obtained as in (5.3).*

Let  $\hat{x}^{(j)}$  be the discretization of  $\hat{w}^{(j)}(t)$  with the sampling period associated with  $P_{j+1}$ , i.e.,  $\tau_{s,j+1}$ . Combining Lemma 5.1 and Corollary 5.1, an important proposition can be stated as follows.

**Proposition 5.1** *If  $\hat{x}_i^{(j)} > \alpha_j$ , then we have*

$$\hat{y}_i^{(j+1)} = u_i^{(j)}.$$

*On the other hand, if  $\hat{x}_i^{(j)} < -\alpha_j$ , then*

$$\hat{y}_i^{(j+1)} = l_i^{(j)}.$$

*Proof.* Let  $t = i\tau_{s,j+1}$ . Given  $\hat{w}^{(j)}(t) > \alpha_j$ , it is clear that  $\hat{x}_i^{(j)} > \alpha_j$ . From Lemma 5.1, we must have  $\hat{w}^{(j+1)}(t) > 0$ , which means  $\hat{x}^{(j+1)}$ . By means of Corollary 5.1, it can be seen that  $\hat{y}_i^{(j+1)} = u_i^{(j)}$ . For the case that  $\hat{w}^{(j)}(t) < -\alpha_j$ , the proposition can be verified in the same fashion.  $\square$

Proposition 5.1 says that the worst-case input obtained via solving  $P_j$  yields a guideline to determine a solution of  $P_{j+1}$ , that is,  $\hat{y}^{(j+1)}$ . So as to formalize our discussion and bypass all the interpolated quantities, a theorem that furnishes a clear relation between elements of  $\hat{y}^{(j+1)}$  and  $\hat{x}^{(j)}$  must be presented. Firstly, a lemma that matches the placements of elements in  $\hat{x}^{(j)}$  with those in  $\hat{x}_i^{(j)}$ , with respect to sampling instants is given.

**Lemma 5.3** *If  $i$  is even, then  $\hat{x}_i^{(j)}$  coincides with  $\hat{x}_{i/2}^{(j)}$ , that is, they locate at the same sampling instant along the time axis. Moreover, we have*

$$\hat{x}_i^{(j)} = \begin{cases} \hat{x}_{(i+1)/2}^{(j)}, & \text{if } i \text{ is odd,} \\ \frac{1}{2} \left( \hat{x}_{i/2}^{(j)} + \hat{x}_{i/2+1}^{(j)} \right), & \text{if } i \text{ is even.} \end{cases}$$

*Proof.* Recall that  $\hat{x}^{(j)} \in \mathbb{R}^{1+N_1 2^{j-1}}$  is associated with  $\tau_{s,j}$ , and  $\hat{x}^{(j)} \in \mathbb{R}^{N_1 2^j}$  are associated with  $\tau_{s,j+1}$ . Next, let  $\hat{i}$  and  $\tilde{i}$  be indices into elements of  $\hat{x}^{(j)}$  and  $\hat{x}^{(j)}$ , respectively. Suppose also that  $\tilde{i}$  is odd. If  $\hat{x}_{\tilde{i}}^{(j)}$  is sampled at the same instant as  $\hat{x}_i^{(j)}$ , then  $(\hat{i} - 1)\tau_{s,j} = (\tilde{i} - 1)\tau_{s,j+1}$ , which implies that

$$\hat{i} = \frac{1}{2}(\tilde{i} + 1)$$

This is because  $\tau_{s,j} = 2\tau_{s,j+1}$ . Hence,  $\hat{x}_{(\tilde{i}+1)/2}^{(j)}$  and  $\hat{x}_{\tilde{i}}^{(j)}$  are both sampled at  $\tilde{i}\tau_{s,j+1}$ . In addition, as  $\hat{x}^{(j)}$  is discretized from  $\hat{w}^{(j)}(t)$  which is interpolated from points of  $\hat{x}^{(j)}$ , we have  $\hat{x}_{(\tilde{i}+1)/2}^{(j)} = \hat{x}_{\tilde{i}}^{(j)}$  if  $\tilde{i}$  is odd. The proof follows by replacing the index  $\tilde{i}$  with  $i$ . In the case that  $i$  is even, we can infer

$$\hat{x}_i^{(j)} = \frac{1}{2} \left( \hat{x}_{i-1}^{(j)} + \hat{x}_{i+1}^{(j)} \right). \quad (5.4)$$

As  $i-1$  and  $i+1$  are odd, we have

$$\begin{aligned} \hat{x}_{i-1}^{(j)} &= \hat{x}_{i/2}^{(j)}, \\ \hat{x}_{i+1}^{(j)} &= \hat{x}_{i/2+1}^{(j)}. \end{aligned}$$

Substituting these into (5.4) completes the proof.  $\square$

The main theorem that provides the explicit relationship between  $\hat{x}^{(j)}$  and  $\hat{y}^{(j+1)}$  can now be given.

**Theorem 5.1** *Given a solution  $\hat{x}^{(j)}$  pertaining to  $P_j$ , we can draw its relationship to a solution  $\hat{y}^{(j+1)}$  of  $P_{j+1}$  as follows:*

- (i) If  $\hat{x}_i^{(j)} > \alpha_j$ , then  $\hat{y}_{2i-1}^{(j+1)} = u_{2i-1}^{(j+1)}$ ;
- (ii) If  $\hat{x}_i^{(j)} < -\alpha_j$ , then  $\hat{y}_{2i-1}^{(j+1)} = l_{2i-1}^{(j+1)}$ ;
- (iii) If  $(\hat{x}_i^{(j)} + \hat{x}_{i+1}^{(j)})/2 > \alpha_j$ , then  $\hat{y}_{2i}^{(j+1)} = u_{2i}^{(j+1)}$ ;
- (iv) If  $(\hat{x}_i^{(j)} + \hat{x}_{i+1}^{(j)})/2 < -\alpha_j$ , then  $\hat{y}_{2i}^{(j+1)} = l_{2i}^{(j+1)}$ ;

*Proof.* From Proposition 5.1, we have

$$\begin{aligned} \hat{x}_i^{(j)} > \alpha_j &\longrightarrow \hat{y}_i^{(j+1)} = u_i^{(j+1)}, \\ \hat{x}_i^{(j)} < -\alpha_j &\longrightarrow \hat{y}_i^{(j+1)} = l_i^{(j+1)}. \end{aligned}$$

If  $\hat{i}$  is odd, write  $\hat{i} = 2i - 1$ , and from Lemma 5.3, it is obvious that

$$\begin{aligned} \hat{x}_i^{(j)} > \alpha_j &\longrightarrow \hat{y}_{2i-1}^{(j+1)} = u_{2i-1}^{(j+1)}, \\ \hat{x}_i^{(j)} < -\alpha_j &\longrightarrow \hat{y}_{2i-1}^{(j+1)} = l_{2i-1}^{(j+1)}. \end{aligned}$$

Hence, (i) and (ii) follow. On the other hand, if  $\hat{i}$  is even, write  $\hat{i} = 2i$ . From Lemma 5.3, we have

$$\begin{aligned} \frac{1}{2}(\hat{x}_i^{(j)} + \hat{x}_{i+1}^{(j)}) > \alpha_j &\longrightarrow \hat{y}_{2i}^{(j+1)} = u_{2i}^{(j+1)}, \\ \frac{1}{2}(\hat{x}_i^{(j)} + \hat{x}_{i+1}^{(j)}) < -\alpha_j &\longrightarrow \hat{y}_{2i}^{(j+1)} = l_{2i}^{(j+1)}. \end{aligned}$$

Then, (iii) and (iv) follow.  $\square$

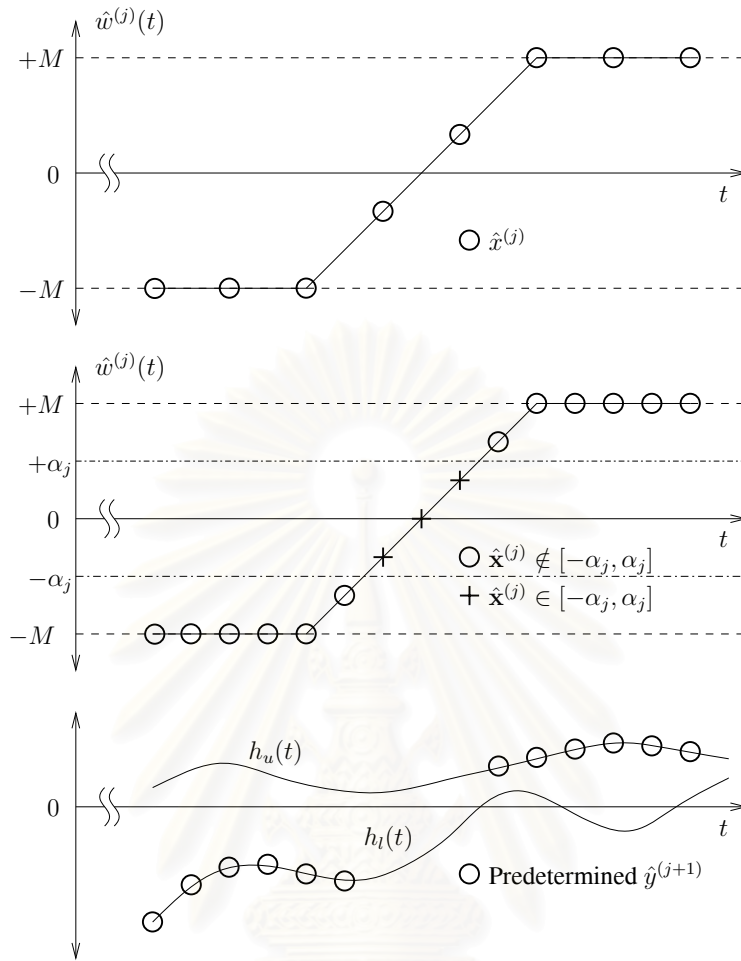


Figure 5.8: The top subplot displays locations of  $\hat{x}_i^{(j)}$  (marked with circle) along the graph of  $\hat{w}^{(j)}(t)$ , while the middle subplot displays the locations of  $\hat{x}_i^{(j)}$  where the circle markers represent the points with  $|\hat{x}_i^{(j)}| > \alpha_j$  and the plus markers represent those with  $|\hat{x}_i^{(j)}| < \alpha_j$ . The bottom subplot displays  $\hat{y}_i^{(j+1)}$  at the same instants where  $|\hat{x}_i^{(j)}| > \alpha_j$ . Some points lie along  $h_u(t)$  and others along  $h_l(t)$ , depending on the sign of  $\hat{x}_i^{(j)}$ .

Following this theorem, one can determine parts of  $\hat{y}^{(j+1)}$  using elements of  $\hat{x}^{(j)}$  before even solving  $P_{j+1}$ . Shortly speaking, the more elements of  $\hat{x}^{(j)}$  satisfy  $|\hat{x}_i^{(j)}| < \alpha_j$ , the more dimensions can be disregarded. Figure 5.8 illustrates an example of the process to obtain elements of  $y^{(j+1)}$  partially by considering the magnitudes and signs of  $\hat{x}^{(j)}$ 's, which is an interpolation of  $\hat{x}^{(j)}$ . Most of the time, this prediction reduce the problem dimension significantly. Our next task is to study how much this technique would accelerate the BB computation.

#### 5.4 Upper Bound on $\{\alpha_j\}$

The discussion in the previous section is based on the assumption that the sequence  $\{\alpha_j\}$  exists and is known to us. However, even though  $\{\alpha_j\}$  may exist, we have not had any method sofar to obtain the sequence in practise. One possible issue to handle this is by estimating an upper bound of  $\alpha_j$ .

Based on our observation,  $\alpha_j$  should be related in a complex manner to  $\tau_{s,j}$  and other problem parameters, namely,  $M$ ,  $D$ ,  $h_u(t)$  and  $h_l(t)$ . As a consequence, the workable upper bound of  $\alpha_j$  should also account for these dependencies. Our hypothesis is that the upper bound for  $\alpha_j$  should be around  $D\tau_{s,j}/2$ . The reason behind this assumption can be roughly described as follows. Let  $\hat{w}(t)$  denote the actually worst-case input, which is a solution to (4.16). Intuitively,  $\hat{w}^{(j)}(t)$  should locate somewhere in the vicinity of  $\hat{w}(t)$ , and should get closer to line up with it as  $j$  becomes larger. Seeing from this view, we may think of  $\hat{w}^{(j)}(t)$  as an estimate of the interpolated signal obtained by discretizing  $\hat{w}(t)$  with  $\tau_{s,j}$ . Now let us consider  $\hat{x}^{(j)}$  and  $\hat{x}^{(j+1)}$  in  $\mathbb{R}^{N_1 2^j}$ . As  $P_{j+1}$  employs higher precision, we are convinced to trust the accuracy of  $\hat{x}^{(j+1)}$  (and this makes more sense when  $j$  is large). Thus, it is preferable to use  $\hat{x}^{(j+1)}$  as a standard and compared  $\hat{x}^{(j)}$  with it along the time axis.

Notice that each point of  $\hat{x}^{(j+1)}$  can change its value at every  $\tau_{s,j+1}$  second, but  $\hat{x}^{(j)}$  can do so at every  $\tau_{s,j}$  second, which is slower. Suppose that  $\hat{x}_i^{(j+1)}$  starts to change its value at a certain sampling instant between two adjacent elements of  $\hat{x}^{(j)}$ , for example, when  $i$  is even (see Lemma 5.3 for an idea). Then,  $\hat{x}^{(j)}$  has to wait for  $\tau_{s,j+1}$  to follow. According to the problem (4.23), two contiguous elements of  $\hat{x}^{(j+1)}$  may not be farther than  $D\tau_{s,j+1}$ . This means that the change in magnitude of  $\hat{x}_i^{(j+1)}$  cannot be greater than  $D\tau_{s,j+1}$ , which consequently implies that the value of  $\hat{x}^{(j+1)}$  can differ from  $\hat{x}^{(j)}$  for at most  $D\tau_{s,j+1} = D\tau_{s,j}/2$ . This explain why the upper bound for  $\alpha_j$  should be around  $D\tau_{s,j}/2$ . Figure 5.9 gives a pictorial perspective of our speculation where  $\hat{w}^{(j)}(t)$  leads  $\hat{w}^{(j+1)}(t)$  by an amount of  $D\tau_{s,j}/2$  in magnitude after 2 second. In this example, the points of  $\hat{x}^{(j)}$  locates at the sampling instants  $t = 1, 3, 5, 7$  (depicted as cross marks), whereas the points of  $\hat{x}^{(j+1)}$  locates at  $t = 1, 2, \dots, 7$  (depicted as circle marks). Here  $\tau_{s,j} = 2$ , and hence,  $\tau_{s,j+1} = 1$ . The input  $\hat{w}^{(j+1)}(t)$  starts to rise at  $t = 2$  while  $\hat{w}^{(j)}(t)$  has to wait until  $t = 3$  to start changing its value in order to catch up with  $\hat{w}^{(j+1)}(t)$ . Both signals climb up at the same rate  $D$ . Within this time horizon, it is clear that the maximum difference of the worst-case inputs, that is,  $\|\hat{w}^{(j+1)}(t) - \hat{w}^{(j)}(t)\|_\infty$ , is equal to  $D\tau_{s,j}/2$ .

It should be note that the aforementioned surmise is loosely based on some assumptions that may not be true in general, so we would want to add up the discarded parts. Specifically, the upper bound should equal  $D\tau_{s,j}/2$  multiplied by a scaling factor greater than one. Let  $\bar{\alpha}_j$  denotes the upper bound of  $\alpha_j$ . Our discussion above culminates in the following assumption.

**Assumption 5.2** Assume that for each  $j$ ,  $\alpha_j$  is bounded by

$$\bar{\alpha}_j = \beta \left( \frac{D\tau_{s,j}}{2} \right) = \beta \left( \frac{TD}{N_1 2^j} \right).$$

where  $1 \leq \beta \leq 2$  depending on the respective WCN computational problems.

The sequence  $\{\bar{\alpha}_j\}$  as a whole will be referred to as the *ambiguity magnitude threshold* with an acronym: AMT. The upper bound  $\bar{\alpha}_j$  at the  $j$ th iteration will also be conveniently mentioned as the AMT with specific  $j$ . The proposed name is intended to carry a meaning of a threshold for  $\hat{x}_i^{(j)}$  whereby we can sort out the elements of  $\hat{y}^{(j+1)}$  which cannot be predetermined (the ambiguous

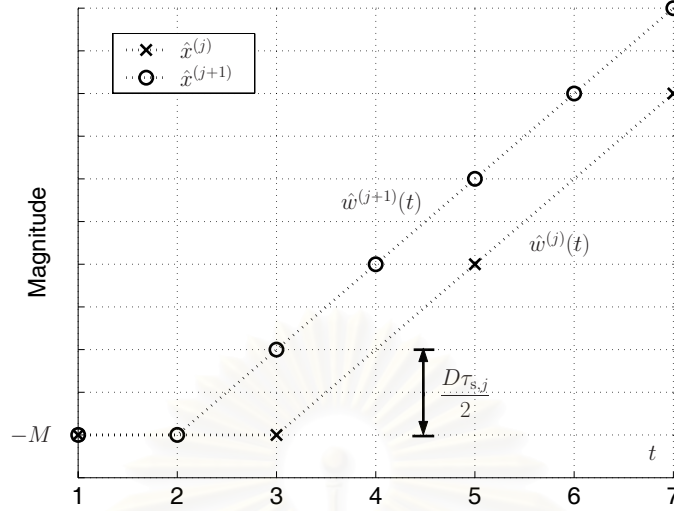


Figure 5.9: Comparison between the worst-case inputs  $\hat{w}^{(j)}(t)$  and  $\hat{w}^{(j+1)}(t)$ . The cross marks represent points of  $\hat{x}^{(j)}$ , while the circle marks represent those of  $\hat{x}^{(j+1)}$ .

elements). Note that, from Assumption (5.2), the first AMT, which is used in the first outer iteration, has the form

$$\bar{\alpha}_1 = \beta \left( \frac{TD}{2N_1} \right). \quad (5.5)$$

According to Assumption 5.2, Theorem 5.1 can be restated here in terms of  $\bar{\alpha}_j$  as follows.

**Proposition 5.2** Assume that  $\bar{\alpha}_j > \alpha_j$  for every  $j$ . Let  $\hat{x}^{(j)}$  be the solution of  $P_j$ , a solution  $\hat{y}^{(j+1)}$  of  $P_{j+1}$  can be predetermined as

- (i) If  $\hat{x}_i^{(j)} > \bar{\alpha}_j$ , then  $\hat{y}_{2i-1}^{(j+1)} = u_{2i-1}^{(j+1)}$ ;
- (ii) If  $\hat{x}_i^{(j)} < -\bar{\alpha}_j$ , then  $\hat{y}_{2i-1}^{(j+1)} = l_{2i-1}^{(j+1)}$ ;
- (iii) If  $(\hat{x}_i^{(j)} + \hat{x}_{i+1}^{(j)})/2 > \bar{\alpha}_j$ , then  $\hat{y}_{2i}^{(j+1)} = u_{2i}^{(j+1)}$ ;
- (iv) If  $(\hat{x}_i^{(j)} + \hat{x}_{i+1}^{(j)})/2 < -\bar{\alpha}_j$ , then  $\hat{y}_{2i}^{(j+1)} = l_{2i}^{(j+1)}$ ;

*Proof.* The proof follows directly from Theorem 5.1 and Assumption 5.2 that  $\bar{\alpha}_j \geq \alpha_j$ .  $\square$

Remark that the parameter  $\beta$  in Assumption 5.2 represents neglected effects of other variables including the difference between the elements of  $u^{(j)}$ ,  $l^{(j)}$  and the corresponding elements of  $u^{(j+1)}$ ,  $l^{(j+1)}$ . In general, to guarantee that Assumption 5.2 is sound and Proposition 5.2 is practical, we want to make the AMT large to make sure that it really bounds  $\alpha_j$  from above for each  $j$ . Nonetheless, the drawback of  $\beta$  being large is that AMT can be more conservative. From Proposition 5.2, this leads to fewer elements of  $\hat{y}^{(j+1)}$  being determined and more are left unambiguous, so the problem dimension of  $P_{j+1}$  has not been much reduced. This would more or less impede the attempt to accelerate the BB computation. The suitable choice of  $\beta$  depends on parameters of individual WCN computational problem.

To support our choice of AMT in Assumption 5.2, we have set up a special experiment that obtain  $\alpha_1$  from a random collection of WCN computational problems. In particular, there are 1,000 problems randomly formulated. Our constructed sets of problem parameters are the combinations of

1. five different problem dimensions, *i.e.*,  $N_1 \in \{8, 9, 10, 11, 12\}$ ;
2. four different derivative bounds that characterized  $\mathcal{W}$ , *i.e.*,  $D \in \{0.4, 0.8, 1.5, 2.5\}$ ;
3. fifty random choices of uncertain systems each described by a randomly-specified stable impulse envelope  $(h_u(t), h_l(t))$ .

In every problem, we set  $M = 1$ . For each problem with the assigned value of  $N_1$ , we solve  $P_1$  and  $P_2$ , *i.e.*, the WCN computational problems when  $\tau_s = T/N_1$  and  $= T/(2N_1)$ , respectively. Then  $\hat{w}^{(1)}(t)$  and  $\hat{w}^{(2)}(t)$  are compared, and the following quantity is computed

$$\hat{\beta} = \frac{2}{D\tau_{s,1}} \left\| \hat{w}^{(2)}(t) - \hat{w}^{(1)}(t) \right\|_{\infty}.$$

This signifies the maximum deviation of the worst-case inputs of two problems normalized by  $D\tau_{s,1}/2$ . The shorthand notation  $\hat{\beta}$  has been purposefully used here to serve as an analogue of the factor  $\beta$  appears in Assumption 5.2. If the assumption holds for a particular WCN computational problem, then  $2\|\hat{w}^{(2)}(t) - \hat{w}^{(1)}(t)\|_{\infty}/(D\tau_{s,1})$  should be bounded from above by  $\beta$ . Hence,  $\hat{\beta}$  can be regarded as empirical counterpart of  $\beta$  experimentally obtained from these 1,000 problems. After solving all the random problems and record  $\hat{\beta}$ , a histogram is plotted to show the range and the distribution of  $\hat{\beta}$  as depicted in Figure 5.10.

In almost every case of our particular WCN problems, the computed values of  $\hat{\beta}$  vary between 1 and 2, which supports Assumption 5.2. Specifically, there are 98.3% of all test problems that Assumption 5.2 holds for  $\beta = 2$ , and this percentage reduces to 91.5% if  $\beta = 1$ . We plot only those problems that yield  $\hat{\beta}$  within the interval  $[0, 2]$ . Note that not all the problems give  $\hat{\beta} \leq 2$ . In fact, there are 17 problems that yield  $\hat{\beta} > 2$  ( $\hat{\beta}$  ranges from 2.13 to 11). We have examine further the problems of which  $\hat{\beta}$  is greater than two (17 out of 1,000 problems), and learn some evidence that provides a good explanation for this instance. It is found that, in each of the 17 problems, there are two or more local solutions which are apart from each other to some degree, but whose optimal objective functions are nearly equal. From those 17 problems, these differences are less than 1%. This is almost like the case where there are multiple limit points of a sequence  $\{\hat{w}^{(j)}(t)\}$ . Since the optimal objective functions associated with these limit points are about the same, at one sampling rate  $\hat{w}^{(j)}(t)$  may be close to one limit point; at another sampling rate it may be close to another limit point, and hence, Assumption 5.2 (and probably Assumption 5.1) does not hold for this specific problem. Nevertheless, this rare phenomenon is not so adverse; despite the two limit points being different, fortunately, their optimal objective functions are close. Thus, if enforcing Assumption 5.2 in addition to Assumption 5.1 causes the sequence  $\{\hat{w}^{(j)}(t)\}$  to converge to a local limit point different from the global one, we can still hope that the optimal objective function corresponding to this local solution



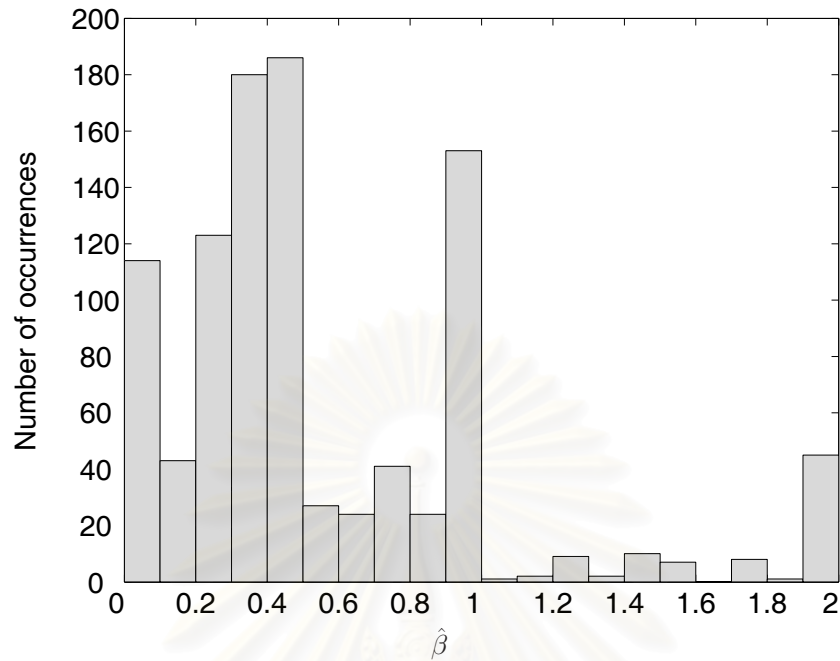


Figure 5.10: Distribution of  $\hat{\beta}$ , which is computed as  $2\|\hat{w}^{(2)}(t) - \hat{w}^{(1)}(t)\|_{\infty}/(D\tau_{s,1})$ , among totally 1,000 test problems.

is very close to that of the global solution, *i.e.*,  $p_j^*$ , and it can be used as a good estimate of the true WCN. In the following section, we firmly introduce the algorithm to expedite our BB process. In Section 5.5.1, we will discuss how to choose proper  $\beta$ .

## 5.5 Hierarchical Branch-and-Bound Algorithm

In this section, we will formulate a hierarchical optimization, which consists of a proposed accelerating technique that invokes a BB algorithm as its subroutine. For convenience, we will call this accelerating technique as *Reduction of AMT* or *RAMT* and call the whole algorithm as *Hierarchical Branch-and-Bound* algorithm, or shortly, *HBB* algorithm. The technique name originates from an idea of accelerating the BB algorithm via reducing AMT by half, performing the BB computation, and exploiting the known solution of the former problem in solving the latter problem. Actually, this is the reason behind the definitions of problem dimension  $N_j = N_1 2^{j-1}$  and of the sampling period  $\tau_{s,j} = T/(N_1 2^{j-1})$  as in (5.2). Let us refer to the iterations of the HBB algorithm as *outer iterations* and to the iterations of the standard BB algorithm as *inner iterations*. Furthermore, let us denote the reduced dimension problem corresponding to  $P_j$  as  $P'_j$ . At the end of  $j$  iteration, the termination criterion for HBB is computed from the relative difference between the optimal values of  $P_j$  and  $P_{j+1}$ , that is,

$$\left| \frac{p_{j+1}^* - p_j^*}{p_j^*} \right|.$$

We update the AMT in every iteration by dividing it by 2. This is the direct result of Assumption (5.2). The procedure of HBB is illustrated in Figure 5.11, and is described as follows.

**begin**

$P'_1 \leftarrow P_1$ ;

solve  $P'_1$  to get  $(\hat{x}^{(1)}, \hat{y}^{(1)})$  and  $p_1^*$ ;

obtain  $\bar{\alpha}_1$ ;

use  $\hat{x}^{(1)}$  and  $\bar{\alpha}_1$  to fix elements of  $\hat{y}^{(2)}$  and obtain  $P'_2$ ;

solve  $P'_2$  via BB algorithm to get  $(\hat{x}^{(2)}, \hat{y}^{(2)})$  and  $p_2^*$ ;

$d_1 = |(p_2^* - p_1^*)/p_1^*|$ ;

$\bar{\alpha}_2 = \bar{\alpha}_1/2$ ;

$j \leftarrow 2$ ;

**while**  $d_j > \epsilon$  **do**

use  $\hat{x}^{(j)}$  and  $\bar{\alpha}_j$  to fix elements of  $\hat{y}^{(j+1)}$  and obtain  $P'_{j+1}$ ;

solve  $P'_{j+1}$  via BB algorithm to get  $(\hat{x}^{(j+1)}, \hat{y}^{(j+1)})$  and  $p_{j+1}^*$ ;

$d_j = |(p_{j+1}^* - p_j^*)/p_j^*|$ ;

$\bar{\alpha}_{j+1} = \bar{\alpha}_j/2$ ;

$j \leftarrow j + 1$ ;

**end**

**end**

In each outer iteration, the BB algorithm is executed. The computation period is saved due to the fact that the dimension of  $P'_j$  has been reduced. For better understanding, suppose that the computation of the BB algorithm takes exponential time<sup>3</sup>. Specifically, suppose that this computation time is  $\mathcal{O}(2^N)$  where  $N$  is the problem dimension. Roughly speaking, the HBB algorithm can be viewed as an attempt to avoid solving one large-scale problem by solving other smaller-scale problems. Let say that it divides a problem with dimension  $N$  into two problems with lower dimensions  $N_1, N_2$ . Assuming<sup>4</sup> that  $N_1 + N_2 = N$  and  $N_1 \geq N_2$ , the sum of computation times of the two subproblems should be bounded by  $\mathcal{O}(2^{N_1}) + \mathcal{O}(2^{N_2})$ . Due to the fact that

$$2^{N_1} + 2^{N_2} \leq 2^{N_1+1} \leq 2^N,$$

we should expect the computation time consumed by HBB to be much faster than the standard BB algorithm (and we will see later in this chapter that it is very much faster). However, up to this point, HBB is still not complete as the way to acquire  $\bar{\alpha}_1$  is not given. This will be the topic of the next section. After that, we present the experiment to verify the accelerated computation and investigate

<sup>3</sup>The computation time of the BB algorithm is unpredictable, but the worst-case is guarantee to be exponential time. The time consumed by each WCN computational problem depends critically on problem parameters.

<sup>4</sup>The first assumption actually depends on several factors. However, in real WCN computations, we have observed that, when HBB algorithm divides a large-scale problem into small-scale problems, the sum of dimensions of these divided problems is much less than that of the original large-scale problem.

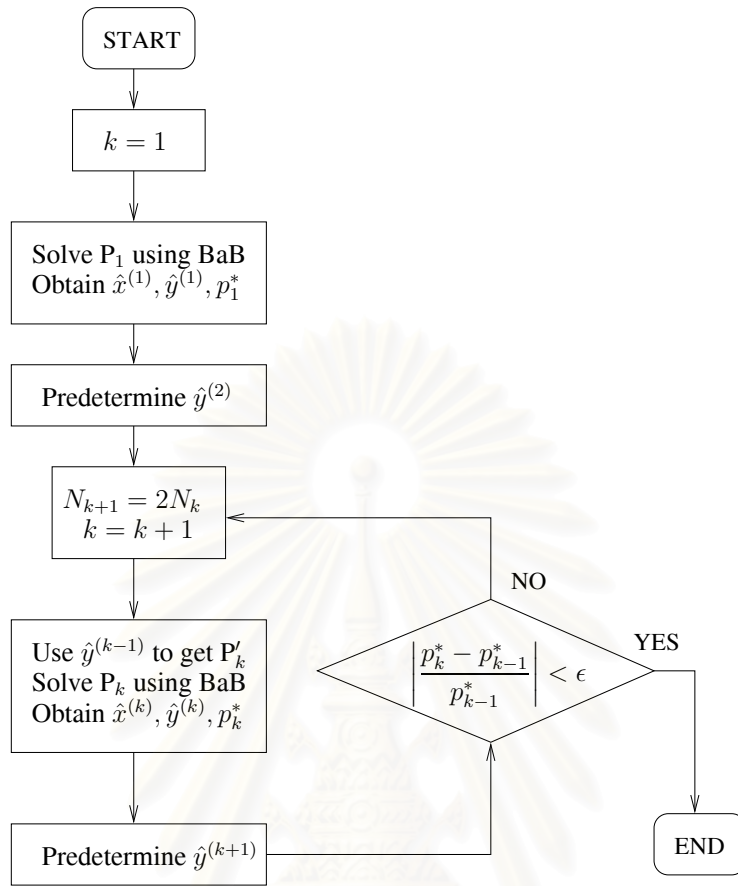


Figure 5.11: Flowchart of HBB algorithm.

its efficiency.

### 5.5.1 Initiation of HBB

To start HBB, we need to determine  $\bar{\alpha}_1$ , which is given in (5.5). Hence, our task is actually to find appropriate  $\beta$  and  $N_1$ . To have HBB function properly, it is necessary that  $\bar{\alpha}_1$  is less than  $M$ . Otherwise, after the first outer iteration we will not be able to determine any element of  $\hat{y}^{(2)}$  via Theorem 5.2 because  $|\hat{x}^{(1)}|$  is always less than  $\bar{\alpha}_1$ , which is greater than  $M$ . In practise, not only we need  $\bar{\alpha}_1 < M$ , but we also prefer to have  $\bar{\alpha}_1$  less than  $M$  to some extent, which will allow more elements of  $\hat{y}^{(2)}$  to be predetermined, making  $P'_2$  to be of lower dimension. This specification induces the following constraint on  $\bar{\alpha}_1$ :

$$\bar{\alpha}_1 \leq \left( \frac{\gamma}{\gamma + 1} \right) M \quad (5.6)$$

where  $\gamma$  is any positive number. Note that  $\gamma/(\gamma + 1)$  is a scaling factor that ranges from zero to one. We particularly introduce this form of scaling factor because the bounding constraint on  $\bar{\alpha}_1$  can be conveniently adjusted; that is, if we prefer to gradually raise  $\bar{\alpha}_1$  towards  $M$ , we can specify  $\gamma$  as an increasing sequence of positive integer: 1, 2, 3, ... and so on. This constraint on the first AMT

as in (5.6) directly suggests another equivalent constraint on  $N_1$ . By combining (5.5) and (5.6), we obtain

$$N_1 \geq \beta \left( \frac{\gamma + 1}{\gamma} \right) \left( \frac{TD}{2M} \right). \quad (5.7)$$

This relationship reveals the explicit trade-off between three free parameters, namely,  $N_1$ ,  $\beta$  and  $\gamma$ .

Firstly, as our primary goal is to speed-up the BB computation, we prefer  $N_1$  to be as small as possible. Secondly, we prefer to have Assumption 5.2 valid for a particular WCN computational problem. This is related to the requirement of  $\beta$  being large. However, this conflicts with the first objective as it can be seen from (5.7) that larger  $\beta$  gives rise to larger  $N_1$ . Another point to discuss is the choice of  $\gamma$ . This parameter directly effects the dimensions of  $P_1$  (equals  $N_1$ ) and  $P'_2$ . If  $\gamma$  is large, from (5.7),  $N_1$  can be made small but  $\bar{\alpha}_1$  will be large so that fewer elements of  $\hat{y}^{(2)}$  can be predetermined leaving  $P'_2$  to be of high dimension. On the contrary, as mentioned earlier, if  $\gamma$  is small, the dimension of  $P'_2$  could be made smaller, but this would require  $N_1$  be somewhat large. These specifications on  $N_1$ ,  $\beta$  and  $\gamma$  can be summarized in order of their priorities as follows:

1.  $N_1$  is relatively small;
2.  $\gamma$  should not be too large or small;
3.  $\beta$  is close to 2.

With the relation (5.7) available as a guideline, we can choose the proper parameters in accordance with the WCN computational problem. In this work, we give a simple heuristic way to specify these parameters for user convenience. After several executions of the random problems used in Section 5.4, it is observed that the computation times for solving the problems with dimensions less than 20 are within a minute or two on 2.8 GHz Pentium IV PC with 512 MB of RAM, which are acceptable to work with. Hence, we prefer to make  $N_1$  less than 20, if possible. However, if  $N_1$  can be made lower, we will either increase  $\beta$  or decrease  $\gamma$ . Let  $\lceil \cdot \rceil$  stands for rounding to the nearest integer larger than itself. With reference to (5.7), the parameter selection is established in five cases as follows

1.  $\frac{TD}{2M} > 20$ ; in this case,  $N_1$  must be greater than 20. To keep  $N_1$  small for the sake of computation time, we need to set  $\beta = 1$ , and  $\gamma$  to be extremely large. From (5.7), this causes  $N_1$  to approach  $TD/2M$ . As  $N_1$  has to be an integer, we select

$$N_1 = \left\lceil \frac{TD}{2M} \right\rceil + 1.$$

Then, from (5.5), we have  $\bar{\alpha}_1 = TD/2N_1$ . Note that we need not to obtain  $\gamma$  numerically as it no longer plays any role since  $\bar{\alpha}_1$  has been determined already. It should be noted that because  $\gamma$  is large, then perhaps only elements of  $\hat{y}^{(2)}$  that are associated with elements of  $\hat{x}^{(1)}$  on the boundary, may be determined.

2.  $15 < \frac{TD}{2M} \leq 20$ ; in this case,  $N_1$  can be made lower than or equal to 20. Thus, this leaves some margin for reducing  $\gamma$ , but this may cause  $N_1$  to exceed 20 for certain values of  $TD/2M$ . Here,

we set  $\gamma = 10$  and still  $\beta = 1$ . Thus,  $\bar{\alpha}_1$  can be obtained from (5.5) where  $N_1$  is calculated as

$$N_1 = \left\lceil \frac{11}{20} \left( \frac{TD}{M} \right) \right\rceil.$$

In this case  $N_1$  range from 17 to 22.

3.  $10 < \frac{TD}{2M} \leq 15$ ;  $N_1$  is moderate in this case, so we can both raise  $\beta$  and reduce  $\gamma$  a bit. The parameters are chosen as  $\beta = 1.2$  and  $\gamma = 9$  so that  $\bar{\alpha}_1$  can be obtained from (5.5) where

$$N_1 = \left\lceil \frac{2}{3} \left( \frac{TD}{M} \right) \right\rceil.$$

In this case  $N_1$  range from 14 to 20.

4.  $5 < \frac{TD}{2M} \leq 10$ ; in this case  $N_1$  can be made small, which gives more room for increasing  $\beta$  while decreasing  $\gamma$ . Here, we set  $\beta = 1.5$  and  $\gamma = 5$ . Then,  $\bar{\alpha}_1$  is determined from (5.5) where

$$N_1 = \left\lceil \frac{9}{10} \left( \frac{TD}{M} \right) \right\rceil.$$

In this case  $N_1$  range from 9 to 18.

5.  $\frac{TD}{2M} \leq 5$ ; for this last case  $N_1$  can be made very small so we decrease  $\gamma$  down to 3 while raising  $\beta$  up to 2. Then,  $\bar{\alpha}_1$  is obtained from (5.5) where

$$N_1 = \left\lceil \frac{4}{3} \left( \frac{TD}{M} \right) \right\rceil.$$

In this case  $N_1$  is at most 14.

It is remarked that this is only a guideline for parameter selection. In Cases 3 and 4, users may reduce  $\beta$  so as to decrease  $N_1$  further. However, as mentioned in Section 5.4, it should be kept in mind that the optimized value may not be the actual global optimum if the chosen value of  $\beta$  does not make WCN computational problem comply with Assumption 5.2. What we can say is that, from the Figure 5.10, the coverage of the WCN computational problems given by Assumption 5.2 with  $\beta = 2$  is 98.3% while the coverage given with  $\beta = 1$  is 91.5%.

## 5.6 Numerical Examples

The three examples of uncertain linear systems employed in Section 5.1 are considered again. Their impulse envelopes are depicted in Figures 4.3(a)–4.3(c). We exploit the standard BB algorithm developed in Section 5.1 to seek for the benchmark solution in order to validate HBB algorithm. Suppose that HBB algorithm returns a vertex as the one corresponding to the global solution. If such vertex is the same as that given by the standard BB algorithm, this evidently implies that HBB algorithm functions correctly.

In particular, a validating experiment has been set up as follows. The bounds on input resemble those in Section 5.2, that is,  $M = 1$  and  $D = 1.2$ . The tested WCN computational problems are obtained from combinations of these circumstances:

- Both BB and HBB algorithms are used to solve the problems.
- WCNs are computed for system 1, 2, and 3.
- Problem dimensions  $N$  are selected from  $\{18, 20, 22, 24, 26\}$ .

This makes up 30 tested problems. The reason for choosing  $N$  to be even because we aim to run HBB algorithm for at least two outer iterations; otherwise, if there is only one outer iteration, HBB algorithm would function exactly like a standard BB computation. Hence, at least we need to have  $N = 2N_1$ , which makes  $N$  even. For  $N = 18, \dots, 26$ , we start with  $N_1 = 9, \dots, 13$ , respectively. For such cases, HBB algorithm takes only two outer iterations. For  $N = 48$ , we start with  $N = 12$ , and then, HBB algorithm takes three outer iterations, that is,  $48 = 12 \times 2^{(3-1)}$ . It is important to note that for  $N$  beyond 52, the standard BB computation will suffer from data storage problem. This is because the number of active nodes is getting larger plus that one node requires a space of  $8N$  bytes of storage. So, we choose  $N$  to be at most 48 in this experiment.

The computation times are recorded, and the solutions of HBB algorithm are checked with those of the BB algorithm. We find that HBB algorithm issues the solutions similar to those given by standard BB computation, and hence, it is justified. Let us consider especially the case when  $N = 48$ . Table 5.2 displays the computation times consumed by each method when applied to three uncertain systems. For the BB computation, Figures 5.12(a)–5.12(f) illustrate the convergence of the upper and lower bounds and the number of active nodes for the WCN computations of system 1 to system 3. Notice how large the number of iteration and the total number of active nodes for the BB computation of the WCN of system 1. This gives rise to its strongly impractical computation time as in Table 5.2. The convergence of the two bound for the case of system 1 is magnified within the range of 1,000 iteration, and revealed in Figure 5.13 in order to clarify the magnitudes of these bounds during the beginning of the algorithm.

Table 5.2: Comparison between the computation times of the BB algorithm and HBB algorithm used in computing the WCN of the three uncertain systems when  $N = 48$ .

Methods	System 1	System 2	System 3
<b>BB algorithm</b>	5 days	7.2 mins	7.4 mins
<b>HBB algorithm</b>	17.8 secs	10.1 secs	10.0 secs

It can be seen from Table 5.2 that HBB algorithm greatly saves CPU times in computing the WCN of all example systems. The computation time of 5 days for system 1 is reduced to just 17 seconds. To reveal the effectiveness of HBB algorithm, we perform a very high-dimensional WCN computation, *i.e.*,  $N = 1280$ , starting with  $N_1 = 10$ , for system 1. The computation takes 1 minute 48 seconds and the WCN is computed to be 5.192. The results are shown in Figure 5.14(a)–5.14(d). The trend of the WCN in Figure 5.14(a) explicitly converges to 5.192. The termination criteria shown in Figure 5.14(b) decreases in linear rate. The worst-case signals should be compared to Figure 5.7 as when  $N = 50$ .

## 5.7 Summary

In this chapter, we apply the BB method, which is a global optimization approach, to compute the WCN of uncertain linear systems. Our branching strategy proceeds along the most implausible component first, which is more likely to give fast convergent rate. In numerical examples, the BB algorithm efficiently yields global solutions for the WCN problems with dimensions up to 20. Moreover, these solutions are consistent with the WCN given by the exhaustive search. For the systems satisfying certain assumptions, the acceleration of the BB method is proposed. Specifically, we invent a double-level optimization, called HBB algorithm, which contains the BB computation as an inner loop. The parameter selection is determined heuristically and the numerical examples are given to exhibit the great improvement and advantage of the WCN computation through HBB algorithm over the standard BB algorithm.



สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย

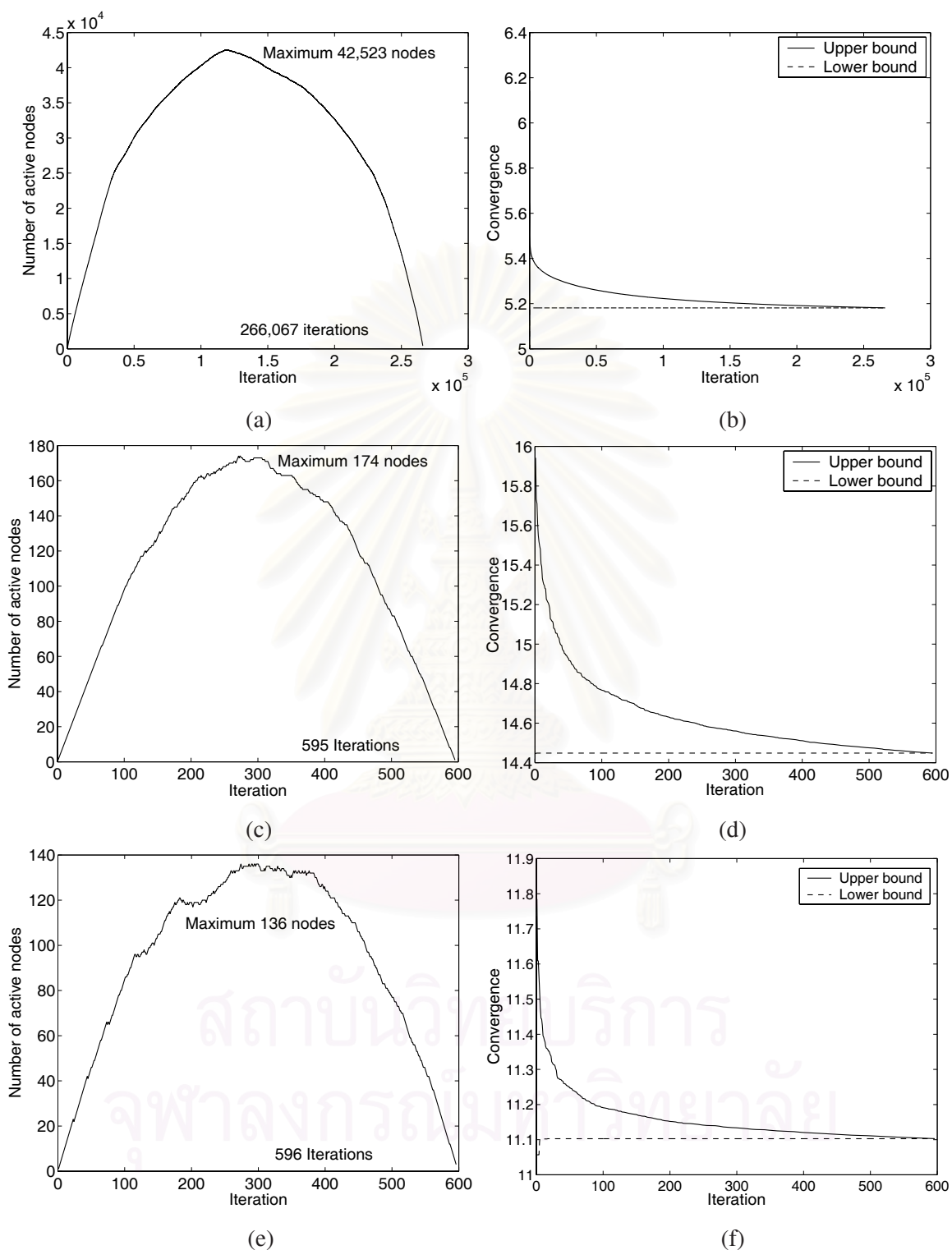


Figure 5.12: For WCN computation with  $N = 48$ , the number of active nodes at each iteration: (a) system 1, (c) system 2, and (e) system 3, and the convergence of upper and lower bounds: (b) system 1, (d) system 2, (f) system 3. For the case of system 1, the number iteration is excessively high; the relevant computation time is about 5 days while the other two cases take only about 7 minutes.



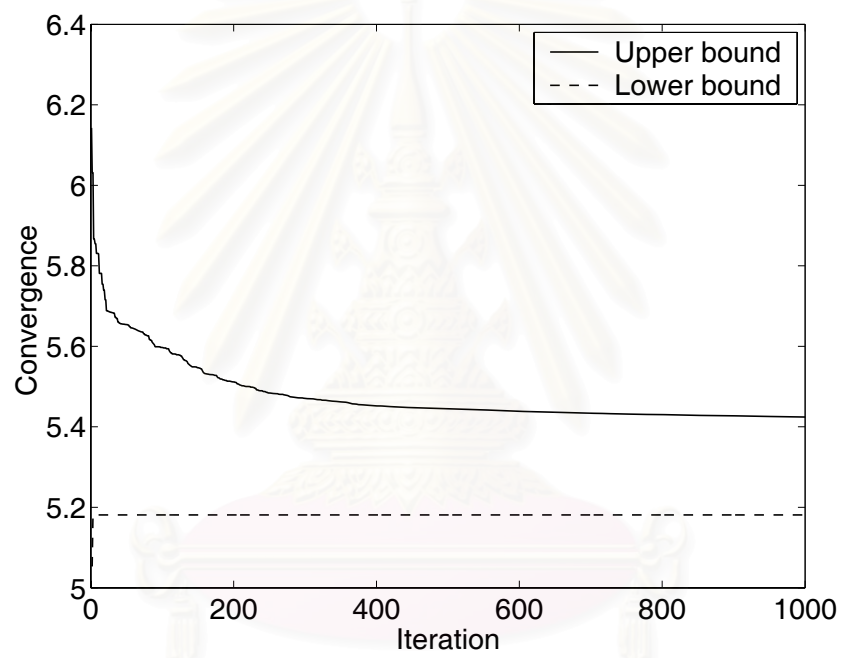


Figure 5.13: The convergence of the upper and the lower bounds of the WCN computation for system 1 when  $N = 48$ , redrawn within the range from the first iteration through the 1,000th iteration.

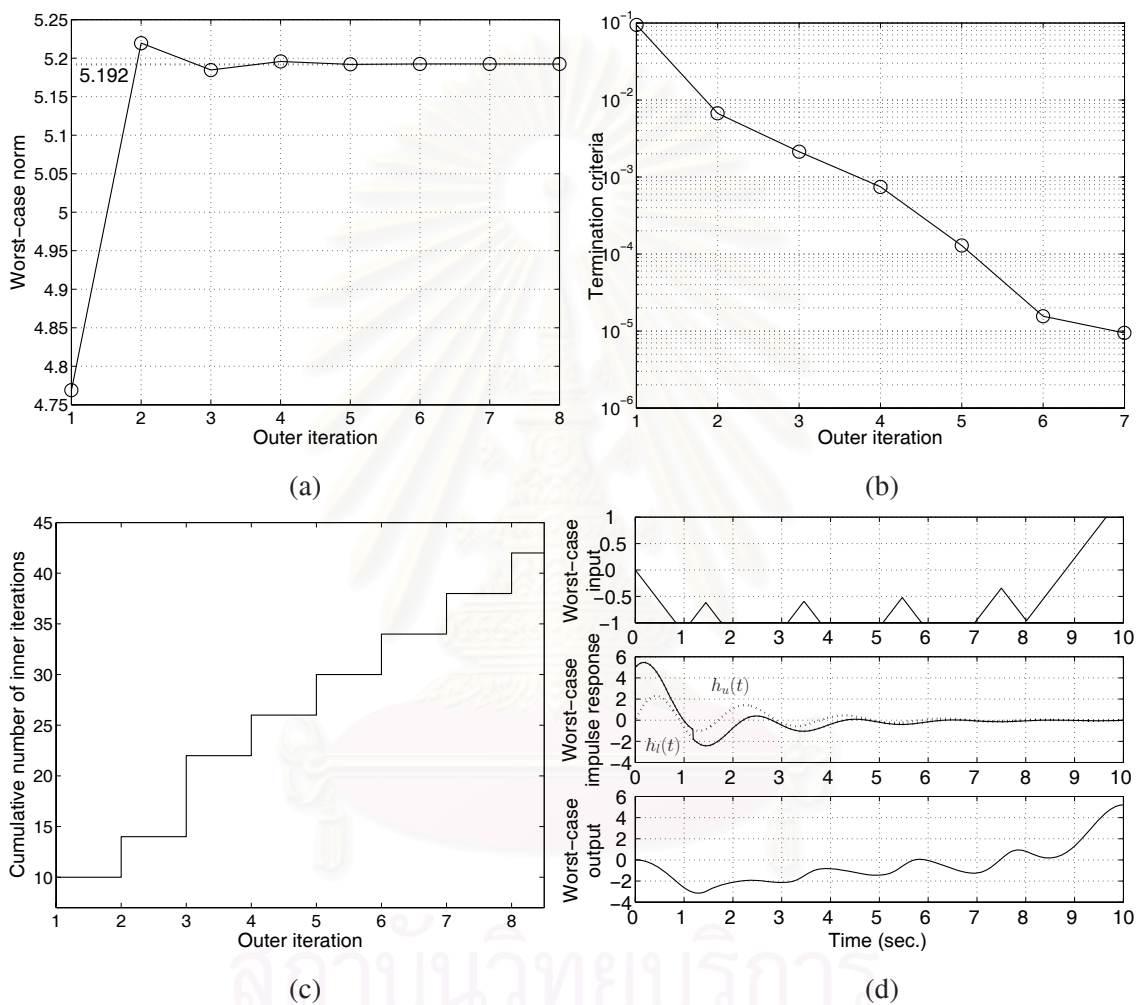


Figure 5.14: Computation of the WCN of system 1 via HBB algorithm when  $N = 1280$ : (a) the values of the WCN converges to certain limit, (b) the value of  $|(p_{j+1}^* - p_j^*)/p_j^*|$  at  $j$ th iteration, (c) the cumulative number of inner iterations (BB iterations) at each outer iteration (HBB iteration), and (d) the input, impulse response, and output in the worst-case scenario.

## CHAPTER VI

### APPLICATION TO ACTIVE SUSPENSION SYSTEM

The research on vehicle suspension control has been studied by system engineers and researchers for a few decades to achieve smooth suspension performance. The early approach employed by several forerunners is the passive suspension control for which the suspension spring and damper are carefully designed to serve passenger comfort. Nonetheless, the absence of energy supplied to suspension system has become the major drawback of the passive suspension technique, hence, the demand for the active counterpart. The active suspension control, a modification of the common passive suspension control system by augmenting a force actuator, exerts additional control effort to the system which has brought in an improvement to counteract road roughness.

In a vehicle active suspension system, changes in road condition disturb vehicle travel, and thus, regulatory control is required to suppress vertical acceleration for riding comfort, while the relative movement between suspension structures is maintained at a safe distance to avoid structural damage [61–63]. Typically, the disturbance generated by road roughness is observed as a finite energy signal comes in ripples modelled as multiple pulses or salient portions of a sinusoidal curve [63–66]. However, in this dissertation, a harsher terrain is of interest which is modelled as a finite power signal with bounded magnitude and rate. In such circumstance, the road surface deviates over the time from the flat reference level. In addition to this disturbance, our suspension system is also incorporate a load mass uncertainty due to unknown-yet-limited passengers' weights in various situations. This uncertain load mass often causes significant difficulty in suspension control [66].

In this chapter, we consider the WCNs, which are the maximum magnitude of vehicle chassis acceleration and suspension deflection when the suspension system is subjected to road disturbances and uncertainty in total vehicle mass. It indicates the performance of control system in rejecting disturbances. Recall that the WCN generally represents the worst-case performance for which the lower the value implies the better performance. Thus, it is usually desirable to keep the WCN at minimal. As the dynamic model of the active suspension system considered here is linear time-invariant (LTI), we are dealing with the WCN of LTI systems driven by disturbances with bounded magnitudes and rates. The main contribution of this chapter is the application of the WCN to the active suspension control system whose disturbance and uncertainty follow the performance analysis framework. Specifically, a PID controller design method is employed in a multi-objective design problem based on the WCN.

In Section 6.1, the dynamics of active suspension system and characteristics of road disturbances are described. The control design specifications are given in Section 6.2. In Section 6.3, a ready-to-use search algorithm is employed to design a multi-objective PID controller for the active suspension system. Lastly, the concluding remarks are given in Section 6.4.

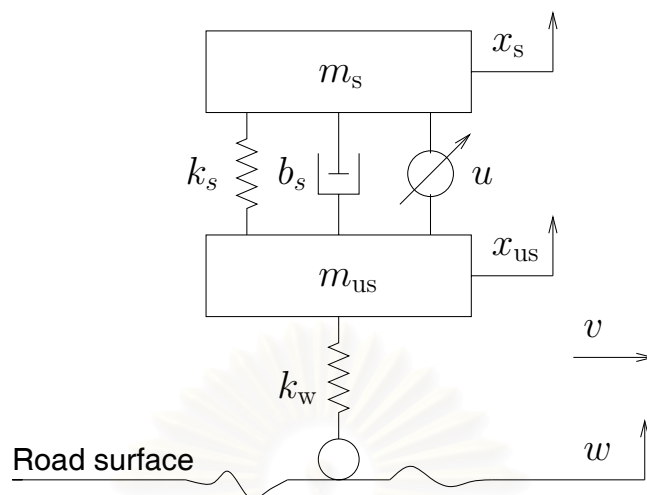


Figure 6.1: The schematic diagram of a vehicle active suspension system.

## 6.1 Dynamic Model of Active Suspension Systems

A simplified version of four-wheel vehicle suspension system, which is known as a quarter-car model, is displayed in Figure 6.1 where solely one-dimensional movement is considered as that one wheel is assumed to serve a quarter of a vehicle [61, 67]. In this picture,  $k_s$  is the passive spring, and  $b_s$  is the damper representing a shock absorber. The spring  $k_t$  stands for the tire compressibility. The unsprung mass  $m_{us}$  denotes the mass of one wheel, while the total chassis mass (including load mass) are divided by four to obtain the sprung mass  $m_s$ . Part of the sprung mass is the load mass  $m_l$  that is assumed to vary from 10 to 200 kg, that is, the total load mass range from 40 to 800 kg per a full vehicle.

In the same figure,  $u$  stands for the actuator force supplied to the system. The displacements  $x_{us}$  and  $x_s$  are associated with the unsprung and sprung masses, respectively. The vehicle velocity is denoted as  $v$ . Later in this section, this velocity will be used to elaborate the profile of road disturbance. The parameters of the active suspension system are obtained from [67] and are given below:

- The spring coefficient:  $k_s = 130,000$  N/m,
- The damper coefficient:  $b_s = 9,800$  N·s/m,
- The spring coefficient:  $k_t = 1,000,000$  N/m,
- The unsprung mass:  $m_{us} = 20$  kg,
- The sprung mass:  $m_s = m_l + 375$  kg,
- The load mass:  $10 \leq m_l \leq 200$  kg.

The deflections  $x_s, x_{us}$  are measured from their static positions. In addition, the contact of the tire with the road surface is assumed to remain at all times. The dynamic equation of the active suspension

is then of the form:

$$\begin{aligned}\dot{x}(t) &= Ax(t) + B \begin{bmatrix} u(t) \\ w(t) \end{bmatrix}, \\ y(t) &= Cx(t)\end{aligned}$$

where  $A$ ,  $B$ , and  $C$  are respectively given by

$$\begin{aligned}A &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{k_s}{m_s} & -\frac{b_s}{m_s} & \frac{k_s}{m_s} & \frac{b_s}{m_s} \\ 0 & 0 & 0 & 1 \\ \frac{k_s}{m_{us}} & \frac{b_s}{m_{us}} & -\frac{k_s + k_t}{m_{us}} & -\frac{b_s}{m_{us}} \end{bmatrix}, \\ B &= \begin{bmatrix} 0 & 0 \\ 0 & \frac{1}{m_s} \\ 0 & 0 \\ \frac{k_t}{m_{us}} & -\frac{1}{m_{us}} \end{bmatrix}, \\ C &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ -\frac{k_s}{m_s} & -\frac{b_s}{m_s} & \frac{k_s}{m_s} & \frac{b_s}{m_s} \\ 1 & 0 & -1 & 0 \end{bmatrix}.\end{aligned}$$

The state vector  $x = [x_s, \dot{x}_s, x_{us}, \dot{x}_{us}]^T$ , and the performance output vector  $y = [x_s, \ddot{x}_s, x_s - x_{us}]^T$ .

The physical meanings of these output elements are as follows:

- $x_s$ : chassis (passenger) travel,
- $\ddot{x}_s$ : chassis (passenger) acceleration,
- $x_s - x_{us}$ : suspension deflection.

It is assumed that only the first output element, the chassis displacement, is available for measurement to compensate the system dynamics. As mentioned earlier, the characteristics of the road roughness is modelled as a signal with bounded magnitude and rate. Consider earlier works, it is reasonable to assume that the deviation of road surface does not exceed 10 cm, and its rate of change is within 1 m [64, 65]. For better insight, let the vehicle travel forwards with the velocity around  $v = 30$  km/h. This means that the road disturbance would displace from zero to its peak at 10 cm in at least about 83 cm of horizontal distance. This suggests that the maximum slope of the road surface is approximately  $\tan^{-1}(0.1/0.83) = 7$  degree for this assumption of  $v$ .

From the above discussion, the set of all admissible disturbance inputs, representing road roughness, denoted by  $\mathcal{W}$  which is expressed as

$$\mathcal{W} \triangleq \{w(t) : |w(t)| \leq 0.1, |\dot{w}(t)| \leq 1, \forall t \geq 0\} \quad (6.1)$$

where  $w(t) = 0, \forall t < 0$ .

## 6.2 Control Design Specifications

The control design specifications for active suspension system are listed as follows.

- I In general, the control design aims to simultaneously minimize chassis acceleration and suspension deflection for passenger comfort. However, it is a standard fact that small transfer functions of both outputs cannot be achieved at the same time [68]. This brings about the trade-off between both performance objectives. In this work, a specification is added to reduce the passenger travel; hence, in brief, it is desirable to keep all performance outputs relatively small.
- II As mentioned previously, the suspension deflection is subjected to a hard constraint to avoid suspension *bottoming* which may lead to serious vehicle damage. The limit is selected to be 8 cm according to [66]. Thus, we have

$$|x_s(t) - x_{us}(t)| < 0.08, \quad \forall t \geq 0. \quad (6.2)$$

- III As opposed to passive suspension, active suspension needs to take into account the maximum control effort which can be produced by the force actuator. The maximum control signal is assumed to be around 5 kN, which consequently places a hard constraint on control input as

$$|u(t)| < 5,000, \quad \forall t \geq 0. \quad (6.3)$$

This practical saturation limit usually degrades the plausible regulatory performance that would be achieved theoretically. Here, the WCN can be applied to control signal to address its limit. However, for the sake of practicality, instead of computing the WCN of the control signal, we compute the worst-case magnitude  $\xi(T)$  as defined in (4.2) at  $T = 5$ . Even though the road roughness keep fluctuating around the nominal level, we may assume that it should end at some finite time on actual road surface. If the vehicle takes the same speed as earlier, *i.e.*,  $v = 50$  km/h, the period of 5 second implies the rough surface of the distance around 70 meters. It should be noted that using  $T = 5$  will not affect the computations of WCN of other signals as time constants of the suspension systems linked with these signals are relatively small, *i.e.*, all less than 1 second. An attempt to attain a compromise between these performance specifications for the uncertain active suspension system bring forth to a multi-objective robust controller design problem. The analysis of the worst-case performance of uncertain linear systems in the next section would yield a means for possible resolution.

### 6.3 PID Controller Design Procedure

A basic PID controller is used here to manifest an application of the WCN in controller design. The block diagram of a classical PID control scheme is depicted in Figure 6.2. The active suspension system transfer function matrix is represented by  $G(s, m_1)$  where the second argument indicates the dependency of the transfer function matrix on the uncertain passenger's load mass. The PID controller has a conventional parallel form with approximate derivative for practical implementation. Its transfer function is as follows.

$$K \left( 1 + \frac{1}{\tau_I s} + \frac{\tau_D s}{\epsilon \tau_D s + 1} \right). \quad (6.4)$$

The controller parameters to be designed are

- $K$ : the proportional gain,
- $\tau_I$ : the reset time,
- $\tau_D$ : the derivative time.

The approximation factor  $\epsilon$  for the derivative action is fixed at 0.1. Let the controller of the form (6.4) be parameterized with  $(K, \tau_I, \tau_D)$ . Suppose that  $H(s, m_1)$  stands for a SISO closed-loop system, from  $w(t)$  to either  $u(t)$  or any of performance outputs, evaluated at a fixed load mass  $m_1$ . Let  $h(t, m_1)$  is the corresponding impulse response of  $H(s, m_1)$  at each  $m_1$ . To compute the WCN as described in the foregoing chapter, the impulse envelope is first determined. Theoretically, the formulas for  $h_u(t)$  and  $h_l(t)$  at each  $t$  are given by

$$\begin{aligned} h_u(t) &= \max_{10 \leq m_1 \leq 200} h(t, m_1), \\ h_l(t) &= \min_{10 \leq m_1 \leq 200} h(t, m_1) \end{aligned} \quad (6.5)$$

In this work,  $h(t, m_1)$  is evaluated at selected values of load mass, namely,  $m_1 = 10, 100, 150$ , and 200 kg. Then, the impulse response bounds at any time instant  $t$  are simply estimated as

$$\begin{aligned} h_u(t) &= \max\{h(t, 10), h(t, 100), h(t, 150), h(t, 200)\}, \\ h_l(t) &= \min\{h(t, 10), h(t, 100), h(t, 150), h(t, 200)\}. \end{aligned}$$

It should be noted that more samples of  $m_1$  can be taken to compute impulse responses, yet the estimations of  $h_u(t)$  and  $h_l(t)$  are improved only marginally. For the WCN computation, we choose  $N = 384$  because it is high enough to yield acceptable accuracy while the computation still consumes reasonable time. The WCN is computed using HBB as described in Section 5.5. A procedure to design the PID controller for the suspension system consists of three steps:

- Obtain a rough initial guess through Ziegler-Nichols (Z-N) open-loop test.
- Tune the controller roughly via trial-and-error method.
- Fine tune using a numerical search algorithm called the *moving boundary process* or MBP [69].

The MBP can be used in nonlinear feasibility problem. We must first supply a good initial point to start this algorithm, and it will proceed by generating a series of trial points that may eventually be feasible. If the process is trapped by local minimum, a new starting point needs to be supplied. The algorithm, however, is not guaranteed to find the feasible point.

To begin PID controller design, the Ziegler-Nichols open-loop test is conducted by injecting a unit-step control force  $u(t)$  into  $G(s, m_1)$  and obtain the DC-gain of 1, the deadtime of 0.01 second, and the time constant of 0.03 second. These gives the following controller parameters: (3.6, 0.02, 0.005). Nonetheless, this is not an appropriate choice as it causes violation in suspension deflection constraint (6.2). This problem is resolved by increasing  $\tau_D$  and  $\tau_I$  to 1 second, and then raise  $K$  significantly to 2000. To start the MBP, we roughly search for a suitable initial point via

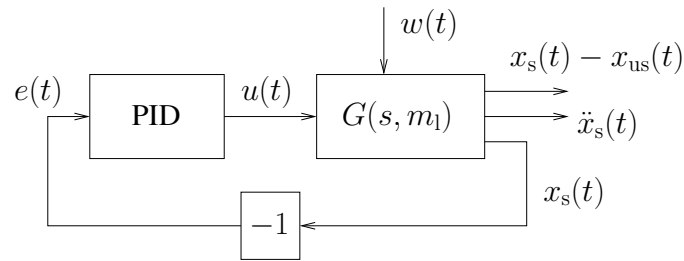


Figure 6.2: The block diagram of classical PID control for an active suspension system.

trial-and-error method by observing the WCN of closed-loop mappings associated with the performance outputs and adjust the PID parameters. It is found that both passenger travel and suspension deflection exhibit similar trends against the changing controller parameters, while the passenger acceleration take the inverse tendency. Furthermore, adjusting  $\tau_D$  seems not much influence any control performance except that of the suspension deflection. Hence, we try to tune  $K$  and  $\tau_I$  first to achieve a relatively good performance with control effort kept under its bound. Then, we adjust  $\tau_D$  to correct the suspension deflection. Note that it makes sense to raise both  $K$  and  $\tau_I$  to have better performance while keeping the control effort minimum. During the tuning process, we also pay some attention to transient behaviors, *e.g.*, settling time and rise time, as they help improve overall performance, but are not explicitly related to the WCN. As a result, we come up with a good starting point (3000, 30, 1). Then, we input it in to the MBP. The search algorithm returns the PID parameters (4465, 32, 0.8). Table 6.1 compares the performances between two suspension control systems with the PID parameters obtained from the Z-N tuning method against that obtained from the MBP. These performances are the WCNs of closed-loop mappings from  $w(t)$  to each of  $y(t)$  and to  $u(t)$ . It can be seen that the Z-N tuning method yields the closed-loop system that violates the design specification of the suspension deflection (6.2). On the other hand, the MBP yields the closed-loop system that satisfies all the design specifications with overall performances better than that via the Z-N tuning method.

Table 6.1: The WCNs of the PID-control active suspension systems when the PID parameters are obtained from the Z-N tuning method and the MBP.

Design Method	Control force (kN)	Chassis acceleration ( $\text{m/s}^2$ )	Chassis travel (cm)	Suspension deflection (cm)
Z-N	0.09	72.9	17.9	9.4
MBP	4.30	75.6	14.9	6.7

According to the PID parameters given by the MBP, a simulation is conducted for the closed-loop active suspension system in comparison with a basic passive suspension system, of which control force is absent. The testing road disturbance in  $\mathcal{W}$  is randomly selected as depicted in Figure 6.3. We



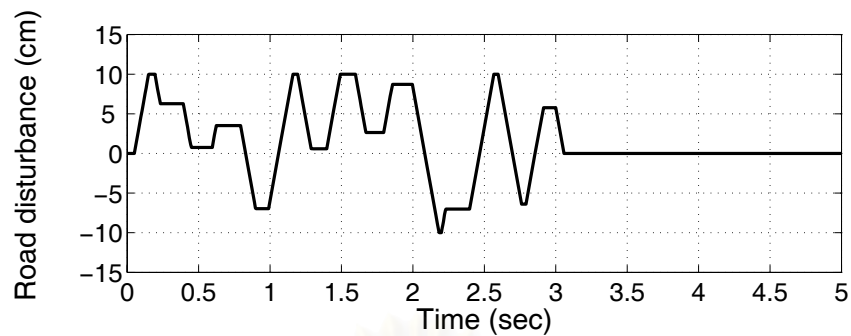


Figure 6.3: Disturbance representing road roughness profile used in simulation of suspension systems.

intend to have  $w(t)$  vanish after three second so as to observe the settling behaviors of the closed-loop responses. In Figure 6.4(a) to 6.4(c), responses of chassis acceleration and suspension deflection are shown for two load masses, namely,  $m_1 = 10$  kg and 200 kg. It can be seen from the figure that suspension deflections of all load masses lie within  $\pm 8$  cm, satisfying (6.2). The control efforts also conform with the bound (6.3) as depicted in Figure 6.5(a) to 6.5(c). The overall responses by the active suspension control reveal superior performance than that of the passive counterpart.

#### 6.4 Summary

A PID controller design for active suspension system is presented based on the WCN of uncertain LTI systems under disturbances with magnitude and rate bounds. The design procedure comprises the initial use of the Z-N tuning rule, the trial-and-error method for rough tuning, and the fine tuning using MBP. The designed controller enhances the passengers' comfort under road roughness disturbances and provides robustness against load mass uncertainty in active suspension system. The simulation results confirm that the performances satisfy the multi-objective design specifications with the robustness against uncertain load mass. When the disturbance model is closely matched the actual road condition, the closed-loop performance becomes realistic, and hence, the control design becomes effective.

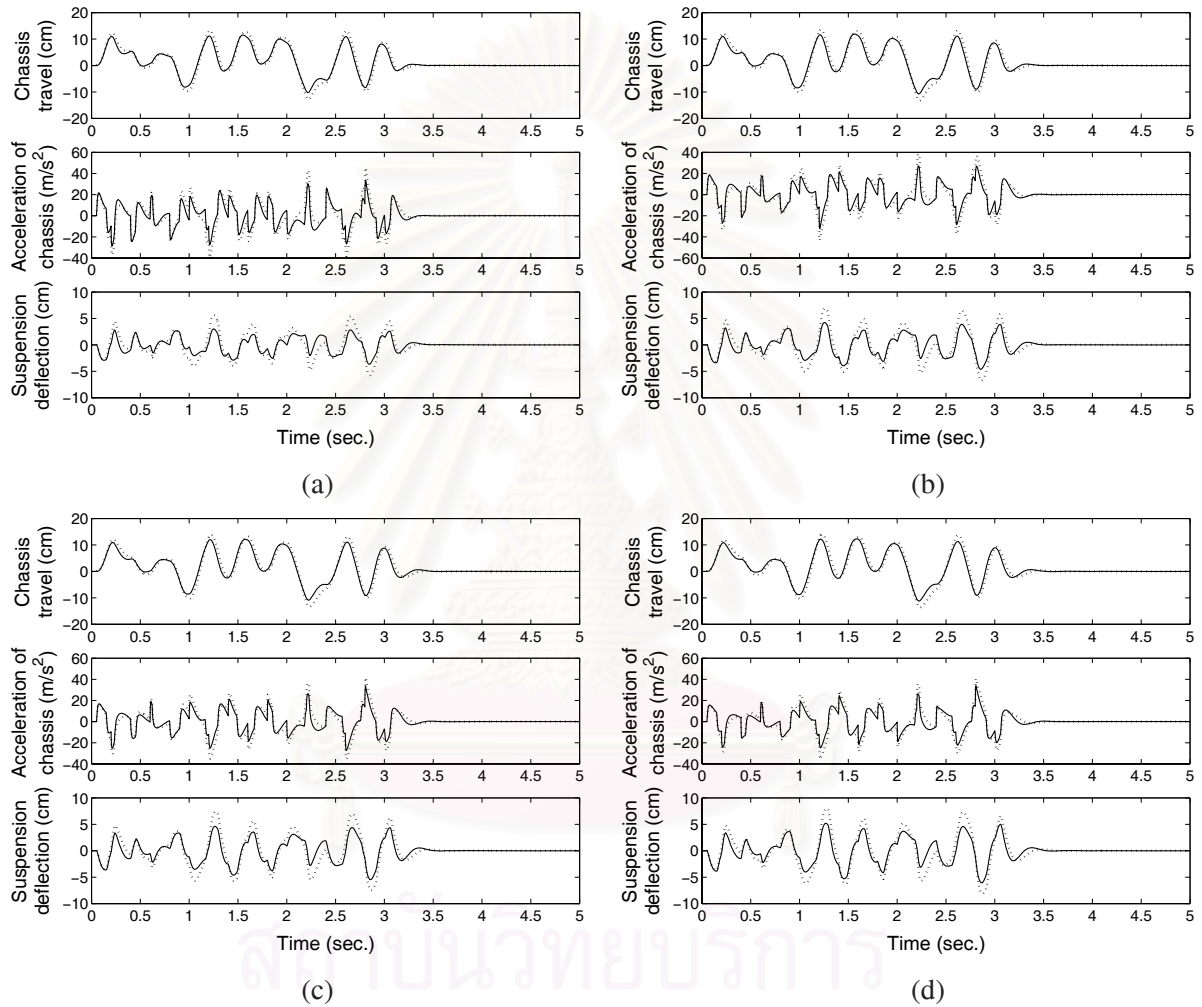


Figure 6.4: Comparison of vehicle chasis accelerations and suspension travels between passive systems (dotted line), and PID-control active systems (solid line): (a)  $m_1 = 10$  kg, (b)  $m_1 = 100$  kg, (c)  $m_1 = 150$  kg, (d)  $m_1 = 200$  kg.

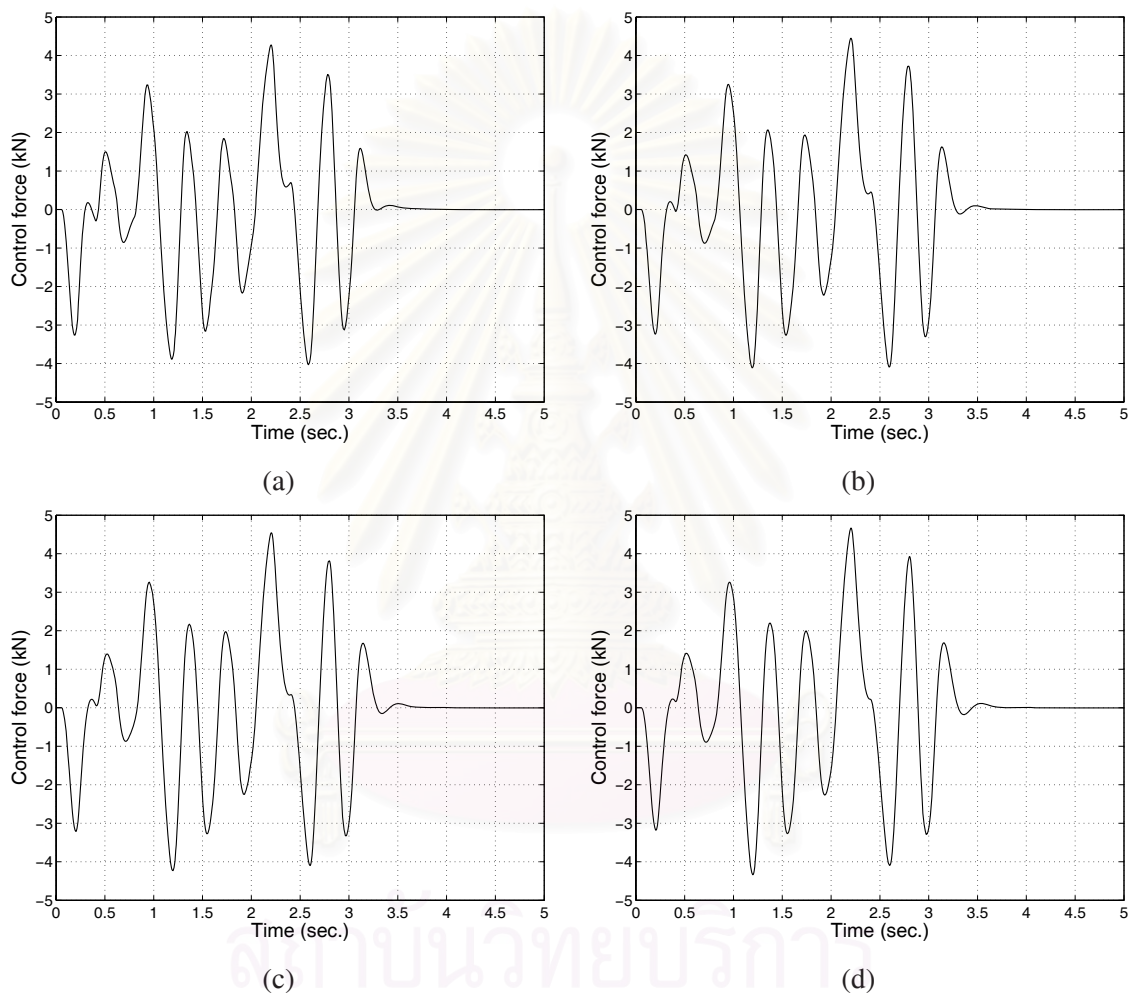


Figure 6.5: Control forces, provided by the PID controller with  $K = 3794$ ,  $\tau_I = 46$  and  $\tau_D = 1.62$ , acting on the active suspension control systems: (a)  $m_1 = 10$  kg, (b)  $m_1 = 100$  kg, (c)  $m_1 = 150$  kg, (d)  $m_1 = 200$  kg. The force magnitudes are always maintained within the allowable limit of  $\pm 5$  kN.

## CHAPTER VII

### CONCLUSIONS

#### 7.1 Summary

In this dissertation, the WCN computational problem is discussed in twofold: when a linear system of interest includes and does not include uncertainty. The latter case is considered first in Chapter 2 and 3, and the former case which is of major concern is considered in Chapter 4 and 5. When there is no uncertainty the WCN computational problem is formulated as an optimal control problem and the algorithm SPIS is introduced to compute the WCN. The algorithm is validated with the second-order systems whose WCNs are known analytically. The accuracy of SPIS is also compared with another standard computational method via discrete-time formulation. The validation results are positive.

In the presence of uncertainty, the optimal control formulation does not appear to be promising for WCN computation; hence, the discrete-time formulation is applied instead. The result is an  $\mathcal{NP}$ -hard problem with requiring function comparison at local solutions whose number grows exponentially with problem dimension. The upper and lower bounds of the WCN are presented and are applied to the standard BB algorithm, which is later validated with the exhaustive search. The results show that the BB computation functions correctly, but its execution speed is still impractical for problem with high dimension. For such reason, the algorithm HBB is developed based on the BB algorithm to accelerate the computation. HBB is in turn validated with the BB algorithm, resulting as positive. The major achievement of this dissertation is the development of HBB itself. In this Section 5.6, it is revealed that *HBB remarkably reduces the computation time of five days to less than half a minute*. This makes the WCN computation of high-dimensional problem practical, thereby control design problem based on the WCN objectives/constraints can be realized. The example of PID controller tuning is illustrated in Chapter 6. The closed-loop responses of the PID control active suspension system designed based on the WCN performance specifications are compared with the (open loop) responses of the passive suspension system. The former control system gives performances better than the latter, which proves the benefit of the WCN application in control system design.

#### 7.2 Further Improvements

There are several improvements that can be introduced to increase the efficiency of HBB in terms of computation time and accuracy. These improvements are listed as follows:

1. In the discrete-time formulation, the method used in estimating the convolution integral

$$\int_0^T h(T-t)w(t)dt$$

can be replaced with other technique that yields higher estimation accuracy. The simple one is the Simpson's rule whose estimation error is bounded by  $\mathcal{O}(\tau_s^5 \phi^{(4)})$  where

$$\phi(t) = h(T - t) * w(t).$$

In this case,  $w^{(3)}(t)$  and  $w^{(4)}(t)$  need to be defined properly for the inputs in  $\mathcal{W}$ . However, this may add only marginal improvement to the accuracy.

2. The upper bound of the WCN in Theorem 4.1 can be refined by adding other necessary conditions. From (4.23), we have the following constraints on adjacent components of  $x_i$

$$\begin{aligned} x_{i+1} - x_i &\leq \tau D, \\ x_i - x_{i+1} &\leq \tau D, \end{aligned}$$

for  $k = 1, \dots, N - 1$ . Combining these inequalities, It is obvious to see that, for any  $i$  and  $j$

$$\begin{aligned} \tilde{x}_i - \tilde{x}_{i+j} &\leq j\tau D \\ \tilde{x}_i - \tilde{x}_{i-j} &\leq j\tau D. \end{aligned}$$

By multiplying both sides of both equations with  $\tilde{y}_i$  and re-arrange them, we obtain

$$\begin{aligned} \tilde{x}_i \tilde{y}_i &\leq \tilde{x}_{i+j} \tilde{y}_i + j\tau D \\ \tilde{x}_i \tilde{y}_i &\leq \tilde{x}_{i-j} \tilde{y}_i + j\tau D, \end{aligned}$$

which consequently leads to

$$\begin{aligned} \tilde{x}_k \tilde{y}_k &\leq (u_k - l_k) \tilde{x}_{k+m} + m\tau D \\ \tilde{x}_k \tilde{y}_k &\leq (u_k - l_k) \tilde{x}_{k-m} + m\tau D. \end{aligned}$$

Hence, the constraints on  $\tilde{z}_i$  are derived as

$$\begin{aligned} \tilde{z}_i &\leq (u_i - l_i) \tilde{x}_{i+j} + j\tau D \\ \tilde{z}_i &\leq (u_i - l_i) \tilde{x}_{i-j} + j\tau D, \end{aligned}$$

for each  $i$  and for a positive index  $j$  which is less than  $2M/D$ . Let  $\hat{j}$  be the greatest integer less than or equal to  $2M/D$ . The refined upper bound is the optimal value of the following LP:

$$\begin{aligned} \max \quad & \tau(\mathbf{1}^T \tilde{z} + l^T \tilde{x} - M\mathbf{1}^T \tilde{y} - M\mathbf{1}^T l) \\ \text{s.t.} \quad & 0 \leq \tilde{x}_i \leq 2M, & i = 1, \dots, N, \\ & M - \tau D \leq \tilde{x}_1 \leq M + \tau D, \\ & \tilde{x}_{i+1} - \tilde{x}_i \leq \tau D, & i = 1, \dots, N - 1, \\ & \tilde{x}_i - \tilde{x}_{i+1} \leq \tau D, & i = 1, \dots, N - 1, \\ & 0 \leq \tilde{y}_i \leq u_i - l_i, & i = 1, \dots, N, \\ & \tilde{z}_i \leq (u_i - l_i) \tilde{x}_i, & i = 1, \dots, N, \\ & \tilde{z}_i \leq 2M \tilde{y}_i, & i = 1, \dots, N, \\ & \tilde{z}_i \leq (u_i - l_i) \tilde{x}_{i+j} + j\tau D, & i = 1, \dots, N, \quad j = 1, \dots, \hat{j} \\ & \tilde{z}_i \leq (u_i - l_i) \tilde{x}_{i-j} + j\tau D, & i = 1, \dots, N, \quad j = 1, \dots, \hat{j} \end{aligned} \tag{7.1}$$

The total number of constraints in this LP is counted to be  $(2\hat{j} + 8)N$  constraints. Even though the refined upper bound is tighter, the computation time of this LP may be unnecessarily lengthy compared to the LP in Theorem 4.1.

3. In solving for the upper or lower bound of WCN, the LP solver, *i.e.*, the interior point method, can be modified. In particular, the solver of the unconstrained logarithmic-barrier optimization may be changed from the standard Newton's method to other Quasi-Newton methods that reduce computational burden in each iteration by estimating the Hessian matrix instead of obtaining the exact one.
4. By taking  $h_u(t)$  and  $h_l(t)$  into account during the theoretical setup for the HBB algorithm, we may be able to improve the accuracy of  $\bar{\alpha}_j$ , *i.e.*, the AMT at the  $j$ th iteration. Specifically, the quantities  $u^{(j)}$ ,  $l^{(j)}$ ,  $u^{(j+1)}$ , and  $l^{(j+1)}$  should be exploited in Assumption 5.1 and 5.2. In fact, it is preferable to construct a primary assumption on  $h_u(t)$  and  $h_l(t)$  that implies Assumption 5.1 and 5.2. As a result, Assumption 5.1 and 5.2 can then be restated as propositions or theorems based on such a primary assumption.
5. To start the HBB algorithm, a more systematic way to choose  $N_1, \beta$  and  $\gamma$  can be constructed so that each term will be adjusted in order of their priority with respect to the size of  $TD/2M$ .

### 7.3 Possible Extensions

1. **MIMO systems:** The concept of the WCN can be applied to *multiple-input multiple-output (MIMO)* strictly proper, finite-dimensional, causal, uncertain linear time-invariant systems with proper definition of the WCN. We will briefly reveal that the WCN of MIMO systems can be obtained from the WCN of SISO systems. Let  $\mathbf{h}(t) \in \mathbb{R}^{m \times p}$  be the impulse response matrix associated with a MIMO uncertain linear system with an input vector  $\mathbf{w}(t) \in \mathbb{R}^p$ , and output vector  $\mathbf{z}(t) \in \mathbb{R}^m$ . Each element of  $\mathbf{w}(t)$ , denoted by  $w_j(t)$  for  $j = 1, \dots, p$ , is assumed to fall in the input set  $\mathcal{W}_j$  defined as

$$\mathcal{W}_j \triangleq \{w_j(t) : w_j(t) = 0, \forall t \leq 0; |w_j(t)| \leq M_j, |\dot{w}_j(t)| \leq D_j, \forall t > 0\}$$

where  $M_j, D_j$  are finite positive numbers. Let  $h_{ij}(t)$  represent the element of  $\mathbf{h}(t)$  in the  $i$ th row and the  $j$ th column. Furthermore, assume that  $h_{ij}(t)$  be in the set  $\mathbb{H}_{ij}$  defined as

$$\mathbb{H}_{ij} \triangleq \{h_{ij}(t) : \underline{h}_{ij}(t) \leq h_{ij}(t) \leq \bar{h}_{ij}(t), \forall t\}$$

where  $\underline{h}_{ij}(t) \leq \bar{h}_{ij}(t)$ , and  $\underline{h}_{ij}(t), \bar{h}_{ij}(t) \in \mathbb{H}_0$ . The WCN of MIMO systems is defined as the maximum of the worst-case peak magnitude among the elements of  $\mathbf{z}(t)$ , denoted by  $z_i(t)$ . This suggests that

$$\begin{aligned} \|\mathbf{h}\|_{\text{wc}} &\triangleq \max_{1 \leq i \leq m} \sup_{t \geq 0} \sup_{w_j \in \mathcal{W}_j} \sup_{h_{ij} \in \mathbb{H}_{ij}} |z_i(t)| \\ &= \max_{1 \leq i \leq m} \sup_{t \geq 0} \sup_{w_j \in \mathcal{W}_j} \sup_{h_{ij} \in \mathbb{H}_{ij}} \left| \sum_{j=1}^p [h_{ij}(t) * w_j(t)] \right|. \end{aligned}$$

If  $\underline{h}_{ij}(t)$  and  $\bar{h}_{ij}(t)$  are stable for every  $(i, j)$ , it can be shown that

$$\begin{aligned}
\|\mathbf{h}\|_{\text{wc}} &= \max_{1 \leq i \leq m} \lim_{t \rightarrow \infty} \max_{w_j \in \mathcal{W}_j} \max_{h_{ij} \in \mathbb{H}_{ij}} \left| \sum_{j=1}^p [h_{ij}(t) * w_j(t)] \right| \\
&\leq \max_{1 \leq i \leq m} \lim_{t \rightarrow \infty} \max_{w_j \in \mathcal{W}_j} \max_{h_{ij} \in \mathbb{H}_{ij}} \sum_{j=1}^p |h_{ij}(t) * w_j(t)| \\
&= \max_{1 \leq i \leq m} \sum_{j=1}^p \left\{ \lim_{t \rightarrow \infty} \max_{w_j \in \mathcal{W}_j} \max_{h_{ij} \in \mathbb{H}_{ij}} |h_{ij}(t) * w_j(t)| \right\} \\
&= \max_{1 \leq i \leq m} \sum_{j=1}^p \|h_{ij}\|_{\text{wc}}.
\end{aligned}$$

This concludes that

$$\|\mathbf{h}\|_{\text{wc}} \leq \max_{1 \leq i \leq m} \sum_{j=1}^p \|h_{ij}\|_{\text{wc}}. \quad (7.2)$$

However, if we consider  $\hat{w}_j(t)$  and  $\hat{h}_{ij}(t)$ , which are the worst-case input and the worst-case impulse response associated with the worst-case magnitude  $\xi_{ij}(t)$  at  $t$ , it can be seen that

$$\lim_{t \rightarrow \infty} \hat{h}_{ij}(t) * \hat{w}_j(t) = \|h_{ij}\|_{\text{wc}} \quad (7.3)$$

for each  $(i, j)$ . Since  $\hat{w}_j(t) \in \mathcal{W}_j$  and  $\hat{h}_{ij}(t) \in \mathbb{H}_{ij}$ , (7.3) suggests that

$$\begin{aligned}
\max_{1 \leq i \leq m} \sum_{j=1}^p \|h_{ij}\|_{\text{wc}} &= \max_{1 \leq i \leq m} \sum_{j=1}^p \left\{ \lim_{t \rightarrow \infty} \hat{h}_{ij}(t) * \hat{w}_j(t) \right\} \\
&\leq \max_{1 \leq i \leq m} \sum_{j=1}^p \left\{ \lim_{t \rightarrow \infty} \max_{w_j \in \mathcal{W}_j} \max_{h_{ij} \in \mathbb{H}_{ij}} [h_{ij}(t) * w_j(t)] \right\} \\
&= \max_{1 \leq i \leq m} \lim_{t \rightarrow \infty} \max_{w_j \in \mathcal{W}_j} \max_{h_{ij} \in \mathbb{H}_{ij}} \sum_{j=1}^p [h_{ij}(t) * w_j(t)] \\
&\leq \max_{1 \leq i \leq m} \lim_{t \rightarrow \infty} \max_{w_j \in \mathcal{W}_j} \max_{h_{ij} \in \mathbb{H}_{ij}} \left| \sum_{j=1}^p [h_{ij}(t) * w_j(t)] \right| \\
&= \|\mathbf{h}\|_{\text{wc}}.
\end{aligned} \quad (7.4)$$

From (7.2) and (7.4), it follows that

$$\|\mathbf{h}\|_{\text{wc}} = \max_{1 \leq i \leq m} \sum_{j=1}^p \|h_{ij}\|_{\text{wc}}. \quad (7.5)$$

Hence, the WCN of MIMO systems,  $\|\mathbf{h}\|_{\text{wc}}$ , can then be computed from that of the SISO systems,  $\|h_{ij}\|_{\text{wc}}$ . The algorithm to obtain the WCN of SISO uncertain linear systems is already given in this dissertation.

2. **Delayed systems:** The WCN can also be extended to delayed systems, which is a type of infinite dimensional systems. For time delay systems, the WCN can be computed directly from that of the delay-free systems. Let  $h(t)$  be the strictly proper, causal, uncertain linear time-invariant systems. Suppose that it is contained in  $\mathbb{H}$  with the impulse envelope of  $(h_u(t), h_l(t))$ . Let  $h_d(t)$  be defined in terms of  $h(t)$  with the time delay of  $\tau_d$ , that is,

$$h_d(t) = h(t - \tau_d)$$

It can be easily seen that the uncertain delayed system  $h_d(t)$  can be represented by the set  $\mathbb{H}_d$ , which contains all systems  $h_d(t)$  such that

$$h_l(t - \tau_d) \leq h_d(t) \leq h_u(t - \tau_d), \quad \forall t.$$

The impulse envelope  $(h_u(t - \tau_d), h_l(t - \tau_d))$  can be directly processed through the discrete-time formulation as described in Section 4.2, and the WCN can be computed accordingly.

3. **Biproper systems:** When an uncertain linear system of interest is proper but not strictly proper, the computation of the WCN becomes slightly different. The impulse response of biproper systems comprises two parts: the strictly proper part, and the feedthrough term. Specifically, let the impulse response of the biproper system be denoted by  $h_b(t)$ . We can always express this response as

$$h_b(t) = h(t) + d\delta(t) \tag{7.6}$$

where  $h(t)$  is an impulse response of an uncertain strictly proper system whose WCN can be computed,  $\delta(t)$  is the Dirac delta function, and  $d$  is the feedthrough constant. The uncertain impulse response  $h(t)$  is assumed to be in a set  $\mathbb{H}$ , and the constant  $d$  is assumed to be unknown but lie in an interval  $[d_{\min}, d_{\max}]$ . The convolution integral  $h_b(T) * w(T)$  can be expressed as

$$\begin{aligned} h_b(T) * w(T) &= \int_0^T h_b(T-t)w(t)dt \\ &= \int_0^T h(T-t)w(t)dt + d \int_0^T \delta(T-t)w(t)dt \\ &= h(T) * w(T) + dw(T). \end{aligned}$$

With this expression, the discretized problem can be formulated to compute  $\xi(T)$  as follows:

$$\begin{aligned} \max \quad & \tau_s(x^T y) + dx_N \\ \text{s.t.} \quad & -M \leq x_i \leq M, \quad i = 1, \dots, N, \\ & -\tau_s D \leq x_1 \leq \tau_s D, \\ & -\tau_s D \leq x_{i+1} - x_i \leq \tau_s D, \quad i = 1, \dots, N-1, \\ & l_i \leq y_i \leq u_i, \quad i = 1, \dots, N, \\ & d_{\min} \leq d \leq d_{\max}. \end{aligned}$$

where the vector quantities are defined from the discretized variables as

$$\begin{aligned} x_i &= w[i], \quad i = 1, \dots, N, \\ y_i &= h[N-i], \quad i = 1, \dots, N-1, \quad y_N = \frac{1}{2}h[0], \\ u_i &= h_u[N-i], \quad i = 1, \dots, N-1, \quad u_N = \frac{1}{2}h_u[0], \\ l_i &= h_l[N-i], \quad i = 1, \dots, N-1, \quad l_N = \frac{1}{2}h_l[0]. \end{aligned}$$



4. **Asymmetric magnitude bound:** In this dissertation, the disturbance magnitude is assumed to be within the symmetric range  $[-M, +M]$ . To generalize the WCN, we consider the situation when the upper bound and the lower bound of the input magnitude are not symmetric, that is,

$$M_1 \leq w(t) \leq M_2, \quad \forall t$$

where  $M_1 \leq M_2$ , but  $M_1, M_2$  can have the same sign, and  $|M_1|$  may not equal  $|M_2|$ . In this case, the absolute value operator in the worst-case magnitude definition (4.2) cannot be omitted, and hence, we have

$$\xi(T) = \max_{w \in \mathcal{W}} \max_{h \in \mathbb{H}} |h(T) * w(T)|.$$

Since the input bound is not symmetric, this maximization must be calculated on both negative and positive sides. This means that

$$\xi(T) = \max \left\{ \max_{w \in \mathcal{W}} \max_{h \in \mathbb{H}} [h(T) * w(T)], \max_{w \in \mathcal{W}} \max_{h \in \mathbb{H}} -[h(T) * w(T)] \right\}.$$

Thus, to compute  $\xi(T)$ , we need to solve two maximization problems where the objective functions have the opposite signs, but the feasible sets are the same. Therefore, it suffices to discuss only

$$\max_{w \in \mathcal{W}} \max_{h \in \mathbb{H}} [h(T) * w(T)], \quad (7.7)$$

while the other maximization can be obtained in a parallel way by adding the minus sign to the objective function of (7.7). The maximization of the convolution integral  $\int_0^T h(T-t)w(t)dt$  in (7.7) can be written as

$$\begin{aligned} \max \quad & \tau_s x^T y \\ \text{s.t.} \quad & M_1 \leq x_i \leq M_2, \quad i = 1, \dots, N, \\ & -\tau_s D \leq x_1 \leq \tau_s D, \\ & -\tau_s D \leq x_{i+1} - x_i \leq \tau_s D, \quad i = 1, \dots, N-1, \\ & l_i \leq y_i \leq u_i, \quad i = 1, \dots, N \end{aligned}$$

where the vector quantities are defined from the discretized variables as

$$\begin{aligned} x_i &= w[i], \quad i = 1, \dots, N, \\ y_i &= h[N-i], \quad i = 1, \dots, N-1, \quad y_N = \frac{1}{2}h[0], \\ u_i &= h_u[N-i], \quad i = 1, \dots, N-1, \quad u_N = \frac{1}{2}h_u[0], \\ l_i &= h_l[N-i], \quad i = 1, \dots, N-1, \quad l_N = \frac{1}{2}h_l[0]. \end{aligned}$$

The convex maximization problem (4.31) can be formulated in the similar fashion. However, to compute the upper bound of the WCN, the change of coordinate of  $x$  in (4.32) should be modified as

$$\tilde{x} = x - M_2 \mathbf{1}.$$

The computation of upper and lower bounds can then be proceeded as mentioned in Chapter 4.

5. **Nonzero input initial condition:** When computing the WCN, the uncertain linear system is assumed to be at rest, that is, all initial conditions are zero. In addition, the disturbance is assumed to start from zero, *i.e.*,  $w(0) = 0$ . In many practical situations, we may want to compute the WCN when disturbances start from nonzero initial conditions. Before we proceed, an additional assumption regarding the WCN computation should be made. Given the continuity of the disturbance signal, nonzero initial condition implies that  $w(t)$  cannot equal zero all the time before  $t = 0$ . Nevertheless, we assume that the information on disturbance input is available from  $t = 0$  onwards, and the convolution  $h(T) * w(T)$  starts from  $t = 0$ , *i.e.*,  $h(T) * w(T) = \int_0^T h(T-t)w(t)dt$ , which means that the contribution from part of  $w(t)$  where  $t < 0$  is omitted. To compute the WCN in this case, we rewrite the convolution integral as

$$\begin{aligned} h(T) * w(T) &= \int_0^T h(T-t)w(t)dt \\ &= \int_0^T h(T-t)\tilde{w}(t)dt + \int_0^T h(T-t)w_0dt \\ &= h(T) * \tilde{w}(T) + \int_0^T h(t)w_0dt \\ &= h(T) * \tilde{w}(T) + w_0s(T) \end{aligned}$$

where  $\tilde{w}(t) = w(t) - w_0$ ,  $w_0 = w(0)$ , and  $s(t) = \int_0^t h(\tau)d\tau$  is the step response of the system. The temporary input  $\tilde{w}$  has the zero initial condition, *i.e.*,  $\tilde{w}(0) = 0$ , but the bound on magnitude of  $\tilde{w}(t)$  becomes asymmetric

$$-M - w_0 \leq \tilde{w}(t) \leq M - w_0, \quad \forall t.$$

Following the argument in the previous topic, the worst-case magnitude can be obtained as

$$\xi(T) = \max \left\{ \max_{w \in \mathcal{W}} \max_{h \in \mathbb{H}} \psi(T), \max_{w \in \mathcal{W}} \max_{h \in \mathbb{H}} -\psi(T) \right\}$$

where  $\psi(t) = |h(t) * \tilde{w}(t) + w_0s(t)|$ . Consider only the maximization on the positive side, that is,  $\max_{w \in \mathcal{W}} \max_{h \in \mathbb{H}} \psi(T)$ . This maximization problem can be formulated as

$$\begin{aligned} \max \quad & \tau_s(x + w_0\mathbf{1})^T y \\ \text{s.t.} \quad & -M - w_0 \leq x_i \leq M - w_0, \quad i = 1, \dots, N, \\ & -\tau_s D \leq x_1 \leq \tau_s D, \\ & -\tau_s D \leq x_{i+1} - x_i \leq \tau_s D, \quad i = 1, \dots, N-1, \\ & l_i \leq y_i \leq u_i, \quad i = 1, \dots, N \end{aligned}$$

where the vector quantities are defined from the discretized variables as

$$\begin{aligned} x_i &= \tilde{w}[i], \quad i = 1, \dots, N, \\ y_i &= h[N-i], \quad i = 1, \dots, N-1, \quad y_N = \frac{1}{2}h[0], \\ u_i &= h_u[N-i], \quad i = 1, \dots, N-1, \quad u_N = \frac{1}{2}h_u[0], \\ l_i &= h_l[N-i], \quad i = 1, \dots, N-1, \quad l_N = \frac{1}{2}h_l[0]. \end{aligned}$$

The convex maximization problem (4.31) can be formulated in the same manner, and the upper bound of the WCN can be calculated with the coordinate shift of  $x$  in (4.32) being modified as  $\tilde{x} = x + (M + w_0)\mathbf{1}$ .

6. **Other control design methods:** There are several systems which are subjected to the disturbance and uncertainty with the forms considered in this dissertation. Control problems can be set up for these systems to investigate control performances that can be improved when the WCN specifications are employed. Furthermore, more rigorous control design techniques can be applied to these control problems instead of the PID controller tuning proposed in Chapter 6 for more systematic ways to solve for the controllers.



สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย

## REFERENCES

- [1] G. Zames. On the Input-Output Stability on Nonlinear Time-Varying Feedback Systems—Part II: Conditions Involving Circles in the Frequency Plane and Sector Nonlinearities. *IEEE Trans. Aut. Control* 11, 3, (1966): 465–476.
- [2] K. S. Narendra and R. M. Goldwyn. A Geometrical Criterion for the Stability of Certain Nonlinear Nonautonomous Systems. *IEEE Trans. Circuit Theory* 11, 3, (1964): 406–408.
- [3] I. W. Sandberg. A Frequency-Domain Condition for the Stability of Feedback Systems Containing a Single Time-Varying Nonlinear Element. *Bell Syst. Tech. J.* 43, 3, (1964): 1601–1608.
- [4] M. A. Dahleh and I. J. Diaz-Bobillo. *Control of Uncertain Systems*. New Jersey: Prentice-Hall, 1995.
- [5] B. Birch and R. Jackson. The Behaviour of Linear Systems with Inputs Satisfying Certain Bounding Conditions. *J. of Electron. and Control* 6, 4, (1959): 366–375.
- [6] S. S. L. Chang. Minimal Time Control with Multiple Saturation Limits. *IRE International Convention Record* 10, 2, (1962): 143–151.
- [7] S. S. L. Chang. Minimal Time Control with Multiple Saturation Limits. *IEEE Trans. Aut. Control* 8, 1, (1963): 35–42.
- [8] I. M. Horowitz. *Synthesis of Feedback Systems*. London: Academic Press, 1963.
- [9] J. J. Bongiorno Jr.. On the Response of Linear Systems to Inputs with Limited Amplitudes and Slopes. *SIAM Rev.* 9, 3, (1967): 554–563.
- [10] S. Boyd and C. Barratt. *Linear Controller Design: Limits of Performance*. New Jersey: Prentice-Hall, 1991.
- [11] A. E. Bryson and Y. C. Ho. *Applied Optimal Control*. Hemisphere Publishing, 1975.
- [12] E. Polak. An Historical Survey of Computational Methods in Optimal Control. *SIAM Rev.* 15, 2, (1973): 553–584.
- [13] L. S. Pontryagin, V. Boltyanskii, R. Gamkrelidze, and E. Mishehenko. *The Mathematical Theory of Optimal Processes*. New Jersey: John Wiley & Sons, 1962.
- [14] L. D. Berkovitz. On Control Problems with Bounded State Variables. *J. Math. Anal. Appl.* 3, (1961): 145–169.
- [15] R. McGill. Optimal Control, Inequality State Constraints, and the Generalized Newton-Raphson Algorithm. *SIAM J. on Control* 3, (1965): 291–298.

- [16] D. E. Kirk. *Optimal Control Theory: An Introduction*. New Jersey: Prentice-Hall, 1970.
- [17] A. P. Sage and C. C. White III. *Optimum System Control*. New Jersey: Prentice-Hall, 1977.
- [18] G. Saridis and Z. V. Rekasius. Investigation of Worst-Case Errors when Inputs and Their Rate of Change are Bounded. *IEEE Trans. Aut. Control* 11, 2, (1966): 296–300.
- [19] P. G. Lane. *Design of Control Systems with Inputs and Outputs Satisfying Certain Bounding Conditions*. PhD thesis, UMIST, Manchester, UK, October 1992.
- [20] W. Reinelt. Maximum Output Amplitude of Linear Systems for Certain Input Constraints. in *Proc. IEEE Conf. on Decision and Control*, (2000): 1075–1080.
- [21] W. Khaisongkram and D. Banjerdpongchai. An Optimal Control Approach to Compute the Performance of Linear Systems under Disturbances with Bounded Magnitudes and Bounded Derivatives. in *Proc. American Control Conf.*, (2004): 28–33.
- [22] W. Khaisongkram and D. Banjerdpongchai. On Computing the Worst-Case Norm of Convolution Systems: A Comparison of Continuous-Time and Discrete-Time Approaches. in *Proc. the 16<sup>th</sup> IFAC World Congress*, (2005): n/a.
- [23] W. Reinelt. Worst Case Output of Uncertain Systems. Technical report, Linköpings Universitet, Linköpings, Sweden, May 2001.
- [24] K. G. Murty. Solving the Fixed-Charge Problem by Ranking the Extreme Points. *Operations Research* 16, (1968): 268–279.
- [25] B. Kalantari. A Linear Max-Min problem. *Math. Prog.* 5, (1973): 169–188.
- [26] H. Tuy. Concave Programming under Linear Constraints. *Soviet Math. Dokl.* 5, (1968): 1437–1440.
- [27] P. Zwart. Nonlinear Programming: Counterexample to Two Global Optimization Algorithms. *Operations Research* 21, (1973): 1260–1266.
- [28] A. J. Majthay and A. Whinston. Quasi-Concave Minimization Subject to Linear Constraints. *Discrete Maths.* 9, (1974): 35–59.
- [29] H. D. Sherali and C. M. Shetty. A Finitely Convergent Algorithm for Bilinear Programming Problems using Polar Cuts and Disjunctive Face Cuts. *Math. Prog.* 19, (1980): 14–31.
- [30] R. Horst and H. Tuy. *Global Optimization: Deterministic Approaches*. Berlin: Springer, 3<sup>rd</sup> ed., 1996.
- [31] J. P. Snectman and N. V. Sahinidis. A Finite Algorithm for Global Minimization of Separable Concave Function. *J. of Global Optimization* 12, (1998): 1–36.
- [32] M. Locatelli and N. V. Thoai. Finite Exact Branch-and-Bound Algorithms for Concave Minimization over Polytopes. *J. of Global Optimization* 18, (2000): 107–130.

- [33] H. Konno. Maximization of a Convex Quadratic Function Subject to Linear Constraints. *Math. Prog.* 11, (1976): 117–127.
- [34] H. Konno. Maximizing a Convex Quadratic Function Over a Hypercube. *J. Operations Research Society of Japan* 23, 2, (1980): 171–189.
- [35] P. Hansen, B. Jaumard, M. Ruiz, and J. Xiong. Global Minimization of Indefinite Quadratic Functions Subject to Box Constraints. *Naval Research Logistics Quarterly* 40, (1993): 373–392.
- [36] L. T. H. An and P. D. Tao. A Branch and Bound Method via D.C. Optimization Algorithms and Ellipsoidal Technique for Box Constrained Nonconvex Quadratic Problems. *J. of Global Optimization* 13, (1998): 171–206.
- [37] C. Dang and L. Xu. A Barrier Function Method for the Nonconvex Quadratic Programming Problem with Box Constraints. *J. of Global Optimization* 18, (2000): 165–188.
- [38] P. L. de Angelis, I. M. Bomze, and G. Toraldo. Ellipsoidal Approach to Box-Constrained Quadratic Problems. *J. of Global Optimization* 28, (2004): 1–15.
- [39] D. Vandebussche and G. L. Nemhauser. A Branch-And-Cut Algorithm for Nonconvex Quadratic Programs with Box Constraints. *Math. Prog.* 102, 3, (2005): 559–575.
- [40] P. M. Pardalos and S. A. Vavasis. Quadratic Programming with One Negative Eigenvalue is  $\mathcal{NP}$ -Hard. *J. of Global Optimization* 1, (1993): 15–23.
- [41] R. Horst and U. Raber. Convergent Outer Approximation Algorithms for Solving Unary Program. *J. of Global Optimization* 43, 7, (2003): 908–917.
- [42] M. Poremski. How to Extend the Concept of Convexity Cuts to Derive Deeper Cutting Planes. *J. of Global Optimization* 15, (1999): 371–404.
- [43] M. Poremski. Finitely Convergent Cutting Planes for Concave Minimization. *J. of Global Optimization* 20, (2001): 113–136.
- [44] A. I. Rusakov. Concave Programming under Simplest Linear Constraints. *Computational Mathematics and Mathematical Physics* 43, 7, (2003): 908–917.
- [45] T. Kuno and T. Utsunomiya. A Lagrangian Based Branch-and-Bound Algorithm for Production-Transportation Problems. *J. of Global Optimization* 18, (2000): 59–73.
- [46] J. Brimberg, P. Hansen, and N. Mladenovic. A Note on Reduction of Quadratic and Bilinear Program with Equality Constraints. *J. of Global Optimization* 22, (2002): 39–47.
- [47] W. Khaisongkram. *Performance Analysis and Controller Design for a Binary Distillation Column under Disturbances with Bounded Magnitudes and Bounded Derivatives*. Master's thesis, Dept. of Elec. Eng., Chulalongkorn Univ., Bangkok, Thailand, May 2003.

- [48] R. G. Bartle. *The Elements of Real Analysis*. New York: Wiley, 1964.
- [49] D. G. Luenberger. *Optimization by Vector Space Methods*. New York: John Wiley and Sons, 1969.
- [50] K. Glover. Model Reduction: A Tutorial on Hankel-Norm Methods and Lower Bounds on  $\mathcal{L}^2$  Errors. in *Proc. the 10<sup>th</sup> IFAC World Congress*, (1987): .
- [51] V. Balakrishnan and S. Boyd. On Computing the Worst-Case Peak Gain of Linear Systems. *Syst. Control Letters* 19, 4, (1992): 265–269.
- [52] M. S. Bazaraa and C. M. Shetty. *Nonlinear Programming: Theory and Algorithms*. New York: John Wiley and Sons, 2<sup>nd</sup> ed., 1993.
- [53] W. Khaisongkram, S. Boyd, and D. Banjerdpongchai. A Linear Programming Approach for the Worst-Case Norm of Uncertain Linear Systems Subject to Disturbances with Magnitude and Rate Bounds. in *Proc. IEEE Conf. on Decision and Control*, (2006): 4399–4404.
- [54] M. Padberg. The Boolean Quadric Polytope: Some Characteristics, Facets and Relatives, Mathematical Programming. *Math. Prog.* 45, 1–3, (1989): 139–172.
- [55] Y. Yajima and T. Fujie. A Polyhedral Approach for Nonconvex Quadratic Programming Problems with Box Constraints. *J. of Global Optimization* 13, 2, (1998): 151–170.
- [56] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge, UK: Cambridge University Press, 2004.
- [57] P. M. Pardalos and G. Schnitger. Checking Local Optimality in Constrained Quadratic Programming is  $\mathcal{NP}$ -Hard. *Operations Research Letters* 7, (1988): 33–35.
- [58] V. Balakrishnan, S. Boyd, and S. Balemi. Branch and Bound Algorithm for Computing the Minimum Stability Degree of Parameter-Dependent Linear Systems. *Int. J. Robust and Nonlinear Control* 1, 4, (1991): 295–317.
- [59] V. Balakrishnan and S. Boyd. Global Optimization in Control System Analysis and Design. Book Chapter in *Control and Dynamic Systems: Advances in Theory and Applications*, C. T. Leondes, editor. New York: NY Academic, 1992.
- [60] C. L. DeMarco, V. Balakrishnan, and S. Boyd. A Branch and Bound Methodology for Matrix Polytope Stability Problems Arising in Power Systems. in *Proc. IEEE Conf. on Decision and Control*, (1990): 3022–3027.
- [61] J.-S. Lin and I. Kanellakopoulos. Nonlinear Design of Active Suspensions. *IEEE Control Syst. Mag.* 17, 3, (1997): 45–59.
- [62] D. Hrovat. Survey of Advanced Suspension Developments and Related Optimal Control Application. *Automatica* 33, 10, (1997): 1781–1817.

- [63] I. Fialho and G. J. Balas. Road Adaptive Active Suspension Design Using Linear Parameter-Varying Gain-Scheduling. *IEEE Trans. Control Sys. Tech.* 10, 1, (2002): 43–54.
- [64] Y. Sama, J. Osmana, and M. Ghanib. A Class of Proportional-Integral Sliding Mode Control with Application to Active Suspension System. *Syst. Control Letters* 3–4, 51, (2004): 217–223.
- [65] H. Chen and K.-H. Guo. Constrained  $\mathcal{H}_\infty$  Control of Active Suspensions: an LMI Approach. *IEEE Trans. Control Sys. Tech.* 13, 3, (2005): 412–421.
- [66] H. Gao, J. Lam, and C. Wang. Multi-Objective Control of Vehicle Active Suspension Systems via Load-Dependent Controllers. *Journal of Sound and Vibration* 290, 3–5, (2006): 654–675.
- [67] G. F. Franklin, J. D. Powell, and A. Emami-Naeini. *Feedback Control of Dynamic Systems*. New Jersey: Pearson Education, Inc., 5<sup>th</sup> ed., 2006.
- [68] J. K. Hedrick and T. Butsuen. Invariant Properties of Automotive Suspensions. *Proc. of Inst. of Mech. Eng. part D: Journal of Auto. Eng.* 204, (1990): 21–27.
- [69] V. Zakian and U. Al-Naib. Design of Dynamical and Control Systems by the Method of Inequalities. *Proc. of Inst. of Elec. Eng.* 120, (1973): 1421–1427.





APPENDICES

สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย

## APPENDIX A

### A Bound on Truncation Error

We will derive the result of the more general case first, *i.e.*, the case of uncertain systems. Then, the case of no uncertainty can be obtained by specializing this result. Assume that  $h(t)$  is a stable uncertain system associated with  $\mathbb{H}$ . We will show that  $\max_{h \in \mathbb{H}} \int_T^\infty |h(\tau)| d\tau$  is a bound on the truncation error for the estimation of  $\|h\|_{\text{wc}}$  by  $\xi(T)$  where  $T$  is the terminal time. Specifically, we need to show that

$$\left| \|h\|_{\text{wc}} - \xi(T) \right| \leq M \max_{h \in \mathbb{H}} \int_T^\infty |h(\tau)| d\tau. \quad (\text{A.1})$$

To see this, let us consider any  $w(t) \in \mathcal{W}$ ,  $h(t) \in \mathbb{H}$ , and  $t > T$ . We have

$$\begin{aligned} h(t) * w(t) &= \int_0^t h(\tau) w(t - \tau) d\tau \\ &\leq \max_{w \in \mathcal{W}} \max_{h \in \mathbb{H}} \left( \int_0^T h(\tau) w(t - \tau) d\tau + \int_T^t h(\tau) w(t - \tau) d\tau \right) \\ &\leq \max_{w \in \mathcal{W}} \max_{h \in \mathbb{H}} \int_0^T h(\tau) w(t - \tau) d\tau + \max_{w \in \mathcal{W}} \max_{h \in \mathbb{H}} \int_T^t h(\tau) w(t - \tau) d\tau \\ &\leq \xi(T) + M \max_{h \in \mathbb{H}} \int_T^t |h(\tau)| d\tau \end{aligned} \quad (\text{A.2})$$

Recall that  $\xi(t) = \max_{w \in \mathcal{W}} \max_{h \in \mathbb{H}} \int_0^t h(\tau) w(t - \tau) d\tau$ . Since the inequality (A.2) holds for any  $w(t) \in \mathcal{W}$  and  $h(t) \in \mathbb{H}$ , we have

$$\xi(t) = \max_{w \in \mathcal{W}} \max_{h \in \mathbb{H}} h(t) * w(t) \leq \xi(T) + M \max_{h \in \mathbb{H}} \int_T^t |h(\tau)| d\tau.$$

By limiting  $t$  to infinity on both sides of this equation, it is readily seen that

$$\begin{aligned} \|h\|_{\text{wc}} &\leq \xi(T) + M \max_{h \in \mathbb{H}} \int_T^\infty |h(\tau)| d\tau \\ \|h\|_{\text{wc}} - \xi(T) &\leq M \max_{h \in \mathbb{H}} \int_T^\infty |h(\tau)| d\tau. \end{aligned} \quad (\text{A.3})$$

From the monotonicity of  $\xi(T)$  in Section 4.1.1 and the definition of  $\|h\|_{\text{wc}}$  in (4.3), we have  $\|h\|_{\text{wc}} \geq \xi(T)$ , and hence,  $\left| \|h\|_{\text{wc}} - \xi(T) \right| = \|h\|_{\text{wc}} - \xi(T)$ . Thus, the statement (A.1) is verified.

The result for linear systems  $h(t)$  without uncertainty can be obtained by thinking of the characterizing set  $\mathbb{H}$  as containing only a single system  $h(t)$ . This suggests that

$$\max_{h \in \mathbb{H}} \int_T^\infty |h(\tau)| d\tau = \int_T^\infty |h(\tau)| d\tau,$$

and (A.3) becomes

$$\|h\|_{\text{wc}} - \xi(T) \leq M \int_T^\infty |h(\tau)| d\tau.$$

It should be noted that this bound on truncation error cannot be conveniently exploited to determine a suitable terminal time  $T$  because it needs computation of the integration with unbounded limit. Although the integration can be obtained analytically given the closed form of  $h(t)$ , this method is time consuming, and hence, not practical to use with arbitrary choice of  $h(t)$ . For this reason, a computable bound on truncation error for linear systems without uncertainty is given in Section (3.1.2), and that for uncertain linear systems is given in Section (4.1.1).



สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย

## APPENDIX B

### Proofs Relating to the Successive Pang Interval Search (SPIS)

**Proof of Lemma 3.1:** Consider the first case that  $\pi'_i \equiv \pi''_j$  but  $\pi'_i \neq \pi''_j$ . To verify this, we will show that (i) implies (ii), and the negation of (i) implies the negation of (ii).

(i)→(ii) Since  $\pi_c$  is an up segment, then  $\tilde{s}(T - t'_0) > \tilde{s}(T - t''_0)$ . Suppose that the segment  $\pi'_i$  precedes  $\pi''_j$ . If  $\pi'_i$  is an even segment, then from Proposition 3.1,  $\pi'_i$  is an up segment. This means  $\tilde{s}(T - t)$  crosses over  $\tilde{s}(T - t'_0)$  upwards. Since  $\tilde{s}(T - t'_0)$  and  $\tilde{s}(T - t''_0)$  do not cut through any segment between  $\pi'_i$  and  $\pi''_j$ , such segment must be located above  $\tilde{s}(T - t'_0)$  and  $\tilde{s}(T - t''_0)$ . Thus,  $\tilde{s}(T - t)$  cannot cross  $\tilde{s}(T - t''_0)$  in  $\pi''_j$  unless it crosses  $\tilde{s}(T - t'_0)$  first, but this is impossible since  $\tilde{s}(T - t'_0)$  does not cut through segments between  $\pi'_i$  and  $\pi''_j$ . Therefore,  $\pi'_i$  cannot be an even segment.

For this reason,  $\pi'_i$  must be an odd segment which is, in this particular case, a down segment. Then,  $\tilde{s}(T - t)$  crosses over  $\tilde{s}(T - t'_0)$  downwards and stay below  $\tilde{s}(T - t'_0)$  and above  $\tilde{s}(T - t''_0)$  along every segment between  $\pi'_i$  and  $\pi''_j$ . Therefore, if  $\tilde{s}(T - t)$  is about to pass  $\tilde{s}(T - t''_0)$ , it must pass downwards. Thus,  $\pi''_j$  is an odd segment too.

(ii)→(i) Next, suppose contrarily that  $\pi''_j$  precedes  $\pi'_i$ . From Proposition 3.1, if  $\pi''_j$  is an odd segment, then is a down segment. This means  $\tilde{s}(T - t)$  passes over  $\tilde{s}(T - t''_0)$  downwards and stay below  $\tilde{s}(T - t''_0)$ . Therefore, if  $\tilde{s}(T - t)$  is about to cross  $\tilde{s}(T - t'_0)$  in  $\pi'_i$ , it must first cross  $\tilde{s}(T - t''_0)$ , but this violates the assumption that  $\pi'_i \equiv \pi''_j$ . Hence,  $\pi''_j$  cannot be an odd segment. Now suppose that  $\pi''_j$  is an even segment, then it is an up segment. This implies that  $\tilde{s}(T - t)$  passes over  $\tilde{s}(T - t''_0)$  upwards and stays below and  $\tilde{s}(T - t'_0)$ . For this reason, if  $\tilde{s}(T - t)$  is about to cross over  $\tilde{s}(T - t'_0)$ , it must cross upwards. Hence,  $\pi'_i$  is also an even segment.

Now consider the case that  $\pi'_i = \pi''_j$ . This means that  $\pi'_i$  and  $\pi''_j$  are of the same direction. Since we are given that  $\pi'_0 = \pi''_0$ , from Proposition 3.1,  $\pi'_i$  is an odd segment if and only if  $\pi''_j$  is.

Subsequently, we will establish (iii) and (iv). For the case that  $\pi'_i \equiv \pi''_j$  but  $\pi'_i \neq \pi''_j$ , if  $\pi'_i, \pi''_j$  are odd segments, then  $\pi'_i$  precedes  $\pi''_j$ , which means  $t'_i < t''_j$ . In contrast, If  $\pi'_i, \pi''_j$  are even segments, then  $\pi''_j$  precedes  $\pi'_i$ , and  $t'_i > t''_j$ . For the case that  $\pi'_i = \pi''_j$ , because  $\pi'_0 (= \pi''_0)$  is assumed to be an up segment, if both  $\pi'_i, \pi''_j$  are odd segments, then both are down segments. In addition, recall that  $\tilde{s}(T - t'_0) > \tilde{s}(T - t''_0)$ . This implies that  $\tilde{s}(T - t'_i) > \tilde{s}(T - t''_j)$ , and  $t'_i < t''_j$  by the definition of down segments. On the other hand, if both  $\pi'_i, \pi''_j$  are even segments, then both are up segments, which means  $\tilde{s}(T - t'_i) > \tilde{s}(T - t''_j)$  and  $t'_i > t''_j$ .  $\square$

**Proof of Corollary 3.1:** Suppose that  $\exists \pi'_{i_1} \in \Pi', \exists \pi''_{j_1} \in \Pi''$  such that  $\pi'_{i_1} \equiv \pi''_{j_1}$ . First, we will show that  $\forall \pi'_{i_2} \in \Pi', \pi'_{i_2} \not\equiv \pi''_{j_1}$  if  $\pi'_{i_2} \neq \pi'_{i_1}$ . To see this, assume contrarily that  $\exists \pi'_{i_2} \in \Pi'$  such that  $\pi'_{i_2} \neq \pi'_{i_1}$  and  $\pi'_{i_2} \equiv \pi''_{j_1}$ . Then, there are four presumable cases:

1.  $\pi'_{i_2} = \pi''_{j_1} = \pi'_{i_1}$ ,

2.  $\pi'_{i_2} \neq \pi''_{j_1} = \pi'_{i_1}$ ,
3.  $\pi'_{i_2} = \pi''_{j_1} \neq \pi'_{i_1}$ ,
4.  $\pi'_{i_2} \neq \pi''_{j_1} \neq \pi'_{i_1}$ .

The first case contradicts the assumption that  $\pi'_{i_2} \neq \pi'_{i_1}$ . For the second case, since  $\pi''_{j_1} = \pi'_{i_1}$  and  $\pi'_{i_1} \neq \pi'_{i_2}$ , then by Definition 3.5,  $\pi'_{i_2}$  cannot also be equivalent to  $\pi''_{j_1}$  since  $\pi'_{i_1}$  is situated at  $\pi''_{j_1}$  already. This contradicts the assumption that  $\pi'_{i_2} \equiv \pi''_{j_1}$ . Contradiction of the third case can be inferred in the same manner as the second case. For the last case, we make use of the preceding lemma. Note that  $\pi'_{i_2}$  cannot lie between  $\pi'_{i_1}$  and  $\pi''_{j_1}$  since it would invalidate the equivalence between  $\pi'_{i_1}$  and  $\pi''_{j_1}$ . Therefore,  $\pi'_{i_1}$  and  $\pi'_{i_2}$  must be located on the different sides with respect to  $\pi''_{j_1}$ . Now, assume that  $\pi'_{i_1}$  precedes  $\pi''_{j_1}$ , which in turn precedes  $\pi'_{i_2}$ . From Lemma 3.1, it can be concluded that  $\pi''_{j_1}$  is an odd segment since  $\pi'_{i_1}$  precedes  $\pi''_{j_1}$ . However, it can be concluded as well that  $\pi''_{j_1}$  is an even segment since  $\pi''_{j_1}$  precedes  $\pi'_{i_2}$ , hence, a contradiction. The same contradiction arises if we assume that  $\pi'_{i_2}$  precedes  $\pi''_{j_1}$ , which in turn precedes  $\pi'_{i_1}$ .

It can be proved using the same argument that  $\forall \pi''_{j_2} \in \Pi''$ ,  $\pi''_{j_2} \not\equiv \pi'_{i_1}$  if  $\pi''_{j_2} \neq \pi''_{j_1}$ . Finally, we conclude that the equivalent pair  $(\pi'_{i_1}, \pi''_{j_1})$  is unique.  $\square$

**Proof of Lemma 3.2:** As we have obtained a set of unique equivalent pairs  $\Phi_{ab}$ , consider one of those pairs, which are not the first member of  $\Phi_{ab}$ , denoted by  $(\pi'_i, \pi''_j)$ . From Lemma 3.1, the segments  $\pi'_i, \pi''_j$  must be either odd or even segments at the same time. This applies to the segments  $\pi'_k, \pi''_l$  as well. The claim (i) will be verified by contradiction.

(i) First, we examine the case that  $i, j$  are odd. Assume contrarily that  $k, l$  are odd too. We will show that there exists an equivalent pair, in  $\Phi_{ab}$ , between  $(\pi'_i, \pi''_j)$  and  $(\pi'_k, \pi''_l)$ , which contradicts the assumption that these pairs are contiguous in  $\Phi_{ab}$ . Since the only case that  $\pi_c$  is an up segment is considered,  $\pi'_i, \pi''_j, \pi'_k, \pi''_l$  are all down segments, by Proposition 3.1. Furthermore, from Lemma 3.1,  $\pi'_k$  precedes  $\pi''_l$  and  $\pi'_i$  precedes  $\pi''_j$ . Thus, if we want to consider any segment between  $(\pi'_i, \pi''_j)$  and  $(\pi'_k, \pi''_l)$ , it suffices to consider only between  $\pi''_l$  and  $\pi'_i$ .

The response  $\tilde{s}(T-t)$  crosses over  $\tilde{s}(T-t''_0)$  in  $\pi''_l$  downwards, and then crosses over  $\tilde{s}(T-t'_0)$  in  $\pi'_i$ , downwards again. Since  $\tilde{s}(T-t''_0) < \tilde{s}(T-t'_0)$  (because  $\pi_c$  is an up segment), this obviously implies that the response  $\tilde{s}(T-t)$  after  $\pi''_l$  must cross over  $\tilde{s}(T-t'_0)$  upwards, and  $\tilde{s}(T-t'_0)$  before it can cross  $\tilde{s}(T-t'_0)$  downwards in  $\pi'_i$ . This suggests that there exist segments  $\pi''_{j_\nu} \in \Pi''$  and  $\pi'_{i_\mu} \in \Pi'$  inside which  $\tilde{s}(T-t)$  crosses upwards over  $\tilde{s}(T-t''_0)$  and  $\tilde{s}(T-t'_0)$ , respectively.

Now, moving *backwards* from  $\pi'_i$  one segment at a time to  $\pi'_k$ , we may find a finite sequence of  $\{\pi'_{i_\mu}\}$  in which  $\tilde{s}(T-t)$  crosses over  $\tilde{s}(T-t''_0)$  upwards. Then, following the same procedure, yet with forward direction, from  $\pi''_l$  to  $\pi''_j$ , we obtain another finite sequence  $\{\pi''_{j_\nu}\}$ . Start from  $\{\pi''_{j_1}\}$  and  $\{\pi'_{i_1}\}$ , recalling that  $\{\pi'_{i_1}\}$  is enumerated backwards. If there is any element in  $\{\pi'_{i_\mu}\}$ , say  $\{\pi'_{i_2}\}$ , which is located between  $\{\pi''_{j_1}\}$  and  $\{\pi'_{i_1}\}$ , then we pick up that segment and drop out  $\{\pi'_{i_1}\}$ . Similarly, if such element is in  $\{\pi''_{j_\nu}\}$ , say  $\{\pi''_{j_2}\}$ , then we pick up this element and drop out  $\{\pi''_{j_1}\}$ . Continuing this process, we can always find an adjacent pair  $(\pi'_{i_\mu}, \pi''_{j_\nu})$ , between which there is no

other element in either  $\{\pi'_{i_\mu}\}$  or  $\{\pi''_{j_\nu}\}$ . Since both  $\pi'_{i_\mu}$  and  $\pi''_{j_\nu}$  are up segment, by Definition 3.5,  $\pi'_{i_\mu} \equiv \pi''_{j_\nu}$ , and both are situated between  $(\pi'_k, \pi'_l)$  and  $(\pi'_i, \pi'_j)$ , hence, a contradiction.

For the case that  $i, j$  are even, we assume contrarily that  $k, l$  are also even. The concept is analogous to the case that  $i, j$  are odd, so the proof is left to the reader.

(ii) If  $\pi'_i, \pi'_j$  are even, then  $\pi'_k, \pi'_l$  are odd, hence, down segments. By lemma 3.1, we can see that  $\tilde{s}(T-t)$  passes downwards over  $\tilde{s}(T-t'_0)$  and then  $\tilde{s}(T-t''_0)$ . Recall that  $\tilde{s}(T-t''_0) < \tilde{s}(T-t'_0)$ . Therefore, in order for  $\tilde{s}(T-t)$  to cross over  $\tilde{s}(T-t'_0)$  upwards in  $\pi'_{k+1}$ , it must first cross over  $\tilde{s}(T-t''_0)$  upwards in some preceding segments. Let  $\pi''_l$  be the latest one of such segments. This means there is no cutting segment in  $\Pi'$  or  $\Pi''$  located between  $\pi''_l$  and  $\pi'_{k+1}$  (except themselves), which implies that  $\pi'_{k+1} \equiv \pi''_l$ . Since this is the first equivalent pair next to  $(\pi'_k, \pi'_l)$ , we conclude that  $i = k + 1$ .

(iii) The third claim can be verified with the same manner. If  $\pi'_i, \pi'_j$  are odd, then  $\pi'_k, \pi'_l$  are even and are up segments. Hence,  $\tilde{s}(T-t)$  passes over  $\tilde{s}(T-t'_0)$  upwards and then  $\tilde{s}(T-t''_0)$  by Lemma 3.1. Before  $\tilde{s}(T-t)$  can cross over  $\tilde{s}(T-t''_0)$  downwards in  $\pi''_{l+1}$ , it must cross over  $\tilde{s}(T-t'_0)$  downwards in any nearest foregoing segment, say  $\pi'_i$ . This leads to a conclusion that  $\pi'_i \equiv \pi''_j$  where  $j = l + 1$ .  $\square$

**Proof of Proposition 3.2:** Assume that  $(\pi'_i, \pi''_j)$  and  $(\pi'_k, \pi'_l)$  are any two contiguous equivalent pairs in  $\Phi_{ab}$  such that  $(\pi'_k, \pi'_l)$  precedes  $(\pi'_i, \pi''_j)$ . For simplicity of discussion, let  $\mathcal{O}'_{ab}$  and  $\mathcal{E}'_{ab}$  be sets of odd and even *indices* to the first member of each pair in  $\Phi_{ab}$ . Also, define  $\mathcal{O}''_{ab}$  and  $\mathcal{E}''_{ab}$  similarly, but with respect to the second member. By means of these sets, we can decompose  $\theta'_a$  and  $\theta''_b$  as follows.

$$\begin{aligned}\theta'_a &= \sum_{i \in \mathcal{O}'_{ab}} \sum_{m=k+1}^i (-1)^{m+1} \Delta t'_m - \sum_{i \in \mathcal{E}'_{ab}} \sum_{m=k+1}^i (-1)^{m+1} \Delta t'_m, \\ \theta''_b &= \sum_{j \in \mathcal{O}''_{ab}} \sum_{m=l+1}^j (-1)^{m+1} \Delta t''_m - \sum_{j \in \mathcal{E}''_{ab}} \sum_{m=l+1}^j (-1)^{m+1} \Delta t''_m.\end{aligned}$$

Using Lemma 3.2, these equations are simplified as follows.

$$\theta'_a = - \sum_{i \in \mathcal{E}'_{ab}} \Delta t'_i + \sum_{i \in \mathcal{O}'_{ab}} \sum_{m=k+1}^i (-1)^{m+1} \Delta t'_m, \quad (\text{B.1})$$

$$\theta''_b = \sum_{j \in \mathcal{O}''_{ab}} \Delta t''_j - \sum_{j \in \mathcal{E}''_{ab}} \sum_{m=l+1}^j (-1)^{m+1} \Delta t''_m. \quad (\text{B.2})$$

Consider  $\sum_{m=k+1}^i (-1)^{m+1} \Delta t'_m$  where  $i \in \mathcal{O}'_{ab}$ . As discussed in Lemma 3.2, we can show that

$$\sum_{m=k+1}^i (-1)^{m+1} \Delta t'_m \leq \sum_{m=k+1}^i \Delta t'_m = t'_i - t'_k < t''_j - t''_l = t''_j - t''_{j-1} = \Delta t''_j.$$

The last inequality holds due to statements (iii) and (iv) in Lemma 3.1 and statement (iii) in Lemma 3.2. In an analogous fashion, let consider  $j \in \mathcal{E}_{ab}''$ . We can show that

$$\sum_{m=l+1}^j (-1)^{m+1} \Delta t_m'' \leq \sum_{m=l+1}^j \Delta t_m'' = t_j'' - t_l'' < t_i' - t_k' = t_i' - t_{i-1}' = \Delta t_i', \quad (\text{B.4})$$

which also makes use of statements (iii) and (iv) in Lemma 3.1 and statement (ii) in Lemma 3.2. Substitute (B.3) and (B.4) in (B.1) and compare the result with (B.2). This straightforwardly yields (3.5).  $\square$

**Proof of Corollary 3.2:** By Proposition 3.2, we obtain the following fact.

$$\theta_k' < \theta_l''. \quad (\text{B.5})$$

The proof of the previous proposition will be imitated in each case.

(i) Let  $l + 1 \leq \hat{j} \leq j$ . By statement (i) in Lemma 3.2, if  $i, j$  are even, then  $k, l$  are odd. By statements (iii) and (iv) in Lemma 3.1 and statement (ii) in Lemma 3.2, we can show that

$$\sum_{m=l+1}^{\hat{j}} (-1)^{m+1} \Delta t_m'' \leq \sum_{m=l+1}^{\hat{j}} \Delta t_m'' \leq \sum_{m=l+1}^j \Delta t_m'' = t_j'' - t_l'' < t_i' - t_k' = t_i' - t_{i-1}' = \Delta t_i'. \quad (\text{B.6})$$

Since  $l + 1$  are even, from (B.5) and (B.6) we have immediately

$$\theta_j'' = \theta_l'' - \sum_{m=l+1}^{\hat{j}} (-1)^{m+1} \Delta t_m'' > \theta_k' - \Delta t_i' = \theta_{i-1}' - \Delta t_i' = \theta_i'. \quad (\text{B.7})$$

This proves (3.6).

(ii) With similar fashion, let  $k + 1 \leq \hat{i} \leq i$ . By statement (i) in Lemma 3.2, if  $i, j$  are odd, then  $k, l$  are even. By statements (iii) and (iv) in Lemma 3.1 and statement (iii) in Lemma 3.2, we can show that

$$\sum_{m=k+1}^{\hat{i}} (-1)^{m+1} \Delta t_m' \leq \sum_{m=k+1}^{\hat{i}} \Delta t_m' \leq \sum_{m=k+1}^i \Delta t_m' = t_i' - t_k' < t_j'' - t_l'' = t_j'' - t_{j-1}'' = \Delta t_j''. \quad (\text{B.8})$$

Since  $k + 1$  are odd, from (B.5) and (B.8) we have

$$\theta_i' = \theta_k' + \sum_{m=k+1}^{\hat{i}} (-1)^{m+1} \Delta t_m' < \theta_l'' + \Delta t_j'' = \theta_{j-1}'' + \Delta t_j'' = \theta_j''. \quad (\text{B.9})$$

This verifies (3.7).  $\square$

## APPENDIX C

### Closed-Form Solutions for the WCN of Second-Order Linear Systems

From the earlier works, there has not been any closed-form formulas of the WCN. To the best of our knowledge, Lane [19] partially gave a result regarding the closed-form solution of the WCN. He considered the time responses in terms of a specific exponentially-decaying sinusoidal function and gave the analytical solutions of  $\xi(t)$  for fixed  $t$ . However, his solutions appear in the form of a finite summation of integral, which is not an explicit formula of the WCN.

In this chapter, we consider the WCN of second-order convolution systems which are divided into overdamped, underdamped, and critically-damped and lightly-damped systems. The characteristics of the worst-case input follow the previous development in Section 2.3. Then we derive the worst-case inputs and obtain explicit expressions for the WCN.

#### Bounds on the WCN

Let us introduce an upper bound and a lower bound of  $\xi(t)$ , which is defined in (2.2). An upper bound of  $\xi(t)$  denoted by  $\bar{\xi}(t)$  is of the form

$$\bar{\xi}(t) = \max_{w \in \bar{\mathcal{W}}} [h(t) * w(t)] \quad (\text{C.1})$$

where  $\bar{\mathcal{W}}$  is a set similar to  $\mathcal{W}$  but extends to the input with single discontinuity at  $t = 0$ . Since  $\mathcal{W} \subset \bar{\mathcal{W}}$ , it is obvious that  $\xi(t) \leq \bar{\xi}(t)$  for any  $t$ . In addition, let  $\bar{w}(t)$  be the worst-case input in  $\bar{\mathcal{W}}$  that yields  $\bar{\xi}(t)$ . On the other hand, a simple lower bound  $\underline{\xi}(t)$  can be readily obtained as

$$\underline{\xi}(t) = h(t) * \underline{w}(t) \quad (\text{C.2})$$

where  $\underline{w}(t)$  is defined as

$$\underline{w}(t) \triangleq \begin{cases} 0, & t < 0, \\ \frac{D}{M} \bar{w} \left( \frac{M}{D} \right) t, & 0 \leq t < \frac{M}{D}, \\ \bar{w}(t), & t \geq \frac{M}{D}. \end{cases} \quad (\text{C.3})$$

To see that this is a valid lower bound, we will show that  $\underline{w}(t)$  lies in  $\mathcal{W}$ . In the definition (C.3),  $(D/M)\bar{w}(M/D)t$  is the linear function starting from zero when  $t = 0$  and converging to  $\bar{w}(M/D)$  as  $t$  approaches  $M/D$ . This implies that  $\underline{w}(0) = 0$ , and the left and right limits of  $\underline{w}(t)$  at  $t = M/D$  match. Thus,  $\underline{w}(t)$  is continuous for all  $t$ . In addition, for  $0 \leq t < M/D$ , if  $\bar{w}(M/D) > 0$ , then  $\underline{w}(t) > 0$  and

$$\underline{w}(t) = \frac{D}{M} \bar{w} \left( \frac{M}{D} \right) t \leq \bar{w} \left( \frac{M}{D} \right) \leq M.$$

If  $\bar{w}(M/D) \leq 0$ , then  $\underline{w}(t) \leq 0$  and

$$\underline{w}(t) = \frac{D}{M} \bar{w} \left( \frac{M}{D} \right) t \geq \bar{w} \left( \frac{M}{D} \right) \geq -M.$$



Thus,  $|\underline{w}(t)| \leq M$  for  $t \geq 0$ . Furthermore, since the rate of change of  $\underline{w}(t)$  for  $0 \leq t < M/D$  is  $(D/M)\bar{w}(M/D)$ , we have

$$|\dot{\underline{w}}(t)| = \left| \frac{D}{M} \bar{w} \left( \frac{D}{M} \right) \right| \leq D.$$

Thus,  $|\dot{\underline{w}}(t)| \leq D$  for  $t \geq 0$ . Therefore, it can be concluded that  $\underline{w}(t) \in \mathcal{W}$ . By the definition of  $\xi(t)$  in (2.2), we have  $\xi(t) \geq \underline{\xi}(t)$  for all  $t$ .

These upper and lower bounds are useful because they converge to the same value when  $t \rightarrow \infty$  if  $h(t)$  is BIBO stable. To show this, consider the difference  $\bar{\xi}(t) - \underline{\xi}(t)$ . From (C.1), (C.2), and (C.3) it can be shown that, for  $t \geq M/D$ ,

$$\begin{aligned} \bar{\xi}(t) - \underline{\xi}(t) &= h(t) * [\bar{w}(t) - \underline{w}(t)] \\ &= \int_0^t h(t-\tau) [\bar{w}(\tau) - \underline{w}(\tau)] d\tau. \end{aligned}$$

Since the difference between  $\bar{w}(\tau)$  and  $\underline{w}(\tau)$  is not greater than  $M$ , we have

$$\begin{aligned} \bar{\xi}(t) - \underline{\xi}(t) &\leq M \int_0^{\frac{M}{D}} |h(t-\tau)| d\tau \\ &= M \int_{t-\frac{M}{D}}^t |h(\tau)| d\tau. \end{aligned}$$

It is obvious that the integral in the last term vanishes as  $t \rightarrow \infty$ , provided that  $\int_0^\infty |h(\tau)| d\tau < \infty$ . Hence,

$$\lim_{t \rightarrow \infty} \underline{\xi}(t) = \lim_{t \rightarrow \infty} \bar{\xi}(t). \quad (\text{C.4})$$

Since we have

$$\underline{\xi}(t) \leq \xi(t) \leq \bar{\xi}(t),$$

from (2.5) and (C.4), it can be shown that

$$\|h\|_{\text{wc}} = \lim_{t \rightarrow \infty} \xi(t) = \lim_{t \rightarrow \infty} \bar{\xi}(t). \quad (\text{C.5})$$

At this point, we explain why we need the input continuity at  $t = 0$  in our problem formulation. Recall that without the initial condition, the relation (2.5) would not hold, and this would consequently invalidate (C.5).

### Characteristics of the Worst-Case Input for Second-Order Linear Systems

The relation (C.5) is quite useful since it allows us to deal with the characterization of the worst-case input  $\bar{w}(t)$  instead of  $\hat{w}(t)$ . Obviously, the expression of  $\bar{w}(t)$  is analytically simpler to construct for a second-order convolution system. Henceforth, let the definitions and theorems in Chapter 2 be applied to  $\bar{w}(t)$  instead of  $\hat{w}(t)$  for the ease of discussion.

Next, we derive the characteristics of the worst-case input  $\bar{w}(t)$  for the second-order convolution systems by applying Theorem 2.1–2.3. Consider the second-order system in (3.23), which is

$$H(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

where  $\omega_n$  is the natural frequency and  $\zeta$  is the damping ratio. Note that the second-order system in this form has a unit dc-gain. For the system with different gain, its WCN can be computed with scaling. Next, we categorize the system with respect to  $\omega_n$  and  $\zeta$ . Let  $T_c$  equal  $2M/D$ . If  $\omega_n \leq \pi/T_c$ , the system (3.23) can be divided into three common cases as follows:

1. Underdamped case:  $0 < \zeta < 1$ ,
2. Critically-damped case:  $\zeta = 1$ ,
3. Overdamped case:  $\zeta > 1$ ,

If  $\omega_n > \pi/T_c$ , define  $\zeta_c = \sqrt{1 - (\pi/T_c\omega_n)^2}$ . The system (3.23) can be divided into three cases as follows:

1. Lightly-damped case:  $0 < \zeta \leq \zeta_c$
2. Underdamped case:  $\zeta_c < \zeta < 1$ ,
3. Critically-damped and overdamped case:  $\zeta \geq 1$ ,

The second and third cases are the standard classifications, while the first case is purposefully distinguished from the second case. The reason behind this will become clear later.

We would like to point out the approach presented in [19] could not render the formulas of the WCN. In the former work, the step response under consideration has the form  $1 - e^{-t} \sin(\omega t + \pi/2)$ , for some  $\omega$ . This step response cannot be realized by any second-order linear system. In addition, there was no formulas explicitly given for the WCN.

### Lightly-damped Systems

For better understanding, we discuss the lightly-damped case first, and then, move to the underdamped case. The class of lightly-damped systems considered here consists of systems whose damping ratio is less than  $\zeta_c$ . When  $\zeta$  is relatively small, the impulse response  $h(t)$  oscillates more rapidly, which makes it harder for an input to match the oscillatory pattern of  $h(t)$ . Thus, we must sort out the case when  $\zeta$  is *very small* from the typical underdamped case.

We will give the worst-case input  $\bar{w}(t)$ , and then explain how it satisfies Theorem 2.1–2.3. Here, we present  $\bar{w}(t)$  when the final time  $T$  falls into certain time instant which is defined as

$$T_n = \frac{n\pi - \phi}{\omega_d}, \quad n = 1, 2, \dots$$

where  $\omega_d = \omega_n \sqrt{1 - \zeta^2}$  is the damped natural frequency, and  $\phi = \tan^{-1}(\sqrt{1 - \zeta^2}/\zeta)$ . Note that the step response corresponding to the lightly-damped system (and the underdamped system) takes the form

$$s(t) = 1 - \alpha e^{-\sigma t} \sin(\omega_d t + \phi). \quad (\text{C.6})$$

It is obvious to see that

$$s(T_n) = 1, \quad n = 1, 2, \dots$$

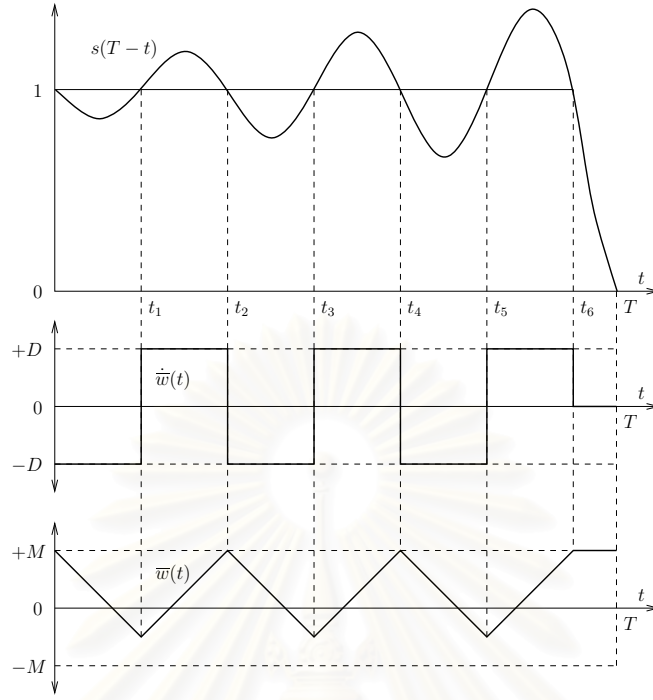


Figure 1: The worst-case input of a lightly-damped second-order system, *i.e.*,  $\omega_d \leq \pi D/2M$ .

As an example, the input  $\bar{w}(t)$  is displayed in Figure 1, when the final time is  $T$  equals  $T_7$ . Notice that

$$s(T) = s(T - t_1) = \dots = s(T - t_6) = 1.$$

In this Figure, the input  $\bar{w}(t)$  has two intervals, starting with a pang interval  $[0, t_6]$  and ending with a bang interval  $(t_6, T]$ . If we reverse the direction of the time axis using  $\tau = T - t$ , and also enumerate backward, *e.g.*, so that  $\tau_1 = t_6$ . It is obvious that in general the initial instant  $\tau'_i$  and the terminal instant  $\tau''_i$  of each sloping segment are

$$\begin{aligned} \tau'_i &= \frac{i\pi - \phi}{\omega_d}, \quad i = 1, \dots, n-1 \\ \tau''_i &= \frac{(i+1)\pi - \phi}{\omega_d}, \quad i = 1, \dots, n-1. \end{aligned} \tag{C.7}$$

This is because  $s(\tau'_i) = s(\tau''_i) = 1$  for  $i = 1, \dots, n-1$ .

At this point, we elaborate how  $\bar{w}(t)$  as shown in Figure 1 complies with the worst-case input characteristics. The boundary condition in Theorem 2.1 is satisfied since  $s(T) = s(T - t_6)$ . The derivative condition in Theorem 2.2 is also satisfied because the sign of  $\dot{\bar{w}}(t)$  is identical to the sign of  $s(T-t) - 1$  in each sloping segment. Lastly, the magnitude of  $\bar{w}(t)$  in the bang interval  $(t_6, T]$  satisfies Theorem 2.3 since its sign is opposite to that of the slope of  $s(T-t)$ . Note that  $ds(T-t)/dt = -h(T-t)$ .

The reason for distinguishing the lightly-damped case from the underdamped case becomes clear as we observe that the input pattern in Figure 1 is no longer valid if the damped natural frequency

$\omega_d$  is so small. If the oscillating period,  $2\pi/\omega_d$ , is large, then the sloping segments of  $\bar{w}(t)$  cannot be contained in the envelope  $\pm M$ . In particular, the system is considered to be lightly-damped if  $\pi/\omega_d$  is less than or equal to  $T_c$ , which is equivalent to saying that  $\zeta \leq \zeta_c$ .

### Underdamped Systems

As mentioned earlier, for the underdamped case that  $\zeta_c < \zeta < 1$ , the damped natural frequency  $\omega_d$  of the system is small enough (or  $\zeta$  is large enough), so that the input with the rate of change equal to  $D$  can catch up with the system response. From the definition of  $T_n$  in the previous section, let us consider an example of response depicted in Figure 2. With  $T = T_5$ , we find the corresponding worst-case input  $\bar{w}(t)$ . As shown in Figure 2,  $\bar{w}(t)$  stays at the magnitude boundaries for some times. There are five bang intervals alternating with four pang intervals. The length of each pang interval is equal to  $T_c$ , which is the time required for  $\bar{w}(t)$  to move from one boundary to the other.

In general, the initial and terminal times of each pang interval can be obtained as follows. Reverse the time axis of  $s(T - t)$  using  $\tau = T - t$ , and then consider  $s(\tau)$  instead. Let  $\tau'$  and  $\tau''$  where  $\tau'' > \tau'$  be time instants representing the two ends of the pang interval. This means that the distance between the two ends is  $T_c$ , i.e.,  $\tau'' - \tau' = T_c$ , and also  $s(\tau') = s(\tau'')$ , following Theorem 2.1. From (C.6), we simply obtain

$$\sin(\omega_d \tau' + \phi) = e^{-\sigma T_c} \sin(\omega_d (\tau' + T_c) + \phi). \quad (\text{C.8})$$

It is straightforward to show that

$$\tan(\omega_d \tau' + \phi) = \left( \frac{\sin(\omega_d T_c)}{e^{\sigma T_c} - \cos(\omega_d T_c)} \right).$$

Let the right-hand side of (C.8) be denoted by  $\psi$ . We have

$$\begin{aligned} \tau'_i &= \frac{1}{\omega_d} (i\pi + \psi - \phi), \quad i = 1, \dots, n-1 \\ \tau''_i &= \tau'_i + T_c. \end{aligned} \quad (\text{C.9})$$

These can be readily converted to get  $t'_1, \dots, t'_n$  and  $t''_1, \dots, t''_n$  by setting  $t = T - \tau$  and varying  $i$  accordingly, but we will not display the results here since (C.9) can be used directly to compute the WCN of the system in the next section.

Since the initial and terminal times of each pang interval is computed under the assumption that  $s(\tau'_i) = s(\tau''_i)$ , the boundary condition in Theorem 2.1 is automatically satisfied. The derivative and magnitude conditions in Theorem 2.2–2.3 are clearly satisfied from Figure 2. Before we proceed, it is interesting to note that, for any  $h(t)$ , the worst-case input  $\bar{w}(t)$  that solves (C.1) is unique [9].

### Overdamped and Critically-damped Systems

We combine the overdamped and critically-damped cases since their step responses share common attribute, that is,  $s(t)$  is non-negative for all time. This is because  $h(t)$  is a non-negative function of

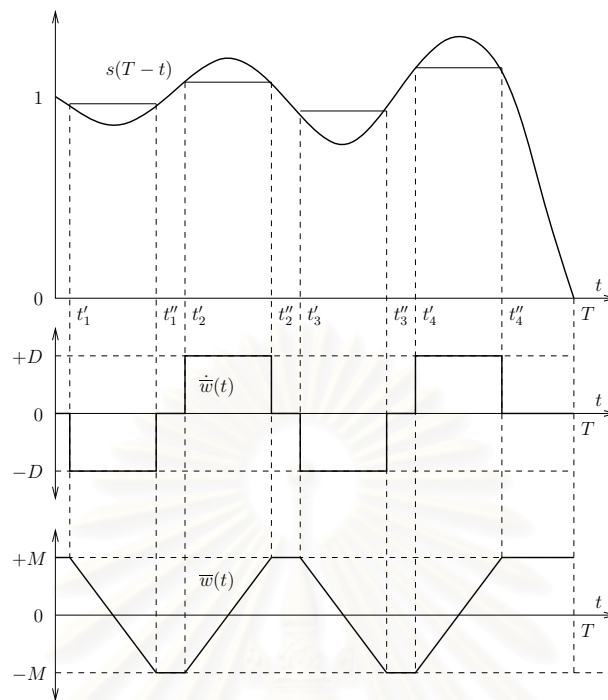


Figure 2: The worst-case input of an underdamped second-order system, *i.e.*,  $\omega_d > \pi D/2M$ .

time. In particular, for an overdamped system,

$$h(t) = \frac{\omega_n(e^{-\sigma_2 t} - e^{-\sigma_1 t})}{2\sqrt{1 - \zeta^2}}$$

where  $\sigma_1 = \sigma + \omega_n/\sqrt{1 - \zeta^2}$ ,  $\sigma_2 = \sigma - \omega_n/\sqrt{1 - \zeta^2}$ , and  $\sigma = \zeta\omega_n$ . In addition, for the critically-damped system,

$$h(t) = \omega_n^2 t e^{-\omega_n t}.$$

This means  $s(T - t)$  must be non-negative, so the input  $\bar{w}(t)$  satisfying Theorem 2.1–2.3 contains only one bang interval  $[0, T]$ . That is

$$\bar{w}(t) = M, \quad \forall t \geq 0$$

In fact, we do not have to consider Theorem 2.1–2.2 since there is no pang interval in  $\bar{w}(t)$ . It is straightforward to see that since  $h(t)$  does not change its sign along the entire interval  $[0, T]$ , so does  $\bar{w}(t)$ . Hence, we get  $\bar{w}(t) = M$  accordingly.

### Formulas for the WCN

We now turn to present formulas for the WCN of the second-order systems which are divided into three cases.

### Lightly-damped Systems

We will begin with computing  $\bar{\xi}(T_n)$  for given  $n$ . Then, the WCN can be obtained via (C.5) as

$$\|h\|_{\text{wc}} = \lim_{n \rightarrow \infty} \bar{\xi}(T_n). \quad (\text{C.10})$$

From the definition (C.1), we have

$$\bar{\xi}(T_n) = \int_0^{T_n} h(T_n - t) \bar{w}(t) dt.$$

This can be integrated by part as

$$\bar{\xi}(T_n) = -[s(T_n - t) - 1] \bar{w}(t) \Big|_0^{T_n} + \int_0^{T_n} [s(T_n - t) - 1] \dot{\bar{w}}(t) dt \quad (\text{C.11})$$

From Figure 1, we can see that  $\bar{w}(T_n) = M$  since the input always terminates in the bang interval. In addition, recall that  $s(0) = 0$  and  $s(T_n) = 1$ . As a result, from (C.11), we have

$$\bar{\xi}(T_n) = M + \int_0^{T_n} [s(T_n - t) - 1] \dot{\bar{w}}(t) dt.$$

By  $\dot{\bar{w}}(t)$  given in Figure 1 and by change of variable,  $\tau = T_n - t$ , this can be expressed as

$$\bar{\xi}(T_n) = M - D \sum_{i=1}^{n-1} (-1)^i \int_{\tau'_i}^{\tau''_i} [s(\tau) - 1] d\tau. \quad (\text{C.12})$$

From the step response in (C.6), we obtain  $\bar{\xi}(T_n)$  as

$$M + \frac{D}{\sqrt{1 - \zeta^2}} \sum_{i=1}^{n-1} (-1)^i \int_{\tau'_i}^{\tau''_i} e^{-\sigma\tau} \sin(\omega_d\tau + \phi) d\tau.$$

Using the initial and final time of each sloping segment given in (C.7), we can show that

$$\begin{aligned} \bar{\xi}(T_n) &= M + \frac{D}{\sqrt{1 - \zeta^2}} \left( \sum_{i=1}^{n-1} (-1)^i e^{-\frac{i\pi - \phi}{\tan \phi}} \int_0^{\pi/\omega_d} e^{-\sigma\tau} \sin(\omega_d\tau + i\pi) d\tau \right), \\ &= M + \frac{DL}{\sqrt{1 - \zeta^2}} \sum_{i=1}^{n-1} e^{-\frac{i\pi - \phi}{\tan \phi}} \end{aligned} \quad (\text{C.13})$$

where  $L = \int_0^{\pi/\omega_d} e^{-\sigma\tau} \sin(\omega_d\tau) d\tau$ , which is equal to

$$\frac{\sqrt{1 - \zeta^2}}{\omega_n} \left( e^{-\frac{\pi}{\tan \phi}} + 1 \right).$$

Computing the geometric series in (C.13), we have

$$\bar{\xi}(T_n) = M + \frac{DL}{\sqrt{1 - \zeta^2}} e^{-\frac{\pi - \phi}{\tan \phi}} \left( \frac{1 - e^{-n\pi \cot \phi}}{1 - e^{-\pi \cot \phi}} \right)$$

In accordance with (C.10), by limiting  $n \rightarrow \infty$ , the WCN of the lightly-damped second-order system is obtained as

$$\|h\|_{\text{wc}} = M + \frac{D \coth\left(\frac{\pi}{2} \cot \phi\right)}{\omega_n e^{(\pi - \phi) \cot \phi}}$$

where  $\coth(\cdot)$  stands for the hyperbolic cotangent function. Note that in this case the WCN is greater than  $M$ . In addition, it can be checked that as  $\zeta$  approaches 0 (as when the system becomes nearly unstable and  $h(t)$  turns purely oscillating), the WCN approaches  $\infty$ .

### Underdamped Systems

The procedure in obtaining the WCN for this case is similar to that of the former case. We can start at (C.12) using  $\tau_i', \tau_i''$  provided in (C.9). As a consequence,  $\bar{\xi}(T_n)$  is computed as

$$\bar{\xi}(T_n) = M + \frac{DL}{\sqrt{1-\zeta^2}} \sum_{i=1}^n e^{-\frac{i\pi+\psi-\phi}{\tan\phi}}. \quad (\text{C.14})$$

where  $L = \int_0^{T_c} e^{-\sigma\tau} \sin(\omega_d\tau + \psi) d\tau$ , which equals

$$\frac{1}{\omega_n} [\sin(\psi + \phi) - e^{-\sigma T_c} \sin(\omega_d T_c + \psi + \phi)].$$

By evaluating the geometric series in (C.14) and limiting  $n \rightarrow \infty$ , the WCN of the underdamped second-order system is as follows:

$$\|h\|_{\text{wc}} = M + \frac{D [\sin(\psi + \phi) - e^{-\sigma T_c} \sin(\omega_d T_c + \psi + \phi)]}{\omega_d e^{(\psi-\phi) \cot\phi} (e^{\pi \cot\phi} - 1)}.$$

When  $\zeta$  approaches 1, the WCN becomes  $M$  as the second term vanishes. On the other hand, when  $\zeta$  approaches  $\zeta_c$ , the WCN exactly approaches that of the lightly-damped case when  $\zeta = \zeta_c$ .

### Overdamped and Critically-damped Systems

For the overdamped and critically-damped cases,  $\bar{w}(t) = M$  for all  $t \geq 0$ . Since  $h(t)$  is non-negative, the WCN of the system is then obtained as a product of  $M$  and its  $\mathcal{L}_1$ -norm. That is

$$\|h\|_{\text{wc}} = M \int_0^{\infty} |h(t)| dt.$$

From the time responses of the second-order systems, we have

$$\int_0^{\infty} |h(t)| dt = 1.$$

As a result,  $\|h\|_{\text{wc}} = M$ .

## Biography

Wathanyoo Khaisongkram was born in Bangkok, Thailand, in 1980. He received B.Eng. and M.Eng. degrees in Electrical Engineering at Chulalongkorn University, in 2001 and 2003, respectively. His undergraduate study, from 1997 to 2001, was supported by the Mitsubishi Electric (Thailand) Foundation Scholarship. During his master study, from 2001 to 2003, he was a scholar of the Honors (Sitkonkuti) Graduate Program for Electrical Engineering Students. Since 2003, he has been a Ph.D. candidate at the Department of Electrical Engineering, Chulalongkorn University. From 2003 to 2006, he was awarded a Scholarship of the Royal Golden Jubilee (RGJ) Ph.D. Program by the Thailand Research Fund (TRF) under Grant No. PHD/0255/2545. Afterwards, in 2007, he serves as a research assistant at Control Systems Research Laboratory, the Department of Electrical Engineering, Chulalongkorn University.

Throughout the undergraduate and graduate studies, Wathanyoo's research has been under the supervision of Dr. David Banjerdpongchai. In summer 2004, he was granted a visiting scholarship by Toshiba International Foundation to enroll in a hands-on training program at the Power and Industrial Systems Research and Development Center, Toshiba Corporation, Tokyo. In summer and autumn 2005, he was a visiting scholar at Information Systems Laboratory, Stanford University, California. The short-term visiting program was part of the RGJ Ph.D. Program, financially supported by the TRF. The research was supervised by Professor Stephen P. Boyd, a renowned professor in convex optimization and its engineering applications. Wathanyoo's field of interest is control systems; his research interests include linear controller analysis and design using convex optimization, computer-aided design tools, worst-case norm of uncertain linear systems and related robust control problems. During the Ph.D. program, Wathanyoo has published several research works listed as follows.

### Journal Publications

1. W. Khaisongkram and D. Banjerdpongchai, "On Computing the Worst-case Norm of Linear Systems Subject to Inputs with Magnitude Bound and Rate Limit", *International Journal of Control*, vol. 80, no. 2, pp. 190–219, February 2007.
2. W. Khaisongkram and D. Banjerdpongchai, "Linear Controller Design and Performance Limits of Binary Distillation Column Subject to Disturbances with Bounds on Magnitude and Rates of Change", *Journal of Process Control*, vol. 16, no. 8, pp. 845–854, September 2006.

### International Conference Proceeding Papers

1. W. Khaisongkram, S. Boyd, and D. Banjerdpongchai, "A Linear Programming Approach for the Worst-Case Norm of Uncertain Linear Systems Subject to Disturbances with Magnitude and Rate Bounds", In *Proceedings of the 45th IEEE Conference on Decision and Control*, pp. 4399–4404, Dec 2006.
2. W. Khaisongkram and D. Banjerdpongchai, "A Branch-and-Bound Algorithm to Compute the Worst-Case Norm of Uncertain Linear Systems under Inputs with Magnitude and Rate Constraints", In *Proceedings of SICE-ICASE International Joint Conference*, pp. 978–983, Oct 2006. (Young Author Award)



3. W. Khaisongkram and D. Banjerdpongchai, "On Improving Performance of Linear Control Systems under Disturbances with Magnitude Bound and Rate Limit", In *Proceedings of the 5th International Conference on Information, Communications and Signal Processing*, pp. 477–481, Dec 2005.
4. W. Khaisongkram and D. Banjerdpongchai, "On Computing the Worst-Case Norm of Convolution Systems: A Comparison of Continuous-Time and Discrete-Time Approaches", In *Proceedings of the 6th IFAC World Congress*, July 2005.
5. W. Khaisongkram and D. Banjerdpongchai, "Performance Limits of Linear Control Distillation Column under Disturbances with Bounds on Magnitudes and Derivatives", In *Proceedings of the 2004 IEEE TENCON*, pp. 605–608, Sep 2004.
6. W. Khaisongkram and D. Banjerdpongchai, "A MATLAB Based GUI for Combined Geometric-Volumetric Calibration of an Inclined Cylindrical Underground Storage Tank Using Regularized Least-Square Method", In *Proceedings of the 2004 IEEE Conference on Control Applications*, pp. 1515–1520, Sep 2004.
7. W. Khaisongkram, D. Banjerdpongchai, and S. Arunsawatwong, "Controller Design for a Binary Distillation Column under Disturbances with Bounds on Magnitudes and Derivatives Using Zakian's Framework", In *Proceedings of the 5th Asian Control Conference*, pp. 1686–1694, July 2004.
8. W. Khaisongkram and D. Banjerdpongchai, "An Optimal Control Approach to Compute the Performance of Linear Systems under Disturbances with Bounded Magnitudes and Bounded Derivatives", In *Proceedings of the 2004 American Control Conference*, pp. 28–33, June–July 2004.

#### **National Conference Proceeding Papers**

1. W. Khaisongkram and D. Banjerdpongchai, "Multi-Objective PID Tuning of Active Suspension System Subject to Load Mass Uncertainty and Road Disturbances with Magnitude and Rate Bounds", In *Proceedings of the 29th Electrical Engineering Conference*, pp. 1057–1060, Nov 2006. (Best Paper Award)
2. W. Khaisongkram and D. Banjerdpongchai, "Linear Controller Design and Performance Limits of Binary Distillation Column Subject to Disturbances with Bounds on Magnitudes and Rates of Change", In *Proceedings of the 27th Electrical Engineering Conference*, pp. 9–12, Nov 2004. (In Thai)

Wathanyoo also gave several other seminar talks and poster presentations. He received awards from local and international conferences as well as Chulalongkorn University. These awards are listed as follows.

#### **International**

- 2006                      *Young Author Award*, SICE-ICASE International Joint Conference,  
 Title: A Branch-and-bound algorithm to compute the worst-case norm of  
 uncertain linear systems under inputs with magnitude and rate constraints.  
 Authors: W. Khaisongkram and D. Banjerdpongchai.

#### **Local**

- 2007                      *The Outstanding Oral Presentation Award* in Engineering,  
 RGJ-Ph.D. Congress VIII.
- 2006                      *The Best Paper Award* in Control and Instrumentation,  
 and *The Outstanding Presentation Award*,

- The 29<sup>th</sup> Electrical Engineering Conference.  
Title: Multi-objective PID tuning of active suspension system subject to load mass uncertainty and road disturbances with magnitude and rate bounds.  
Authors: W. Khaisongkram, D. Banjerdpongchai.
- 2006 *The Outstanding Poster Presentation Award* in Science and Technology,  
The 6th National Symposium on Graduate Research.
- 2005 *The Outstanding Poster Presentation Award* in Engineering, Technology,  
and Environmental Science, RGJ-Ph.D. Congress VI.
- 2004 The Outstanding Electrical Engineering Student Award,  
Faculty of Engineering, Chulalongkorn University.



สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย