โปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้สำหรับวิธีการควบคุมแบบเรียนรู้ซ้ำ

GRAPHICAL USER INTERFACE FOR ITERATIVE LEARNING CONTROL ALGORITHM

นายการุณย์ มะทิมา รหัส 49362727

นายวิโรจน์ พวงมั่ง รหัส 49381292

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ 19 ม.ค. 2555
เลขทะเบียน 1674542
เลขเรียกหนังสือ วิ.
มหาวิทยาลัยนเรศวร ว534ป
2552

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2552

ใบรับรองปริญญานิพนธ์

ชื่อหัวข้อโครงงาน	โปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้สำหรับวิธีการควบคุมแบบเรียนรู้ซ้ำ
ผู้ดำเนินโครงงาน	นายการุณย์ มะทิมา รหัส 49362727
	นายวิโรจน์ พวงมั่ง รหัส 49381292
ที่ปรึกษาโครงงาน	ดร. มุฑิตา สงฆ์จันทร์
สาขาวิชา	วิศวกรรมไฟฟ้า
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2552

.....

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้า

.....ที่ปรึกษาโครงงาน
(ดร. มุฑิตา สงฆ์จันทร์)

.....กรรมการ
(ดร. ศุภวรรณ พลพิทักษ์ชัย)

.....กรรมการ
(ดร. นิพัทธ์ จันทรมินทร์)

ชื่อหัวข้อโครงงาน	โปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้สำหรับวิธีการควบคุมแบบเรียนรู้ซ้ำ
ผู้ดำเนินโครงงาน	นายการุณย์ มะทิมา รหัส 49362727
	นายวิโรจน์ พวงมั่ง รหัส 49381292
ที่ปรึกษาโครงงาน	ดร. มุฑิตา สงฆ์จันทร์
สาขาวิชา	วิศวกรรมไฟฟ้า
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2552

บทคัดย่อ

ปัจจุบันระบบควบคุมมีรูปแบบและพฤติกรรมที่แตกต่างกันออกไป เพื่อที่จะให้ระบบมีผลตอบสนองเป็นไปตามผู้ควบคุมต้องการ จึงต้องมีการเลือกวิธีการควบคุมให้เหมาะสมกับระบบ อย่างไรก็ตามในการออกแบบและคำนวณหาผลตอบสนองของระบบควบคุมจำเป็นต้องมีการใช้โปรแกรมในการคำนวณ ซึ่งมีความยุ่งยากสำหรับผู้ที่เริ่มศึกษาหรือไม่มีพื้นฐานการใช้งานโปรแกรม

โครงงานนี้ได้สร้างโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้สำหรับการควบคุมแบบเรียนรู้ซ้ำ เพื่อลดความยุ่งยาก และสะดวกต่อการใช้งานของผู้ใช้ ซึ่งจากการทดลองพบว่าผลลัพธ์ที่ได้จากโปรแกรมมีความถูกต้อง และมีความสะดวกต่อผู้ใช้โปรแกรม

Project title	Graphical User Interface for Iterative Learning Control Algorithm		
Name	Mr. Karun	Mathima	ID. 49362727
	Mr. Wiroj	Phuangmung	ID. 49381292
Project advisor	Ms. Mutita	Songjun, Ph.D.	
Major	Electrical Engineering		
Department	Electrical and Computer Engineering		
Academic year	2009		

Abstract

Nowadays, control systems have different forms and behavior. In order to obtain the responses as controller's requirement, the appropriate methods should have been selected. However, it is necessary to use any specific program for calculating the responses which is probably difficult for beginners.

Consequently, to reduce the difficulty and complication for calculating the responses of iteration learning control, GUI is chosen. The results from GUI show that the calculation from GUI is accurate and more convenient for users.

กิตติกรรมประกาศ

โครงงานวิศวกรรมไฟฟ้า เรื่องโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้สำหรับวิธีการควบคุมแบบเรียนรู้ซ้ำนี้ สำเร็จลุล่วงลงได้ เนื่องด้วยความอนุเคราะห์จาก อาจารย์มุฑิตา สงฆ์จันทร์ ที่ปรึกษาโครงงานที่คอยช่วยให้คำปรึกษาและแนะนำสิ่งที่เป็นประโยชน์ต่อโครงงานนี้เรื่อยมา ขอบคุณอาจารย์ศุภวรรณ พลพิทักษ์ชัยและอาจารย์นิพัทธ์ จันทรมินทร์ที่กรุณามาเป็นกรรมการในการสอบโครงงาน พร้อมด้วยคณาจารย์ทุกท่านที่ได้ถ่ายทอดความรู้ที่เป็นประโยชน์ให้แก่ศิษย์ทุกคน อีกทั้งกองทุนให้กู้ยืมเพื่อการศึกษา (กยศ.) ที่เปิดโอกาสทางการศึกษาให้แก่นิสิตนักศึกษาทุกคน จึงขอแสดงความขอบคุณเป็นอย่างสูง ณ ที่นี้ด้วย

สุดท้ายขอขอบพระคุณบิดามารดาที่เป็นกำลังใจให้ผู้ดำเนินโครงงานเสมอมา ส่งผลทำให้โครงงานนี้เสร็จสมบูรณ์ไปด้วยดี

นายการุณย์ มะทิมา

นายวิโรจน์ พวงมั่ง

สารบัญ

หน้า

ใบรับรองปริญญานิพนธ์ ..... ก

บทคัดย่อภาษาไทย ..... ข

บทคัดย่อภาษาอังกฤษ ..... ค

กิตติกรรมประกาศ ..... ง

สารบัญ ..... จ

สารบัญรูป ..... ช

บทที่ 1 บทนำ 1

1.1 ที่มาและความสำคัญ 1

1.2 วัตถุประสงค์ของโครงงาน 1

1.3 ขอบเขตของโครงงาน 2

1.4 ขั้นตอนการดำเนินงาน 2

1.5 ผลประโยชน์ที่ได้รับ 3

1.6 งบประมาณ 3

บทที่ 2 ทฤษฎีและหลักการที่เกี่ยวข้อง 4

2.1 วิธีการควบคุมแบบเรียนรู้ซ้ำ (Iterative Learning Control) 4

2.2 โปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้ (Graphical User Interface) 20

บทที่ 3 ขั้นตอนการดำเนินโครงงาน 33

3.1 ขั้นตอนการควบคุมแบบเรียนรู้ซ้ำ 33

3.2 การทำงานของโปรแกรม 38

3.3 การออกแบบและการเขียนโปรแกรม 39

สารบัญ (ต่อ)

หน้า

บทที่ 4 ผลการดำเนินโครงงาน 49

4.1 การเริ่มต้นใช้งานโปรแกรม 49

4.2 ขั้นตอนการใช้งานโปรแกรม 49

4.3 การทดสอบความถูกต้องของโปรแกรม 58

บทที่ 5 สรุปผลการดำเนินโครงงาน 88

5.1 สรุปและวิเคราะห์ผล 88

5.2 ปัญหาที่พบ 89

5.2 ข้อเสนอแนะ 90

เอกสารอ้างอิง 91

ประวัติผู้ดำเนินโครงงาน 92

สารบัญรูป

รูปที่		หน้า
2.1	แผนภาพองค์ประกอบของวิธีการควบคุมแบบเรียนรู้ซ้ำ	5
2.2	แผนภาพองค์ประกอบของ graphics object	20
2.3	หน้าต่างของการเรียกใช้ GUIDE ของโปรแกรม MATLAB	21
2.4	หน้าต่างของเครื่องมือจากคำสั่ง GUIDE ของโปรแกรม MATLAB	22
2.5	ตัวอย่างการเรียกใช้ตัวแก้ไขคุณสมบัติ	23
2.6	ตัวอย่างตัวแก้ไขคุณสมบัติ	23
2.7	ตัวอย่างการเรียกใช้ฟังก์ชันเรียกกลับ	24
2.8	ตัวอย่างฟังก์ชันเรียกกลับเมื่อมีการเรียกใช้เพื่อเขียนคำสั่งการทำงาน	24
2.9	ตัวอย่างลักษณะและการทำงานของปุ่มกด	25
2.10	ตัวอย่างลักษณะและการทำงานของกล่องตรวจสอบ	26
2.11	ตัวอย่างลักษณะและการทำงานของปุ่ม Radio	26
2.12	ตัวอย่างลักษณะและการทำงานของปุ่ม Toggle	27
2.13	ตัวอย่างลักษณะและการทำงานกล่องรายการ	27
2.14	ตัวอย่างลักษณะการทำงานของ Popup menus	28
2.15	ตัวอย่างลักษณะและการทำงานของตัวเลื่อน	28
2.16	ตัวอย่างลักษณะของกรอบ	29
2.17	ตัวอย่างลักษณะตัวอักษรที่อยู่กับที่	29
2.18	ตัวอย่างลักษณะและการทำงานของกล่องแก้ไขตัวอักษร	30
2.19	ตัวอย่างการแจ้งสถานะการทำงานโปรแกรม	30
2.20	ตัวอย่างความพร้อมในการทำงานของโปรแกรม	31
2.21	ตัวอย่างข้อความเตือนเมื่อมีข้อผิดพลาดโปรแกรม	31
3.1	แผนภาพการทำงานของวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้ค่านอร์มที่เหมาะสมที่สุด	35
3.2	แผนภาพการทำงานของวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้ตัวแปรที่เหมาะสมที่สุด	36

สารบัญรูป (ต่อ)

รูปที่		หน้า
3.3	แผนภาพการทำงานของวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้โดยใช้พหุนามของเมทริกซ์ G	37
3.4	แผนภาพขั้นตอนการทำงานของโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้	38
3.5	ตัวอย่างการวางวัตถุบนหน้าต่างหลักของโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้	39
3.6	ตัวอย่างการแก้ไขคุณสมบัติวัตถุบนหน้าต่างหลักของโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้	40
3.7	ตัวอย่างการวางวัตถุบนหน้าต่างของ Norm Optimization ILC	41
3.8	ตัวอย่างการแก้ไขคุณสมบัติวัตถุบนหน้าต่างของ Norm Optimization ILC	42
3.9	ตัวอย่างการวางวัตถุบนหน้าต่างของ Parameter Optimization ILC	44
3.10	ตัวอย่างการแก้ไขคุณสมบัติวัตถุบนหน้าต่างของ Parameter Optimization ILC	44
3.11	ตัวอย่างการวางวัตถุบนหน้าต่างของ Polynomial of matrix G ILC	46
3.12	ตัวอย่างการแก้ไขคุณสมบัติวัตถุบนหน้าต่างของ Polynomial of matrix G ILC	47
4.1	หน้าต่างหลักของโปรแกรม	49
4.2	หน้าต่างวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้ค่าประจำนอร์มที่เหมาะสมที่สุด	50
4.3	หน้าต่างวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้ตัวแปรที่เหมาะสมที่สุด	52
4.4	หน้าต่างวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้พหุนามของเมทริกซ์ G	55
4.5	กราฟแสดงค่าสัญญาณขาเข้าของระบบอันดับหนึ่งที่ได้จากวิธี NOILC	58
4.6	กราฟแสดงค่าสัญญาณขาออกของระบบอันดับหนึ่งที่ได้จากวิธี NOILC	59
4.7	กราฟแสดงค่าสัญญาณความผิดพลาดของระบบอันดับหนึ่งที่ได้จากวิธี NOILC	59
4.8	กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับหนึ่งที่ได้จากวิธี NOILC	60
4.9	กราฟแสดงค่าสัญญาณขาเข้าของระบบอันดับหนึ่งที่ได้จากวิธี POILC	60
4.10	กราฟแสดงค่าสัญญาณขาออกของระบบอันดับหนึ่งที่ได้จากวิธี POILC	61

สารบัญรูป (ต่อ)

รูปที่ หน้า

4.11 กราฟแสดงค่าสัญญาณความผิดพลาดของระบบอันดับหนึ่งที่ได้จากวิธี POILC 61

4.12 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับหนึ่ง ที่ได้จากวิธี POILC 62

4.13 กราฟแสดงค่าตัวแปรอิสระ (Gamma) ของระบบอันดับหนึ่งที่ได้จากวิธี POILC 62

4.14 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับหนึ่ง ในกรณีที่มีการแปรเปลี่ยนตัวแปรค่าถ่วงน้ำหนักแบบคงที่ที่ได้จากวิธี POILC 63

4.15 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับหนึ่ง ในกรณีที่มีตัวแปรค่าถ่วงน้ำหนักแบบคงที่และแบบปรับค่าได้ที่ได้จากวิธี POILC 63

4.16 กราฟแสดงค่าสัญญาณขาเข้าของระบบอันดับหนึ่งที่ได้จากวิธี Polynomial of matrix G ILC 64

4.17 กราฟแสดงค่าสัญญาณขาออกของระบบอันดับหนึ่งที่ได้จากวิธี Polynomial of matrix G ILC 64

4.18 กราฟแสดงค่าสัญญาณความผิดพลาดของระบบอันดับหนึ่งที่ได้จากวิธี Polynomial of matrix G ILC 65

4.19 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับหนึ่ง ที่ได้จากวิธี Polynomial of matrix G ILC 65

4.20 กราฟแสดงค่าตัวแปรอิสระ (Beta) ของระบบอันดับหนึ่งที่ได้จากวิธี Polynomial of matrix G ILC 66

4.21 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับหนึ่ง ในกรณีที่มีการแปรเปลี่ยนจำนวนพจน์ของพหุนามที่ได้จากวิธี Polynomial of matrix G ILC 66

สารบัญรูป (ต่อ)

รูปที่		หน้า
4.22	กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับหนึ่ง ในกรณีที่มีการแปรเปลี่ยนตัวแปรค่าถ่วงน้ำหนักแบบคงที่ที่ได้จากวิธี Polynomial of matrix G ILC	67
4.23	กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับหนึ่ง ในกรณีที่มีตัวแปรค่าถ่วงน้ำหนักแบบคงที่และแบบปรับค่าได้ที่ได้จากวิธี Polynomial of matrix G ILC	67
4.24	กราฟแสดงค่าสัญญาณขาเข้าของระบบอันดับสองที่ได้จากวิธี NOILC	68
4.25	กราฟแสดงค่าสัญญาณขาออกของระบบอันดับสองที่ได้จากวิธี NOILC	69
4.26	กราฟแสดงค่าสัญญาณความผิดพลาดของระบบอันดับสองที่ได้จากวิธี NOILC	69
4.27	กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับสอง ที่ได้จากวิธี NOILC	70
4.28	กราฟแสดงค่าสัญญาณขาเข้าของระบบอันดับสองที่ได้จากวิธี POILC	70
4.29	กราฟแสดงค่าสัญญาณขาออกของระบบอันดับสองที่ได้จากวิธี POILC	71
4.30	กราฟแสดงค่าสัญญาณความผิดพลาดของระบบอันดับสองที่ได้จากวิธี POILC	71
4.31	กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับสอง ที่ได้จากวิธี POILC	72
4.32	กราฟแสดงค่าตัวแปรอิสระ (Gamma) ของระบบอันดับสองที่ได้จากวิธี POILC	72
4.33	กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับสอง ในกรณีที่มีการแปรเปลี่ยนตัวแปรค่าถ่วงน้ำหนักแบบคงที่ที่ได้จากวิธี POILC	73
4.34	กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับสอง ในกรณีที่มีตัวแปรค่าถ่วงน้ำหนักแบบคงที่และแบบปรับค่าได้ที่ได้จากวิธี POILC	73
4.35	กราฟแสดงค่าสัญญาณขาเข้าของระบบอันดับสองที่ได้จากวิธี Polynomial of matrix G ILC	74

สารบัญรูป (ต่อ)

รูปที่ หน้า

4.36 กราฟแสดงค่าสัญญาณขาออกของระบบอันดับสองที่ได้จากวิธี Polynomialof matrix G ILC 74

4.37 กราฟแสดงค่าสัญญาณความผิดพลาดของระบบอันดับสองที่ได้จากวิธี Polynomial of matrix G ILC 75

4.38 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับสอง ที่ได้จากวิธี Polynomial of matrix G ILC 75

4.39 กราฟแสดงค่าตัวแปรอิสระ (Beta) ของระบบอันดับสองที่ได้จากวิธี Polynomial of matrix G ILC 76

4.40 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับสอง ในกรณีที่มีการแปรเปลี่ยนจำนวนพจน์ของพหุนามที่ได้จากวิธี Polynomial of matrix G ILC 76

4.41 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับสอง ในกรณีที่มีการแปรเปลี่ยนตัวแปรค่าถ่วงน้ำหนักแบบคงที่ที่ได้จากวิธี Polynomial of matrix G ILC 77

4.42 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับสอง ในกรณีที่มีตัวแปรค่าถ่วงน้ำหนักแบบคงที่และแบบปรับค่าได้ที่ได้จากวิธี Polynomial of matrix G ILC 77

4.43 กราฟแสดงค่าสัญญาณขาเข้าของระบบใช้งานจริงที่ได้จากวิธี NOILC 78

4.44 กราฟแสดงค่าสัญญาณขาออกของระบบใช้งานจริงที่ได้จากวิธี NOILC 79

4.45 กราฟแสดงค่าสัญญาณความผิดพลาดของระบบใช้งานจริงที่ได้จากวิธี NOILC 79

4.46 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบใช้งานจริง ที่ได้จากวิธี NOILC 80

4.47 กราฟแสดงค่าสัญญาณขาเข้าของระบบใช้งานจริงที่ได้จากวิธี POILC 80

สารบัญรูป (ต่อ)

รูปที่		หน้า
4.48	กราฟแสดงค่าสัญญาณขาออกของระบบใช้งานจริงที่ได้จากวิธี POILC	81
4.49	กราฟแสดงค่าสัญญาณความผิดพลาดของระบบใช้งานจริงที่ได้จากวิธี POILC	81
4.50	กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบใช้งานจริง ที่ได้จากวิธี POILC	82
4.51	กราฟแสดงค่าตัวแปรอิสระ (Gamma) ของระบบใช้งานจริงที่ได้จากวิธี POILC	82
4.52	กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบใช้งานจริง ในกรณีที่มีการแปรเปลี่ยนตัวแปรค่าถ่วงน้ำหนักแบบคงที่ที่ได้จากวิธี POILC	83
4.53	กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบใช้งานจริง ในกรณีที่มีตัวแปรค่าถ่วงน้ำหนักแบบคงที่และแบบปรับค่าได้ที่ได้จากวิธี POILC	83
4.54	กราฟแสดงค่าสัญญาณขาเข้าของระบบใช้งานจริงที่ได้จากวิธี Polynomial of matrix G ILC	84
4.55	กราฟแสดงค่าสัญญาณขาออกของระบบใช้งานจริงที่ได้จากวิธี Polynomial of matrix G ILC	84
4.56	กราฟแสดงค่าสัญญาณความผิดพลาดของระบบใช้งานจริงที่ได้จากวิธี Polynomial of matrix G ILC	85
4.57	กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบใช้งานจริง ที่ได้จากวิธี Polynomial of matrix G ILC	85
4.58	กราฟแสดงค่าตัวแปรอิสระ (Beta) ของระบบใช้งานจริงที่ได้จากวิธี Polynomial of matrix G ILC	86
4.59	กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบใช้งานจริง ในกรณีที่มีการแปรเปลี่ยนจำนวนพจน์ของพหุนามที่ได้จากวิธี Polynomial of matrix G ILC	86

สารบัญรูป (ต่อ)

รูปที่ หน้า

4.60 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบใช้งานจริง ในกรณีที่มีการแปรเปลี่ยนตัวแปรค่าถ่วงน้ำหนักแบบคงที่ที่ได้จากวิธี Polynomial of matrix G ILC 87

4.61 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบใช้งานจริง ในกรณีที่มีตัวแปรค่าถ่วงน้ำหนักแบบคงที่และแบบปรับค่าได้ที่ได้จากวิธี Polynomial of matrix G ILC 87

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญ

วิธีการควบคุมแบบเรียนรู้ซ้ำ (Iterative learning control: ILC) เป็นวิธีการควบคุมที่ถูกคิดค้นขึ้นมาใหม่โดยประยุกต์ใช้ทฤษฎีการเรียนรู้ (Learning) และการทำซ้ำ (Iterative) มีกระบวนการที่มีการทำงานวนซ้ำหลายๆรอบ เพื่อต้องการปรับปรุงผลตอบสนองชั่วครู่ (Transient response) ลดค่าความผิดพลาดของระบบควบคุม และมีการติดตามสัญญาณอ้างอิงได้ดี ทำให้ระบบควบคุมมีความถูกต้องและแม่นยำยิ่งขึ้น สำหรับผู้ที่เริ่มศึกษาเป็นเรื่องค่อนข้างยุ่งยากในการเขียนคำสั่งในโปรแกรมช่วยเหลือเพื่อออกแบบการควบคุมและคำนวณหาผลตอบสนอง จึงต้องมีการสร้างโปรแกรมเพื่อลดความยุ่งยาก และสะดวกต่อผู้ใช้งาน ซึ่งเรียกว่า ส่วนต่อประสานกราฟิกกับผู้ใช้ (Graphical user interface: GUI)

โปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้เป็นวิธีการเชื่อมต่อกันระหว่างผู้ใช้กับคอมพิวเตอร์ โดยผู้ใช้จะป้อนข้อมูลที่ต้องการผ่านทางคีย์บอร์ด, เมาส์ หรือไมโครโฟนให้กับคอมพิวเตอร์ ส่วนคอมพิวเตอร์จะแสดงผลเป็นตัวอักษรหรือกราฟิกต่างๆผ่านทางจอภาพ รูปแบบของวัตถุ (Object) ที่ใช้ติดต่อกันเช่น หน้าต่าง ไอคอน ปุ่ม และตัวอักษรต่างๆ เป็นต้น

โครงงานนี้ จึงมุ่งเน้นการสร้างโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้สำหรับการควบคุมแบบเรียนรู้ซ้ำเพื่อการใช้งานขึ้น

1.2 วัตถุประสงค์ของโครงงาน

เพื่อสร้างโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้สำหรับวิธีการควบคุมแบบเรียนรู้ซ้ำ เพื่อให้ผู้ที่ต้องการศึกษาหรือทดสอบระบบควบคุมสามารถทำได้ง่ายยิ่งขึ้น

1.3 ขอบเขตของโครงงาน

1.3.1 ศึกษาระบบสัญญาณขาเข้าและสัญญาณขาออกเดียว (Single input single output)

1.3.2 ศึกษาระบบเชิงเส้นที่ไม่แปรผันตามเวลา (Linear time invariance system)

1.3.3 ใช้โปรแกรม MATLAB ในการจำลองระบบควบคุมและสร้างโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้

1.3.4 ใช้การควบคุมแบบเรียนรู้ซ้ำด้วยวิธี Norm optimization iterative learning control, Parameter optimization iterative learning control และ Polynomial of matrix G iterative learning control

1.4 ขั้นตอนการดำเนินงาน

รายละเอียด	ปี 2554				
	มิ.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.
1.ศึกษาวิธีการควบคุมแบบเรียนรู้ซ้ำ	←				
2.ศึกษาโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้		←			
3.สร้างโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้ สำหรับวิธีการควบคุมแบบเรียนรู้ซ้ำ		←	→		
4.ทดสอบโปรแกรมและแก้ไข			←	→	
5.สรุปผลและจัดทำรูปเล่มปริญญานิพนธ์					→

1.5 ผลประโยชน์ที่ได้รับ

1.5.1 ใช้เป็นสื่อการเรียนการสอนของวิธีการควบคุมแบบเรียนรู้ซ้ำได้

1.5.2 สะดวกต่อการใช้งาน สำหรับผู้ที่สนใจหรือกำลังศึกษาวิธีการควบคุมแบบเรียนรู้ซ้ำ

1.5.3 ลดขั้นตอนและประหยัดเวลาในการทดสอบระบบด้วยวิธีการควบคุมแบบเรียนรู้ซ้ำ

1.6 งบประมาณ

1.6.1 ค่าถ่ายเอกสาร	500 บาท
1.6.2 ค่าวัสดุคอมพิวเตอร์	500 บาท
1.6.3 ค่าจัดทำรูปเล่มปริญญานิพนธ์	1,000 บาท
รวมเป็นเงินทั้งสิ้น (สองพันบาทถ้วน)	2,000 บาท

หมายเหตุ: ถัวเฉลี่ยทุกรายการ

บทที่ 2
ทฤษฎีและหลักการที่เกี่ยวข้อง

ในระบบควบคุม การออกแบบตัวควบคุมมีหลักการที่สำคัญคือ ทำให้ผลตอบสนองของระบบเป็นไปตามความต้องการของผู้ออกแบบ เมื่อมีการป้อนสัญญาณเข้าที่เหมาะสมให้กับระบบ ซึ่งการออกแบบตัวควบคุมมีอยู่หลายประเภท ซึ่งขึ้นอยู่กับระบบที่ต้องการควบคุม อย่างไรก็ตาม การเขียนคำสั่งในการออกแบบตัวควบคุมและคำนวณหาผลตอบสนองมีความยุ่งยาก จึงมีการสร้างโปรแกรมเพื่อลดความยุ่งยากของผู้ใช้งาน ในบทนี้จะอธิบายทฤษฎีการควบคุมด้วยวิธีการควบคุมแบบเรียนรู้ซ้ำและโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้โดยรายละเอียดมีดังนี้

2.1 วิธีการควบคุมแบบเรียนรู้ซ้ำ

การควบคุมแบบเรียนรู้ซ้ำ เป็นวิธีการแบบใหม่ในระบบควบคุม ซึ่งการควบคุมแบบเรียนรู้ซ้ำจะพิจารณาระบบที่มีกระบวนการทำซ้ำและมีการแก้ไขเพื่อให้ได้ผลลัพธ์ที่มีความแม่นยำ การควบคุมแบบเรียนรู้ซ้ำเป็นวิธีการแก้ปัญหาเกี่ยวกับการปรับปรุงผลตอบสนองชั่วครู่ และติดตามสมรรถนะของระบบที่มีการทำงานซ้ำๆ ได้ เช่น กระบวนการที่ทำงานหรือเคลื่อนที่ซ้ำตำแหน่งเดิม เป็นต้น การควบคุมแบบเรียนรู้ซ้ำสามารถอธิบายแยกเป็นสองคำคือ "การทำซ้ำ" คือการนำเอาสัญญาณและการวัดค่าที่ได้จากการทำงานในรอบที่ผ่านมาใช้เป็นข้อมูลในการสร้างสัญญาณขาเข้าควบคุมในการทำงานในรอบปัจจุบัน "การเรียนรู้" คือการปรับปรุงสัญญาณขาเข้าโดยการบันทึกค่าสัญญาณขาเข้าในรอบก่อนหน้าเอาไว้และนำข้อมูลที่ได้มาสร้างฟังก์ชันสัญญาณขาเข้าควบคุมในรอบใหม่ การควบคุมแบบเรียนรู้ซ้ำได้ถูกนำไปประยุกต์ใช้กับงานอย่างหลากหลาย เช่น แขนกลที่ทำงานซ้ำที่มีต้องการความแม่นยำสูง กระบวนการทางเคมี และงานที่เกี่ยวกับระบบที่มีการติดตามสัญญาณ เป็นต้น

การควบคุมแบบเรียนรู้ซ้ำสามารถแสดงในแบบจำลองทางคณิตศาสตร์ได้ดังนี้

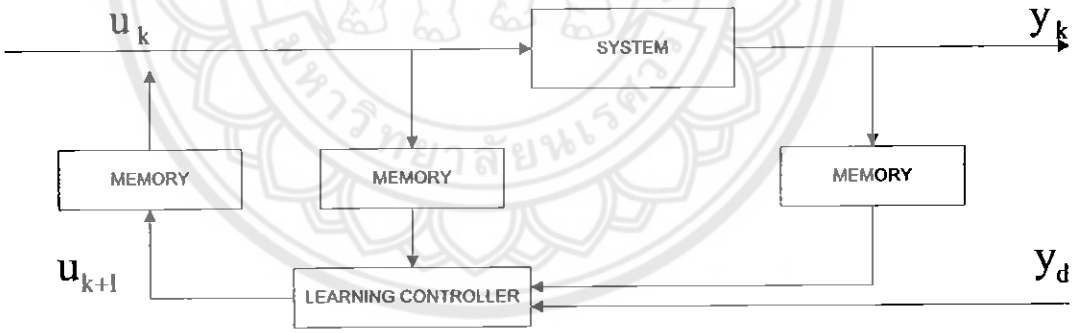
$$\dot{x}(t) = Ax(t) + Bu(t), \;\; x(0) = x_0 \tag{2.1}$$

$$y(t) = Cx(t) + Du(t)$$

ระบบนี้ถูกกำหนดการทำงานในช่วงเวลาจำกัด $t \in [0,T]$ โดยพื้นฐานแล้ววิธีการควบคุมแบบเรียนรู้ซ้ำจะสร้างสัญญาณขาเข้าควบคุมในรอบการทำงานใหม่โดยการบวกเพิ่มพจน์ปรับแก้เข้าไปที่สัญญาณขาเข้าควบคุมในรอบที่ผ่านมา สามารถเขียนสัญญาณขาเข้าควบคุมที่มีการปรับแก้ให้อยู่ในรูปอย่างง่ายได้ดังนี้

$$u_{k+1} = u_k + \Delta u_{k+1} \tag{2.2}$$

สัญญาณขาเข้าควบคุมจะถูกปรับแก้ในทุกรอบการทำงานจนกระทั่งได้สัญญาณขาออกที่ต้องการ และสัญญาณขาเข้าควบคุมที่เหมาะสมที่สุดในรอบการทำงานสุดท้ายควรจะตรงกับสัญญาณขาออกที่ต้องการ วิธีการควบคุมแบบเรียนรู้ซ้ำได้สร้างขึ้นมีสาเหตุมาจากการสังเกตตัวควบคุมระบบมีค่าคงที่และเงื่อนไขในกระบวนการของระบบเหมือนเดิม แล้วค่าความผิดพลาดในผลตอบสนองของสัญญาณขาออกจะได้ค่าเดิมวนซ้ำ ในแต่ละครั้งของการทำงาน พจน์ปรับแก้ที่ถูกเพิ่มเข้าไปนั้น โดยปกติจะเป็นฟังก์ชันค่าความผิดพลาดในรอบที่ผ่านมา เช่น ความแตกต่างระหว่างสัญญาณขาออกจริงกับสัญญาณขาออกที่ต้องการ แนวคิดพื้นฐานของการควบคุมแบบเรียนรู้ซ้ำจะปรับปรุงสมรรถนะของระบบจากรอบการทำงานที่ติดกัน โดยค่าความผิดพลาดในแต่ละรอบควรจะลดลง แนวคิดการควบคุมแบบเรียนรู้ซ้ำสามารถแสดงไว้ในแผนภาพดังรูปที่ 2.1



รูปที่ 2.1 แผนภาพองค์ประกอบของวิธีการควบคุมแบบเรียนรู้ซ้ำ

จากแผนภาพรูปที่ 2.1 สามารถอธิบายได้นี้ ในการทำงานรอบที่ k สัญญาณขาเข้า $u_k(t)$ ถูกป้อนเข้าระบบและทำให้เกิดสัญญาณขาออก $y_k(t)$ สัญญาณขาเข้าและสัญญาณขาออกจะถูกเก็บไว้ที่หน่วยความจำจนกระทั่งสิ้นสุดรอบการทำงาน วิธีการควบคุมแบบเรียนรู้ซ้ำจะคำนวณสัญญาณขาเข้าควบคุมถูกปรับแก้ $u_{k+1}(t)$ และจะเก็บค่าไว้ในหน่วยความจำจนกระทั่งระบบจะเริ่มกระบวนการทำงานอีกครั้งสำหรับการทำงานรอบถัดไป ในเวลานี้สัญญาณขาเข้าควบคุมที่ได้ใหม่จะเป็นค่าเริ่มต้นให้กับระบบ

2.1.1 นิยามการควบคุมแบบเรียนรู้ซ้ำ

การนิยามทางคณิตศาสตร์ที่ถูกต้องของปัญหาการควบคุมแบบเรียนรู้ซ้ำ จะต้องพิจารณาตามแบบจำลองในปริภูมิสถานะของระบบเชิงเส้นไม่แปรผันตามเวลาแบบเวลาต่อเนื่อง ที่ถูกกำหนดในช่วงเวลาจำกัด $t \in [0, T]$ เป็นจุดเริ่มต้น จากสมการที่ (2.1) สามารถแปลงให้อยู่ในแบบจำลองในระบบเชิงเส้นไม่แปรผันตามเวลาแบบเวลาไม่ต่อเนื่อง โดยกำหนดเวลา $t_s = h$ จะได้สมการดังนี้

$$x(t+h) = \emptyset x(t) + \Gamma u(t), \;\; x(0) \tag{2.3}$$

$$y(t) = Cx(t)$$

เมื่อ $A, B, \emptyset, \Gamma, C$ คือ ตัวดำเนินการเป็นเมทริกซ์ที่มีขนาดเหมาะสม

วิธีการควบคุมแบบเรียนรู้ซ้ำยังมีรูปแบบเฉพาะ ถ้า $u_k(t)$ คือ สัญญาณที่ป้อนเข้าไปในระบบในรอบที่ $k \in N$ และ $e_k(t) := r(t) - y_k(t)$ คือ ผลของค่าความผิดพลาด และกฎของการควบคุมสามารถถูกสร้างได้จากสัญญาณค่าความผิดพลาดและ สัญญาณขาเข้า จะได้สมการดังนี้

$$u_{k+1} = f(e_{k+1}, e_k, \dots, e_{k-s}, u_k, u_{k-1}, \dots, u_{k-r}) \tag{2.4}$$

กฎของการควบคุมต้องการให้มีคุณสมบัติ $\lim_{k\to\infty} u_k = \tilde{u}$ และ $\lim_{k\to\infty} e_k = 0$ นอกจากนี้กฎของการควบคุมแบบเรียนรู้ซ้ำยังถูกกำหนดให้สัญญาณอินพุต $u_{k+1}(t)$ เป็นฟังก์ชันสัญญาณความผิดพลาด $e_{k+1}(s)$ เมื่อ $s \le t$ การนิยามปัญหาเริ่มต้นของการควบคุมแบบเรียนรู้ซ้ำจะสมมุติให้สัญญาณขาเข้า $\tilde{u}$ เป็นสัญญาณที่สมบูรณ์ แต่ถ้าไม่เป็นดังกรณีนี้ สามารถแก้ปัญหาโดยวิธีการปรับแก้ให้ลู่เข้าค่าคงที่ค่าหนึ่งคือ u^*และ u^* เป็นผลเฉลยของปัญหาของความเหมาะสมที่สุดดังสมการ

$$u^* = \arg\min_{u\in U} \|r - Gu\|^2 \tag{2.5}$$

เมื่อ U คือ เซตของสัญญาณขาเข้าที่เป็นไปได้

$\|\cdot\|$ คือ ค่านอร์ม

ถ้าระบบสามารถติดตามสัญญาณอ้างอิงได้อย่างสมบูรณ์แบบแล้ววิธีการควบคุมแบบเรียนรู้ซ้ำสามารถหาได้จาก $r = Gu^*$ และ G สามารถผกผันได้ เมื่อ u^* สามารถหาได้จากสัญญาณอ้างอิงที่กำหนดกับค่าผกผันของระบบซึ่งรูปแบบสามารถเขียนแสดงได้ว่า $u^* = G^{-1}r$

วิธีการนี้ทำให้การควบคุมแบบเรียนรู้ซ้ำมีคุณสมบัติการลู่เข้าในหนึ่งรอบการทำงาน โดยเริ่มพิจารณาจากวิธีผกผันของระบบเชิงเส้นดังสมการ

$$u_{k+1} = u_k + G_e^{-1} e_k \tag{2.6}$$

ค่าความผิดพลาดในรอบที่หนึ่งสามารถคำนวณได้ง่ายจาก

$$e_1 = r - Gu_1 = G[u_0 + G^{-1} e_0] = 0 \tag{2.7}$$

อย่างไรก็ตามความผกผันของฟังก์ชัน G ของระบบไม่สามารถเป็นจริงในทางปฏิบัติ ดังนั้นผลเฉลยที่ได้เป็นไปในทางทฤษฎีเท่านั้น และจำนวนรอบการทำงานที่เป็นอนันต์ก็ไม่สามารถยอมรับได้ในทางปฏิบัติเช่นกัน

2.1.2 ระบบการควบคุมแบบเรียนรู้ซ้ำ

จากหัวข้อที่ผ่านมา ระบบเชิงเส้นในสมการที่ (2.3) การพิจารณากระบวนการวิธีการควบคุมแบบเรียนรู้ซ้ำ ในกรณีที่ระบบเป็นสัญญาณข้าเข้าและสัญญาณขาออกเดียว ตัวกระทำการ $\emptyset, \Gamma$และ C เป็นเมทริกซ์ที่มีมิติเท่ากับ $n \times n, n \times 1$ และ $1 \times n$ ตามลำดับ นอกจากนี้วิธีการควบคุมแบบเรียนรู้ซ้ำยังมีจุดมุ่งหมายอื่นอีกคือ มีความแม่นยำในการติดตามค่าสัญญาณอ้างอิงที่กำหนดจะต้องมีแนวโน้มเพิ่มขึ้นในทุกๆรอบการทำงาน และความถูกต้องแม่นยำในการติดตามค่าสัญญาณอ้างอิงควรจะเกิดขึ้นเมื่อจำนวนรอบการทำงานเพิ่มขึ้นจนถึงรอบการทำงานอนันต์

ฟังก์ชันถ่ายโอนของระบบที่แสดงในสมการที่ (2.3) สามารถเขียนได้ดังนี้

$$G(z) = \frac{N(z)}{D(z)} = \frac{b_1 z^{n-1} + \cdots + b_n}{z^n + a_1 z^{n-1} + \cdots + a_n} \tag{2.8}$$

และผลเฉลยของระบบเชิงเส้นแบบเวลาไม่ต่อเนื่องแสดงได้ ดังนี้

$$y(t) = C\emptyset^t x_0 + \sum_{i=0}^{t-1} C\, \emptyset^{t-1-i} \Gamma u(i) \tag{2.9}$$

ในการวิเคราะห์หาผลเฉลยด้วยวิธีการควบคุมแบบเรียนรู้ซ้ำจะใช้วิธีการแทนค่าแบบจำลองทางคณิตศาสตร์นี้ในรูปแบบของเมทริกซ์ที่สัมพันธ์กันระหว่างเวกเตอร์แสดงจุดเวลาต่อเนื่องด้านสัญญาณขาเข้าและด้านสัญญาณขาออกในแต่ละรอบการทำงาน และในกรณีที่องศาสัมพัทธ์ของเมทริกซ์ฟังก์ชันถ่ายโอน $m \geq 1$ สัญญาณขาเข้าควบคุม สัญญาณขาออก และสัญญาณอ้างอิงสามารถระบุได้ดังนี้

$$u_k = \left(u_k(0), u_k(1), \dots, u_k(n-m)\right)$$

$$y_k = \left(y_k(m), y_k(m+1), \dots, y_k(n)\right) \quad (2.10)$$

$$r = \left(r(m), r(m+1), \dots, r(N)\right)$$

เมื่อดัชนี k แสดงจำนวนรอบการทำงาน เพื่อความง่ายในการคำนวณจะสมมุติให้องศาสัมพัทธ์ของเมทริกซ์ฟังก์ชันถ่ายโอนมีค่าเท่ากับหนึ่ง $m = 1$ ในวิธีการควบคุมแบบเรียนรู้ซ้ำจะพิจารณาเฉพาะในช่วงเวลาจำกัดเท่านั้น ($N < \infty$) ดังนั้นสามารถเขียนสัญญาณขาออก สัญญาณขาเข้าควบคุม และสัญญาณอ้างอิงให้อยู่ในรูปของเวกเตอร์ได้ดังนี้

$$y = \begin{bmatrix} y(1) \\ y(2) \\ \vdots \\ y(N) \end{bmatrix}, u = \begin{bmatrix} u(0) \\ u(1) \\ \vdots \\ u(N-1) \end{bmatrix}, r = \begin{bmatrix} r(1) \\ r(2) \\ \vdots \\ r(N) \end{bmatrix} \quad (2.11)$$

ระบบเชิงเส้นในสมการที่ (2.3) สามารถเขียนให้อยู่ในรูปแบบผลเฉลยของสัญญาณขาออกได้ดังนี้

$$y = Gu + d \quad (2.12)$$

เมื่อนำค่า y, u และ r ในสมการที่ (2.11) แทนลงในสมการที่ (2.12) สามารถเขียนสมการให้อยู่ในรูปแบบของเมทริกซ์ที่สมมูลกันได้ดังนี้

$$\begin{bmatrix} y(1) \\ y(2) \\ y(3) \\ \vdots \\ y(N) \end{bmatrix} = \begin{bmatrix} C\Gamma & 0 & 0 & \cdots & 0 \\ C\Phi\Gamma & C\Gamma & 0 & \cdots & 0 \\ C\Phi^2\Gamma & C\Phi\Gamma & C\Gamma & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ C\Phi^{N-1}\Gamma & C\Phi^{N-2}\Gamma & C\Phi^{N-3}\Gamma & \cdots & C\Gamma \end{bmatrix} \begin{bmatrix} u(0) \\ u(1) \\ u(2) \\ \vdots \\ u(N-1) \end{bmatrix} + \begin{bmatrix} C\Phi x_0 \\ C\Phi^2 x_0 \\ C\Phi^3 x_0 \\ \vdots \\ C\Phi^N x_0 \end{bmatrix} \quad (2.13)$$

เมื่อเมทริกซ์ $G \in R^{(pN)\times(mN)}$ เป็นเมทริกซ์แบบสามเหลี่ยมล่างคือ $G_{ij} = G_{(i+1)(j+1)}$ เมื่อ $0 \le i, j \le N-1$ ที่เป็นระบบเชิงเส้นแบบไม่แปรผันตามเวลาซึ่งแสดงได้ดังนี้

$$\begin{bmatrix} C\Gamma & 0 & 0 & \cdots & 0 \\ C\Phi\Gamma & C\Gamma & 0 & \cdots & 0 \\ C\Phi^2\Gamma & C\Phi\Gamma & C\Gamma & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ C\Phi^{N-1}\Gamma & C\Phi^{N-2}\Gamma & C\Phi^{N-3}\Gamma & \cdots & C\Gamma \end{bmatrix} \quad (2.14)$$

ตัวประกอบ $C\emptyset^j\Gamma$ของเมทริกซ์ G นี้เรียกว่า ตัวแปรมาร์คอฟของระบบเชิงเส้นในสมการ (2.3) และ d แสดงถึงผลกระทบจากผลตอบสนองของเงื่อนไขเริ่มต้นที่แสดงในรูปแบบของเมทริกซ์ได้ดังนี้

$$\begin{bmatrix} C\emptyset x_0 \\ C\emptyset^2 x_0 \\ C\emptyset^3 x_0 \\ \vdots \\ C\emptyset^N x_0 \end{bmatrix} \tag{2.15}$$

ต่อจากนี้จะสมมุติให้ $C\emptyset^j\Gamma \neq 0$ เมื่อ $j \geq 0$ และในการเพิ่มเมทริกซ์ G หรือ G^T เป็นเมทริกซ์จัตุรัส เป็นเมทริกซ์ไม่เอกฐานและสามารถหาฟังก์ชันผกผันได้ก็ต่อเมื่อ $C\Gamma \neq 0$ และข้อกำหนดนี้จะเป็นข้อกำหนดที่ใช้สำหรับการวิเคราะห์ระบบที่มีองศาสัมพัทธ์ $m = 1$ แต่ถ้า $C\Gamma = 0$ จะเห็นว่าสัญญาณขาออกตัวแรก $y(1)$ จะไม่ได้รับผลกระทบจากสัญญาณขาเข้าตัวแรก u_0 แต่สัญญาณขาเข้าตัวแรกจะไปมีผลต่อสัญญาณขาออกตัวต่อไป $y(t)$ แทน และสัญญาณขาเข้าที่จุดเวลาสุดท้าย $u(N-1)$ ก็จะถูกเลื่อนหายออกไปจากเวกเตอร์ในนิยามของ y และ u

พิจารณาปัญหาการติดตามสัญญาณ เมื่อสัญญาณอ้างอิงหรือสัญญาณขาออกที่ต้องการแสดงด้วยตัวแปร $r(t)$ เมื่อเวลาอยู่ในช่วง $1 \leq t \leq N$ แล้วนั้น สัญญาณค่าความผิดพลาดของการติดตามในรอบการทำงานที่ k ดังแสดงในสมการต่อไปนี้

$$e_k = r - Gu_k - d = (r - d) - Gu_k \tag{2.16}$$

เมื่อสัญญาณค่าความผิดพลาด e_k และสัญญาณอ้างอิง r เป็นเวกเตอร์ที่ถูกนิยามให้มีมิติที่สัมพันธ์กับสัญญาณขาออก y สำหรับกรณีที่ไม่มีความสูญเสียทั่วไป สัญญาณอ้างอิง r จะถูกแทนด้วย $r - d$ และสมมุติให้ $d = 0$ ซึ่งข้อสมมุตินี้สมมูลกับการสมมุติเงื่อนไขเริ่มต้นของตัวแปรสถานะ x_0 ให้เท่ากับศูนย์

<u>นิยามที่ 2.1</u> พิจารณาระบบที่มีสัญญาณขาเข้า u และสัญญาณขาออก y โดยให้ Y และ U เป็นปริภูมิของฟังก์ชันสัญญาณขาออกและฟังก์ชันสัญญาณขาเข้าตามลำดับและกำหนดให้ $r \in Y$ เป็นสัญญาณอ้างอิงที่ต้องการจากระบบ กระบวนการวิธีการควบคุมแบบเรียนรู้ซ้ำจะสำเร็จ ก็ต่อเมื่อสามารถสร้างลำดับของสัญญาณขาเข้าควบคุม $\{u_k\}_{k\geq 0}$ ให้เกิดการสร้างลำดับของสัญญาณขาออก $\{y_k\}_{k\geq 0}$ ที่มีคุณสมบัติของการเรียนรู้แบบลู่เข้าดังนี้

$$\lim_{k\to\infty} y_k = r, \ \lim_{k\to\infty} u_k = u^* \tag{2.17}$$

การหาค่าที่น้อยที่สุดของสัญญาณขาเข้าควบคุม u^* สำหรับปัญหาค่าที่เหมาะสมที่สุดได้ดังนี้

$$u^* = \arg\min_u\{\|e\|^2 : e = r - y, y = Gu\} \tag{2.18}$$

นิยามที่ 2.2 วิธีการควบคุมแบบเรียนรู้ซ้ำจะเป็นระบบที่มีคุณสมบัติแบบเหตุและผล (causal) ถ้าค่าของสัญญาณขาเข้าควบคุมที่จุดเวลา t ในรอบการทำงานที่ $k+1$ ถูกคำนวณจากข้อมูลที่ได้มาจากการคำนวณในรอบการทำงานที่ $k+1$ ภายในช่วงเวลา $[0,t]$ และจากการคำนวณในรอบที่ผ่านมาในช่วงเวลาทั้งหมด $[0,T]$ เท่านั้น

นิยามที่ 2.3 ปริภูมิเชิงเส้นของค่านอร์มเป็นปริภูมิเวกเตอร์ V บนปริภูมิจำนวนจริง R และการส่งค่าของค่านอร์ม $\|\cdot\|: V \to R$ จะต้องมีคุณสมบัติต่อไปนี้

1. $\|v\| \geq 0$ $\quad \forall v \in V$
2. $\|v\| = 0$ $\quad v = 0$
3. $\|av\| = |a|\|v\|$ $\quad \forall v \in V,\ \forall v \in R$ (2.19)
4. $\|v + w\| = \|v\| + \|w\|$ $\quad \forall v, w \in V$

2.1.3 คุณสมบัติการลู่เข้าของวิธีการควบคุมแบบเรียนรู้ซ้ำ

จากที่ได้กล่าวมาจะเห็นได้ว่าในกลไกกระบวนการการเรียนรู้นั้นได้แนะนำตัวแปรใหม่เกี่ยวกับจำนวนรอบการทำงานนั่นคือตัวแปร k ซึ่งการเพิ่มตัวแปรใหม่เข้าไปนี้เป็นผลทำให้วิธีการควบคุมแบบเรียนรู้ซ้ำเป็นระบบแบบสองมิติที่มีตัวแปรสองตัวที่เป็นอิสระต่อกันหรือมีสองแกนคือแกนของเวลาที่จำกัด $t \in [0,T]$ และแกนของรอบการทำงานที่ไม่จำกัด $k \in N$ สำหรับการวิเคราะห์คุณสมบัติการลู่เข้าพิจารณาจากวิธีการควบคุมแบบเรียนรู้ซ้ำต่อไปนี้

$$u_{k+1}(t) = u_k(t) + [Ke_k](t) \tag{2.20}$$

โดยที่ k คือ ตัวกระทำการเรียนรู้

แบบจำลองทางคณิตศาสตร์ของระบบแสดงได้ดังนี้

$$y_{k+1}(t) = [Gu_{k+1}](t) + z_0(t) \tag{2.21}$$

เมื่อ $t \in [0,T]$ ในการวิเคราะห์หาคุณสมบัติการลู่เข้าของวิธีการควบคุมแบบเรียนรู้ซ้ำนี้จำเป็นจะต้องทราบถึงค่าความผิดพลาด $e_k(t) = r(t) - y_k(t)$ ซึ่งเป็นฟังก์ชันของรอบการทำงานที่ k เสียก่อน สมการวิวัฒนาการของค่าความผิดพลาดสามารถหาได้โดยง่ายจากการประยุกต์วิธีการควบคุมในสมการที่ (2.20) และแบบจำลองคณิตศาสตร์ของระบบในสมการที่ (2.21) พร้อมกับนำนิยามของค่าความผิดพลาด $e_k(t)$ มาใช้ จะได้ผลลัพธ์สุดท้ายดังนี้

$$e_{k+1}(t) = [Le_k](t) \tag{2.22}$$

หรือ

$$e_{k+1} = Le_k + b, \quad k = 0,1,2,\ldots \tag{2.23}$$

กำหนดให้ $L = (I - GK)$ เป็นตัวกระทำการที่ส่งผ่านค่าจาก $e_k(\cdot)$ ไปยัง $e_{k+1}(\cdot)$ ดังนั้นคุณสมบัติของสมการที่ (2.23) จะเป็นเงื่อนไขที่ระบุว่าวิธีการควบคุมจะมีคุณสมบัติการลู่เข้าหรือไม่

ในหัวข้อที่ผ่านมาจะเห็นว่าแนวคิดพื้นฐานของวิธีการควบคุมแบบเรียนรู้ซ้ำก็คือการเรียนรู้ฟังก์ชันสัญญาณขาเข้าแบบซ้ำไปซ้ำมาจนได้ผลลัพธ์ในการติดตามที่สมบูรณ์แบบ นอกจากนี้ค่าความผิดพลาดในแต่ละรอบการทำงานไม่ควรที่จะมีค่ามากเกินไปเพราะค่าความผิดพลาดที่มีค่ามากจะเป็นผลให้คุณภาพของระบบต่ำลงและยังเป็นการทำลายระบบด้วย แต่อย่างไรก็ตามค่านอร์มที่เหมาะสมสามารถใช้วัดค่าความผิดพลาดได้ ดังนั้นในการแก้ปัญหานี้ก็คือจะต้องทำให้ค่า $\|e_{k+1}\|^2$ มีค่าน้อยที่สุดในแต่ละรอบการทำงาน การควบคุมแบบเรียนรู้ซ้ำที่ใช้หลักการของความเหมาะสมที่สุดที่ปรากฏให้เห็นเด่นชัดมีด้วยกันสองแบบคือ การควบคุมแบบเรียนรู้ซ้ำโดยค่านอร์มที่เหมาะสมที่สุด (Norm Optimization Iterative Learning Control: NOILC) และการควบคุมแบบเรียนรู้ซ้ำโดยใช้ค่าตัวแปรที่เหมาะสมที่สุด (Parameter Optimization Iterative Learning Control: POILC) นอกจากนี้ยังได้มีการประยุกต์หลักการของการควบคุมแบบเรียนรู้ซ้ำโดยใช้ตัวแปรที่เหมาะสมที่สุดได้กล่าวไว้ข้างต้น ได้แก่ การควบคุมแบบเรียนรู้ซ้ำอันดับสูง (High Order Iterative Learning Control: HOILC) การควบคุมแบบเรียนรู้ซ้ำโดยใช้พหุนาม (Polynomial Iterative Learning Control) ซึ่งจะกล่าวในหัวข้อต่อไปนี้

2.1.4 การควบคุมแบบเรียนรู้ซ้ำโดยค่านอร์มที่เหมาะสมที่สุด

การควบคุมแบบเรียนรู้ซ้ำโดยค่านอร์มที่เหมาะสมที่สุดเป็นวิธีใหม่ที่ถูกพัฒนาโดย ศาสตราจารย์อ้อมาน โดยจะระบุฟังก์ชันที่ซึ่งผลเฉลยที่เหมาะสมที่สุดได้รวมแบบจำลองภายในของสัญญาณอ้างอิงโดยอัตโนมัติไว้แล้ว ดังนั้นในการเริ่มต้นของการควบคุมแบบเรียนรู้ซ้ำจะพิจารณาจากสมการของแบบจำลองดังนี้

$$y = Gu + z_0 \tag{2.24}$$

โดยที่ G คือ ตัวกระทำการของสัญญาณขาเข้า-สัญญาณขาออก ; $u \in U, y, z_0 \in Y$

U, Y คือ ปริภูมิสัญญาณขาเข้าและปริภูมิสัญญาณขาออกตามลำดับ

โดยสมมุติให้ G เป็นระบบเชิงเส้นและมีขอบเขตจาก Y ไป U โดยค่า z_0 ในแบบจำลองจะอธิบายถึงผลกระทบที่เกิดจากเงื่อนไขเริ่มต้นมีค่าไม่เท่ากับศูนย์ ดังนั้นถ้าพิจารณาไม่มีความสูญเสียทั่วไปจะสมมุติให้ $z_0 = 0$ และตามที่ได้อธิบายไปแล้วในตอนต้น จะเห็นว่าเป้าหมายสูงสุดของวิธีการควบคุมแบบเรียนรู้ซ้ำคือจะต้องแก้ปัญหาค่าความเหมาะสมที่สุด ดังนี้

$$\min_{u \in U} \|e\|^2 \tag{2.25}$$

โดยมีสมการเงื่อนไขบังคับคือ $e = r - Gu$ และสัญญาณอ้างอิงจะถูกสมมุติให้อยู่ในพิสัยของ G ดังนั้นผลเฉลยที่เหมาะสมที่สุด u^* จะสอดคล้องกับ $e^* = r - Gu^* = 0$ ถ้า G และ r เป็นตัวแปรที่ทราบค่า และ r อยู่ในพิสัยของ G และ G เป็นฟังก์ชันผกผัน ผลเฉลยที่เหมาะสมที่สุดเพียงค่าเดียว u^* สามารถหาได้โดยตรงตามทฤษฎีจากสมการ $u^* = G^{-1}r$ แต่ในทางปฏิบัติแล้วจะไม่สามารถทราบค่า G ได้อย่างถูกต้องสมบูรณ์ ดังนั้นจำเป็นต้องใช้ทางเลือกอื่นในการแก้ปัญหานี้ ในการควบคุมแบบเรียนรู้ซ้ำโดยใช้ค่านอร์มนั้น ค่าสัญญาณขาเข้าที่เหมาะสมที่สุดคือ u^* สามารถหาได้จากสมการหาค่าเหมาะสมที่สุดดังนี้

$$\min_{u \in U} J_{k+1}(u_{k+1}) \tag{2.26}$$

เมื่อกำหนดให้ฟังก์ชันของดัชนีสมรรถนะเป็นดังนี้

$$J(u_{k+1}) = \|e_{k+1}\|^2 + \|u_{k+1} - u_k\|^2 \tag{2.27}$$

โดยมีเงื่อนไขสมการบังคับคือ $e_{k+1} = r - Gu_{k+1}$ และค่านอร์มที่ใช้นี้จะสมมุติให้เป็นค่านอร์มแบบเหนี่ยวนำจากผลคูณภายใน สามารถเขียนให้อยู่ในรูปของผลคูณภายในได้เป็น

$\|e_{k+1}\|^2 = \langle e_{k+1}, e_{k+1}\rangle$ และ $\|u_{k+1} - u_k\|^2 = \langle u_{k+1} - u_k, u_{k+1} - u_k\rangle$ โดยที่พจน์ที่หนึ่งนั้นจะมีผลทำให้เกณฑ์ของ $\|e_{k+1}\|^2$ ที่มีค่าน้อยในทุกรอบการทำงาน และพจน์ที่สองจะปรับปรุงค่าของฟังก์ชันสัญญาณขาเข้า u_{k+1}

เพื่อที่จะแสดงถึงคุณสมบัติที่เป็นประโยชน์ของวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้ค่านอร์มที่เหมาะสมที่สุดนั้น จะเริ่มต้นจากการกำหนดให้ u_{k+1}^* เป็นผลเฉลยที่เหมาะสมที่สุดที่รอบการทำงานที่ $k+1$ และให้ e_{k+1}^* เป็นค่าความผิดพลาดที่เหมาะสมที่สุดที่รอบการทำงานที่ $k+1$ ที่สมนัยกับสัญญาณขาเข้า u_{k+1}^* ตัวเลือกอื่นของสัญญาณขาเข้าที่ไม่ใช่ค่าที่เหมาะสมที่สุด $u_{k+1} = u_k$ จะนำไปสู่ความสัมพันธ์ที่ว่า $J_{k+1}(u_k) = \|e_k^*\|^2$ และประโยชน์ที่ได้จากขั้นตอนนี้ทำให้ได้ผลลัพธ์อย่างง่ายคือ

$$\|e_{k+1}^*\|^2 \le J_{k+1}(u_{k+1}^*) \le \|e_k^*\|^2, \quad \forall k \ge 0 \tag{2.28}$$

จะได้ว่าค่านอร์มของค่าความผิดพลาดจะมีคุณสมบัติลู่เข้าแบบลดลงทางเดียวคือ $\|e_{k+1}^*\| \le \|e_k^*\|$ นอกจากนี้ถ้า $\|e_{k+1}^*\| = \|e_k^*\|$ นั่นหมายความว่าขั้นตอนการทำงานสิ้นสุดลงแล้วคือ จะไม่มีการเปลี่ยนแปลงสัญญาณขาเข้าเมื่อรอบการทำงานเพิ่มขึ้น

ในการแสดงให้เห็นว่าวิธีการควบคุมแบบเรียนรู้ซ้ำสามารถลู่เข้าสู่ค่าผลเฉลยที่ถูกต้อง u^* สำหรับกรณีของระบบเชิงเส้นนั้นจะต้องเริ่มแก้ปัญหาของค่าความเหมาะสมที่สุดในสมการที่ (2.26) เป็นอันดับแรก โดยใช้อนุพันธ์ของเฟรเช (Frechet derivative) ในการคำนวณจะได้ผลลัพธ์เงื่อนไขที่จำเป็นสำหรับการหาค่าที่เหมาะสมที่สุดดังสมการต่อไปนี้

$$\langle J_{k+1}(u_{k+1}), \delta u_{k+1}\rangle = \frac{d}{d\epsilon} J_{k+1}(u_{k+1} + \epsilon \delta u_{k+1}) = 0 \tag{2.29}$$

หลังจากที่ทำการคำนวณแล้วค่าผลเฉลยที่เหมาะสมที่สุดสามารถหาได้จากสมการ

$$\frac{d}{d\epsilon} J_{k+1}(u_{k+1} + \epsilon \delta u_{k+1}) = \langle \delta u_{k+1}, u_{k+1} - u_k - G^T e_{k+1}\rangle = 0 \tag{2.30}$$

และผลเฉลยที่ได้จากสมการที่ (2.30) จะเป็นเงื่อนไขที่จำเป็นสำหรับผลเฉลยที่เหมาะสมที่สุด u_{k+1} นั่นคือ

$$u_{k+1} = u_k + G^T e_{k+1} \tag{2.31}$$

ทำการคูณสมการที่ (2.31) ด้วย G และจัดรูปแบบสมการในรูปแบบของความสัมพันธ์ระหว่าง e_{k+1} กับ e_k จะได้สมการใหม่เป็น

$$(I + GG^T)e_{k+1} = e_k \quad (2.32)$$

จะเห็นได้ว่า ตัวกระทำการ $I + GG^T$ ในสมการที่ (2.32) เป็นค่าบวกและสามารถหาฟังก์ชันผกผันได้ ดังนั้นสมการวิวัฒนาการของค่าความผิดพลาดจะแสดงได้ดังนี้

$$e_{k+1} = (I + GG^T)^{-1}e_k \quad (2.33)$$

เมื่อ $(I + GG^T)^{-1}$ คือ ตัวกระทำการการเรียนรู้

จากที่กล่าวถึงทฤษฎีโดยคร่าวๆ ของวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้ค่านอร์มที่เหมาะสมที่สุดจะเห็นว่า ผลลัพธ์ที่ได้จากวิธีการควบคุมแบบนี้จะทำให้ค่าความผิดพลาดค่อยๆ ลดลงจนเป็นศูนย์เมื่อพิจารณาระบบเชิงเส้นไม่แปรผันตามเวลาใดๆ นอกจากนี้ถ้าระบบเป็นแบบเวลาไม่ต่อเนื่องแล้วคุณสมบัติการลู่เข้าจะเป็นแบบเรขาคณิต แต่วิธีการควบคุมแบบนี้ก็มีข้อเสียคือขั้นตอนการทำงานให้สำเร็จผลค่อนข้างมีความยุ่งยากและซับซ้อน

2.1.5 การควบคุมแบบเรียนรู้ซ้ำโดยใช้ตัวแปรที่เหมาะสมที่สุด

วิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้ตัวแปรที่เหมาะสมที่สุด จะเริ่มพิจารณาจากสมการควบคุมที่ใช้หาค่าสัญญาณขาเข้าควบคุมในรอบการทำงานถัดไปแสดงได้ดังนี้

$$u_{k+1}(t) = u_k(t) + \gamma_{k+1}e_k(t+1) \quad (2.34)$$

ขั้นตอนวิธีนี้เป็นขั้นตอนแบบป้อนไปข้างหน้า กล่าวคือไม่จำเป็นต้องใช้ส่วนประกอบอื่นๆเพิ่มเติมในการคำนวณ และพจน์ปรับแก้ที่ใช้ในสมการที่ (2.34) คือ พจน์ที่ประกอบด้วยอัตราขยายของการเรียนรู้คูณกับสัญญาณค่าความผิดพลาดที่มีการเลื่อนทางเวลา ซึ่งสัญญาณค่าความผิดพลาดนี้มาจากการคำนวณในรอบการทำงานที่ผ่านมา จากสมการที่ (2.34) ถ้าไม่สามารถเลือกค่า γ_{k+1} ที่เหมาะสมที่สุดได้จะทำให้ผลลัพธ์ที่ไม่น่าพอใจ ดังนั้นในการหาค่า γ_{k+1} จะต้องคำนวณจากปัญหาค่าความเหมาะสมที่สุดแสดงไว้ดังนี้

$$\gamma_{k+1} = arg\min_{u_{k+1}}\{J_{k+1}(\gamma_{k+1}): e_{k+1} = r - y_{k+1}, y_{k+1} = Gu_{k+1}\} \quad (2.35)$$

เมื่อฟังก์ชันของดัชนีสมรรถนะ $J(\gamma_{k+1})$ ถูกนิยามว่า

$$J(\gamma_{k+1}) = \|e_{k+1}\|^2 + w\gamma_{k+1}^2 \quad (2.36)$$

เมื่อกำหนดให้ $w > 0$ เป็นตัวแปรค่าถ่วงน้ำหนักที่ใช้ในการจำกัดค่าของ γ_{k+1}

จากสมการเงื่อนไขบังคับในสมการที่ (2.36) สามารถเขียนได้เป็น $y_k = Gu_k$ และจากนิยามของค่าความผิดพลาด $e_k = r - Gu_k$ จะได้สมการของความสัมพันธ์ระหว่างค่าความผิดพลาดในรอบการทำงานใหม่กับรอบการทำงานปัจจุบันดังนี้

$$e_{k+1} = (I - \gamma_{k+1}G)e_k, \ \forall k \geq 0 \tag{2.37}$$

สามารถเขียนให้อยู่ในรูปคอส์ทฟังก์ชันได้ ดังนี้

$$J_{k+1}(\gamma_{k+1}) = e_k^T(I - \gamma_{k+1}G)^T(I - \gamma_{k+1}G)e_k + w\gamma_{k+1}^2 \tag{2.38}$$

เงื่อนไขจำเป็นสำหรับความเหมาะสมที่สุด คือจะต้องหาอนุพันธ์ของสมการที่ (2.38) เทียบกับ γ_{k+1} และกำหนดให้เท่ากับศูนย์ดังนี้

$$\frac{1}{2}\frac{dJ_{k+1}(\gamma_{k+1})}{d\gamma_{k+1}} = -e_k^T G e_K + \gamma_{k+1}(w + e_k^T G^T G e_K) = 0 \tag{2.39}$$

แก้สมการที่ (2.39) จะได้ผลเฉลยที่เป็นค่า γ_{k+1} ที่เหมาะสมที่สุด คือ

$$\gamma_{k+1} = \frac{e_k^T G e_k}{w + e_k^T G^T G e_k} \tag{2.40}$$

กล่าวโดยสรุปคือวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้ตัวแปรที่เหมาะสมที่สุดเป็นโครงสร้างแบบใหม่ของวิธีการควบคุมแบบเรียนรู้ซ้ำที่ใช้รูปแบบขั้นตอนวิธีการควบคุมที่คงที่ในการแก้ปัญหา นอกจากนี้จะเห็นว่าขั้นตอนวิธีการควบคุมแบบป้อนไปข้างหน้าจะให้ผลลัพธ์ของค่าความผิดพลาดลู่เข้าสู่ศูนย์แบบลดลงทางเดียว ถ้าระบบเป็นไปตามเงื่อนไขของฟังก์ชันบวกไม่เช่นนั้นค่าความผิดพลาดจะลู่เข้าสู่ค่าที่ไม่เป็นศูนย์

2.1.6 การควบคุมแบบเรียนรู้ซ้ำอันดับสูง

วิธีการควบคุมแบบเรียนรู้ซ้ำอันดับสูงนี้เป็นวิธีที่ถูกพัฒนามาจากวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้ตัวแปรที่เหมาะสมที่สุดโดยที่การควบคุมด้วยวิธีนี้จะใช้ข้อมูลซึ่งอาจจะเป็นค่าความผิดพลาดหรือค่าสัญญาณขาเข้าควบคุมจากรอบการทำงานที่ผ่านมามากว่าหนึ่งรอบเพื่อที่จะใช้ในการปรับปรุงสัญญาณขาเข้าควบคุมในรอบการทำงานปัจจุบัน วิธีการควบคุมแบบเรียนรู้ซ้ำอันดับสูงนี้จะเพิ่มความยืดหยุ่นให้กับการทำงานเมื่อเปรียบเทียบกับวิธีการควบคุมแบบเรียนรู้ซ้ำอันดับหนึ่ง เพราะวิธีนี้จะใช้ค่าตัวแปร M ค่าในการพิจารณา โดยศาสตราจารย์โอเว่น และมิสเตอร์เฟง ได้เสนอวิธีการควบคุมแบบเรียนรู้ซ้ำที่ใช้ข้อมูลเป็นค่าความผิดพลาดจาก M รอบการทำงานที่ผ่านมา ดังแสดงในสมการ

$$u_{k+1}(t) = u_k(t) + \sum_{i=1}^{M} \beta_{k+1}(i) e_{k-i+1}(t+1); 1 \le M \le k \qquad (2.41)$$

เมื่อ $e_i(t) = r(t) - y_i(t)$ เป็นสัญญาณค่าความผิดพลาด และ M เป็นจำนวนรอบของสัญญาณค่าความผิดพลาด e_{k-i+1} ที่ได้จากรอบการทำงานที่ผ่านมา โดยที่เวกเตอร์ของอัตราขยายการเรียนรู้ $\beta_{k+1} := [\beta_{k+1}(1) \quad \beta_{k+1}(2) \quad \cdots \quad \beta_{k+1}(M)]^T$ จะเป็นผลเฉลยของปัญหาความเหมาะสมที่สุด

2.1.7 การควบคุมแบบเรียนรู้ซ้ำโดยใช้พหุนามของเมทริกซ์ G

ที่มาสำหรับวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้พหุนามของเมทริกซ์ G มาจากแนวคิดของวิธีการควบคุมแบบเรียนรู้ซ้ำแบบผกผันซึ่งมีสมการปรับปรุงสัญญาณขาเข้าดังนี้

$$u_{k+1}(t) = u_k(t) + G^{-1} e_k(t+1) \qquad (2.42)$$

- ขั้นตอนของวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้พหุนามของเมทริกซ์ G

วิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้พหุนามของเมทริกซ์ G สามารถอธิบายได้ในรูปแบบทั่วไปดังนี้

$$u_{k+1}(t) = u_k(t) + \sum_{j=1}^{M} \beta_{k+1}(j) G^{j-1} e_k(t+1); \; 1 \le j \le M \le k \qquad (2.43)$$

เมื่อ G คือ เมทริกซ์จัตุรัสที่แสดงข้อมูลของระบบที่กำหนด

M คือ จำนวนพจน์ของพหุนาม และเวกเตอร์ของตัวแปร

$\beta_{k+1}(j)$ คือ ตัวแปรอัตราขยาย

$\beta_{k+1}(j)$ นี้จะมีค่าเปลี่ยนแปลงในทุกรอบการทำงาน โดยที่ค่า $\beta_{k+1}(j)$ ที่เหมาะสมที่สุดสามารถคำนวณได้จากผลเฉลยของปัญหาความเหมาะสมที่สุดของค่านอร์มที่น้อยที่สุด ดังนี้

$$[\beta_{k+1}(1), \dots, \beta_{k+1}(M)] = arg \min_{\beta_{k+1}} \{J_{k+1}(\beta_{k+1}(i)) : e_{k+1} = r - G u_{k+1}\} \qquad (2.44)$$

เมื่อ

$$J(\beta_{k+1}(1), \beta_{k+1}(1), \dots, \beta_{k+1}(M)) \ge 0 \qquad (2.45)$$

ดังนั้นเกณฑ์ความเหมาะสมที่สุดจะถูกใช้ในการหาค่าตัวแปร $\beta_{k+1}(j)$ ดังแสดงได้ในสมการต่อไปนี้

$$J(\beta_{k+1}(1), \ldots, \beta_{k+1}(M)) = \|e_{k+1}\|^2 + \beta_{k+1}^T W \beta_{k+1} \quad (2.46)$$

เมื่อ $W = W^T > 0$ เป็นตัวแปรค่าถ่วงน้ำหนัก

ผลเฉลยของปัญหาค่าความเหมาะสมที่สุดในสมการที่ (2.46) เป็นส่วนสำคัญที่ทำให้ขั้นตอนการคำนวณสำหรับวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้พหุนามเมทริกซ์ G เสร็จสมบูรณ์ และการที่จะให้ได้มาซึ่งผลเฉลยนี้สามารถแสดงให้เห็นได้ด้วยการคำนวณตามลำดับขั้นต่อไปนี้ โดยจะเริ่มต้นที่สมการวิวัฒนาการค่าความผิดพลาด ซึ่งถ้าใช้สมการปรับปรุงค่าสัญญาณขาเข้าควบคุมในสมการ (2.43) และ $e_k = r - y_k$ จะได้สมการวิวัฒนาการค่าความผิดพลาดดังนี้

$$e_{k+1} = e_k - \sum_{j=1}^{M} \beta_{k+1}(j) G^j e_k, \forall k \geq 0, 1 \leq M \leq k \quad (2.47)$$

จากสมการวิวัฒนาการค่าความผิดพลาดในสมการที่ (2.47) สามารถเขียนให้อยู่ในรูปแบบเมทริกซ์ได้ ดังนี้

$$e_{k+1} = e_k - N(e_k)\beta_{k+1} \quad (2.48)$$

เมื่อ $N(e_k) = [Ge_k \quad G^2 e_k \quad \cdots \quad G^M e_k]$ และ $\beta_{k+1} = \begin{bmatrix} \beta_{k+1}(1) \\ \beta_{k+1}(2) \\ \vdots \\ \beta_{k+1}(M) \end{bmatrix}$

ค่าตัวแปร β_{k+1} ที่เหมาะสมที่สุดสามารถคำนวณได้จากดัชนีสมรรถนะในสมการที่ (2.46) และเงื่อนไขที่จำเป็น $\frac{dJ}{d\beta_{k+1}} = 0$ ซึ่งจะได้ค่า β_{k+1} ที่เหมาะสมที่สุดในแต่ละรอบการทำงาน ดังนี้

$$\beta_{k+1} = [W + N(e_k)^T N(e_k)]^{-1} N(e_k)^T e_k \quad (2.49)$$

เมื่อ $W = W^T > 0$

ดังนั้นจะได้เมทริกซ์จตุรัสทแยงมุมขนาด $M \times M$ แสดงได้ดังนี้

$$W = \begin{bmatrix} w_1 & 0 & \cdots & 0 \\ 0 & w_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & w_M \end{bmatrix} \quad (2.50)$$

- **ตัวแปรถ่วงน้ำหนัก**

ในการวิเคราะห์วิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้พหุนามของเมทริกซ์ G นั้นไม่ว่าจะเป็นอันดับหนึ่งหรืออันดับที่สูงกว่าจะต้องเลือกตัวแปรค่าถ่วงน้ำหนักอย่างไรให้มีค่าที่เหมาะสมที่สุดเพื่อใช้ในการคำนวณ โดยตัวแปรค่าถ่วงน้ำหนักจะเป็นตัวแปรอิสระที่สามารถพิจารณาได้เป็นสองกรณีดังนี้

1. ตัวแปรค่าถ่วงน้ำหนักแบบคงที่ $w \geq 0$
2. ตัวแปรค่าถ่วงน้ำหนักแบบปรับค่าได้ $w = w_1 + w_2\|e_k\|^2$

โดยกรณีแรกเป็นตัวแปรถ่วงน้ำหนักแบบคงที่ที่มีค่ามากกว่าหรือเท่ากับศูนย์ แต่จากการทดลองจะพบว่าถ้าให้ตัวแปรค่าถ่วงน้ำหนักมีค่าเท่ากับศูนย์ ($w = 0$) แต่ในทางปฏิบัติจะเลือกค่าตัวแปรถ่วงน้ำหนักที่มีค่าน้อยๆแต่มากว่าศูนย์ จากสมการการปรับปรุงค่า β_{k+1} ที่เหมาะสมที่สุดดังแสดงในสมการที่ (2.49) ถ้า $G^T + G > 0$ และ $w \geq 0$ แล้ว $\|e_k\|^2 \gg w$ และค่าความผิดพลาด จะเข้าใกล้ศูนย์ซึ่งจะนำไปสู่ผลลัพธ์ที่ว่าค่าตัวแปร β_{k+1} จะเข้าใกล้ศูนย์ด้วยเช่นกัน

สำหรับในกรณีที่สองเป็นกรณีที่ใช้การปรับปรุงตัวแปรค่าถ่วงน้ำหนักเพื่อทำให้การลู่เข้าของการผลลัพธ์มีความเร็วที่สูงขึ้น โดยตัวแปรค่าถ่วงน้ำหนักจะมีการเปลี่ยนแปลงค่าในทุกรอบการทำงาน ซึ่งทำให้ได้สมการดัชนีสมรรถนะในรูปแบบใหม่ดังนี้

$$J(\beta_{k+1}) = \|e_{k+1}\|^2 + (w_1 + w_2\|e_k\|^2)(\beta_{k+1})^2 \qquad (2.51)$$

เมื่อ $w_1 \geq 0$ และ $w_2 \geq 0$ หรือในอีกทางหนึ่งอาจกล่าวได้ว่าตัวแปรค่าถ่วงน้ำหนักที่ใช้มีค่าดังนี้

$$w = w_1 + w_2\|e_k\|^2 \qquad (2.52)$$

และถ้าแทนตัวแปรค่าถ่วงน้ำหนักแบบปรับค่าได้ในสมการที่ (2.52) ลงในสมการที่ (2.48) จะสามารถคำนวณหาค่า β_{k+1} ที่เหมาะสมที่สุดได้จาก

$$\beta_{k+1} = [(w_1 + w_2\|e_k\|^2)I + N(e_k)^T N(e_k)]^{-1} N(e_k)^T e_k \qquad (2.53)$$

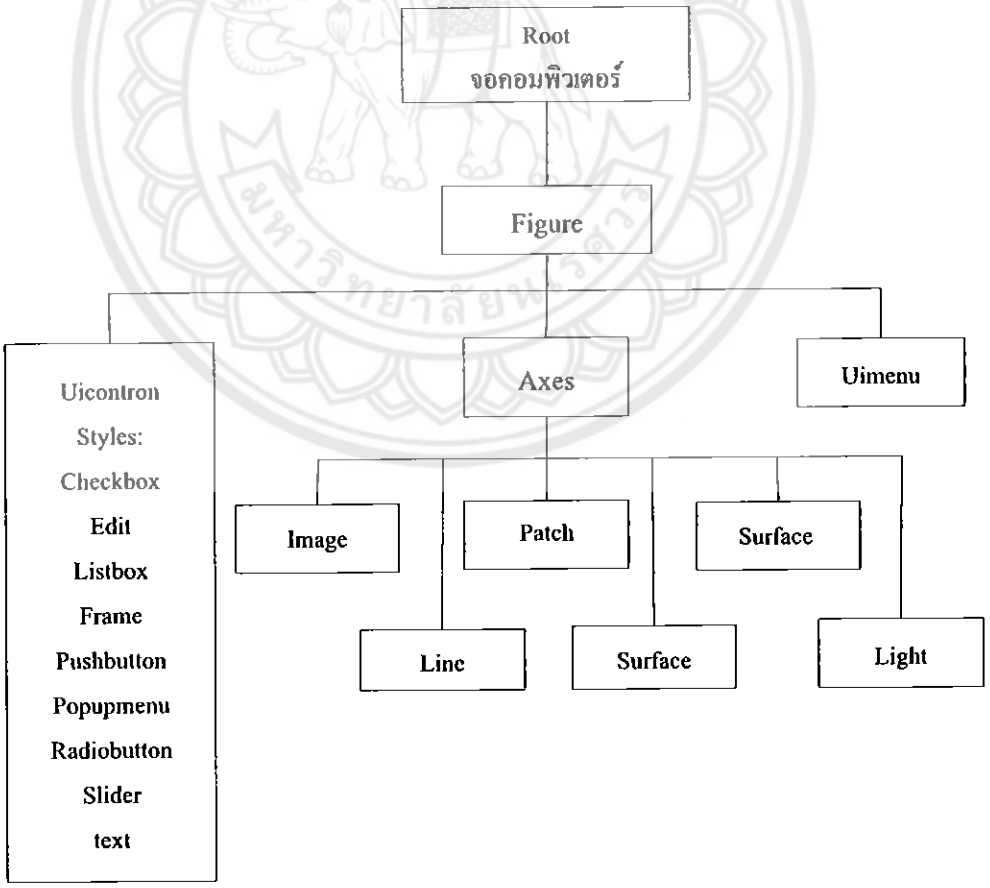
จากที่ได้กล่าวมาแล้วในข้างต้น เห็นได้ว่ากระบวนการวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้พหุนามของเมทริกซ์ G จะมีความยุ่งยากและซับซ้อนกว่าวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้ตัวแปรที่เหมาะสมที่สุด เนื่องจากวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้พหุนามของเมทริกซ์ G จะเป็นวิธีการควบคุมแบบเรียนรู้ซ้ำอันดับสูงขึ้นอยู่กับจำนวนพจน์ของพหุนามที่ใช้ และจำนวนตัวแปรแต่ละตัวที่ปรากฏอยู่ในกระบวนการวิธีเป็นตัวแปรที่อยู่ในรูปแบบของเมทริกซ์ซึ่งทำให้การคำนวณค่าต่างๆในการทำงานแต่ละรอบต้องใช้เวลานาน แต่วิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้พหุนามของเมทริกซ์ G ก็ยังแสดงคุณสมบัติที่สำคัญหลายประการ โดยเฉพาะอย่างยิ่งคุณสมบัติการลู่เข้าซึ่งเป็นคุณสมบัติที่จำเป็นสำหรับวิธีการควบคุมแบบเรียนรู้ซ้ำ และสมรรถนะที่ได้ของวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้พหุนามของเมทริกซ์ G ยังสามารถถูกปรับปรุงได้โดยการปรับปรุงค่าตัวแปรค่าถ่วงน้ำหนักให้เป็นแบบปรับค่าได้ ซึ่งตัวอย่างของสมรรถนะที่ได้จากวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้พหุนามของเมทริกซ์ G เมื่อมีการเปลี่ยนระบบที่ใช้ในการควบคุมและมีการเปลี่ยนค่าสัญญาณอ้างอิง รวมไปถึงการใช้ตัวแปรค่าถ่วงน้ำหนักแบบปรับค่าได้

2.2 โปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้

ส่วนต่อประสานกราฟิกกับผู้ใช้ เป็นวิธีการเชื่อมต่อกันระหว่างผู้ใช้และคอมพิวเตอร์ โดยคอมพิวเตอร์จะถูกใช้งานก็ต่อเมื่อผู้ใช้ป้อนข้อมูลผ่านคีย์บอร์ด เมาส์ ไมโครโฟน เป็นต้น คอมพิวเตอร์จะแสดงผลเป็นตัวอักษรหรือกราฟิกบนจอภาพ การเชื่อมต่อผู้ใช้ทางกราฟิก จะสร้างวัตถุ (Object) ต่างๆที่ใช้สำหรับการติดต่อหรือใช้งานร่วมกันเช่น หน้าต่าง ไอคอน ปุ่ม กรอบ เมนู และตัวอักษรต่างๆ

2.2.1 Handle Graphic

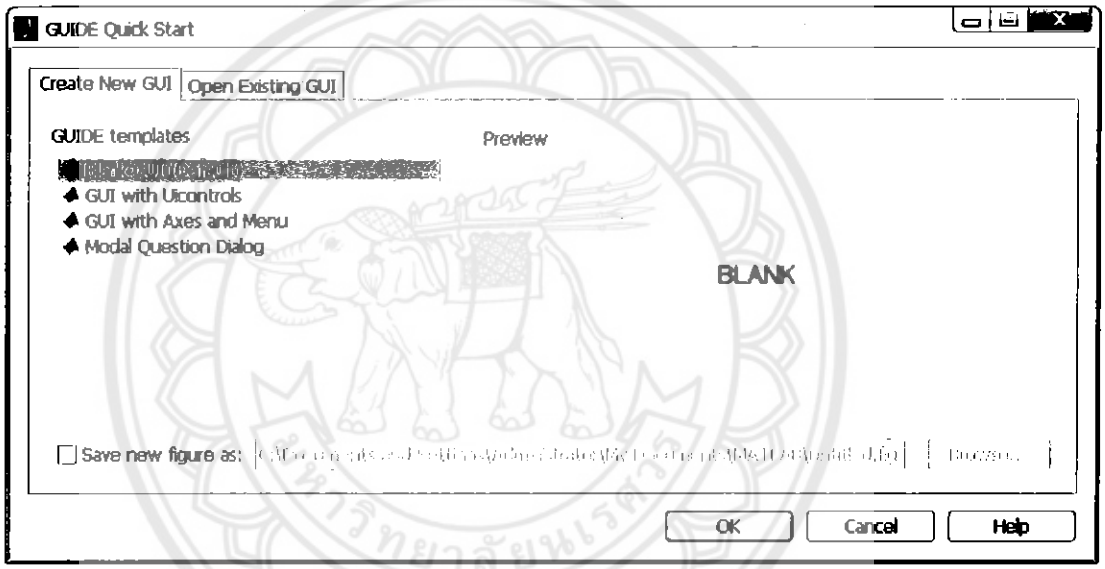
Handle Graphic เป็นพื้นฐานอยู่บนแนวความคิดที่ว่าทุกๆ สิ่งบนหน้าต่างรูปภาพของโปรแกรม MATLAB จะเป็นวัตถุที่อยู่บนจอภาพหรือหน้าต่างรูปภาพจะมีเอกลักษณ์เฉพาะตัว ซึ่งเรียกว่า Handle และแต่ละวัตถุจะมีคุณสมบัติที่สามารถปรับปรุงแก้ไขได้ตลอดเวลา ทุกอย่างที่สร้างโดยใช้คำสั่งทางกราฟิกจะเรียกว่า graphics object ซึ่งมีองค์ประกอบต่างๆ แสดงไว้ดังรูปที่ 2.2



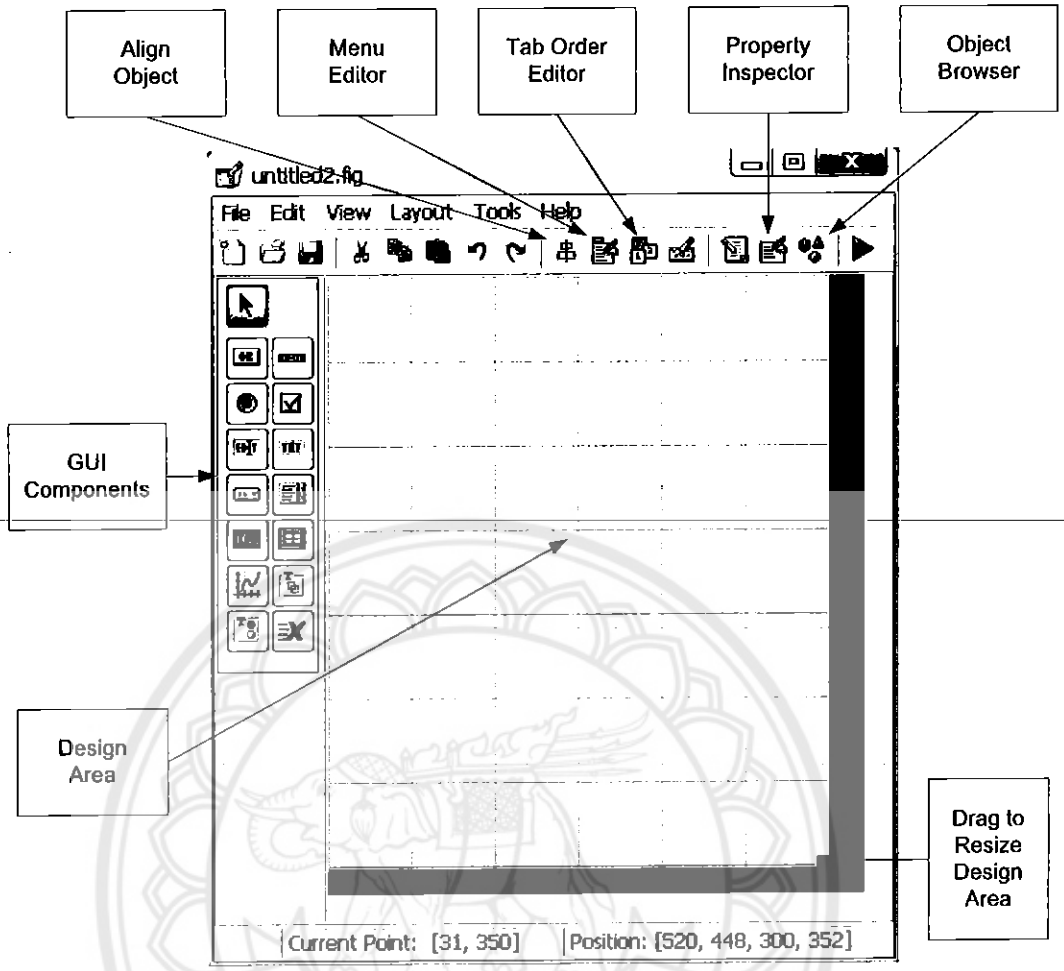
รูปที่ 2.2 แผนภาพองค์ประกอบของ graphics object

2.2.2 การสร้างส่วนต่อประสานกราฟิกกับผู้ใช้ด้วยโปรแกรม MATLAB

ในการสร้างส่วนต่อประสานกราฟิกกับผู้ใช้ด้วยโปรแกรม MATLAB สามารถเขียนได้ 2 รูปแบบ คือ ใช้เครื่องมือช่วยในการเขียนคำสั่งภายใน (Graphical User Interface Development Environment: GUIDE) และ การใช้ชุดคำสั่งหรือฟังก์ชันในการสร้าง ในหัวข้อนี้จะทำการอธิบายการสร้างส่วนต่อประสานกราฟิกกับผู้ใช้โดยใช้ GUIDE โดยทำการเปิดโปรแกรม MATLAB ขึ้นมาแล้วทำการพิมพ์คำว่า GUIDE ใน Command Window หรือคลิกไอคอนไกด์ในแถบเมนูหลักของหน้าต่างของโปรแกรม MATLAB ดังรูปที่ 2.3 และถ้ากดปุ่ม OK จะได้หน้าต่างใหม่พร้อมเครื่องมือในการสร้างส่วนต่อประสานกราฟิกกับผู้ใช้ดังรูปที่ 2.4



รูปที่ 2.3 หน้าต่างของการเรียกใช้ GUIDE ของโปรแกรม MATLAB

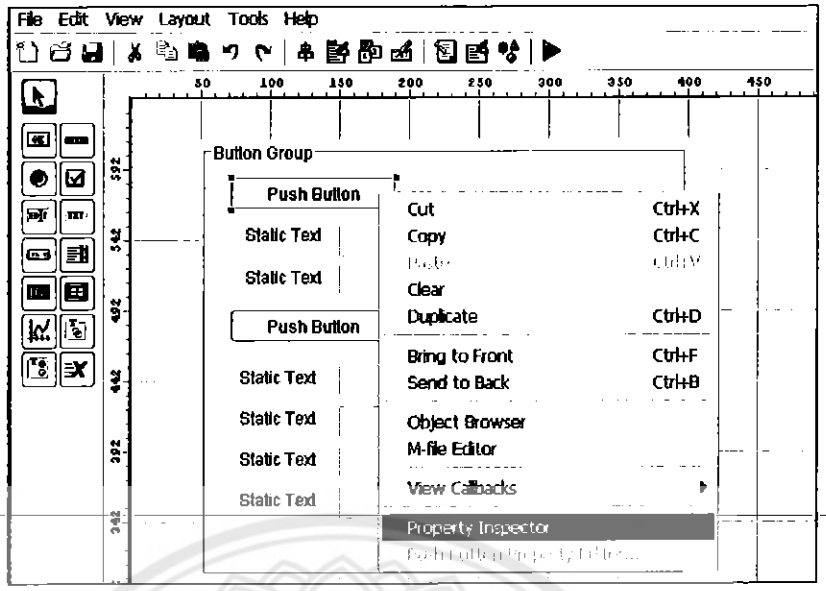


รูปที่ 2.4 หน้าต่างของเครื่องมือจากคำสั่ง GUIDE ของโปรแกรม MATLAB

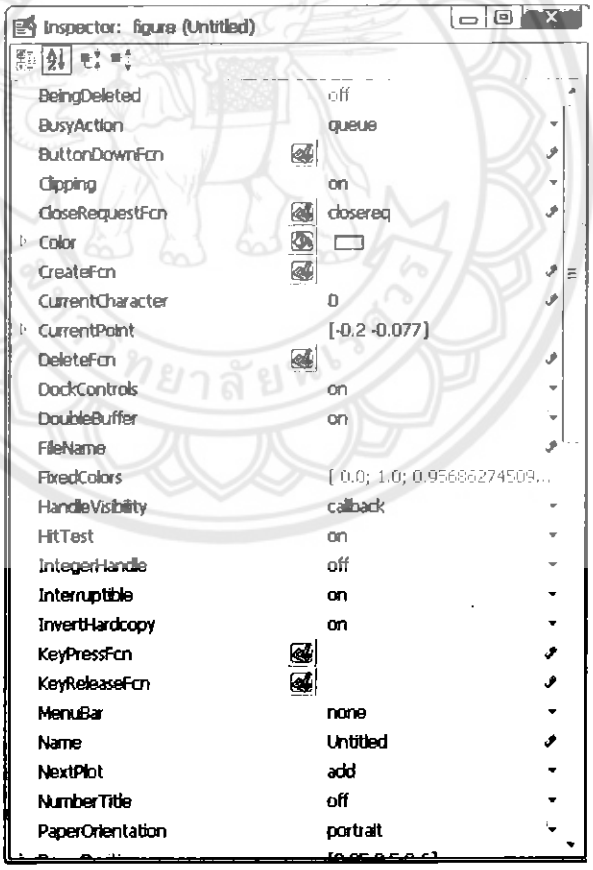
สำหรับการใช้งานวัตถุของโปรแกรมเพื่อให้ได้ตามความต้องการของผู้ออกแบบจำเป็นต้องมีการแก้ไขคุณสมบัติ และการเพิ่มคำสั่งลงในวัตถุนั้นๆ ซึ่งจะอธิบายไว้ดังนี้

1. ตัวแก้ไขคุณสมบัติ (Property Inspector)

ตัวแก้ไขคุณสมบัติ จะทำหน้าที่แก้ไขคุณสมบัติของวัตถุ เช่น การปรับเปลี่ยนขนาดปุ่ม ขนาดของตัวอักษร และการกำหนดชื่อของวัตถุ (Tag) เนื่องจากการเขียนคำสั่งการทำงานของโปรแกรมจำเป็นต้องมีการเรียกใช้ชื่อของวัตถุ ดังนั้นในการตั้งชื่อควรให้มีความสอดคล้องของการทำงานของโปรแกรมเพื่อให้ผู้เขียนโปรแกรมเข้าใจได้ง่ายขึ้น สำหรับการเรียกใช้สามารถทำได้สองวิธีคือ การคลิกปุ่ม Property Inspector ในหน้าต่างของ GUIDE หรือคลิกขวาที่วัตถุเลือก Property Inspector ดังรูปที่ 2.5 เมื่อเลือกจะได้หน้าต่างตัวแก้ไขคุณสมบัติดังรูปที่ 2.6



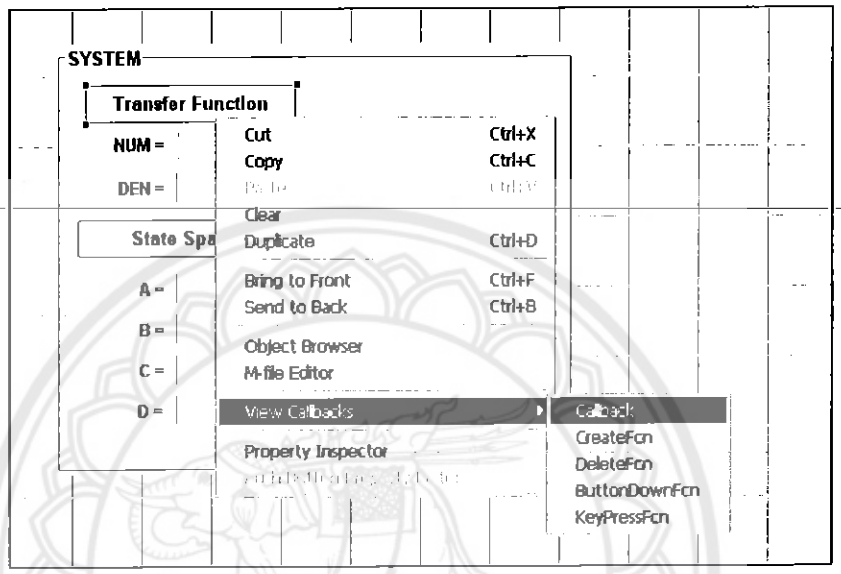
รูปที่ 2.5 ตัวอย่างการเรียกใช้ตัวแก้ไขคุณสมบัติ



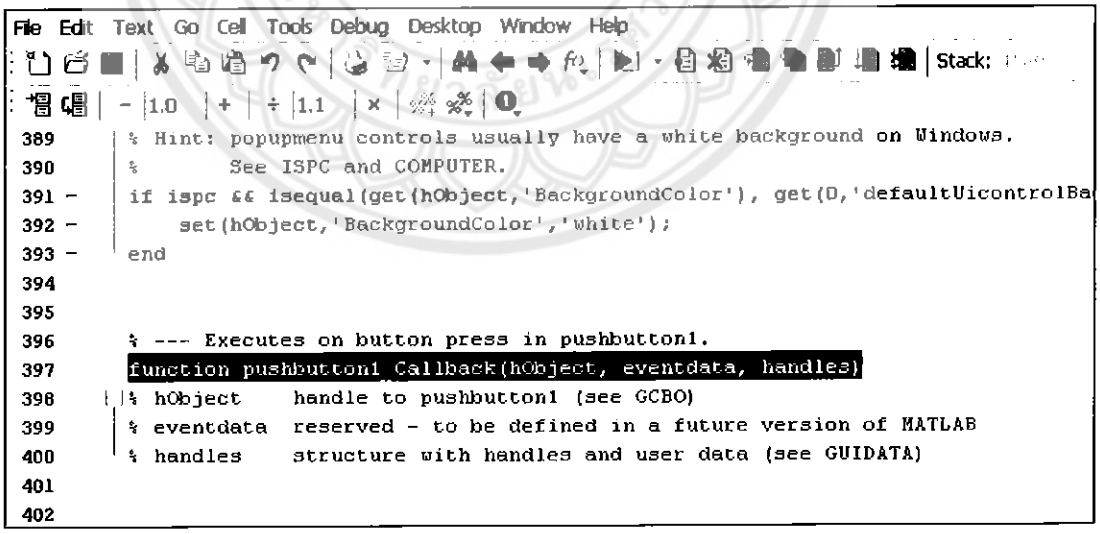
รูปที่ 2.6 ตัวอย่างตัวแก้ไขคุณสมบัติ

2. ฟังก์ชันเรียกกลับ (Callback Function)

การเขียนคำสั่งการทำงานของโปรแกรม ทำได้โดยการเพิ่มคำสั่งลงในฟังก์ชันเรียกกลับ สำหรับการเรียกใช้คำสั่งสามารถทำได้โดยการคลิกขวาที่วัตถุเลือก View Callbacks > Callback ดังรูปที่ 2.7 ซึ่งโปรแกรมจะสร้างฟังก์ชันเรียกกลับขึ้นโดยอัตโนมัติดังรูปที่ 2.8



รูปที่ 2.7 ตัวอย่างการเรียกใช้ฟังก์ชันเรียกกลับ



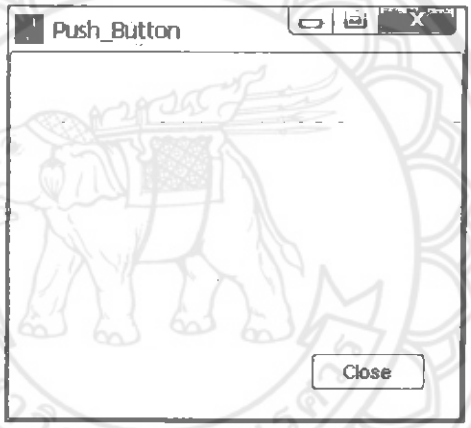
รูปที่ 2.8 ตัวอย่างฟังก์ชันเรียกกลับเมื่อมีการเรียกใช้เพื่อเขียนคำสั่งการทำงาน

ห้องสมุดคณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ต่อไปจะเป็นการอธิบายลักษณะและการทำงานของวัตถุต่างๆ เช่น ปุ่มกด (Push buttons) กล่องตรวจสอบ (Check boxes) กล่องรายการ (List boxes) กรอบ (Frames) ปุ่ม radio (Radio buttons) ตัวเลื่อน (Sliders) เป็นต้น พร้อมทั้งยกตัวอย่างประกอบโดยใช้ด้วยคำสั่ง GUIDE ของโปรแกรม MATLAB

1. ปุ่มกด (Push buttons)

การสร้างปุ่มกดนี้ เพื่อใช้สำหรับการติดต่อกับผู้ใช้โดยให้ผู้ใช้คลิกเมาส์ที่ตำแหน่งปุ่มบนหน้าต่างคำสั่งเพื่อเลือกกระทำตามที่ต้องการ ในหัวข้อนี้จะทำการยกตัวอย่างการสร้างปุ่มกด Close เพื่อสำหรับปิดหน้าต่างรูปภาพ สามารถอธิบายได้ดังนี้ เมื่อใช้เมาส์คลิกที่ปุ่ม Close หน้าต่างรูปภาพก็จะปิดทันทีดังรูปที่ 2.9

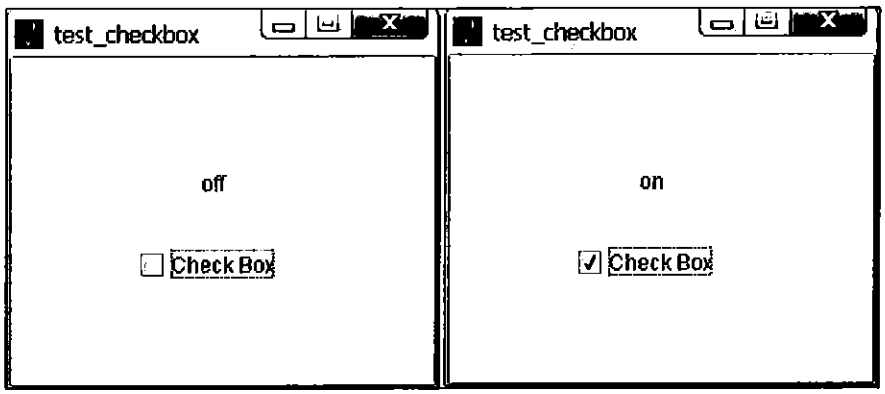


รูปที่ 2.9 ตัวอย่างลักษณะการทำงานของปุ่มกด

2. กล่องตรวจสอบ (Check boxes)

กล่องตรวจสอบ จะใช้สำหรับเลือกกระทำที่คล้ายกับสวิตช์ปิด/เปิด โดยใช้เมาส์เป็นตัวเลือกถ้าคลิกที่กล่องเพื่อเลือกตำแหน่งเปิด แต่ถ้าไม่เลือกก็จะอยู่ตำแหน่งปิด ผู้ออกแบบสามารถสร้างกล่องตรวจสอบได้มากกว่าหนึ่งกล่องก็ได้ โดยแต่ละกล่องจะขึ้นต่อกันหรือไม่ขึ้นต่อกันก็ได้ ในหัวข้อนี้จะทำการยกตัวอย่างโดยสร้างกล่องตรวจสอบ Check Box เมื่อใช้เมาส์คลิกที่กล่อง Check Box บนหน้าต่างรูปภาพ จะพบว่าค่าเปลี่ยนจาก off เป็น on ดังรูปที่ 2.10

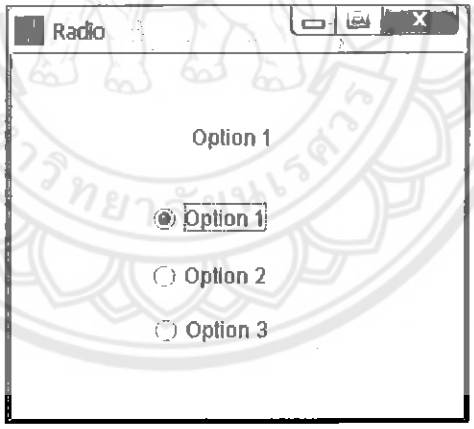
15745342
ม.
ก 534ป
2552



รูปที่ 2.10 ตัวอย่างลักษณะและการทำงานของกล่องตรวจสอบ

3. ปุ่ม Radio (Radio buttons)

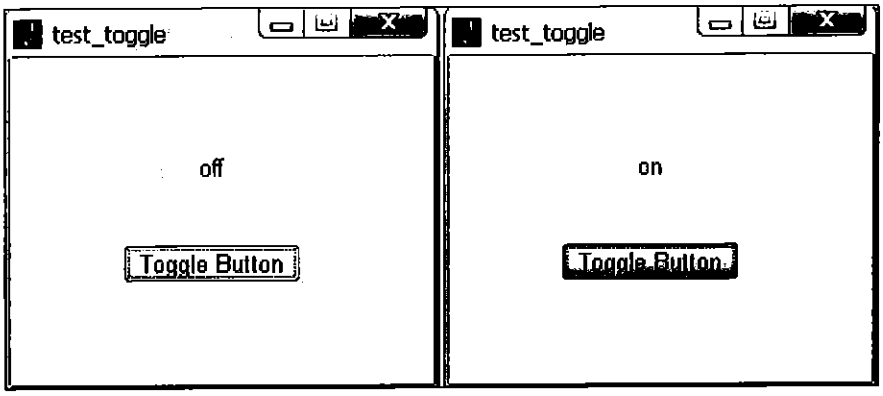
ปุ่ม Radio มีลักษณะการทำงานคล้ายกับ Check boxes แต่ข้อแตกต่างสำคัญก็คือ โดยทั่วไปแล้วการใช้ปุ่ม Radio นี้ผู้เขียนโปรแกรมจะจัดให้เป็นกลุ่มของตัวเลือก และในกลุ่มตัวเลือกจะสามารถเลือกได้เพียงครั้งละตัวเดียวในกลุ่มนั้น ในการทำงานผู้ใช้จะใช้เมาส์กดไปที่วัตถุนั้น จากนั้นสภาพของอุปกรณ์จะเปลี่ยนไปตามที่ผู้ใช้เลือกดังรูปที่ 2.11



รูปที่ 2.11 ตัวอย่างลักษณะและการทำงานของปุ่ม Radio

4. ปุ่ม Toggle (Toggle buttons)

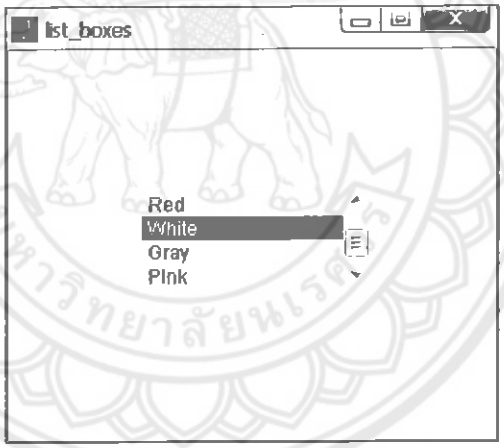
ปุ่ม Toggle เป็นปุ่มที่มีลักษณะการใช้งานสองสถานะคือ เปิดกับปิด โดยปกติมีค่าเป็น 1 (Max) ปุ่มจะอยู่ในสถานะเปิด และมีค่าเป็น 0 (Min) ปุ่มจะอยู่ในสถานะปิด เมื่อผู้ใช้เปลี่ยนค่าโดยการกดเมาส์ลงไปบริเวณของวัตถุนี้ดังรูปที่ 2.12



รูปที่ 2.12 ตัวอย่างลักษณะและการทำงานของ Toggle buttons

5. กล่องรายการ (List boxes)

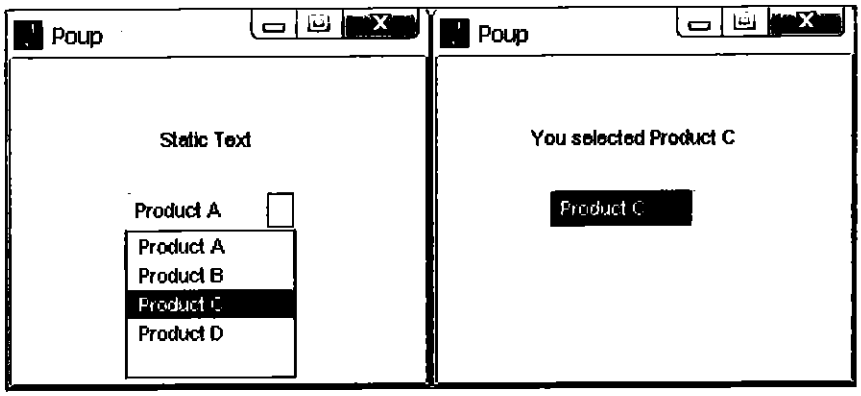
กล่องรายการ จะมีรายการต่างๆให้เลือกใช้งานสิ่งที่อยู่ในกล่องโดยการใช้เมาส์คลิกไปที่รายการที่ต้องการเลือกซึ่งโปรแกรม MATLAB ก็จะปฏิบัติตามสิ่งที่ได้เลือกไว้ดังรูปที่ 2.13



รูปที่ 2.13 ตัวอย่างลักษณะและการทำงานกล่องรายการ

6. Popup menus

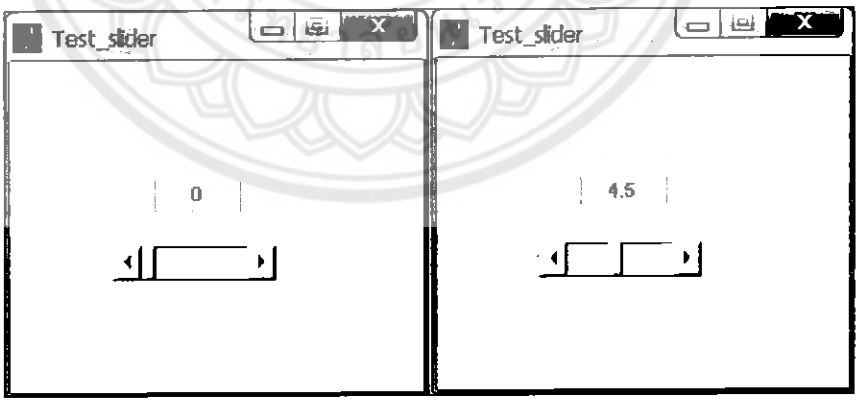
Popup menus เป็นเมนูรายการที่อยู่บนหน้าต่างรูปภาพที่ไม่ใช่เมนูบาร์ ซึ่งมีลักษณะเป็นสี่เหลี่ยมมุมฉากที่แสดงชื่อเมนูปัจจุบัน และมีลูกศรในสี่เหลี่ยมเล็กด้ายขวามือเพื่อให้ผู้ใช้สามารถใช้เมาส์เลื่อนเพื่อเลือกเมนูต่างๆ ในหัวข้อนี้แสดงตัวอย่างผู้ใช้สามารถเลือกรายการ Product จาก Popup menus ได้โดยการใช้เมาส์คลิกที่ลูกศรในสี่เหลี่ยมเล็กด้านขวามือ ก็จะปรากฏรายการ Product ต่างๆ ในตัวอย่างนี้ทำการเลือกรายการ Product C ดังรูปที่ 2.14



รูปที่ 2.14 ตัวอย่างลักษณะและการทำงานของ Popup menus

7. ตัวเลื่อน (Sliders)

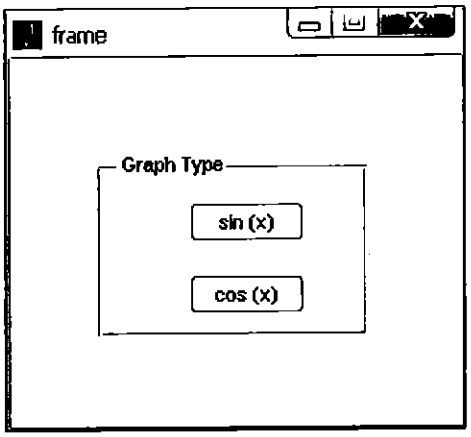
ตัวเลื่อน เป็นการป้อนค่าตัวเลขโดยอาศัยแถบเลื่อน ค่าที่ป้อนจะถูกกำหนดโดยผู้เขียนโปรแกรมและผู้ใช้จะทำการกำหนดค่าโดยการใช้เมาส์กดแล้วเลื่อนแถบบนตัวเลื่อน หรือกดบริเวณลูกศรเพื่อให้แถบค่อยๆเลื่อนไปเป็นลำดับ ตำแหน่งของตัวเลื่อนบนแท่งจะเป็นค่าตัวเลขตามสัดส่วนของระยะบนแท่งเลื่อนนี้ และค่าจะได้รับการคำนวณหลังจากที่เราปล่อยปุ่มเมาส์ ผู้เขียนโปรแกรมสามารถที่จะตั้งค่าสูงสุด ต่ำสุด และค่าปัจจุบันของตัวเลื่อนได้ ในหัวข้อนี้จะยกตัวอย่างให้ค่าต่ำสุด และสูงสุดของตัวเลื่อนเท่ากับ 0 และ 10 ตามลำดับ เมื่อทำการเลื่อนตัวเลื่อนก็จะมีค่าแสดงขึ้นดังรูปที่ 2.15



รูปที่ 2.15 ตัวอย่างลักษณะและการทำงานของตัวเลื่อน

8. กรอบ (Frames)

กรอบ เป็นรูปพื้นที่สี่เหลี่ยม ซึ่งจะแบ่งหน้าต่างรูปภาพออกเป็นส่วนๆ ส่วนใหญ่จะใช้จัดรูปแบบควบคุม หรือข้อความอธิบายต่างๆ ให้อยู่ในกรอบเดียวกัน เพื่อความสะดวกในการใช้งานและเข้าใจง่ายดังรูปที่ 2.16



รูปที่ 2.16 แสดงลักษณะกรอบ

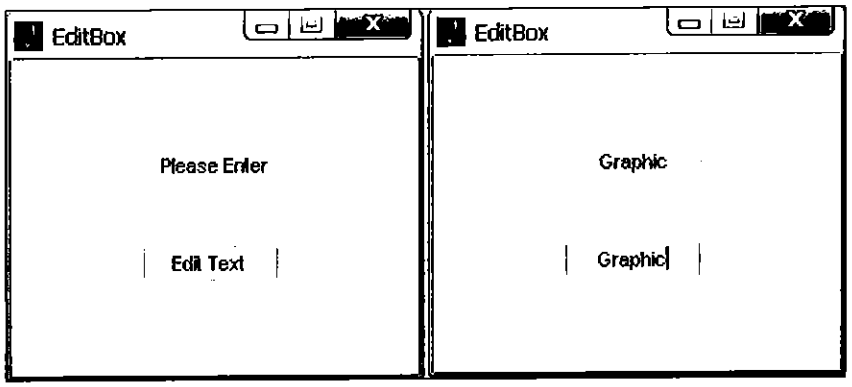
9. ตัวอักษรที่อยู่กับที่ (Static text)

ตัวอักษรที่อยู่กับที่ เป็นตัวหนังสือที่ผู้ใช้ไม่สามารถที่จะแก้ไขได้ แต่ผู้เขียนโปรแกรมอาจแก้ไขได้ โดยทั่วไปจึงใช้ในการเขียนตัวหนังสือเพื่อบอกถึงชื่อส่วนต่างต่างๆ หรือบอกสภาวะการทำงานของโปรแกรมดังรูปที่ 2.17

รูปที่ 2.17 แสดงลักษณะตัวอักษรที่อยู่กับที่

10. กล่องแก้ไขตัวอักษร (Editable text boxes)

กล่องแก้ไขตัวอักษร เป็นกล่องที่ผู้ใช้สามารถแก้ไขตัวอักษรที่อยู่ในกล่องนั้นได้ แสดงตัวอย่างดังนี้ เมื่อพิมพ์คำว่า Graphic ในช่อง Edit text แล้วกด Enter ก็จะแสดงคำว่า Graphic ดังรูปที่ 2.18



รูปที่ 2.18 แสดงลักษณะและการทำงานของกล่องแก้ไขตัวอักษร

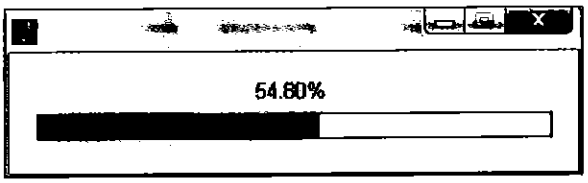
2.2.3 หลักการในการออกแบบส่วนต่อประสาน

การออกแบบส่วนต่อประสานกับผู้ใช้ระหว่างผู้ใช้กับคอมพิวเตอร์ เพื่อการออกแบบพัฒนาส่วนต่อประสานให้ใช้งานได้อย่างมีประสิทธิภาพ โดยมีวัตถุประสงค์หลักคือ สามารถใช้งานได้ง่าย และเพิ่มมาตรฐานการออกแบบที่อยู่ในระบบ การออกแบบส่วนต่อประสานที่ดีจะทำให้งานที่สำเร็จออกมาดีใช้งานได้ง่าย เรียนรู้ได้ง่าย ซึ่งมีหลักการออกแบบดังนี้

1) นำเสนอส่วนที่เป็นข้อความ และส่วนที่เป็นสัญลักษณ์กราฟิกให้ชัดเจน สื่อความหมายว่าต้องการให้ผู้ใช้ทำอะไร
2) ออกแบบระบบที่เสนอโครงสร้างและขั้นตอนของการดำเนินการอย่างชัดเจน
3) ในกรณีที่ผู้ใช้คำสั่งดำเนินการไม่ตรงตามขั้นตอนการทำงานควรให้มีการเตือนหรือแสดงสัญลักษณ์ที่ผู้ใช้ทราบได้ในทันทีว่าควรจะแก้ไขอย่างไร

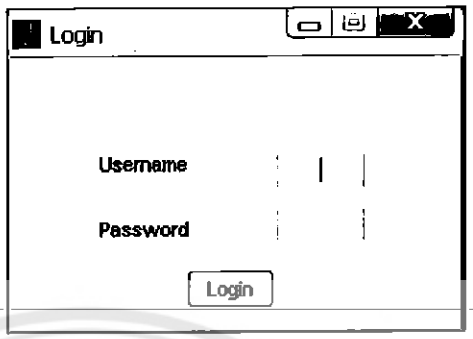
2.2.4 การตอบสนองของระบบ

1) แจ้งสถานะการทำงาน (Status information) แจ้งให้ผู้ใช้ทราบถึงสถานะปัจจุบันในการทำงานของระบบเช่น แสดงหน้าที่กำลังทำงานอยู่ แสดงความคืบหน้าในการประมวลผล ตัวอย่างแสดงในรูปที่ 2.19



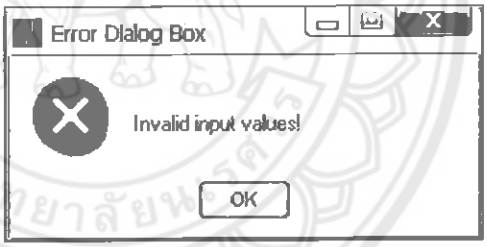
รูปที่ 2.19 ตัวอย่างการแจ้งสถานะการทำงานของโปรแกรม

2) แสดงความพร้อมในการรับคำสั่ง (Prompting cues) เป็นการออกแบบเพื่อแจ้งสถานะความพร้อมในการรับคำสั่ง เช่น หน้า ล็อกอินเข้าระบบ เคอเซอร์กระพริบรอรับชื่อผู้ใช้ดังรูปที่ 2.20



รูปที่ 2.20 ตัวอย่างความพร้อมในการทำงานของโปรแกรม

3) ข้อความแจ้งหรือเตือนเมื่อมีข้อผิดพลาด (Error/Warning messages) การแสดงข้อความเมื่อเกิดข้อผิดพลาด และแนะแนวทางในการแก้ไข เพื่ออธิบายให้ผู้ใช้สามารถแก้ไขปัญหาด้วยตนเองได้ดังรูปที่ 2.21



รูปที่ 2.21 ตัวอย่างข้อความเตือนเมื่อมีข้อผิดพลาดของโปรแกรม

2.2.5 การทดสอบเพื่อประเมินผลการออกแบบ

การทดสอบงานออกแบบ เพื่อค้นหามาตรฐานของผู้ใช้ การทดสอบเป็นสิ่งที่สำคัญมาก เพื่อให้ผู้ออกแบบได้ทราบผลของงานออกแบบที่เกิดขึ้นว่า งานออกแบบมีความเป็นไปได้และมีประสิทธิภาพมากน้อยแค่ไหน ดังนั้นการทดสอบควรทดสอบกับกลุ่มตัวอย่าง ที่เป็นกลุ่มเป้าหมายในช่วงระดับเดียวกัน เช่น กลุ่มที่มีประสบการณ์ใกล้เคียงกัน กลุ่มที่มีทักษะในการใช้งานใกล้เคียงกัน ก่อนที่จะนำเอาไปใช้จริง และนำผลย้อนกลับ จากกลุ่มตัวอย่างมาปรับปรุงรูปแบบการออกแบบต่อไป การออกแบบอย่างมีประสิทธิภาพควรคำนึงถึงดังนี้

1) ผู้ใช้มองหน้าจอก็สามารถคาดเดารูปลักษณ์หรือองค์ประกอบต่างๆของหน้าจอได้ว่า สิ่งต่างๆเหล่านั้นไว้ใช้ทำอะไร และผู้ใช้สามารถทำงานได้โดยง่าย

2) ผู้ใช้จะต้องสามารถเข้าใจการทำงานลำดับขั้นตอนได้โดยง่าย โดยไม่ต้องมาเรียนรู้ใหม่ทุกครั้งเมื่อมีการใช้งาน
3) สร้างข้อความเตือน เช่น เมื่อเกิดมีการสั่งงานที่ผิดพลาด การแจ้งเตือนจะต้องให้ถูกระยะเวลาหรือช่วงเวลาที่เหมาะสม
4) เตรียมวิธีการป้องกันความผิดพลาดของผู้ใช้ เมื่อเกิดมีความผิดพลาดในการใช้ เช่น เมื่อผู้ใช้หลงทาง หรือกดข้อมูลผิด ผู้ออกแบบจะต้องมีทางเลือก เช่น การย้อนกลับ หรือ ออก ให้ผู้ใช้เสมอ เช่น โปรแกรม Web browser จะมีปุ่มย้อนกลับ หรือ ปุ่มไปข้างหน้า เป็นต้น
5) พยายามสร้างตัวช่วย ในการบอกวิธีการใช้ เช่น ถ้าออกแบบหน้าจอโปรแกรมจะต้องมีตัวช่วย (Help) เป็นต้น

บทที่ 3
ขั้นตอนการดำเนินโครงงาน

จากบทที่ 2 เป็นการอธิบายถึงหลักการและทฤษฎีที่เกี่ยวข้องของวิธีการควบคุมแบบเรียนรู้ซ้ำ ในบทนี้จะนำเสนอเกี่ยวกับขั้นตอนของการควบคุมแบบเรียนรู้ซ้ำในแต่ละแบบ และรวมถึงการออกแบบส่วนต่อประสานกราฟิกกับผู้ใช้ ซึ่งมีรายละเอียดดังต่อไปนี้

3.1 ขั้นตอนการควบคุมแบบเรียนรู้ซ้ำ

ในหัวข้อนี้จะนำเสนอถึงขั้นตอนการควบคุมแบบเรียนรู้ซ้ำ ซึ่งได้นำทฤษฎีที่กล่าวไว้ในบทที่ 2 นำมาใช้ในกระบวนการควบคุม มีรายละเอียดดังนี้

3.1.1 ขั้นตอนการควบคุมแบบเรียนรู้ซ้ำโดยค่าเกนอร์มที่เหมาะสมที่สุด

สำหรับขั้นตอนเริ่มต้นของการควบคุมแบบเรียนรู้ซ้ำโดยค่าเกนอร์มที่เหมาะสมที่สุดจะพิจารณาจากสมการเงื่อนไขที่จำเป็นสำหรับผลเฉลยที่เหมาะสมที่สุด ซึ่งมีสมการที่สำคัญที่ใช้ในการคำนวณ ดังนี้

$$u_{k+1} = u_k + G^T e_{k+1} \tag{3.1}$$

จากสมการที่ (3.1) จำเป็นต้องทราบค่าความผิดพลาด (e_{k+1}) จึงจะสามารถคำนวณหาสัญญาณขาเข้าควบคุมได้ ดังนั้นสมการที่ใช้หาค่าความผิดพลาด แสดงไว้ดังนี้

$$e_{k+1} = (I + GG^T)^{-1} e_k \tag{3.2}$$

3.1.2 ขั้นตอนการควบคุมแบบเรียนรู้ซ้ำโดยใช้ตัวแปรที่เหมาะสมที่สุด

สำหรับขั้นตอนเริ่มต้นของการควบคุมโดยใช้ตัวแปรที่เหมาะสมที่สุดจะเริ่มพิจารณาจากสมการควบคุมที่ใช้หาค่าสัญญาณขาเข้าควบคุม ซึ่งมีสมการที่สำคัญที่ใช้ในการคำนวณ ดังนี้

$$u_{k+1}(t) = u_k(t) + \gamma_{k+1} e_k(t+1) \tag{3.3}$$

ผลเฉลยที่เป็นค่า γ_{k+1} ที่เหมาะสมที่สุด คือ

$$\gamma_{k+1} = \frac{e_k^T G e_k}{w + e_k^T G^T G e_k} \quad (3.4)$$

เมื่อ w คือ ค่าตัวแปรถ่วงน้ำหนัก

e_k คือ ค่าความผิดพลาดที่เกิดจากการทำงานรอบที่ k

3.1.3 ขั้นตอนการควบคุมแบบเรียนรู้ซ้ำโดยใช้พหุนามของเมทริกซ์ G

สำหรับการควบคุมแบบเรียนรู้ซ้ำโดยใช้พหุนามของเมทริกซ์ G ซึ่งมีการอธิบายถึงวิธีการควบคุมโดยละเอียดในบทที่ผ่านมา ในหัวข้อนี้จะแสดงสมการที่สำคัญที่ใช้ในการคำนวณดังนี้

$$u_{k+1}(t) = u_k(t) + \sum_{j=1}^{M} \beta_{k+1}(j) G^{j-1} e_k(t+1);\ 1 \le j \le M \le k \quad (3.5)$$

เมื่อ G คือ เมทริกซ์จัตุรัสที่แสดงข้อมูลของระบบที่กำหนด

M คือ จำนวนพจน์ของพหุนาม และเวกเตอร์ของตัวแปร

$\beta_{k+1}(j)$ คือ ตัวแปรอัตราขยาย

และค่า β_{k+1} ที่เหมาะสมที่สุดคือ

$$\beta_{k+1} = [W + N(e_k)^T N(e_k)]^{-1} N(e_k)^T e_k \quad (3.6)$$

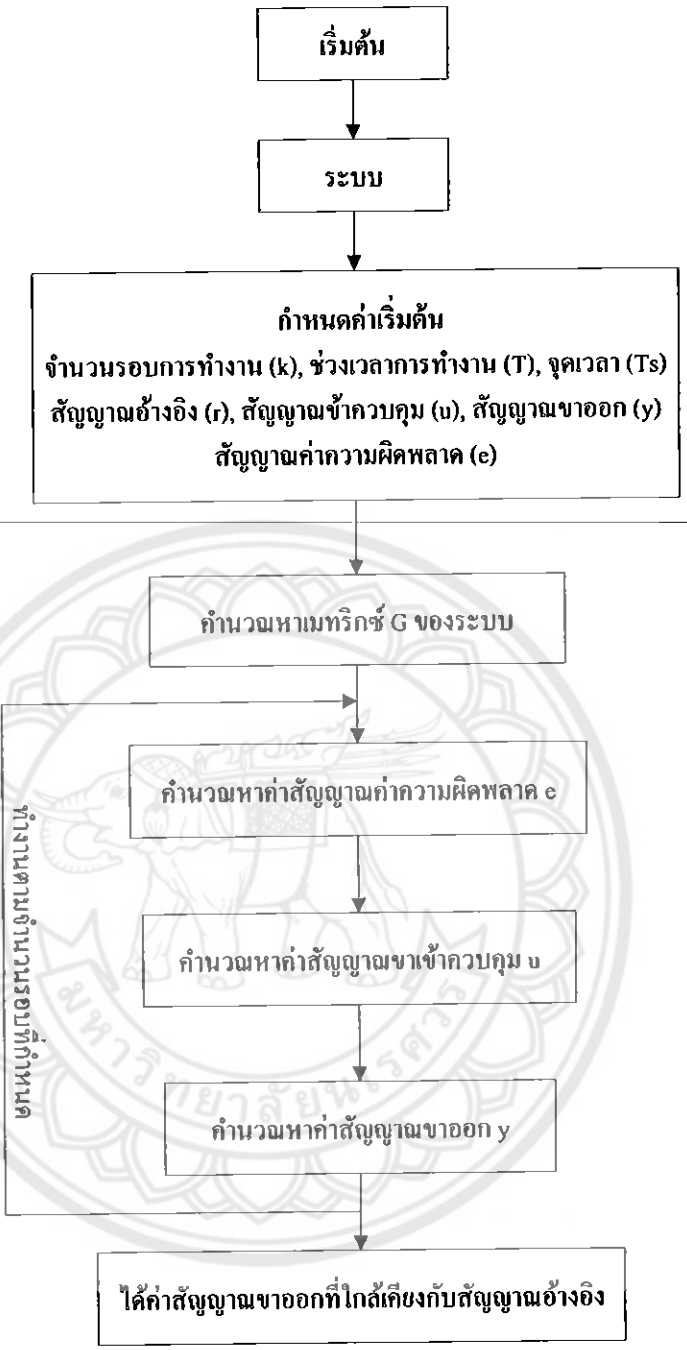
เมื่อ W คือ ค่าตัวแปรถ่วงน้ำหนัก

$N(e_k) = [Ge_k \quad G^2 e_k \quad \cdots \quad G^M e_k]$

และสมการค่าความผิดพลาดที่อยู่ในรูปเมทริกซ์ คือ

$$e_{k+1} = e_k - N(e_k)\beta_{k+1} \quad (3.7)$$

สำหรับขั้นตอนวิธีการควบคุมแบบเรียนรู้ซ้ำในแต่ละวิธีที่ได้กล่าวไว้ข้างต้น ซึ่งได้แก่ การควบคุมแบบเรียนรู้ซ้ำโดยใช้ค่านอร์มที่เหมาะสมที่สุด การควบคุมแบบเรียนรู้ซ้ำโดยใช้ตัวแปรที่เหมาะสมที่สุด และการควบคุมแบบเรียนรู้ซ้ำโดยใช้พหุนามของเมทริกซ์ G จะแสดงดังรูปที่ 3.1, 3.2 และ 3.3 ตามลำดับ มีดังต่อไปนี้



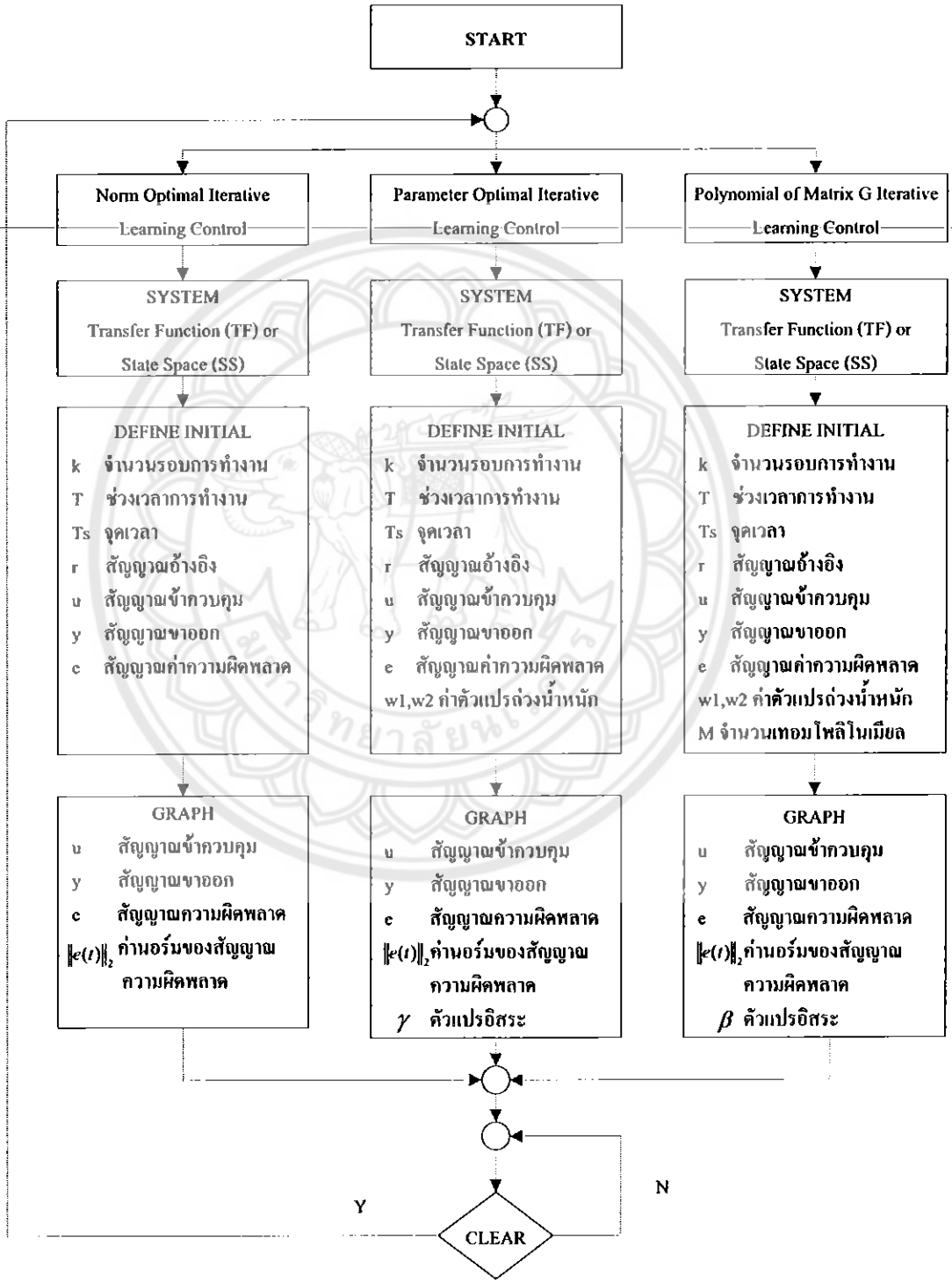
รูปที่ 3.1 แผนภาพการทำงานของวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้ค่านอร์มที่เหมาะสมที่สุด

รูปที่ 3.2 แผนภาพการทำงานของวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้ตัวแปรที่เหมาะสมที่สุด

รูปที่ 3.3 แผนภาพการทำงานของวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้โดยใช้พหุนามของเมทริกซ์ G

3.2 การทำงานของโปรแกรม

สำหรับแผนผังการทำงานของโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้โดยใช้โปรแกรม MATLAB แสดงไว้ดังรูปที่ 3.4



รูปที่ 3.4 แผนภาพขั้นตอนการทำงานของโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้

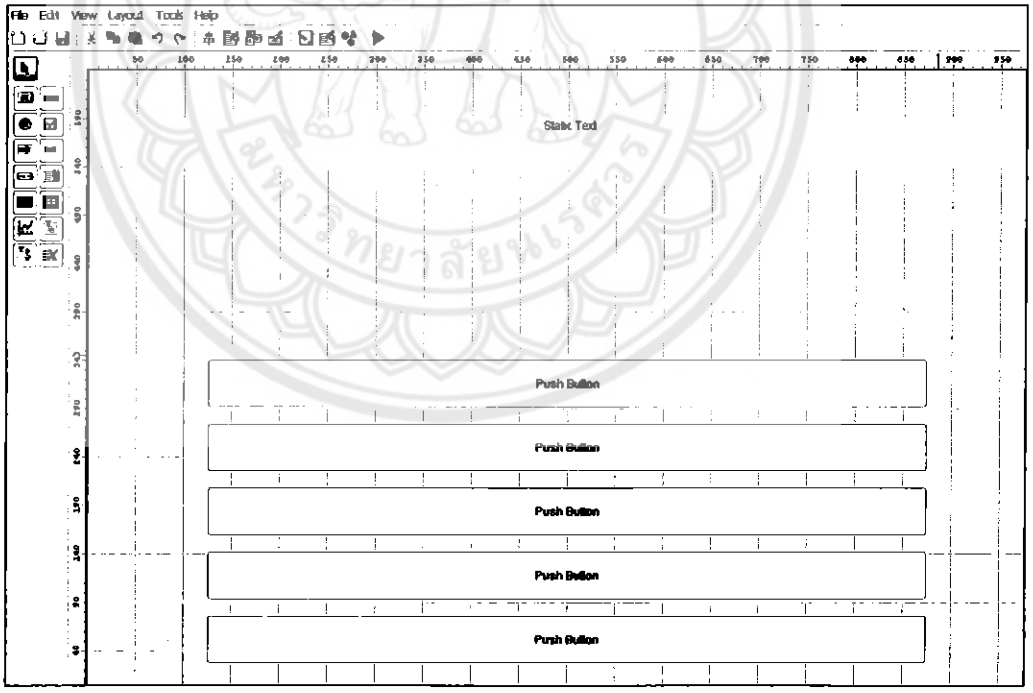
3.3 การออกแบบและการเขียนโปรแกรม

การสร้างส่วนต่อประสานกราฟิกกับผู้ใช้ในโครงงานนี้ ได้แสดงหน้าต่างของโปรแกรมต่างๆ ประกอบด้วย หน้าต่างหลักของโปรแกรม และหน้าต่างแสดงผลของโปรแกรม ซึ่งจะทำการรับค่าจากการป้อนจากผู้ใช้และการแสดงผลลัพธ์ ซึ่งในโครงงานนี้จะทำการเขียนโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้ด้วย GUIDE ของโปรแกรม MATLAB ซึ่งมีรายละเอียดดังนี้

3.3.1 การสร้างหน้าต่างหลักของโปรแกรม

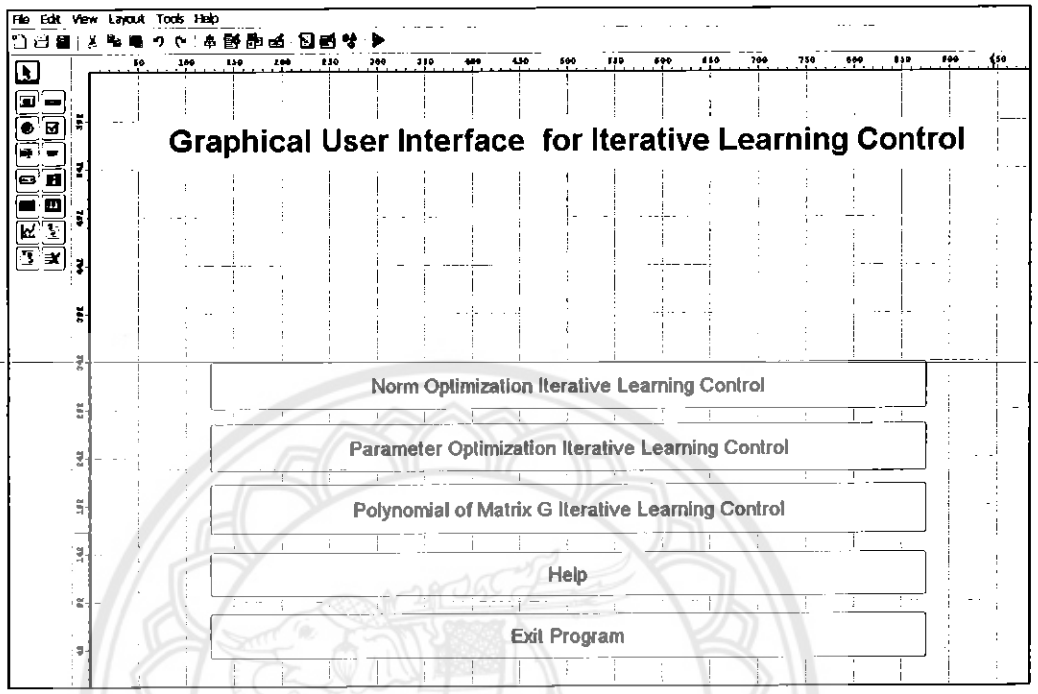
สำหรับหน้าต่างหลักของโปรแกรมจะเป็นการเลือกขั้นตอนของการควบคุมแบบเรียนรู้ซ้ำในแต่ละวิธี ซึ่งมีขั้นตอนดังต่อไปนี้

1. การวางรูปแบบของวัตถุ ในส่วนนี้จะเป็นการออกแบบและวางวัตถุแต่ละชนิดในตำแหน่งต่างๆ ซึ่งประกอบด้วย ตัวอักษรอยู่กับที่ (Static text) และปุ่มกด (Push button) ดังแสดงในรูปที่ 3.5



รูปที่ 3.5 ตัวอย่างการวางวัตถุบนหน้าต่างหลักของโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้

2. การแก้ไขคุณสมบัติของวัตถุ ในส่วนนี้จะเป็นการแก้ไขคุณสมบัติของวัตถุในหน้าต่างโปรแกรม เช่น การแก้ไขชื่อของปุ่ม ขนาดของปุ่ม ขนาดตัวอักษร เป็นต้น จะได้ดังรูปที่ 3.6



รูปที่ 3.6 ตัวอย่างการแก้ไขคุณสมบัติหน้าต่างหลักของโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้

3. การแก้ไขคำสั่งการทำงานของวัตถุ ในส่วนนี้เป็นการแก้ไขคำสั่ง**การทำงานของวัตถุ**โดยการแก้ไขโดยใช้ฟังก์ชันเรียกกลับ จะเป็นการเขียนคำสั่งเพิ่มลงไปเพื่อให้โปรแกรมทำงานตามที่ต้องการ จากรูปที่ 3.6 เมื่อทำการแก้ไขคำสั่งเรียบร้อยแล้ว วัตถุแต่ละชนิดจะมีหน้าที่ดังต่อไปนี้

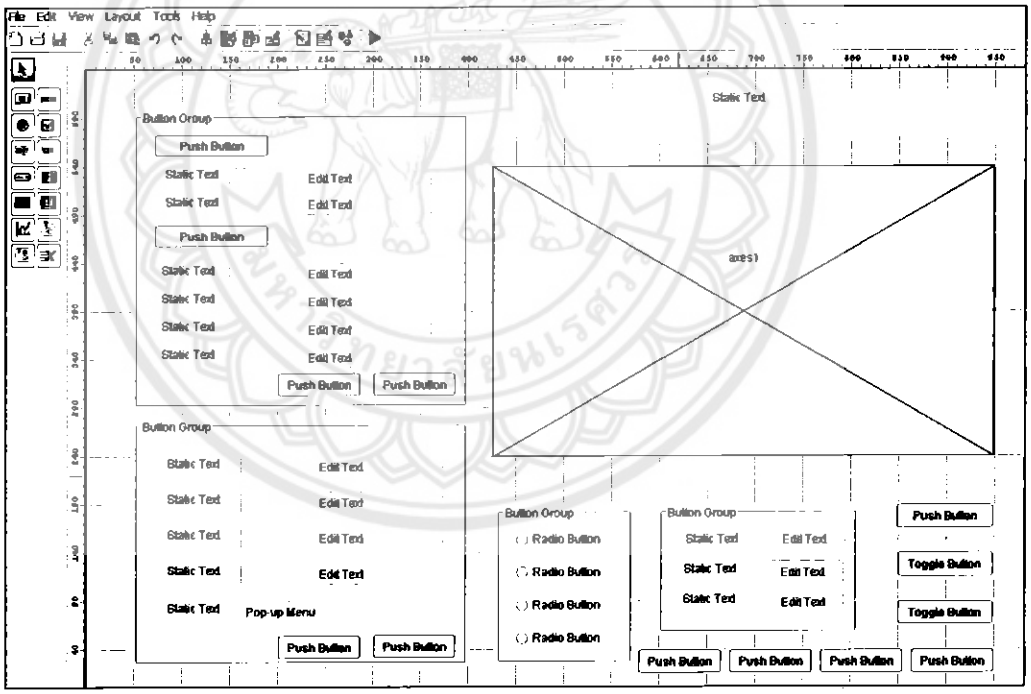
- ปุ่มการควบคุมแบบเรียนรู้ซ้ำโดยใช้ค่านอร์มที่เหมาะสมที่สุด (Norm Optimization Iterative Learning Control) ทำหน้าที่เปิดหน้าต่างของการคำนวณและแสดงผลลัพธ์ของการควบคุมแบบเรียนรู้ซ้ำโดยใช้ค่านอร์มที่เหมาะสมที่สุด สร้างโดยใช้ปุ่มกด
- ปุ่มการควบคุมแบบเรียนรู้ซ้ำโดยใช้ตัวแปรที่เหมาะสมที่สุด (Parameter Optimization Iterative Learning Control) ทำหน้าที่เปิดหน้าต่างของการคำนวณและแสดงผลลัพธ์ของการควบคุมแบบเรียนรู้ซ้ำโดยใช้ตัวแปรที่เหมาะสมที่สุด สร้างโดยใช้ปุ่มกด
- ปุ่มการควบคุมแบบเรียนรู้ซ้ำโดยใช้พหุนามของเมทริกซ์ G (Polynomial of Matrix G Iterative Learning Control) ทำหน้าที่เปิดหน้าต่างของการคำนวณและแสดงผลลัพธ์ของการควบคุมแบบเรียนรู้ซ้ำโดยใช้พหุนามของเมทริกซ์ G สร้างโดยใช้ปุ่มกด

- ปุ่มช่วยเหลือ (Help) ทำหน้าที่แสดงวิธีการใช้งานโปรแกรมในแต่ละขั้นตอน สร้างโดยปุ่มกด
- ปุ่มออกโปรแกรม (Exit Program) ทำหน้าที่ออกจากโปรแกรม สร้างโดยปุ่มกด

3.3.2 การสร้างหน้าต่างการคำนวณและแสดงผลของ NOILC

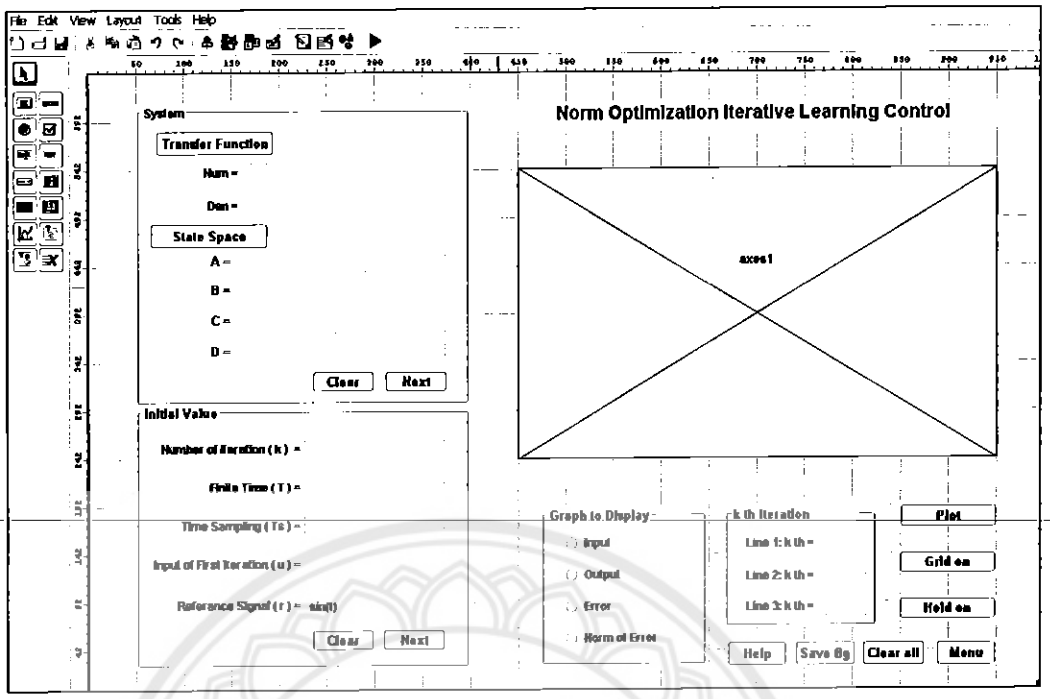
สำหรับหน้าต่างนี้ได้แบ่งเป็นสองส่วนคือ ส่วนที่ผู้ใช้กำหนดตัวแปรต่างๆที่จำเป็นสำหรับการคำนวณ และส่วนแสดงผลลัพธ์ในรูปแบบของกราฟ

1. การวางรูปแบบของวัตถุ ในส่วนนี้จะเป็นการออกแบบและวางวัตถุแต่ละชนิดในตำแหน่งต่างๆ ซึ่งประกอบด้วยตัวอักษรอยู่กับที่ (Static text), กล่องแก้ไขตัวอักษร (Editable text boxes), ปุ่มกด (Push button), ปุ่ม Radio (Radio buttons), ปุ่ม Toggle (Toggle buttons), Popup menu, Axe, และกรอบ (Frame) ดังแสดงในรูปที่ 3.7



รูปที่ 3.7 ตัวอย่างการวางรูปแบบของวัตถุบนหน้าต่างของ Norm Optimization ILC

2. การแก้ไขคุณสมบัติของวัตถุ ในส่วนนี้จะเป็นการแก้ไขคุณสมบัติของวัตถุในหน้าต่างโปรแกรม เช่น การแก้ไขชื่อของปุ่ม ขนาดของปุ่ม ขนาดตัวอักษร เป็นต้น จะได้ดังรูปที่ 3.8



รูปที่ 3.8 ตัวอย่างการแก้ไขคุณสมบัติวัตถุบนหน้าต่างของ Norm Optimization ILC

3. การแก้ไขคำสั่งการทำงานของวัตถุ ในส่วนนี้เป็นการแก้ไขคำสั่งการทำงานของวัตถุโดยการแก้ไขโดยใช้ฟังก์ชันเรียกกลับ จะเป็นการเขียนคำสั่งเพิ่มลงไปเพื่อให้โปรแกรมทำงานตามที่ต้องการ จากรูปที่ 3.8 เมื่อทำการแก้ไขคำสั่งเรียบร้อยแล้ว วัตถุแต่ละชนิดจะมีหน้าที่ดังต่อไปนี้

- ช่องรับค่าจากผู้ใช้ ทำหน้าที่รับค่าเพื่อใช้ในการคำนวณต่อไป สร้างโดยกล่องแก้ไขข้อความ
- Popup menu ทำหน้าที่ให้ผู้ใช้โปรแกรมเลือกสัญญาณอ้างอิงที่ต้องการให้กับระบบ สร้างโดย Popup menu
- ปุ่มฟังก์ชันถ่ายโอน (Transfer Function) ทำหน้าที่ให้ผู้ใช้สามารถป้อนระบบที่เป็นฟังก์ชันถ่ายโอนได้ สร้างโดยปุ่มกด
- ปุ่มปริภูมิสถานะ (State Space) ทำหน้าที่ให้ผู้ใช้สามารถป้อนระบบที่เป็นปริภูมิสถานะได้ สร้างโดยปุ่มกด
- ปุ่มเคลียร์ (Clear) ทำหน้าที่ลบข้อมูลที่ได้กรอกไว้ สร้างโดยปุ่มกด
- ปุ่มถัดไป (Next) ทำหน้าที่ตรวจสอบความถูกต้องของค่าที่รับมาจากการป้อนโดยผู้ใช้ และเริ่มต้นการกำหนดค่าในส่วนถัดไป สร้างโดยปุ่มกด
- ปุ่มสัญญาณขาเข้า (Input) ทำหน้าที่แสดงผลลัพธ์ของสัญญาณขาเข้า สร้างโดยปุ่ม Radio

- ปุ่มสัญญาณขาออก (Output) ทำหน้าที่แสดงผลลัพธ์ของสัญญาณขาออก สร้างโดยปุ่ม Radio

- ปุ่มสัญญาณค่าความผิดพลาด (Error) ทำหน้าที่แสดงผลลัพธ์ของสัญญาณค่าความผิดพลาด สร้างโดยปุ่ม Radio

- ปุ่มนอร์มของสัญญาณค่าความผิดพลาด (Norm of Error) ทำหน้าที่แสดงผลลัพธ์ค่านอร์มของสัญญาณค่าความผิดพลาด สร้างโดยปุ่ม Radio

- กราฟแสดงผล ทำหน้าที่แสดงกราฟที่ผู้ใช้ต้องการ สร้างโดย Axes

- ปุ่มวาด (Plot) ทำหน้าที่คำนวณและวาดกราฟตามที่ผู้ใช้เลือกไว้ สร้างโดยปุ่มกด

- ปุ่มกริดออน (Grid On) ทำหน้าที่แสดงเส้นกริดและปิดเส้นกริด สร้างโดยปุ่ม Toggle

- ปุ่มโฮลด์ออน (Hold On) ทำหน้าที่ให้ผู้ใช้โปรแกรมสามารถเปรียบเทียบเส้นกราฟได้ สร้างโดยปุ่ม Toggle

- ปุ่มบันทึก (Save fig) ทำหน้าที่บันทึกรูปของกราฟที่ได้จากการคำนวณ สร้างโดยปุ่มกด

- ปุ่มลบทั้งหมด (Clear all) ทำหน้าที่ลบข้อมูลที่ได้ป้อนค้างไว้ และรวมถึงข้อมูลที่ได้จากการคำนวณ สร้างโดยปุ่มกด

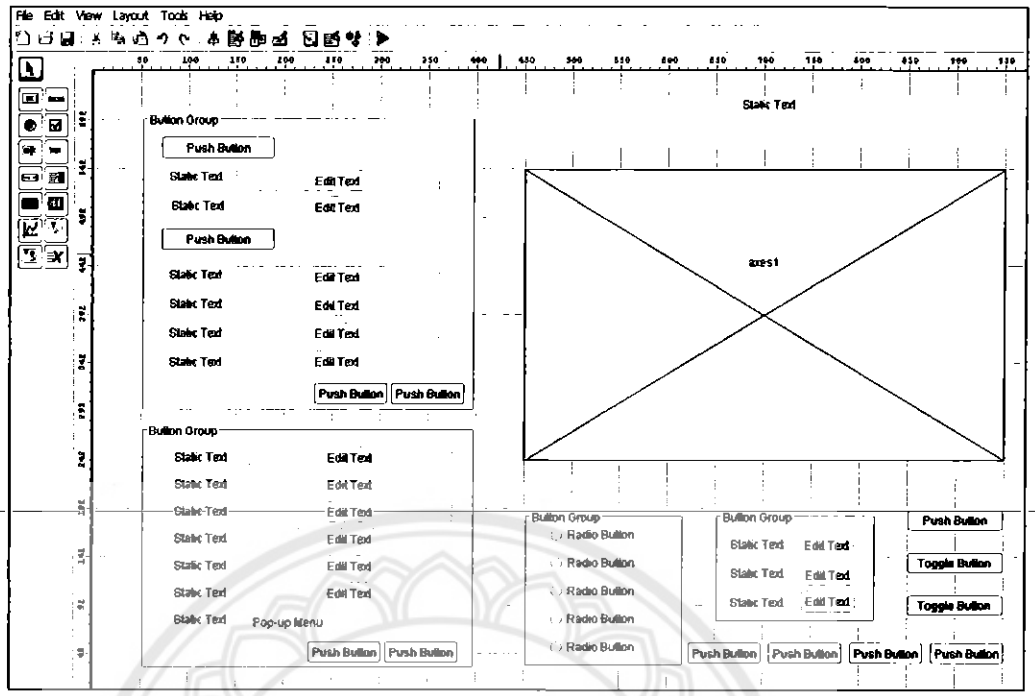
- ปุ่มช่วยเหลือ (Help) ทำหน้าที่แสดงตัวอย่างวิธีการป้อนค่าเริ่มต้นต่างๆ และอธิบายการแสดงผลลัพธ์ สร้างโดยปุ่มกด

- ปุ่มเมนู (Menu) ทำหน้าที่กลับสู่หน้าต่างหลักเพื่อเลือกวิธีการควบคุม สร้างโดยปุ่มกด

3.3.3 การสร้างหน้าต่างการคำนวณและแสดงผลของ POILC

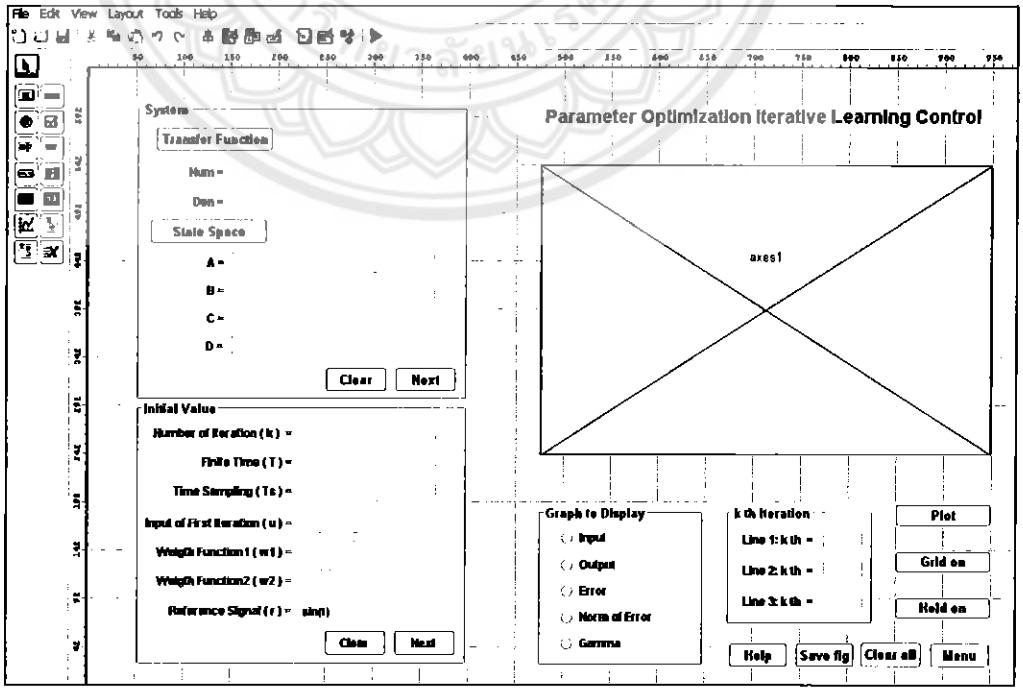
สำหรับหน้าต่างนี้ได้แบ่งเป็นสองส่วนคือ ส่วนที่ผู้ใช้กำหนดตัวแปรต่างๆที่จำเป็นสำหรับการคำนวณ และส่วนแสดงผลลัพธ์ในรูปแบบของกราฟ

1. การวางรูปแบบของวัตถุ ในส่วนนี้จะเป็นการออกแบบและวางวัตถุแต่ละชนิดในตำแหน่งต่างๆ ซึ่งประกอบด้วยตัวอักษรอยู่กับที่ (Static text), กล่องแก้ไขตัวอักษร (Editable text boxes), ปุ่มกด (Push button), ปุ่ม Radio (Radio buttons), ปุ่ม Toggle (Toggle buttons), Popup menu, Axe, และกรอบ (Frame) ดังแสดงในรูปที่ 3.9



รูปที่ 3.9 ตัวอย่างการวางรูปแบบของวัตถุบนหน้าต่างของ Parameter Optimization ILC

2. การแก้ไขคุณสมบัติของวัตถุ ในส่วนนี้จะเป็นการแก้ไขคุณสมบัติของวัตถุในหน้าต่างโปรแกรม เช่น การแก้ไขชื่อของปุ่ม ขนาดของปุ่ม ขนาดตัวอักษร เป็นต้น จะได้ดังรูปที่ 3.10



รูปที่ 3.10 ตัวอย่างการแก้ไขคุณสมบัติวัตถุบนหน้าต่างของ Parameter Optimization ILC

3. การแก้ไขคำสั่งการทำงานของวัตถุ ในส่วนนี้เป็นการแก้ไขคำสั่งการทำงานของวัตถุโดยการแก้ไขโดยใช้ฟังก์ชันเรียกกลับ จะเป็นการเขียนคำสั่งเพิ่มลงไปเพื่อให้โปรแกรมทำงานตามที่ต้องการ จากรูปที่ 3.10 เมื่อทำการแก้ไขคำสั่งเรียบร้อยแล้ว วัตถุแต่ละชนิดจะมีหน้าที่ดังต่อไปนี้

- ช่องรับค่าจากผู้ใช้ ทำหน้าที่รับค่าเพื่อใช้ในการคำนวณต่อไป สร้างโดยกล่องแก้ไขข้อความ
- Popup menu ทำหน้าที่ให้ผู้ใช้โปรแกรมเลือกสัญญาณอ้างอิงที่ต้องการให้กับระบบ สร้างโดย Popup menu
- ปุ่มฟังก์ชันถ่ายโอน (Transfer Function) ทำหน้าที่ให้ผู้ใช้สามารถป้อนระบบที่เป็นฟังก์ชันถ่ายโอนได้ สร้างโดยปุ่มกด
- ปุ่มปริภูมิสถานะ (State Space) ทำหน้าที่ให้ผู้ใช้สามารถป้อนระบบที่เป็นปริภูมิสถานะได้ สร้างโดยปุ่มกด
- ปุ่มเคลียร์ (Clear) ทำหน้าที่ลบข้อมูลที่ได้กรอกไว้ สร้างโดยปุ่มกด
- ปุ่มถัดไป (Next) ทำหน้าที่ตรวจสอบความถูกต้องของค่าที่รับมา และเริ่มต้นการกำหนดค่าในส่วนถัดไป สร้างโดยปุ่มกด
- ปุ่มสัญญาณขาเข้า (Input) ทำหน้าที่แสดงผลลัพธ์ของสัญญาณขาเข้า สร้างโดยปุ่ม Radio
- ปุ่มสัญญาณขาออก (Output) ทำหน้าที่แสดงผลลัพธ์ของสัญญาณขาออก สร้างโดยปุ่ม Radio
- ปุ่มสัญญาณค่าความผิดพลาด (Error) ทำหน้าที่แสดงผลลัพธ์ของสัญญาณค่าความผิดพลาด สร้างโดยปุ่ม Radio
- ปุ่มนอร์มของสัญญาณค่าความผิดพลาด (Norm of Error) ทำหน้าที่แสดงผลลัพธ์ค่านอร์มของค่าสัญญาณความผิดพลาด สร้างโดยปุ่ม Radio
- ปุ่มแกมมา (Gamma) ทำหน้าที่แสดงผลลัพธ์ค่าแกมมา สร้างโดยปุ่ม Radio
- กราฟแสดงผล ทำหน้าที่แสดงกราฟที่ผู้ใช้ต้องการ สร้างโดย Axes
- ปุ่มวาด (Plot) ทำหน้าที่คำนวณและวาดกราฟตามที่ผู้ใช้เลือกไว้ สร้างโดยปุ่มกด
- ปุ่มกริดออน (Grid On) ทำหน้าที่แสดงเส้นกริดและปิดเส้นกริด สร้างโดยปุ่ม Toggle
- ปุ่มโฮลด์ออน (Hold On) ทำหน้าที่ให้ผู้ใช้โปรแกรมสามารถเปรียบเทียบเส้นกราฟได้ สร้างโดยปุ่ม Toggle

- ปุ่มบันทึก (Save fig) ทำหน้าที่บันทึกรูปของกราฟที่ได้จากการคำนวณ สร้างโดยปุ่มกด

- ปุ่มลบทั้งหมด (Clear all) ทำหน้าที่ลบข้อมูลที่ได้ป้อนค้างไว้ และรวมถึงข้อมูลที่ได้จากการคำนวณ สร้างโดยปุ่มกด

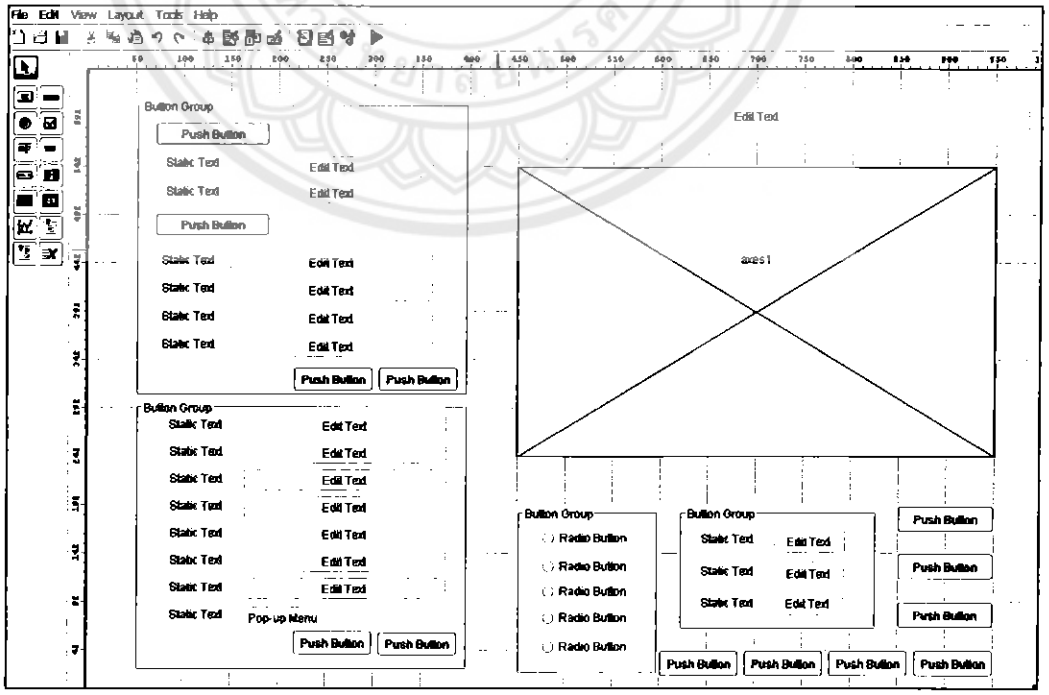
- ปุ่มช่วยเหลือ (Help) ทำหน้าที่แสดงตัวอย่างวิธีการป้อนค่าเริ่มต้นต่างๆ และอธิบายการแสดงผลลัพธ์ สร้างโดยปุ่มกด

- ปุ่มเมนู (Menu) ทำหน้าที่กลับสู่หน้าต่างหลักเพื่อเลือกวิธีการควบคุม สร้างโดยปุ่มกด

3.3.4 การสร้างหน้าต่างการคำนวณและแสดงผลของ Polynomial of matrix G ILC

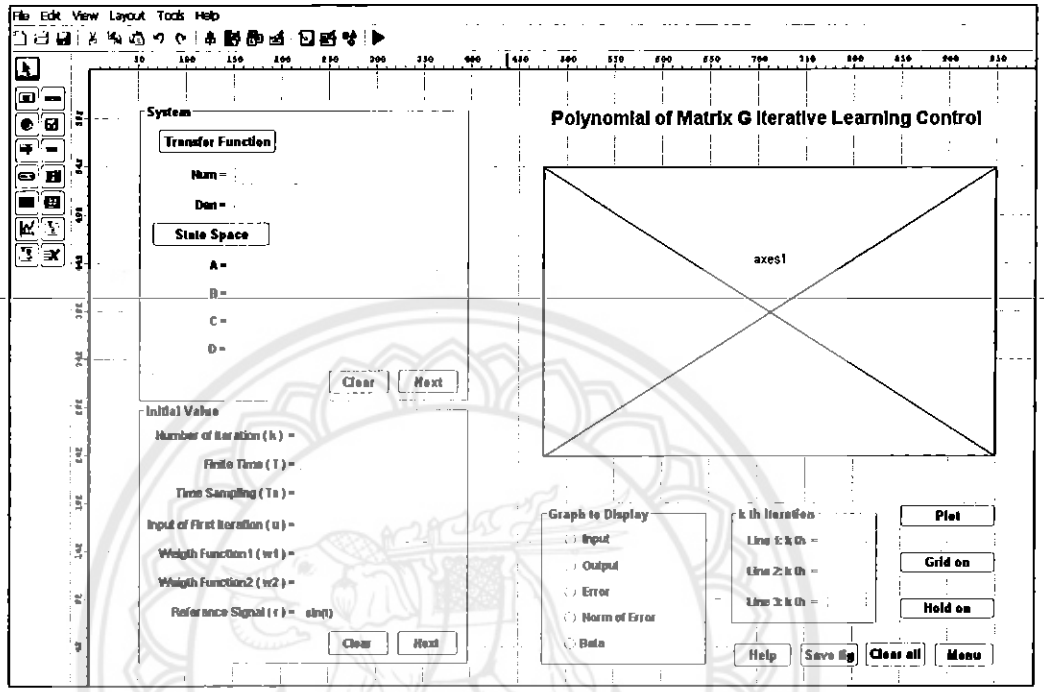
สำหรับหน้าต่างนี้ได้แบ่งเป็นสองส่วนคือ ส่วนที่ผู้ใช้กำหนดตัวแปรต่างๆที่จำเป็นสำหรับการคำนวณ และส่วนแสดงผลลัพธ์ในรูปแบบของกราฟ

1. การวางวัตถุ ในส่วนนี้จะเป็นการออกแบบและวางวัตถุแต่ละชนิดในตำแหน่งต่างๆ ซึ่งประกอบด้วยตัวอักษรอยู่กับที่ (Static text), กล่องแก้ไขตัวอักษร (Editable text boxes), ปุ่มกด (Push button), ปุ่ม Radio (Radio buttons), ปุ่ม Toggle (Toggle buttons), Popup menu, Axe และกรอบ (Frame) ดังแสดงในรูปที่ 3.11



รูปที่ 3.11 แสดงการวางรูปแบบของวัตถุบนหน้าต่างของ Polynomial of matrix G ILC

2. การแก้ไขคุณสมบัติของวัตถุ ในส่วนนี้จะเป็นการแก้ไขคุณสมบัติของวัตถุในหน้าต่างโปรแกรม เช่น การแก้ไขชื่อของปุ่ม ขนาดของปุ่ม ขนาดตัวอักษร เป็นต้น จะได้ดังรูปที่ 3.12



รูปที่ 3.12 แสดงการแก้ไขคุณสมบัติวัตถุบนหน้าต่างของ Polynomial of matrix G ILC

3. การแก้ไขคำสั่งการทำงานของวัตถุ ในส่วนนี้เป็นการแก้ไขคำสั่งการทำงานของวัตถุโดยการแก้ไขโดยใช้ฟังก์ชันเรียกกลับ จะเป็นการเขียนคำสั่งเพิ่มลงไปเพื่อให้โปรแกรมทำงานตามที่ต้องการ จากรูปที่ 3.12 เมื่อทำการแก้ไขคำสั่งเรียบร้อยแล้ว วัตถุแต่ละชนิดจะมีหน้าที่ดังต่อไปนี้

- ช่องรับค่าจากผู้ใช้ ทำหน้าที่รับค่าเพื่อใช้ในการคำนวณต่อไป สร้างโดยกล่องแก้ไขข้อความ
- Popup menu ทำหน้าที่ให้ผู้ใช้โปรแกรมเลือกสัญญาณอ้างอิงที่ต้องการให้กับระบบ สร้างโดย Popup Menu
- ปุ่มฟังก์ชันถ่ายโอน (Transfer Function) ทำหน้าที่ให้ผู้ใช้สามารถป้อนระบบที่เป็นฟังก์ชันถ่ายโอนได้ สร้างโดยปุ่มกด
- ปุ่มปริภูมิสถานะ (State Space) ทำหน้าที่ให้ผู้ใช้สามารถป้อนระบบที่เป็นปริภูมิสถานะได้ สร้างโดยปุ่มกด
- ปุ่มเคลียร์ (Clear) ทำหน้าที่ลบข้อมูลที่ได้กรอกไว้ สร้างโดยปุ่มกด

- ปุ่มถัดไป (Next) ทำหน้าที่ตรวจสอบความถูกต้องของค่าที่รับมา และเริ่มต้นการกำหนดค่าในส่วนถัดไป สร้างโดยปุ่มกด

- ปุ่มสัญญาณขาเข้า (Input) ทำหน้าที่แสดงผลลัพธ์ของสัญญาณขาเข้า สร้างโดยปุ่ม Radio

- ปุ่มสัญญาณขาออก (Output) ทำหน้าที่แสดงผลลัพธ์ของสัญญาณขาออก สร้างโดยปุ่ม Radio

- ปุ่มสัญญาณค่าความผิดพลาด (Error) ทำหน้าที่แสดงผลลัพธ์ของสัญญาณค่าความผิดพลาด สร้างโดยปุ่ม Radio

- ปุ่มนอร์มของสัญญาณค่าความผิดพลาด (Norm of Error) ทำหน้าที่แสดงผลลัพธ์ค่านอร์มของสัญญาณค่าความผิดพลาด สร้างโดยปุ่ม Radio

- ปุ่มเบต้า (Beta) ทำหน้าที่แสดงผลลัพธ์ค่าเบต้า สร้างโดยปุ่ม Radio

- กราฟแสดงผล ทำหน้าที่แสดงกราฟที่ผู้ใช้ต้องการ สร้างโดย Axes

- ปุ่มวาด (Plot) ทำหน้าที่คำนวณและวาดกราฟตามที่ผู้ใช้เลือกไว้ สร้างโดยปุ่มกด

- ปุ่มกริดออน (Grid On) ทำหน้าที่แสดงเส้นกริดและปิดเส้นกริด สร้างโดยปุ่ม Toggle

- ปุ่มโฮลด์ออน (Hold On) ทำหน้าที่ให้ผู้ใช้โปรแกรมสามารถเปรียบเทียบเส้นกราฟได้ สร้างโดยปุ่ม Toggle

- ปุ่มบันทึก (Save fig) ทำหน้าที่บันทึกรูปของกราฟที่ได้จากการคำนวณ สร้างโดยปุ่มกด

- ปุ่มลบทั้งหมด (Clear all) ทำหน้าที่ลบข้อมูลที่ได้ป้อนค้างไว้ และรวมถึงข้อมูลที่ได้จากการคำนวณ สร้างโดยปุ่มกด

- ปุ่มช่วยเหลือ (Help) ทำหน้าที่แสดงตัวอย่างวิธีการป้อนค่าเริ่มต้นต่างๆ และอธิบายการแสดงผลลัพธ์ สร้างโดยปุ่มกด

- ปุ่มเมนู (Menu) ทำหน้าที่กลับสู่หน้าต่างหลักเพื่อเลือกวิธีการควบคุม สร้างโดยปุ่มกด

บทที่ 4
ผลการดำเนินโครงงาน

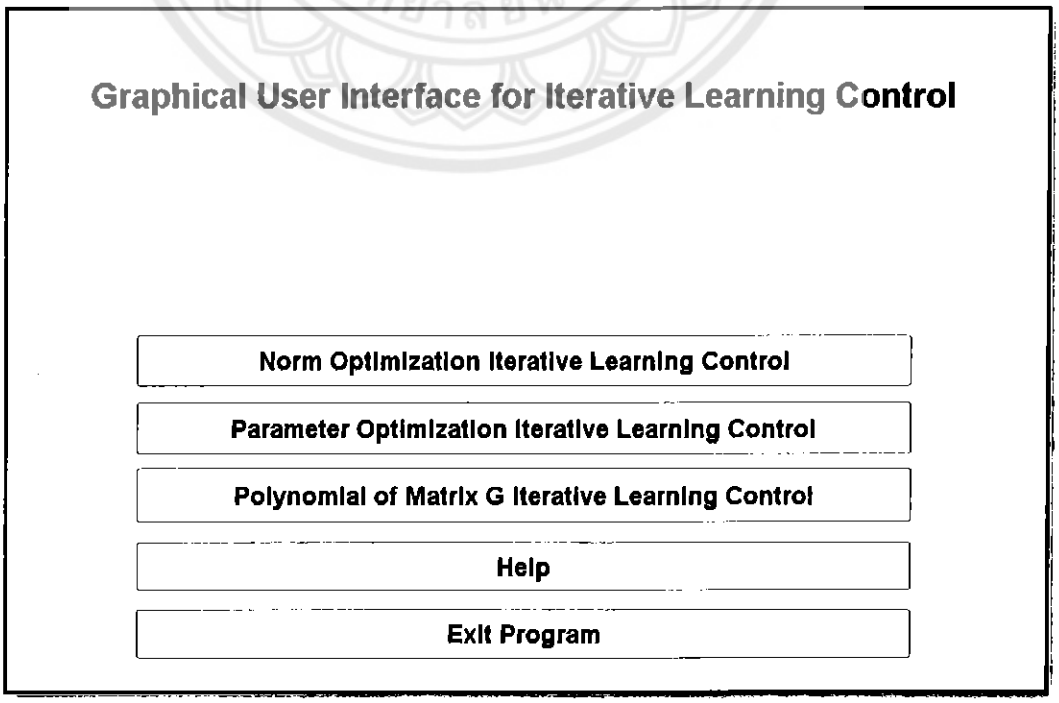
ในบทนี้จะกล่าวถึงวิธีการใช้งานโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้ที่ถูกสร้างขึ้นในบทที่ 3

4.1 การเริ่มต้นใช้งานโปรแกรม

การใช้งานโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้สำหรับวิธีการควบคุมแบบเรียนรู้ซ้ำนี้ สำหรับเครื่องคอมพิวเตอร์ที่ไม่มีโปรแกรม MATLAB ติดตั้งอยู่ในเครื่อง ต้องทำการติดตั้งโปรแกรมแปลภาษาของ MATLAB (MCRInstall) ก่อน จึงจะสามารถเรียกใช้งานโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้สำหรับวิธีการควบคุมแบบเรียนรู้ซ้ำได้ ส่วนเครื่องที่มีโปรแกรม MATLAB ติดตั้งอยู่ในเครื่อง สามารถเรียกใช้โปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้สำหรับวิธีการควบคุมแบบเรียนรู้ซ้ำได้ โดยใช้การเปิดไฟล์ .fig และ.m ในโปรแกรม MATLAB

4.2 ขั้นตอนการใช้งานโปรแกรม

เมื่อทำการเปิดโปรแกรมจะปรากฏหน้าต่างหลักดังรูปที่ 4.1

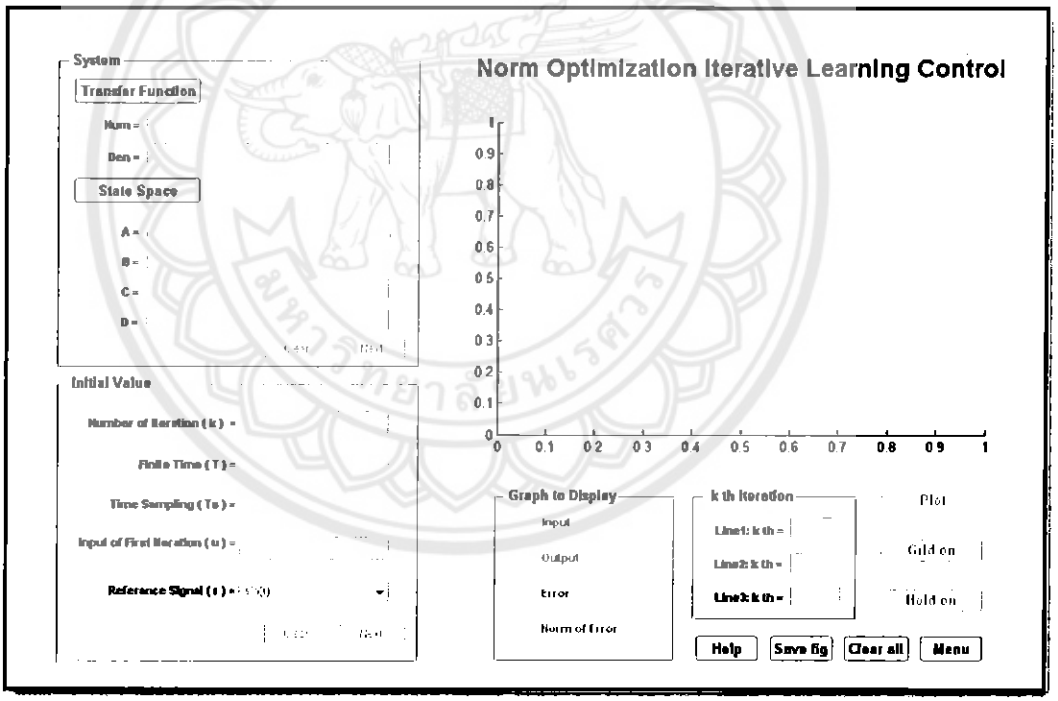


รูปที่ 4.1 หน้าต่างหลักของโปรแกรม

จากรูปที่ 4.1 หน้าต่างหลักประกอบด้วยตัวเลือกดังนี้

- Norm Optimization Iterative Learning Control กดหนึ่งครั้งเพื่อเลือกวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้ค่าประจำนอร์มที่เหมาะสมที่สุด
- Parameter Optimization Iterative Learning Control กดหนึ่งครั้งเพื่อเลือกวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้ตัวแปรที่เหมาะสมที่สุด
- Polynomial of Matrix G Iterative Learning Control กดหนึ่งครั้งเพื่อเลือกวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้พหุนามของเมทริกซ์ G
- Help กดหนึ่งครั้ง เมื่อต้องการเปิดเอกสารแนะนำการใช้งานโปรแกรม
- Exit Program กดหนึ่งครั้ง เมื่อต้องการออกจากโปรแกรม

ก) เมื่อกดปุ่ม Norm Optimization Iterative Learning Control จะปรากฏหน้าต่างดังรูปที่ 4.2



รูปที่ 4.2 หน้าต่างวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้ค่าประจำนอร์มที่เหมาะสมที่สุด

จากรูปที่ 4.2 หน้าต่างมีขั้นตอนการทำงาน ดังต่อไปนี้

1. เลือกชนิดของการป้อนระบบที่ต้องการควบคุม ดังนี้
   - Transfer Function กดหนึ่งครั้ง เพื่อเลือกการป้อนระบบเป็นแบบฟังก์ชันถ่ายโอน (Transfer function) จากนั้นจะปรากฏช่องว่างสำหรับการป้อนเมทริกซ์เศษ (Numerator) และเมทริกซ์ส่วน (Denominator)

- State Space กดหนึ่งครั้ง เพื่อเลือกการป้อนระบบเป็นแบบปริภูมิสถานะ (State space) จากนั้นจะปรากฏช่องว่างสำหรับการป้อนเมทริกซ์ A, B, C และ D

2. เมื่อทำการเลือกชนิดและป้อนเมทริกซ์ของระบบเสร็จเรียบร้อยแล้ว กดปุ่ม Next เพื่อทำการป้อนข้อมูลลำดับต่อไป

   กรณี ต้องการลบค่าเมทริกซ์ของระบบทั้งหมด กรุณากดปุ่ม Clear

3. เมื่อกดปุ่ม Next จะปรากฏช่องว่างสำหรับป้อนค่าเริ่มต้นต่างๆ ดังนี้

   3.1. Number of Iteration ป้อนจำนวนรอบการทำงาน (เป็นจำนวนเต็มบวก) เช่น 10,50,100,500 เป็นต้น

   3.2. Finite Time ป้อนเวลาการทำงาน (เป็นจำนวนเต็มบวก) เช่น 10, 14 20,100 เป็นต้น

   3.3. Time Sampling ป้อนจุดเวลาการทำงาน (มีค่ามากกว่าศูนย์) เช่น 0.01,0.028,0.1,2 เป็นต้น

   3.4. Input of First Iteration ป้อนสัญญาณขาเข้าของการทำงานรอบที่ 1 เช่น 0,1 ,3,5 เป็นต้น

   3.5. Reference Signal เลือกสัญญาณอ้างอิงของระบบ ประกอบด้วย

   - $sin(t)$
   - $5sin(t)$
   - $5sin(0.5\pi t)$
   - $t^2 sin(t)$
   - $e^{t/20} sin(t)$
   - $e^{t/20} sin(t^2/10)$
   - $square\ wave$

4. เมื่อป้อนค่าเริ่มต่างๆครบเรียบร้อยแล้ว กดปุ่ม Next เพื่อทำการเลือกชนิดของกราฟที่ต้องการแสดงผลต่อไป

   กรณี ต้องการลบค่าเริ่มต้นต่างๆที่ป้อนทั้งหมด กรุณากดปุ่ม Clear

5. เมื่อกดปุ่ม Next จะปรากฏตัวเลือกสำหรับเลือกชนิดของกราฟที่ต้องการแสดงผล ดังนี้

   - Input เป็นกราฟที่แสดงค่าสัญญาณขาเข้าของระบบ เมื่อเลือกตัวเลือกนี้จะปรากฏช่องว่างสำหรับป้อนค่ารอบการทำงานที่ต้องการแสดง โดยกำหนดให้ป้อน 3 ค่า เพื่อแสดงค่าสัญญาณขาเข้าจำนวน 3 รอบการทำงาน จากนั้นกดปุ่ม Plot เพื่อแสดงผลในรูปแบบของกราฟต่อไป
   - Output เป็นกราฟที่แสดงค่าสัญญาณขาออกของระบบ เมื่อเลือกตัวเลือกนี้จะปรากฏช่องว่างสำหรับป้อนค่ารอบการทำงานที่ต้องการแสดง โดยกำหนดให้ป้อน 3 ค่า เพื่อแสดงค่าสัญญาณขาออกจำนวน 3 รอบการทำงาน จากนั้นกดปุ่ม Plot เพื่อแสดงผลในรูปแบบของกราฟต่อไป
   - Error เป็นกราฟที่แสดงค่าสัญญาณความผิดพลาดของระบบ เมื่อเลือกตัวเลือกนี้จะปรากฏช่องว่างสำหรับป้อนค่ารอบการทำงานที่ต้องการแสดง โดยกำหนดให้ป้อน 3 ค่า เพื่อแสดง

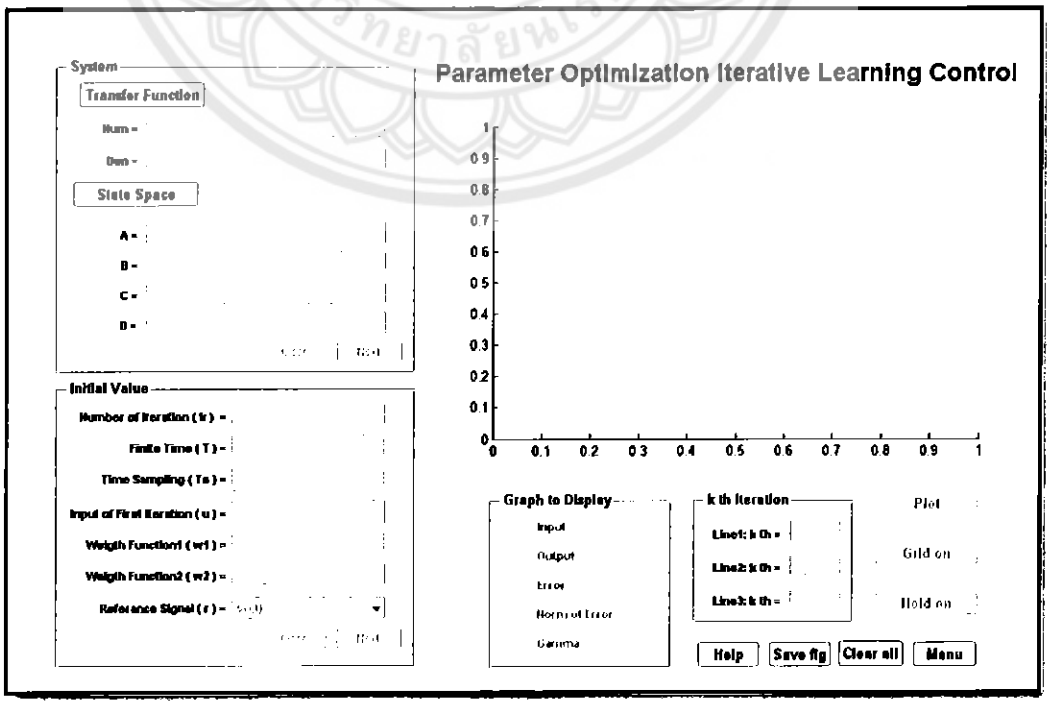
ค่าสัญญาณความผิดพลาดจำนวน 3 รอบการทำงาน จากนั้นกดปุ่ม Plot เพื่อแสดงผลในรูปแบบของกราฟต่อไป

- Norm of Error เป็นกราฟที่แสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบ ซึ่งจะแสดงในรูปแบบของกราฟค่าประจำนอร์มของค่าสัญญาณความผิดพลาดเทียบกับจำนวนรอบการทำงาน เมื่อเลือกตัวเลือกนี้แล้วสามารถกดปุ่ม Plot เพื่อแสดงผลได้ทันที

6. ปุ่มกดอื่น มีการทำงานดังนี้

- Grid on กดหนึ่งครั้งเพื่อแสดงเส้นประรูปตาข่ายตามค่าของแกน กดอีกครั้งเมื่อต้องการลบเส้นประรูปตาข่ายตามค่าของแกน
- Hold on กดหนึ่งครั้ง เมื่อต้องการแสดงกราฟใหม่ในขณะที่ยังแสดงกราฟเดิมอยู่ กดอีกหนึ่งครั้งเมื่อต้องการแสดงกราฟใหม่อย่างเดียว
- Help กดหนึ่งครั้ง เมื่อต้องการเปิดเอกสารแนะนำการใช้งานโปรแกรม
- Save fig กดหนึ่งครั้ง เมื่อต้องการบันทึกรูปกราฟที่แสดงอยู่ในปัจจุบันลงในเครื่องคอมพิวเตอร์
- Clear all กดหนึ่งครั้ง เมื่อต้องการลบข้อมูลทั้งหมดในหน้าต่างและเริ่มการทำงานของหน้าต่างใหม่
- Menu กดหนึ่งครั้ง เมื่อต้องการออกจากหน้าต่างปัจจุบันและกลับไปยังหน้าต่างหลัก

ข) เมื่อกดปุ่ม Parameter Optimization Iterative Learning Control จะปรากฏหน้าต่างดังรูปที่ 4.3



รูปที่ 4.3 หน้าต่างวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้ตัวแปรที่เหมาะสมที่สุด

จากรูปที่ 4.3 หน้าต่างมีขั้นตอนการทำงาน ดังต่อไปนี้

1. เลือกชนิดของการป้อนระบบที่ต้องการควบคุม ดังนี้
   - Transfer Function กดหนึ่งครั้ง เพื่อเลือกการป้อนระบบเป็นแบบฟังก์ชันถ่ายโอน (Transfer function) จากนั้นจะปรากฏช่องว่างสำหรับการป้อนเมทริกซ์เศษ (Numerator) และเมทริกซ์ส่วน (Denominator)
   - State Space กดหนึ่งครั้ง เพื่อเลือกการป้อนระบบเป็นแบบปริภูมิสถานะ (State space) จากนั้นจะปรากฏช่องว่างสำหรับการป้อนเมทริกซ์ A, B, C และ D
2. เมื่อทำการเลือกชนิดและป้อนเมทริกซ์ของระบบเสร็จเรียบร้อยแล้ว กดปุ่ม Next เพื่อทำการป้อนข้อมูลลำดับต่อไป

   กรณี ต้องการลบค่าเมทริกซ์ของระบบทั้งหมด กรุณากดปุ่ม Clear
3. เมื่อกดปุ่ม Next จะปรากฏช่องว่างสำหรับป้อนค่าเริ่มต้นต่างๆ ดังนี้
   3.1. Number of Iteration ป้อนจำนวนรอบการทำงาน (เป็นจำนวนเต็มบวก) เช่น 10,50,100,500 เป็นต้น
   3.2. Finite Time ป้อนเวลาการทำงาน (เป็นจำนวนเต็มบวก) เช่น 10, 14 20,100 เป็นต้น
   3.3. Time Sampling ป้อนจุดเวลาการทำงาน (มีค่ามากกว่าศูนย์) เช่น 0.01,0.028,0.1,2 เป็นต้น
   3.4. Input of First Iteration ป้อนสัญญาณขาเข้าของการทำงานรอบที่ 1 เช่น 0,1 ,3,5 เป็นต้น
   3.5. Weight Function1 ป้อนตัวแปรค่าถ่วงน้ำหนักตัวที่ 1 เช่น 0,10^-1,10^-2,10^-6 เป็นต้น
   3.6. Weight Function2 ป้อนตัวแปรค่าถ่วงน้ำหนักตัวที่ 2 เช่น 0,10^-1,10^-2,10^-6 เป็นต้น
   3.7. Reference Signal เลือกสัญญาณอ้างอิงของระบบ ประกอบด้วย
      - $sin(t)$
      - $5sin(t)$
      - $5sin(0.5\pi t)$
      - $t^2 sin(t)$
      - $e^{t/20} sin(t)$
      - $e^{t/20} sin(t^2/10)$
      - $square\ wave$

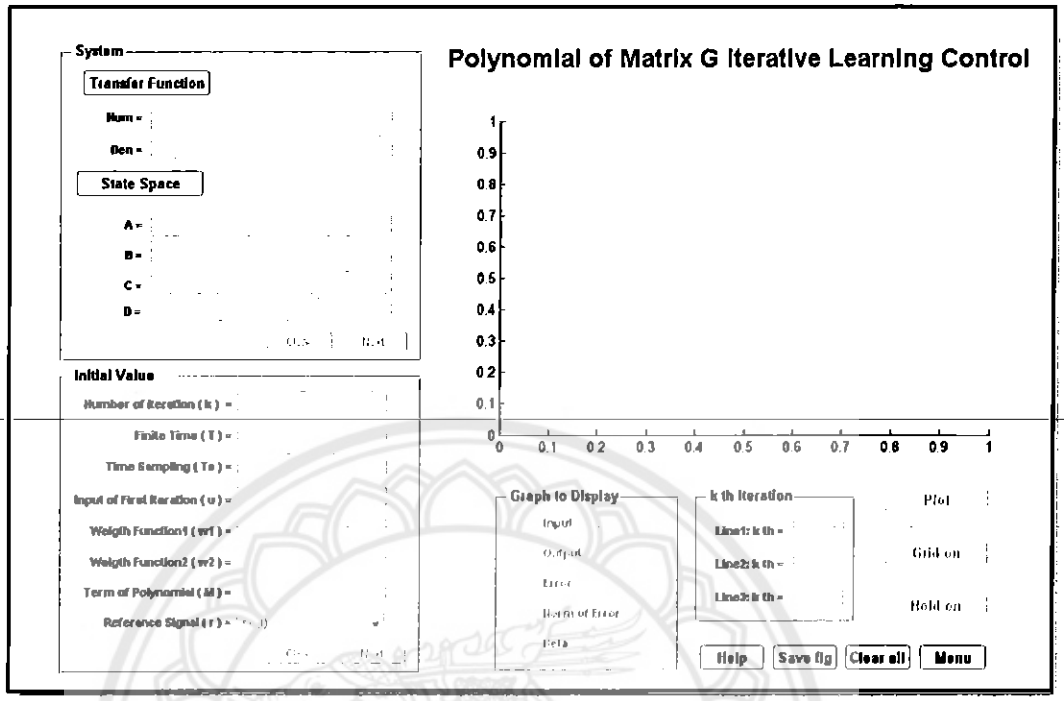
หมายเหตุ กรณีต้องการตัวแปรค่าถ่วงน้ำหนักแบบคงที่ ป้อนตัวแปรค่าถ่วงน้ำหนักตัวที่ 1 มีค่ามากกว่าศูนย์และป้อนตัวแปรค่าถ่วงน้ำหนักตัวที่ 2 เป็นศูนย์ ส่วนกรณีต้องการตัวแปรค่าถ่วงน้ำหนักแบบปรับค่าได้ ป้อนตัวแปรค่าถ่วงน้ำหนักตัวที่ 1และตัวที่ 2 มีค่ามากกว่าศูนย์

4. เมื่อป้อนค่าเริ่มต่างๆครบเรียบร้อยแล้ว กดปุ่ม Next เพื่อทำการเลือกชนิดของกราฟที่ต้องการแสดงผลต่อไป

   กรณี ต้องการลบค่าเริ่มต้นต่างๆที่ป้อนทั้งหมด กรุณากดปุ่ม Clear

5. เมื่อกดปุ่ม Next จะปรากฏตัวเลือกสำหรับเลือกชนิดของกราฟที่ต้องการแสดงผล ดังนี้
   - Input เป็นกราฟที่แสดงค่าสัญญาณขาเข้าของระบบ เมื่อเลือกตัวเลือกนี้จะปรากฏช่องว่างสำหรับป้อนค่ารอบการทำงานที่ต้องการแสดง โดยกำหนดให้ป้อน 3 ค่า เพื่อแสดงค่าสัญญาณขาเข้าจำนวน 3 รอบการทำงาน จากนั้นกดปุ่ม Plot เพื่อแสดงผลในรูปแบบของกราฟต่อไป
   - Output เป็นกราฟที่แสดงค่าสัญญาณขาออกของระบบ เมื่อเลือกตัวเลือกนี้จะปรากฏช่องว่างสำหรับป้อนค่ารอบการทำงานที่ต้องการแสดง โดยกำหนดให้ป้อน 3 ค่า เพื่อแสดงค่าสัญญาณขาออกจำนวน 3 รอบการทำงาน จากนั้นกดปุ่ม Plot เพื่อแสดงผลในรูปแบบของกราฟต่อไป
   - Error เป็นกราฟที่แสดงค่าสัญญาณความผิดพลาดของระบบ เมื่อเลือกตัวเลือกนี้จะปรากฏช่องว่างสำหรับป้อนค่ารอบการทำงานที่ต้องการแสดง โดยกำหนดให้ป้อน 3 ค่า เพื่อแสดงค่าสัญญาณความผิดพลาดจำนวน 3 รอบการทำงาน จากนั้นกดปุ่ม Plot เพื่อแสดงผลในรูปแบบของกราฟต่อไป
   - Norm of Error เป็นกราฟที่แสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบ ซึ่งจะแสดงในรูปแบบของกราฟค่าประจำนอร์มของค่าสัญญาณความผิดพลาดเทียบกับจำนวนรอบการทำงาน เมื่อเลือกตัวเลือกนี้แล้วสามารถกดปุ่ม Plot เพื่อแสดงผลได้ทันที(เมื่อหน้าต่างแสดงกราฟ จะปรากฏรูปกากบาทที่เมาส์เพื่อให้ผู้ใช้เลือกตำแหน่งของการระบุค่าตัวแปรค่าถ่วงน้ำหนักในกราฟ)
   - Gamma เป็นกราฟที่แสดงค่าตัวแปรอิสระของวิธีการควบคุม ซึ่งจะแสดงในรูปแบบของกราฟค่าตัวแปรอิสระเทียบกับจำนวนรอบการทำงาน เมื่อเลือกตัวเลือกนี้แล้วสามารถกดปุ่ม Plot เพื่อแสดงผลได้ทันที
6. ปุ่มกดอื่น มีการทำงานดังนี้
   - Grid on กดหนึ่งครั้งเพื่อแสดงเส้นประรูปตาข่ายตามค่าของแกน กดอีกครั้งเมื่อต้องการลบเส้นประรูปตาข่ายตามค่าของแกน
   - Hold on กดหนึ่งครั้ง เมื่อต้องการแสดงกราฟใหม่ในขณะที่ยังแสดงกราฟเดิมอยู่ กดอีกหนึ่งครั้งเมื่อต้องการแสดงกราฟใหม่อย่างเดียว
   - Help กดหนึ่งครั้ง เมื่อต้องการเปิดเอกสารแนะนำการใช้งานโปรแกรม
   - Save fig กดหนึ่งครั้ง เมื่อต้องการบันทึกรูปกราฟที่แสดงอยู่ในปัจจุบันลงในเครื่องคอมพิวเตอร์
   - Clear all กดหนึ่งครั้ง เมื่อต้องการลบข้อมูลทั้งหมดในหน้าต่างและเริ่มการทำงานของหน้าต่างใหม่
   - Menu กดหนึ่งครั้ง เมื่อต้องการออกจากหน้าต่างปัจจุบันและกลับไปยังหน้าต่างหลัก

ค) เมื่อกดปุ่ม Polynomial of Matrix G Iterative Learning Control จะปรากฏหน้าต่างดังรูปที่ 4



รูปที่ 4.4 หน้าต่างวิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้พหุนามของเมทริกซ์ G

จากรูปที่ 4.4 หน้าต่างมีขั้นตอนการทำงาน ดังต่อไปนี้

1. เลือกชนิดของการป้อนระบบที่ต้องการควบคุม ดังนี้
   - Transfer Function กดหนึ่งครั้ง เพื่อเลือกการป้อนระบบเป็นแบบฟังก์ชันถ่ายโอน (Transfer function) จากนั้นจะปรากฏช่องว่างสำหรับการป้อนเมทริกซ์เศษ (Numerator) และเมทริกซ์ส่วน (Denominator)
   - State Space กดหนึ่งครั้ง เพื่อเลือกการป้อนระบบเป็นแบบปริภูมิสถานะ (State space) จากนั้นจะปรากฏช่องว่างสำหรับการป้อนเมทริกซ์ A, B, C และ D
2. เมื่อทำการเลือกชนิดและป้อนเมทริกซ์ของระบบเสร็จเรียบร้อยแล้ว กดปุ่ม Next เพื่อทำการป้อนข้อมูลลำดับต่อไป

   กรณี ต้องการลบค่าเมทริกซ์ของระบบทั้งหมด กรุณากดปุ่ม Clear
3. เมื่อกดปุ่ม Next จะปรากฏช่องว่างสำหรับป้อนค่าเริ่มต้นต่างๆ ดังนี้
   3.1. Number of Iteration ป้อนจำนวนรอบการทำงาน (เป็นจำนวนเต็มบวก) เช่น 10,50,100,500 เป็นต้น
   3.2. Finite Time ป้อนเวลาการทำงาน (เป็นจำนวนเต็มบวก) เช่น 10, 14 20,100 เป็นต้น
   3.3. Time Sampling ป้อนจุดเวลาการทำงาน (มีค่ามากกว่าศูนย์) เช่น 0.01,0.028,0.1,2 เป็นต้น
   3.4. Input of First Iteration ป้อนสัญญาณขาเข้าของการทำงานรอบที่ 1 เช่น 0,1 ,3,5 เป็นต้น

3.5. Weight function1 ป้อนตัวแปรค่าถ่วงน้ำหนักตัวที่ 1 เช่น 0,10^-1,10^-2,10^-6 เป็นต้น

3.6. Weight function2 ป้อนตัวแปรค่าถ่วงน้ำหนักตัวที่ 2 เช่น 0,10^-1,10^-2,10^-6 เป็นต้น

3.7. Term of polynomial ป้อนค่าจำนวนพจน์ของพหุนาม (เป็นจำนวนเต็มบวก) เช่น 1,2,3,5,100 เป็นต้น

3.8. Reference signal เลือกสัญญาณอ้างอิงของระบบ ประกอบด้วย

- $sin(t)$
- $5sin(t)$
- $5sin(0.5\pi t)$
- $t^2 sin(t)$
- $e^{t/20} sin(t)$
- $e^{t/20} sin(t^2/10)$
- *square wave*

หมายเหตุ กรณีต้องการตัวแปรค่าถ่วงน้ำหนักแบบคงที่ ป้อนตัวแปรค่าถ่วงน้ำหนักตัวที่ 1 มีค่ามากกว่าศูนย์และป้อนตัวแปรค่าถ่วงน้ำหนักตัวที่ 2 เป็นศูนย์ ส่วนกรณีต้องการตัวแปรค่าถ่วงน้ำหนักแบบปรับค่าได้ ป้อนตัวแปรค่าถ่วงน้ำหนักตัวที่ 1และตัวที่ 2 มีค่ามากกว่าศูนย์

4. เมื่อป้อนค่าเริ่มต่างๆครบเรียบร้อยแล้ว กดปุ่ม Next เพื่อทำการเลือกชนิดของกราฟที่ต้องการแสดงผลต่อไป

   กรณี ต้องการลบค่าเริ่มต้นต่างๆที่ป้อนทั้งหมด กรุณากดปุ่ม Clear

5. เมื่อกดปุ่ม Next จะปรากฏตัวเลือกสำหรับเลือกชนิดของกราฟที่ต้องการแสดงผล ดังนี้

   - Input เป็นกราฟที่แสดงค่าสัญญาณขาเข้าของระบบ เมื่อเลือกตัวเลือกนี้จะปรากฏช่องว่างสำหรับป้อนค่ารอบการทำงานที่ต้องการแสดง โดยกำหนดให้ป้อน 3 ค่า เพื่อแสดงค่าสัญญาณขาเข้าจำนวน 3 รอบการทำงาน จากนั้นกดปุ่ม Plot เพื่อแสดงผลในรูปแบบของกราฟต่อไป
   - Output เป็นกราฟที่แสดงค่าสัญญาณขาออกของระบบ เมื่อเลือกตัวเลือกนี้จะปรากฏช่องว่างสำหรับป้อนค่ารอบการทำงานที่ต้องการแสดง โดยกำหนดให้ป้อน 3 ค่า เพื่อแสดงค่าสัญญาณขาออกจำนวน 3 รอบการทำงาน จากนั้นกดปุ่ม Plot เพื่อแสดงผลในรูปแบบของกราฟต่อไป
   - Error เป็นกราฟที่แสดงค่าสัญญาณความผิดพลาดของระบบ เมื่อเลือกตัวเลือกนี้จะปรากฏช่องว่างสำหรับป้อนค่ารอบการทำงานที่ต้องการแสดง โดยกำหนดให้ป้อน 3 ค่า เพื่อแสดงค่าสัญญาณความผิดพลาดจำนวน 3 รอบการทำงาน จากนั้นกดปุ่ม Plot เพื่อแสดงผลในรูปแบบของกราฟต่อไป

- Norm of Error เป็นกราฟที่แสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบ ซึ่งจะแสดงในรูปแบบของกราฟค่าประจำนอร์มของค่าสัญญาณความผิดพลาดเทียบกับจำนวนรอบการทำงาน เมื่อเลือกตัวเลือกนี้แล้วสามารถกดปุ่ม Plot เพื่อแสดงผลได้ทันที(เมื่อหน้าต่างแสดงกราฟ จะปรากฏรูปกากบาทที่เมาส์เพื่อให้ผู้ใช้เลือกตำแหน่งของการระบุค่าจำนวนพจน์ของพหุนามและตัวแปรค่าถ่วงน้ำหนักในกราฟ)
- Beta เป็นกราฟที่แสดงค่าตัวแปรอิสระของวิธีการควบคุม ซึ่งจะแสดงในรูปแบบของกราฟค่าตัวแปรอิสระเทียบกับจำนวนรอบการทำงาน เมื่อเลือกตัวเลือกนี้แล้วสามารถกดปุ่ม Plot เพื่อแสดงผลได้ทันที

6. ปุ่มกดอื่น มีการทำงานดังนี้

- Grid on กดหนึ่งครั้งเพื่อแสดงเส้นปะรูปตาข่ายตามค่าของแกน กดอีกครั้งเมื่อต้องการลบเส้นปะรูปตาข่ายตามค่าของแกน
- Hold on กดหนึ่งครั้ง เมื่อต้องการแสดงกราฟใหม่ในขณะที่ยังแสดงกราฟเดิมอยู่ กดอีกหนึ่งครั้งเมื่อต้องการแสดงกราฟใหม่อย่างเดียว
- Help กดหนึ่งครั้ง เมื่อต้องการเปิดเอกสารแนะนำการใช้งานโปรแกรม
- Save fig กดหนึ่งครั้ง เมื่อต้องการบันทึกรูปกราฟที่แสดงอยู่ในปัจจุบันลงในเครื่องคอมพิวเตอร์
- Clear all กดหนึ่งครั้ง เมื่อต้องการลบข้อมูลทั้งหมดในหน้าต่างและเริ่มการทำงานของหน้าต่างใหม่
- Menu กดหนึ่งครั้ง เมื่อต้องการออกจากหน้าต่างปัจจุบันและกลับไปยังหน้าต่างหลัก

4.3 การทดสอบความถูกต้องของโปรแกรม

ในการทดสอบการทำงานของโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้สำหรับวิธีการควบคุมแบบเรียนรู้ซ้ำนี้ เราเลือกใช้ระบบในการทดสอบจำนวนสามระบบคือ ระบบอันดับหนึ่ง (First order system), ระบบอันดับสอง (Second order system) และระบบที่ใช้งานจริง เพื่อตรวจสอบความถูกต้องในการทำงานและแสดงผลการทดลองของโปรแกรม ดังนี้

ระบบอันดับหนึ่ง (First order system)

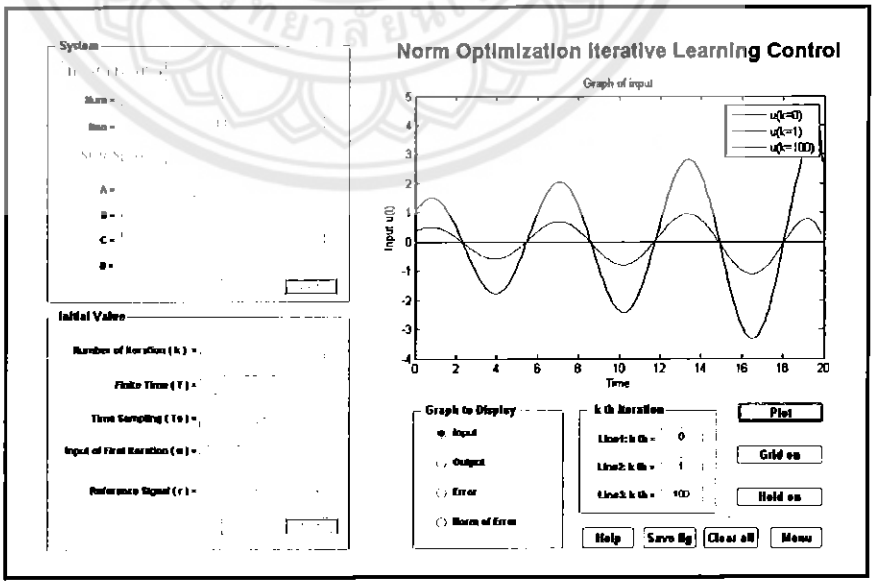
ฟังก์ชันถ่ายโอนของระบบอันดับหนึ่ง คือ

$$G(s) = \frac{1}{s+1}$$

ใช้เวลาในการชักตัวอย่าง $Ts = 0.1$ วินาที ทำงานในช่วงเวลา $T = [0,20]$ มีค่าตัวแปรค่าถ่วงน้ำหนักแบบคงที่ $w = 10^{-6}$ จำนวนพจน์ของพหุนามเท่ากับ $M = 3$ และมีการวนรอบจำนวน $k = 100$ ครั้ง โดยเลือกสัญญาณอ้างอิงเป็น $r = e^{(t/20)}\sin(t)$ จะได้ผลการทดสอบดังนี้

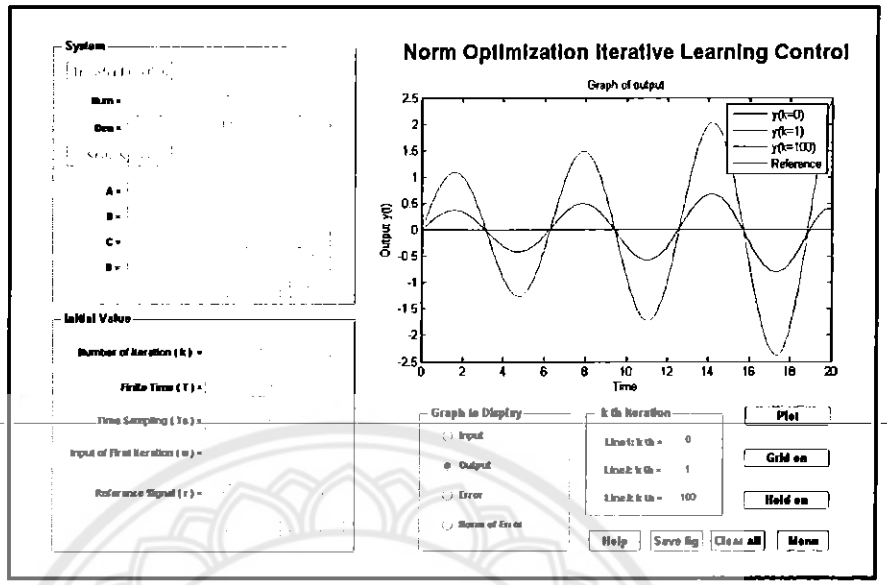
ก. วิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้ค่าประจำนอร์มที่เหมาะสมที่สุด (Norm optimization iterative learning control)

1. จากระบบอันดับหนึ่ง จะได้กราฟแสดงค่าสัญญาณขาเข้าดังรูปที่ 4.5



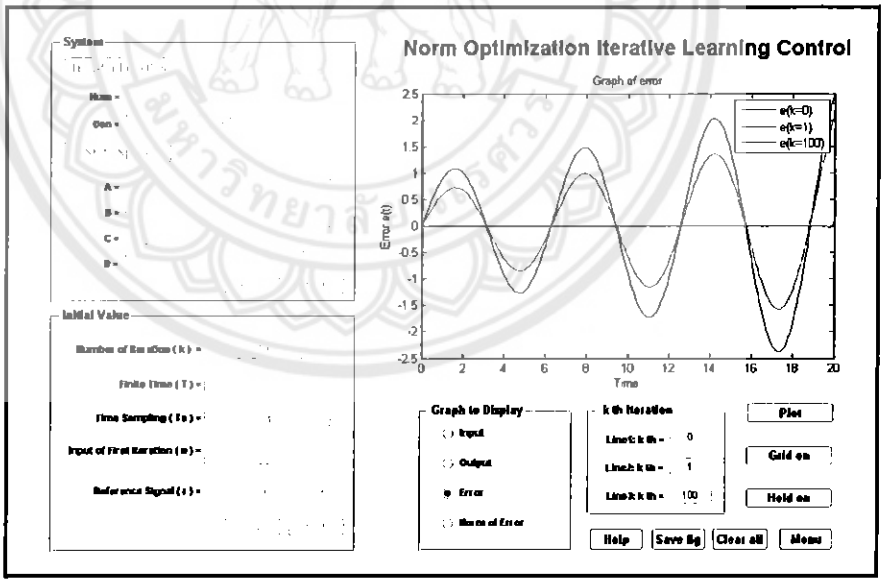
รูปที่ 4.5 กราฟแสดงค่าสัญญาณขาเข้าของระบบอับดับหนึ่งที่ได้จากวิธี NOILC

2. จากระบบอันดับหนึ่ง จะได้กราฟแสดงค่าสัญญาณขาออกดังรูปที่ 4.6



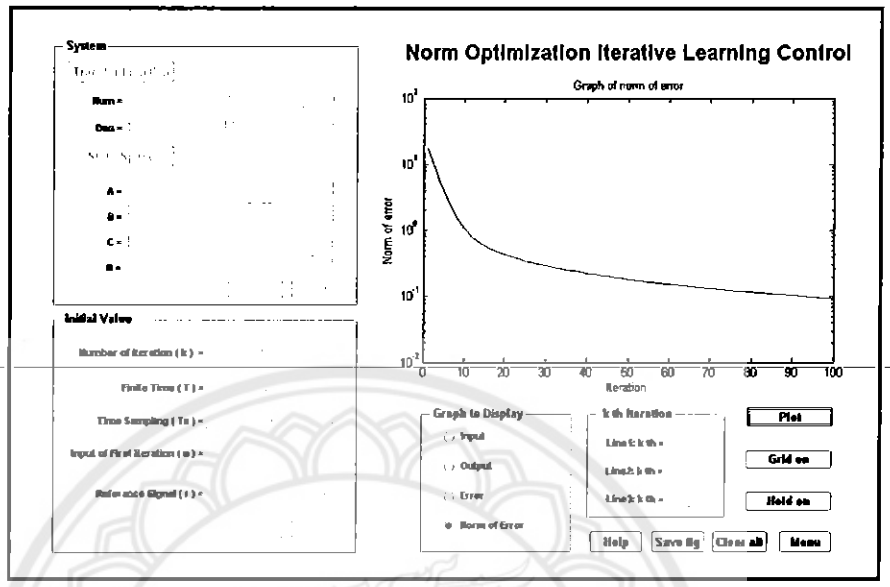
รูปที่ 4.6 กราฟแสดงค่าสัญญาณขาออกของระบบอันดับหนึ่งที่ได้จากวิธี NOILC

3. จากระบบอันดับหนึ่ง จะได้กราฟแสดงค่าสัญญาณความผิดพลาดดังรูปที่ 4.7



รูปที่ 4.7 กราฟแสดงค่าสัญญาณความผิดพลาดของระบบอันดับหนึ่งที่ได้จากวิธี NOILC

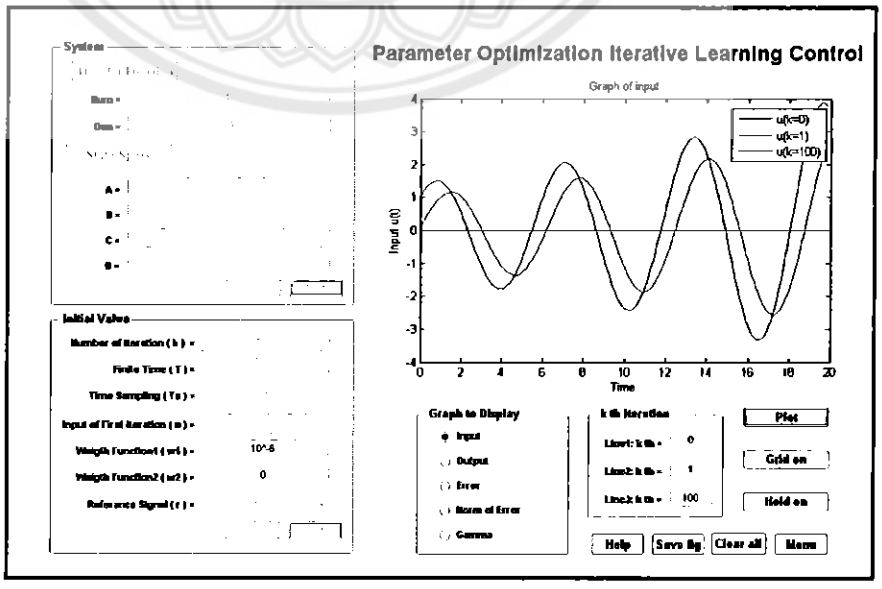
4. จากระบบอันดับหนึ่ง จะได้กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดดังรูปที่ 4.8



รูปที่ 4.8 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับหนึ่งที่ได้จากวิธี NOILC

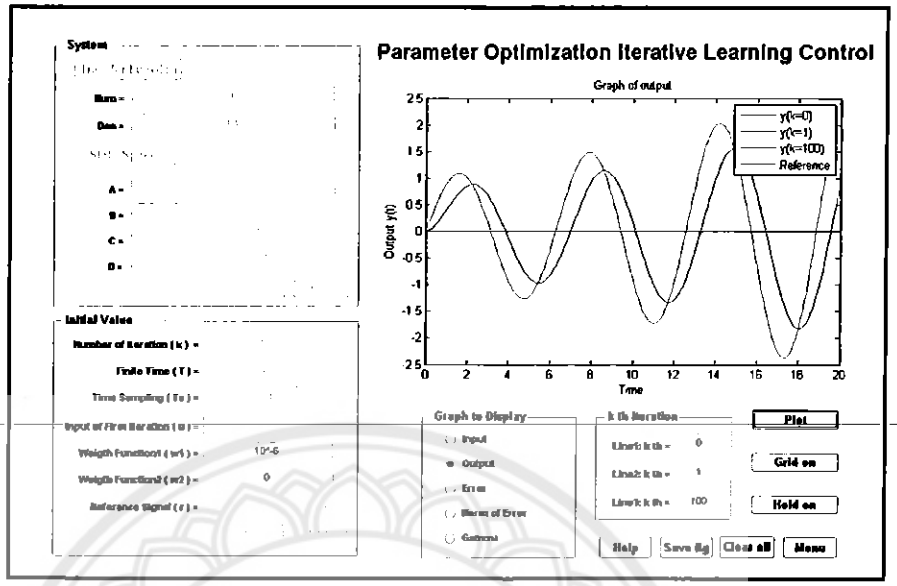
ข. วิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้ตัวแปรที่เหมาะสมที่สุด (Parameter optimization iterative learning control)

1. จากระบบอันดับหนึ่ง จะได้กราฟแสดงค่าสัญญาณขาเข้าดังรูปที่ 4.9



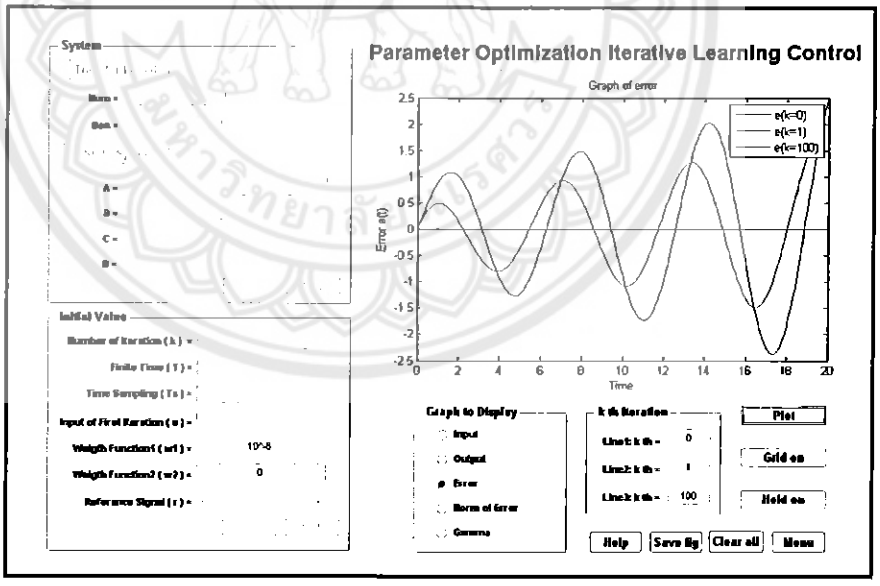
รูปที่ 4.9 กราฟแสดงค่าสัญญาณขาเข้าของระบบอันดับหนึ่งที่ได้จากวิธี POILC

2. จากระบบอันดับหนึ่ง จะได้กราฟแสดงค่าสัญญาณขาออกดังรูปที่ 4.10



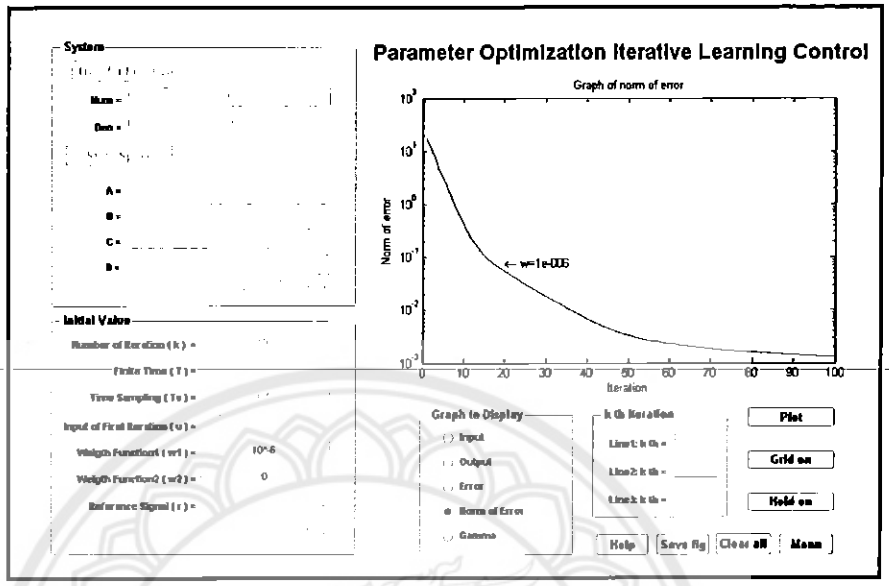
รูปที่ 4.10 กราฟแสดงค่าสัญญาณขาออกของระบบอันดับหนึ่งที่ได้จากวิธี POILC

3. จากระบบอันดับหนึ่ง จะได้กราฟแสดงค่าสัญญาณความผิดพลาดดังรูปที่ 4.11



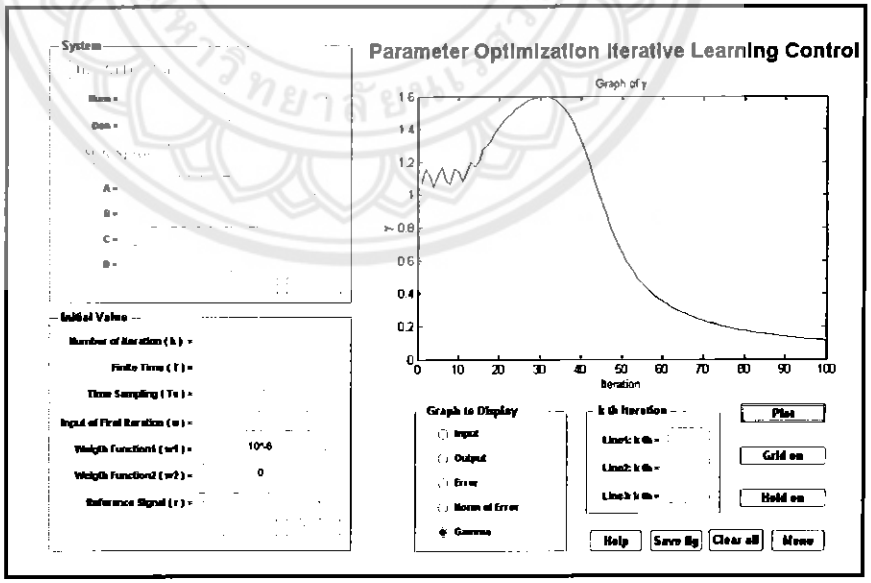
รูปที่ 4.11 กราฟแสดงค่าสัญญาณความผิดพลาดของระบบอันดับหนึ่งที่ได้จากวิธี POILC

4. จากระบบอันดับหนึ่ง จะได้กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดดังรูปที่ 4.12



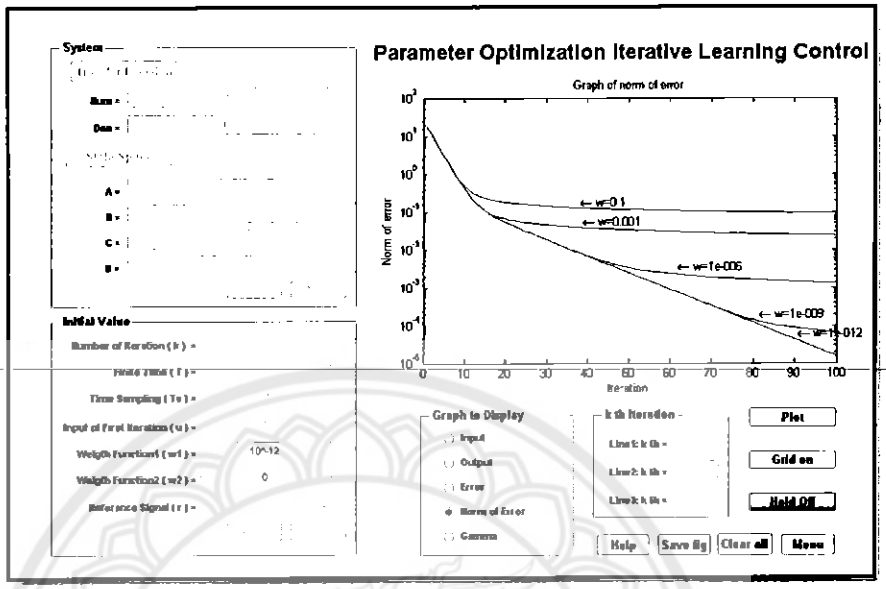
รูปที่ 4.12 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับหนึ่งที่ได้จากวิธี POILC

5. จากระบบอันดับหนึ่ง จะได้กราฟแสดงค่าตัวแปรอิสระดังรูปที่ 4.13



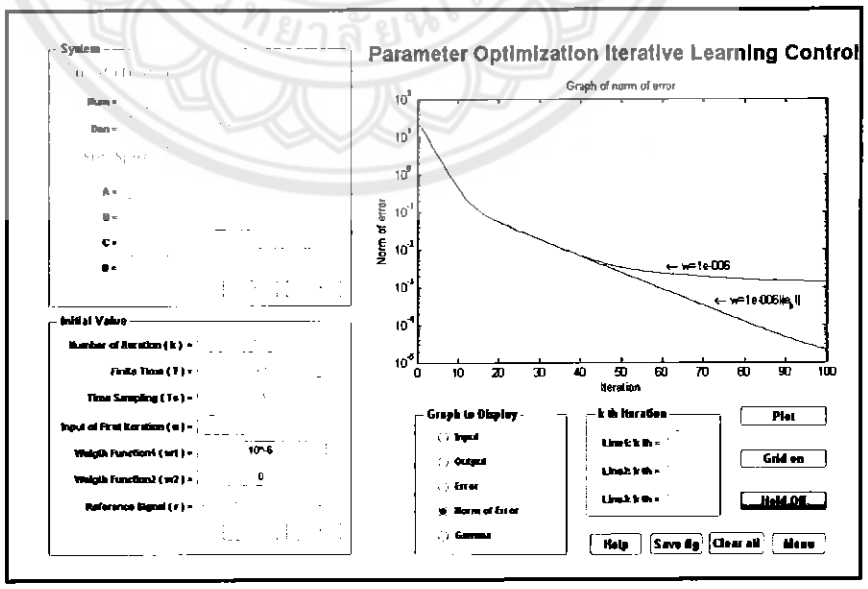
รูปที่ 4.13 กราฟแสดงค่าตัวแปรอิสระ (Gamma) ของระบบอันดับหนึ่งที่ได้จากวิธี POILC

6. จากระบบอันดับหนึ่ง จะได้กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดในกรณีที่มีการแปรเปลี่ยนตัวแปรค่าถ่วงน้ำหนักแบบคงที่ดังรูปที่ 4.14



รูปที่ 4.14 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับหนึ่งในกรณีที่มีการแปรเปลี่ยนตัวแปรค่าถ่วงน้ำหนักแบบคงที่ที่ได้จากวิธี POILC

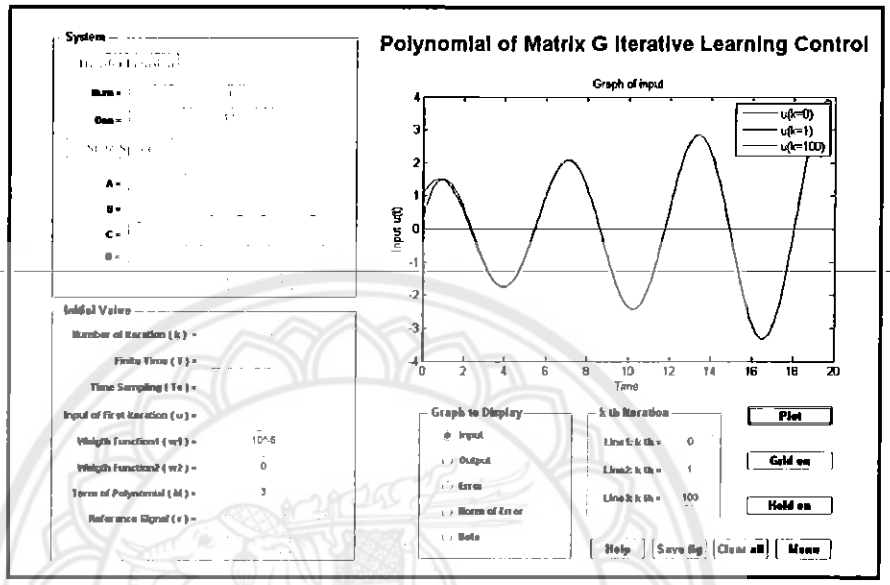
7. จากระบบอันดับหนึ่ง จะได้กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดในกรณีที่มีตัวแปรค่าถ่วงน้ำหนักแบบคงที่และแบบปรับค่าได้ดังรูปที่ 4.15



รูปที่ 4.15 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับหนึ่งในกรณีที่มีตัวแปรค่าถ่วงน้ำหนักแบบคงที่และแบบปรับค่าได้ที่ได้จากวิธี POILC

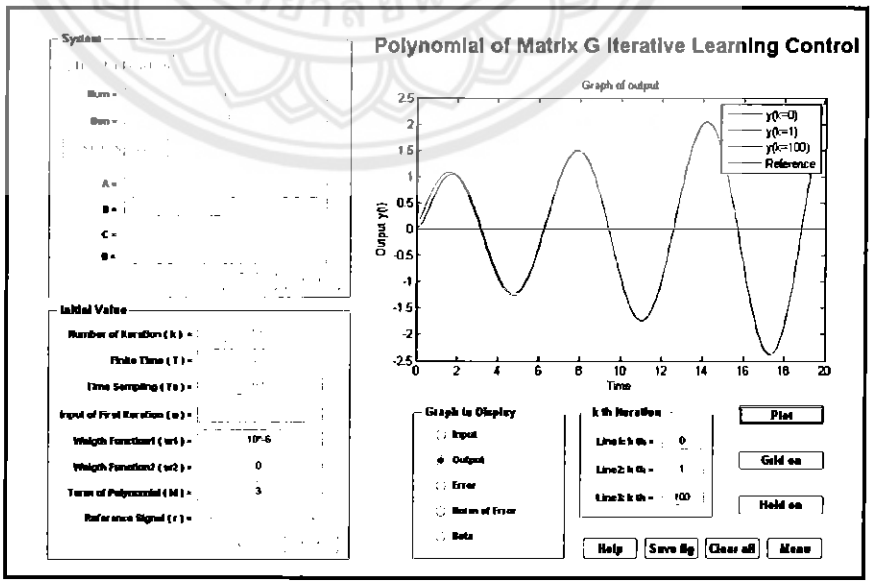
ค. วิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้พหุนามของเมทริกซ์ G (Polynomial of matrix G iterative learning control)

1. จากระบบอันดับหนึ่ง จะได้กราฟแสดงค่าสัญญาณขาเข้าดังรูปที่ 4.16



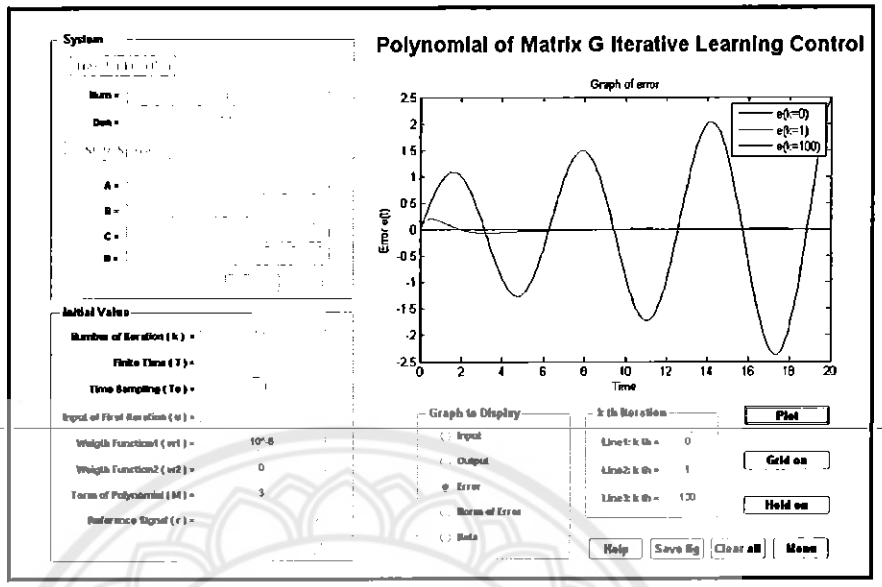
รูปที่ 4.16 กราฟแสดงค่าสัญญาณขาเข้าของระบบอันดับหนึ่งที่ได้จาก Polynomial of matrix G ILC

2. จากระบบอันดับหนึ่ง จะได้กราฟแสดงค่าสัญญาณขาออกดังรูปที่ 4.17



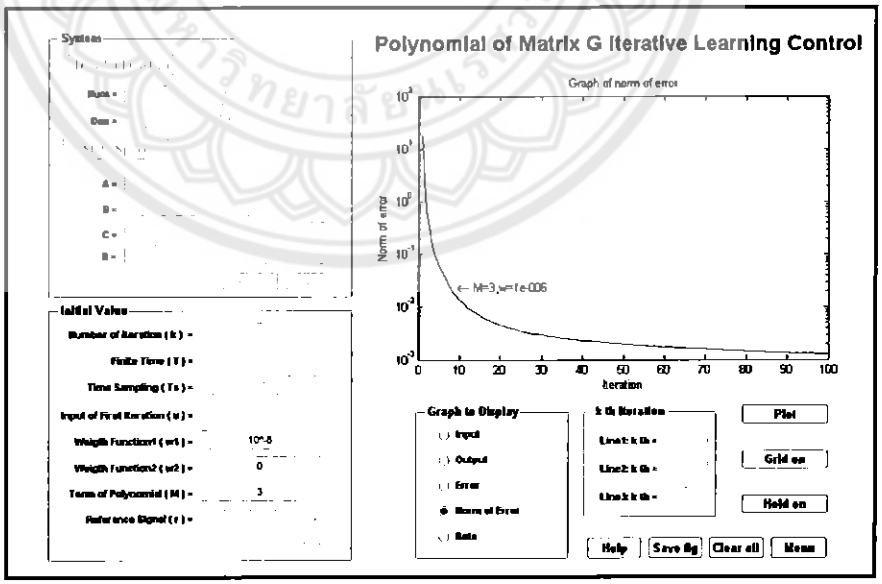
รูปที่ 4.17 กราฟแสดงค่าสัญญาณขาออกของระบบอันดับหนึ่งที่ได้จากวิธี Polynomial of matrix G ILC

3. จากระบบอันดับหนึ่ง จะได้กราฟแสดงค่าสัญญาณความผิดพลาดดังรูปที่ 4.18



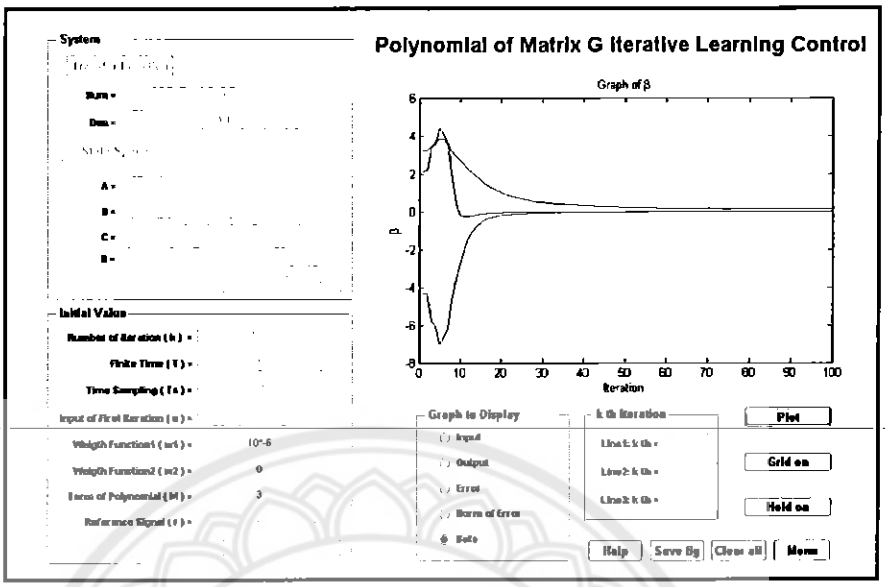
รูปที่ 4.18 กราฟแสดงค่าสัญญาณความผิดพลาดของระบบอันดับหนึ่งที่ได้จากวิธี Polynomial of matrix G ILC

4. จากระบบอันดับหนึ่ง จะได้กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดดังรูปที่ 4.19



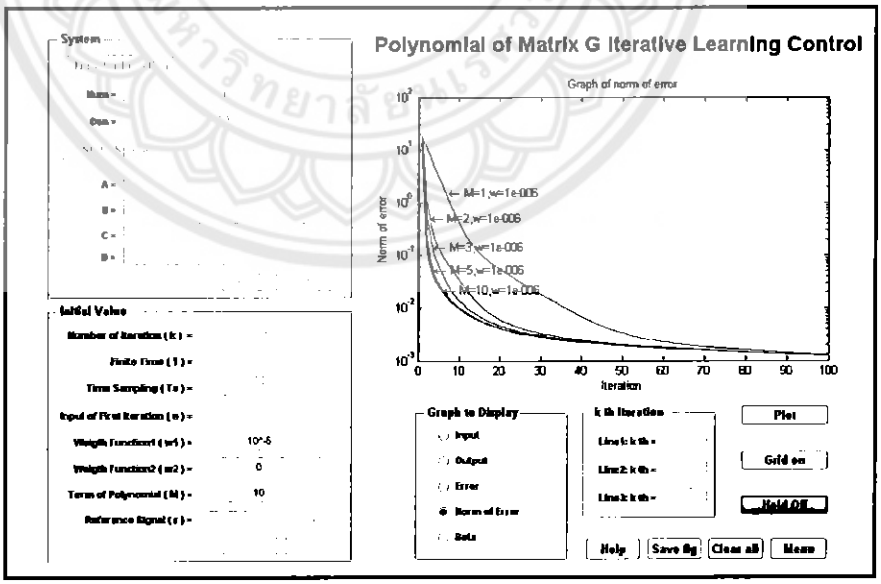
รูปที่ 4.19 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับหนึ่งที่ได้จากวิธี Polynomial of matrix G ILC

5. จากระบบอันดับหนึ่ง จะได้กราฟแสดงค่าตัวแปรอิสระดังรูปที่ 4.20



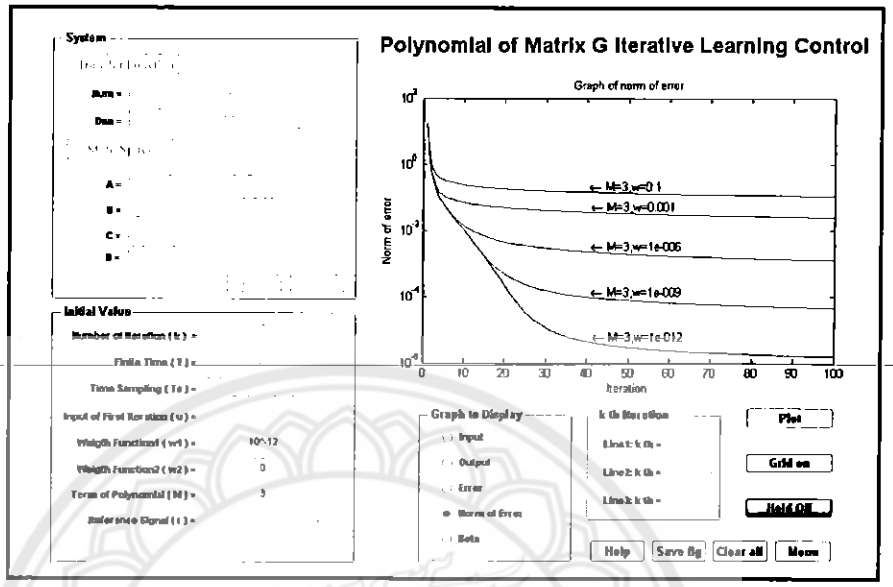
รูปที่ 4.20 กราฟแสดงค่าตัวแปรอิสระ (Beta) ของระบบอันดับหนึ่งที่ได้จากวิธี Polynomial of matrix G ILC

6. จากระบบอันดับหนึ่ง จะได้กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดในกรณีที่มีการแปรเปลี่ยนจำนวนพจน์ของพหุนามดังรูปที่ 4.21



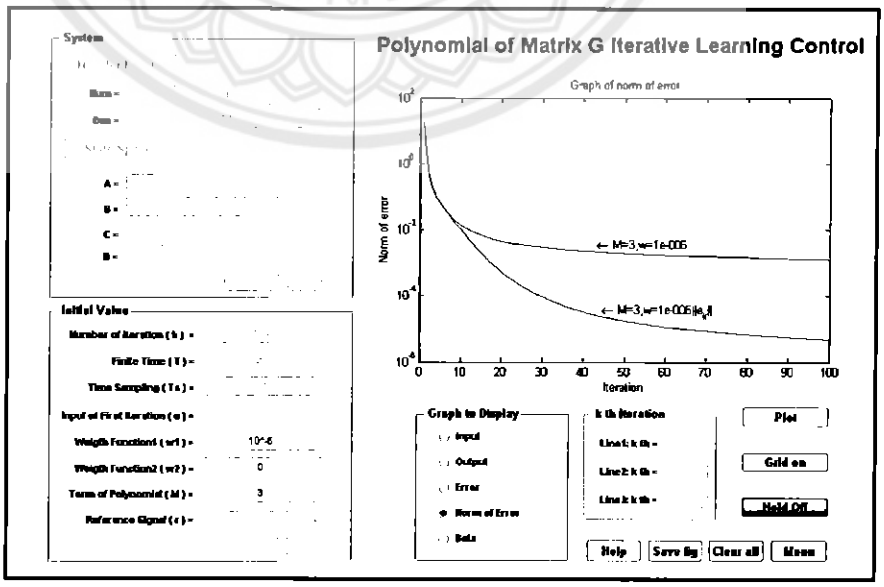
รูปที่ 4.21 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับหนึ่งในกรณีที่มีการแปรเปลี่ยนจำนวนพจน์ของพหุนามที่ได้จากวิธี Polynomial of matrix G ILC

7. จากระบบอันดับหนึ่ง จะได้กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดในกรณีที่มีการแปรเปลี่ยนตัวแปรค่าถ่วงน้ำหนักแบบคงที่ดังรูปที่ 4.22



รูปที่ 4.22 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับหนึ่งในกรณีที่มีการแปรเปลี่ยนตัวแปรค่าถ่วงน้ำหนักแบบคงที่ที่ได้จากวิธี Polynomial of matrix G ILC

8. จากระบบอันดับหนึ่ง จะได้กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดในกรณีที่มีตัวแปรค่าถ่วงน้ำหนักแบบคงที่และแบบปรับค่าได้ดังรูปที่ 4.23



รูปที่ 4.23 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับหนึ่งในกรณีที่มีตัวแปรค่าถ่วงน้ำหนักแบบคงที่และแบบปรับค่าได้ที่ได้จากวิธี Polynomial of matrix G ILC

ระบบอันดับสอง (Second order system)

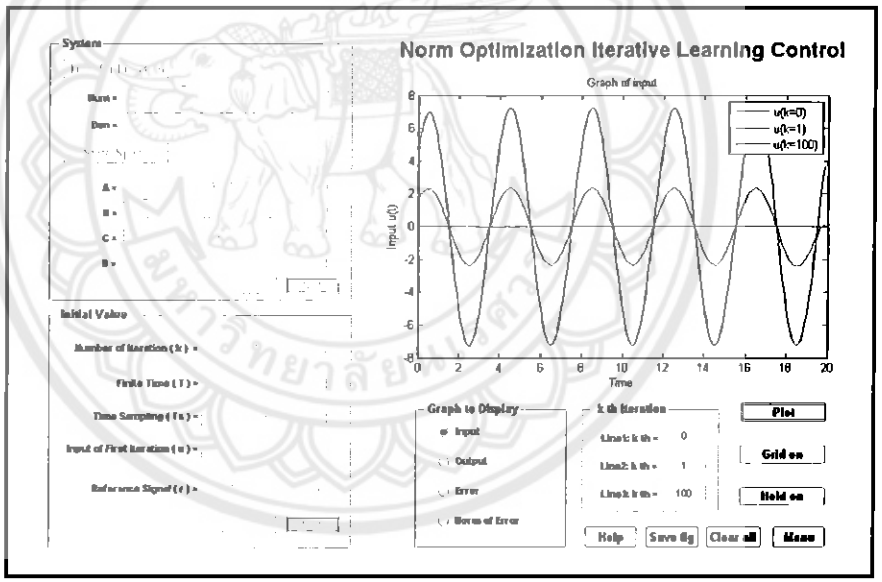
ปริภูมิสถานะของระบบอันดับสองเท่ากับ

$$A = \begin{bmatrix} -4 & -3 \\ 1 & 0 \end{bmatrix}, B = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, C = [2 \quad 3], D = [0]$$

ใช้เวลาในการชักตัวอย่าง $Ts = 0.1$ วินาที ทำงานในช่วงเวลา $T = [0,20]$ มีค่าตัวแปรค่าถ่วงน้ำหนักแบบคงที่ $w = 10^{-6}$ จำนวนพจน์ของพหุนามเท่ากับ $M = 3$ และมีการวนรอบจำนวน $k = 100$ ครั้ง โดยเลือกสัญญาณอ้างอิง $r = 5\sin(0.5\pi t)$ จะได้ผลการทดสอบดังนี้

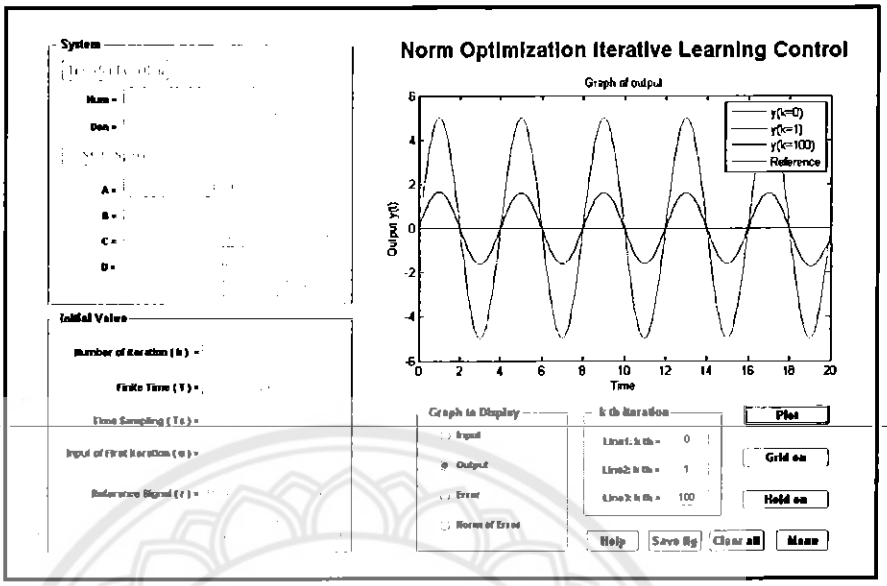
ก. วิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้ค่าประจำนอร์มที่เหมาะสมที่สุด (Norm optimization iterative learning control)

1. จากระบบอันดับสอง จะได้กราฟแสดงค่าสัญญาณขาเข้าดังรูปที่ 4.24



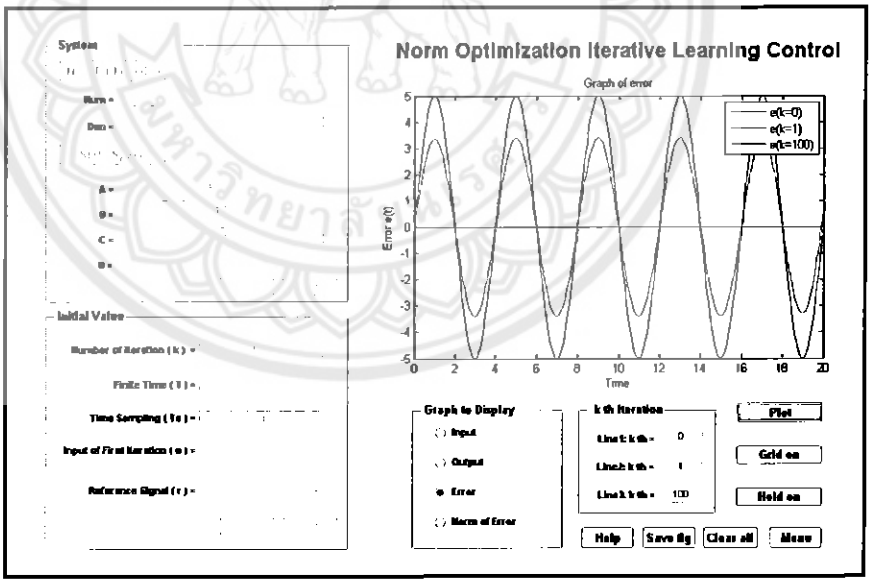
รูปที่ 4.24 กราฟแสดงค่าสัญญาณขาเข้าของระบบอันดับสองที่ได้จากวิธี NOILC

2. จากระบบอันดับสอง จะได้กราฟแสดงค่าสัญญาณขาออกดังรูปที่ 4.25



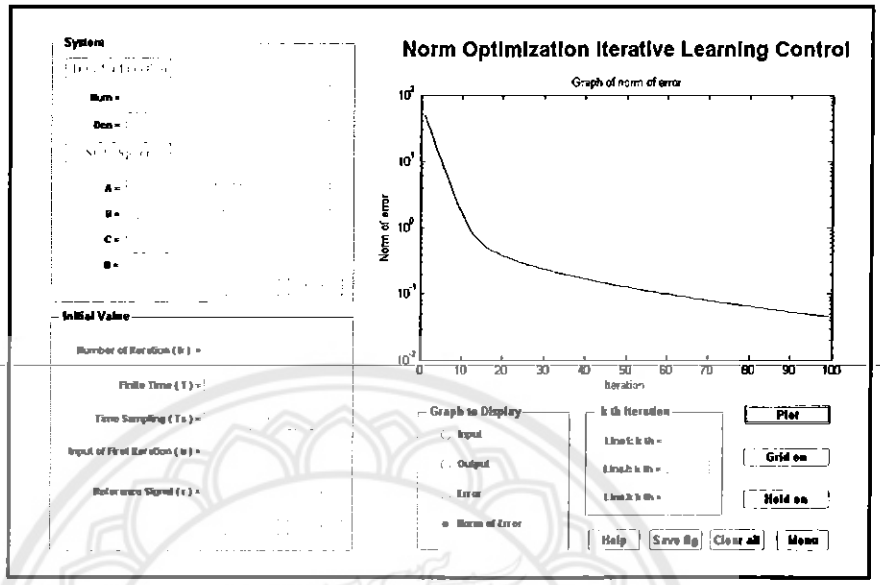
รูปที่ 4.25 กราฟแสดงค่าสัญญาณขาออกของระบบอันดับสองที่ได้จากวิธี NOILC

3. จากระบบอันดับสอง จะได้กราฟแสดงค่าสัญญาณความผิดพลาดดังรูปที่ 4.26



รูปที่ 4.26 กราฟแสดงค่าสัญญาณความผิดพลาดของระบบอันดับสองที่ได้จากวิธี NOILC

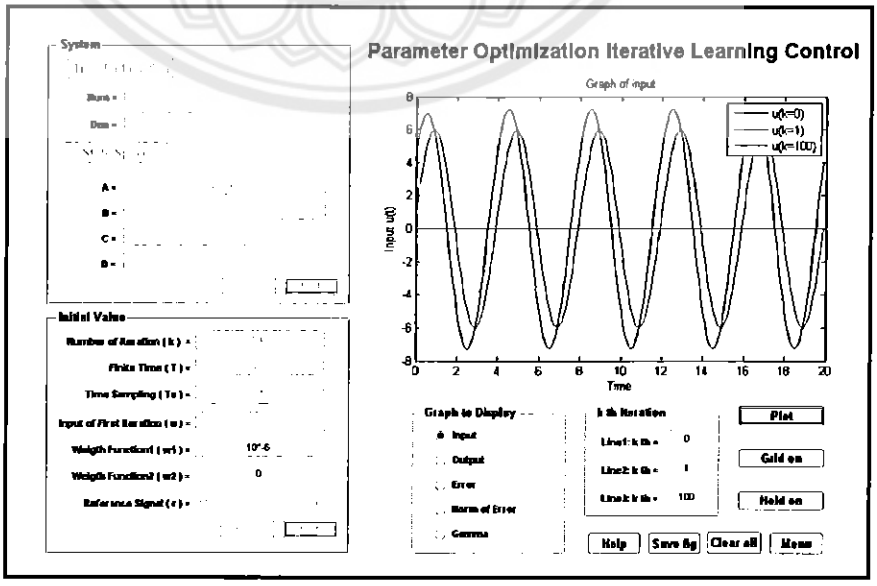
4. จากระบบอันดับสอง จะได้กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดดังรูปที่ 4.27



รูปที่ 4.27 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับสองที่ได้จากวิธี NOILC

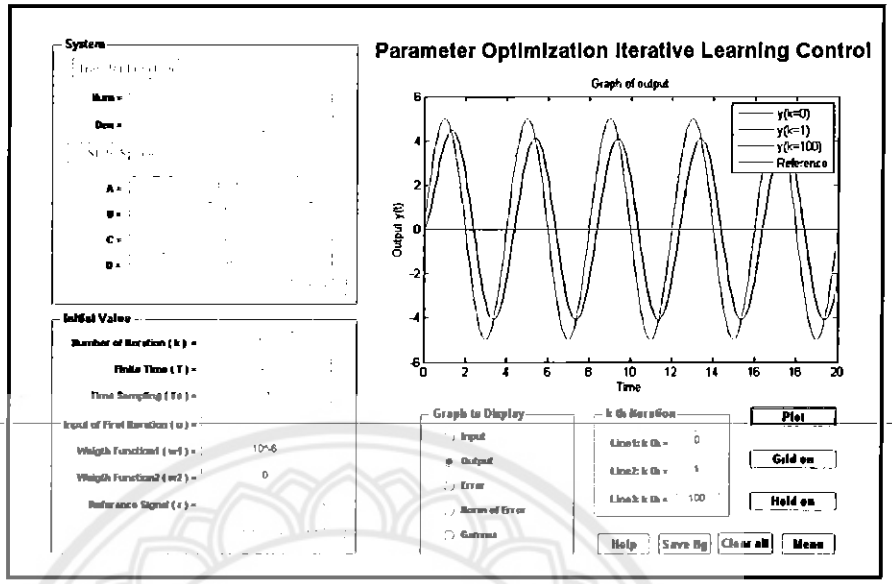
ข. วิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้ตัวแปรที่เหมาะสมที่สุด (Parameter optimization iterative learning control)

1. จากระบบอันดับสอง จะได้กราฟแสดงค่าสัญญาณขาเข้าดังรูปที่ 4.28



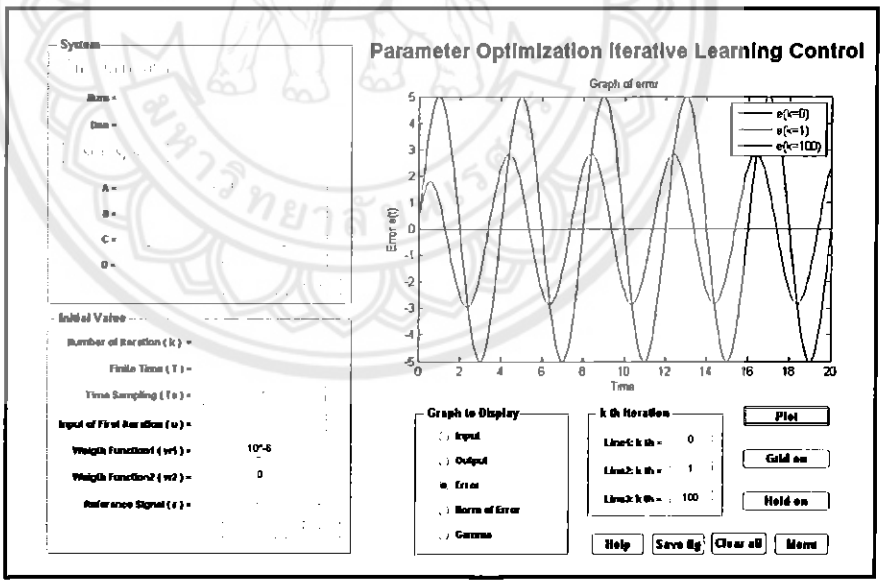
รูปที่ 4.28 กราฟแสดงค่าสัญญาณขาเข้าของระบบอันดับสองที่ได้จากวิธี POILC

2. จากระบบอันดับสอง จะได้กราฟแสดงค่าสัญญาณขาออกดังรูปที่ 4.28



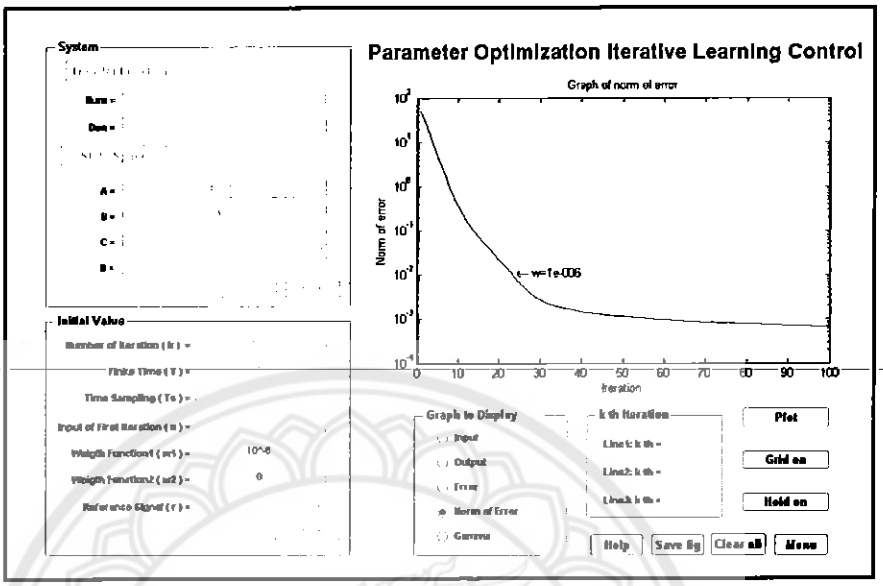
รูปที่ 4.29 กราฟแสดงค่าสัญญาณขาออกของระบบอันดับสองที่ได้จากวิธี POILC

3. จากระบบอันดับสอง จะได้กราฟแสดงค่าสัญญาณความผิดพลาดดังรูปที่ 4.30



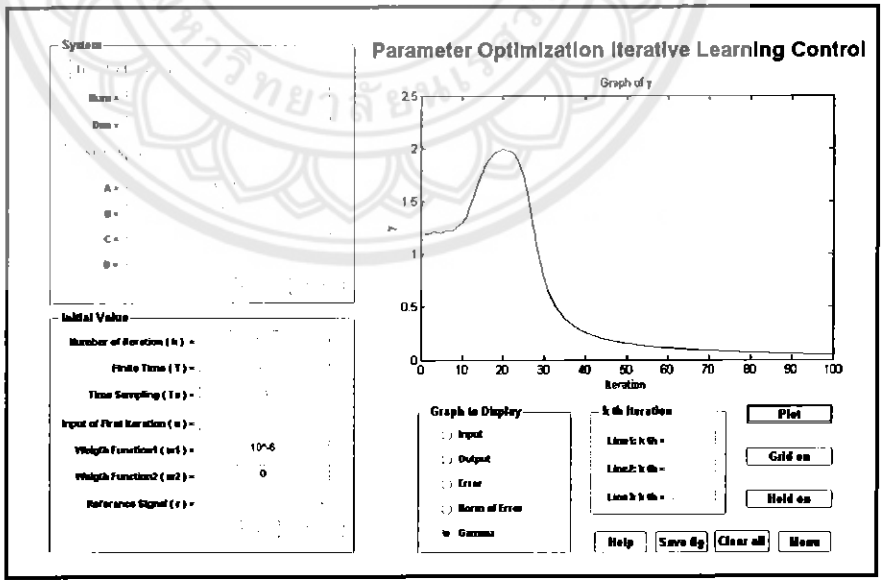
รูปที่ 4.30 กราฟแสดงค่าสัญญาณความผิดพลาดของระบบอันดับสองที่ได้จากวิธี POILC

4. จากระบบอันดับสอง จะได้กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดดังรูปที่ 4.31



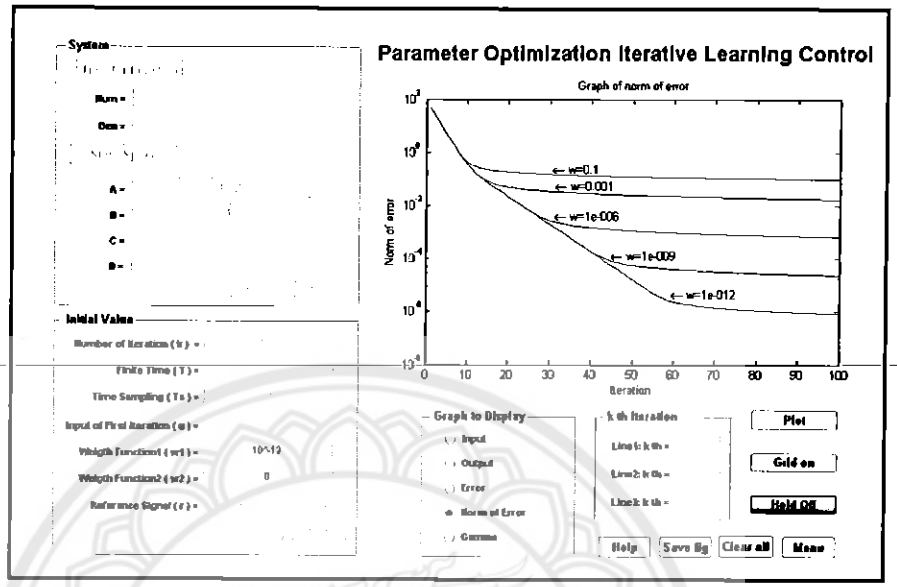
รูปที่ 4.31 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับสองที่ได้จากวิธี POILC

5. จากระบบอันดับสอง จะได้กราฟแสดงค่าตัวแปรอิสระดังรูปที่ 4.32



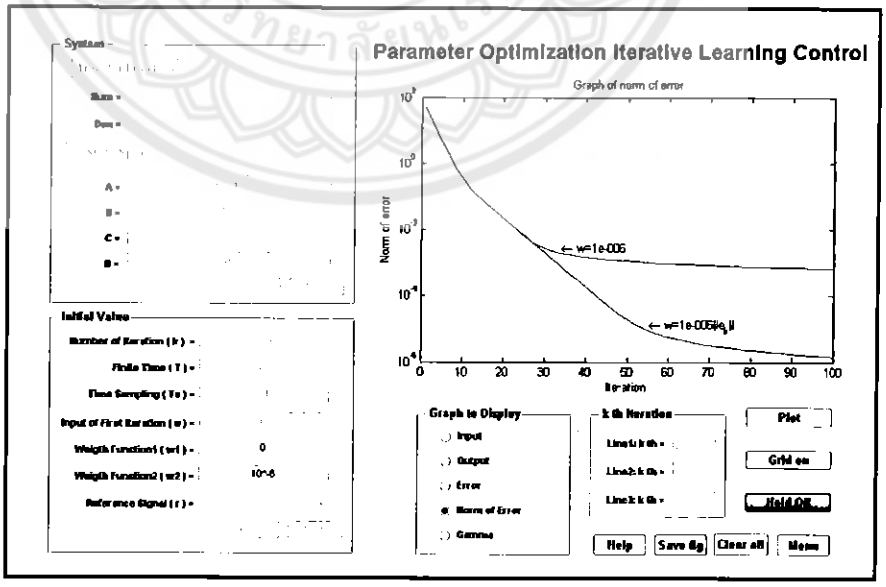
รูปที่ 4.32 กราฟแสดงค่าตัวแปรอิสระ (Gamma) ของระบบอันดับสองที่ได้จากวิธี POILC

6. จากระบบอันดับสอง จะได้กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดในกรณีที่มีการแปรเปลี่ยนตัวแปรค่าถ่วงน้ำหนักแบบคงที่ดังรูปที่ 4.33



รูปที่ 4.33 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับสองในกรณีที่มีการแปรเปลี่ยนตัวแปรค่าถ่วงน้ำหนักแบบคงที่ที่ได้จากวิธี POILC

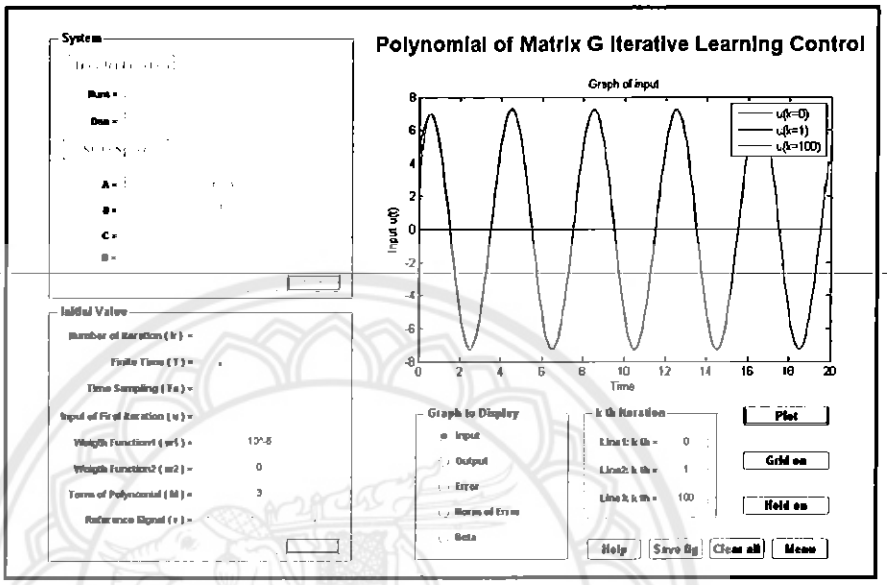
7. จากระบบอันดับสอง จะได้กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดในกรณีที่มีตัวแปรค่าถ่วงน้ำหนักแบบคงที่และแบบปรับค่าได้ดังรูปที่ 4.33



รูปที่ 4.34 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับสองในกรณีที่มีตัวแปรค่าถ่วงน้ำหนักแบบคงที่และแบบปรับค่าได้ที่ได้จากวิธี POILC

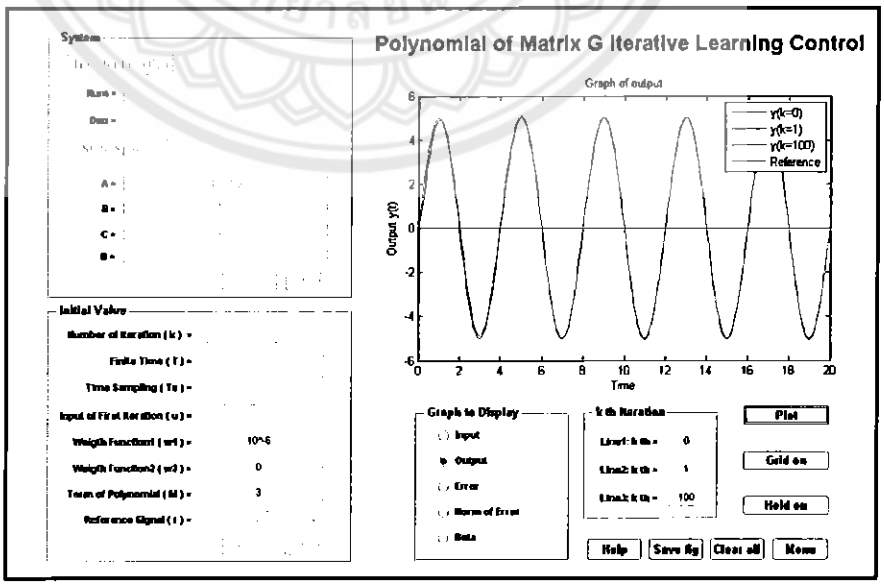
ค. วิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้พหุนามของเมทริกซ์ G (Polynomial of matrix G iterative learning control)

1. จากระบบอันดับสอง จะได้กราฟแสดงค่าสัญญาณขาเข้าดังรูปที่ 4.35



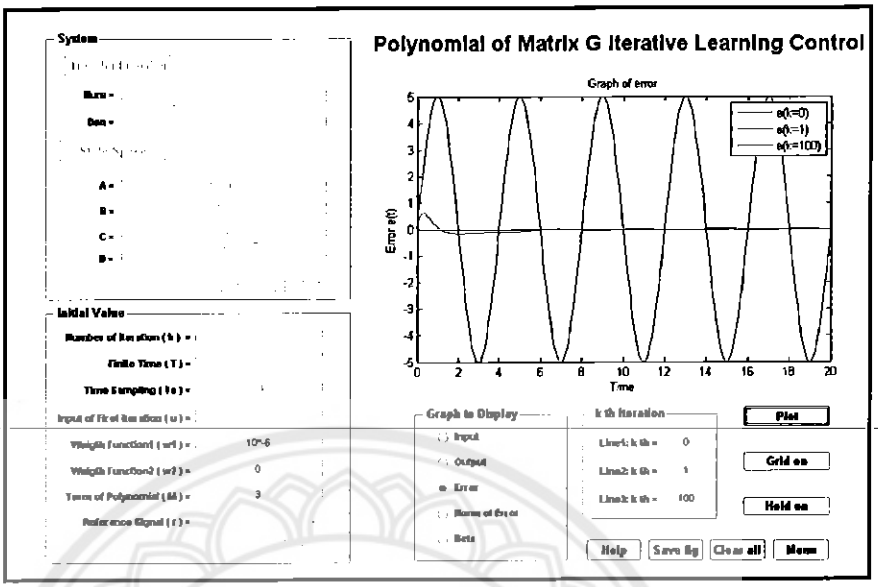
รูปที่ 4.35 กราฟแสดงค่าสัญญาณขาเข้าของระบบอันดับสองที่ได้จากวิธี Polynomial of matrix G ILC

2. จากระบบอันดับสองจะได้กราฟแสดงค่าสัญญาณขาออกดังรูปที่ 4.36



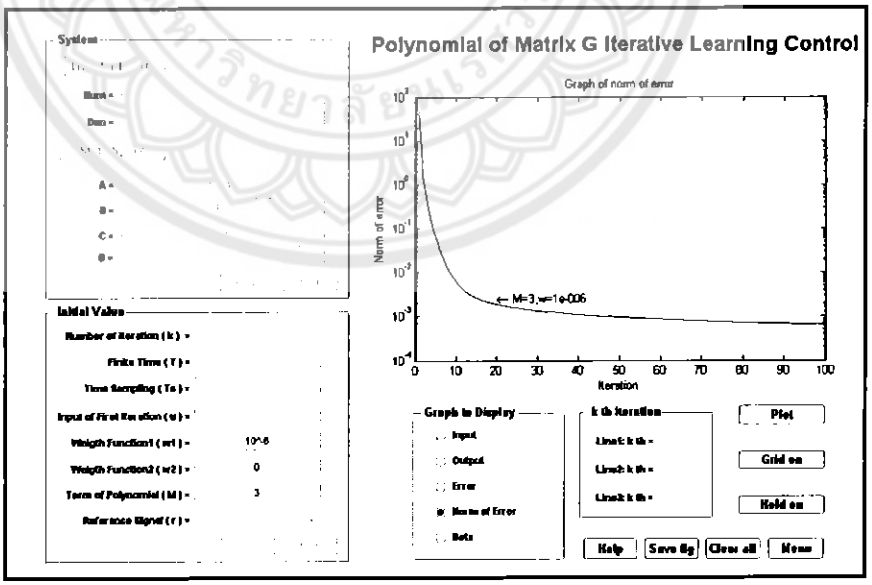
รูปที่ 4.36 กราฟแสดงค่าสัญญาณขาออกของระบบอันดับสองที่ได้จากวิธี Polynomial of matrix G ILC

3. จากระบบอันดับสอง จะได้กราฟแสดงค่าสัญญาณความผิดพลาดดังรูปที่ 4.37



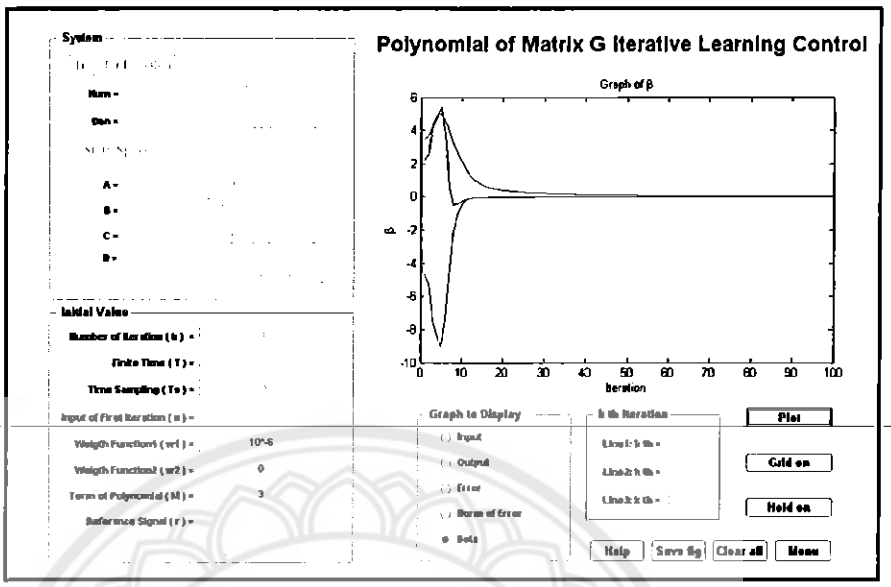
รูปที่ 4.37 กราฟแสดงค่าสัญญาณความผิดพลาดของระบบอันดับสองที่ได้จากวิธี Polynomial of matrix G ILC

4. จากระบบอันดับสอง จะได้กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดดังรูปที่ 4.38



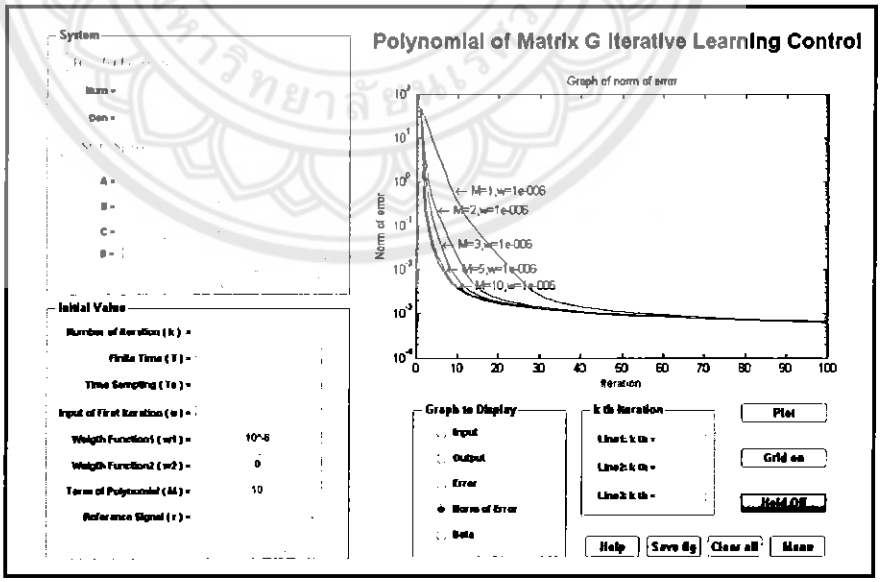
รูปที่ 4.38 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับสองที่ได้จากวิธี Polynomial of matrix G ILC

5. จากระบบอันดับสอง จะได้กราฟแสดงค่าตัวแปรอิสระดังรูปที่ 4.39



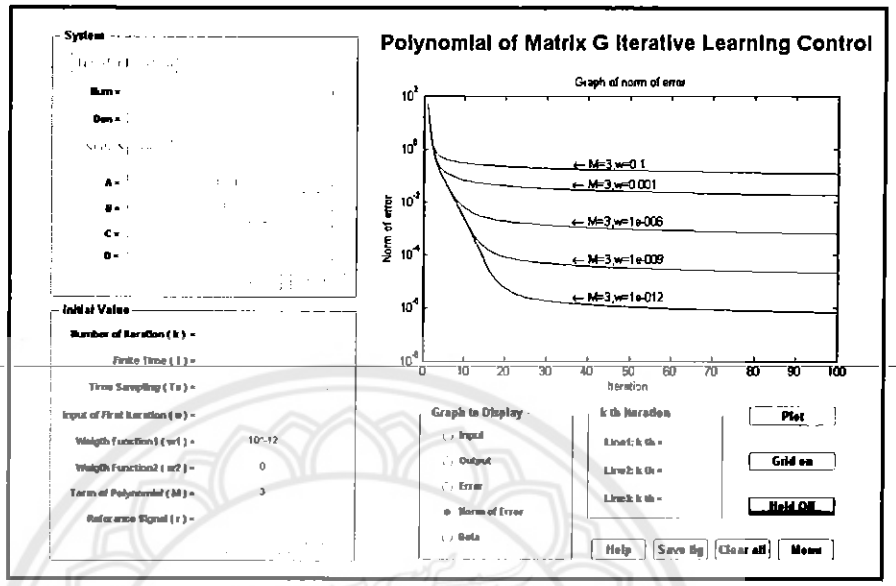
รูปที่ 4.39 กราฟแสดงค่าตัวแปรอิสระ (Beta) ของระบบอันดับสองที่ได้จากวิธี Polynomial of matrix G ILC

6. จากระบบอันดับสอง จะได้กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดในกรณีที่มีการแปรเปลี่ยนจำนวนพจน์ของพหุนามดังรูปที่ 4.40



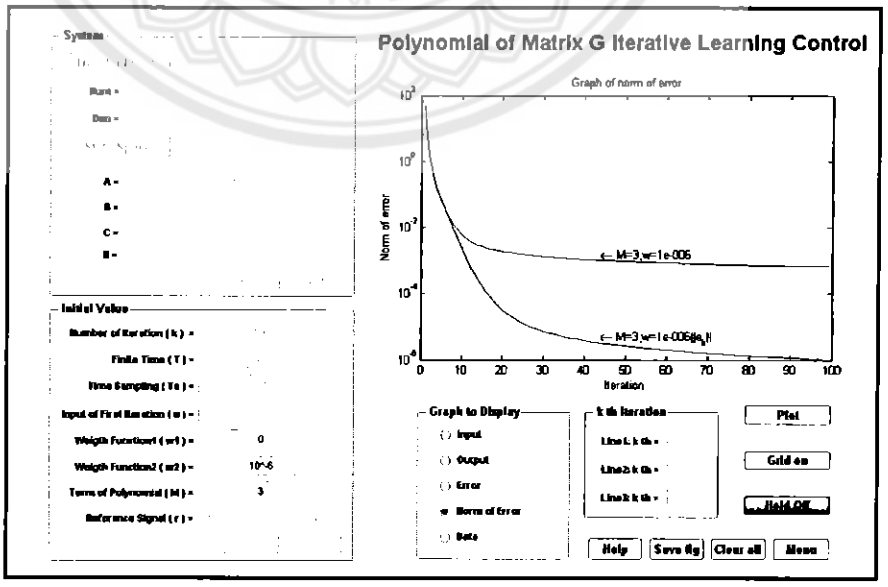
รูปที่ 4.40 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับสองในกรณีที่มีการแปรเปลี่ยนจำนวนพจน์ของพหุนามที่ได้จากวิธี Polynomial of matrix G ILC

7. จากระบบอันดับสอง จะได้กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดในกรณีที่มีการแปรเปลี่ยนตัวแปรค่าถ่วงน้ำหนักแบบคงที่ดังรูปที่ 4.41



รูปที่ 4.41 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับสองในกรณีที่มีการแปรเปลี่ยนตัวแปรค่าถ่วงน้ำหนักแบบคงที่ที่ได้จากวิธี Polynomial of matrix G ILC)

8. จากระบบอันดับสอง จะได้กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดในกรณีที่มีตัวแปรค่าถ่วงน้ำหนักแบบคงที่และแบบปรับค่าได้ดังรูปที่ 4.42



รูปที่ 4.42 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบอันดับสองในกรณีที่มีตัวแปรค่าถ่วงน้ำหนักแบบคงที่และแบบปรับค่าได้ที่ได้จากวิธี Polynomial of matrix G ILC

ระบบใช้งานจริง

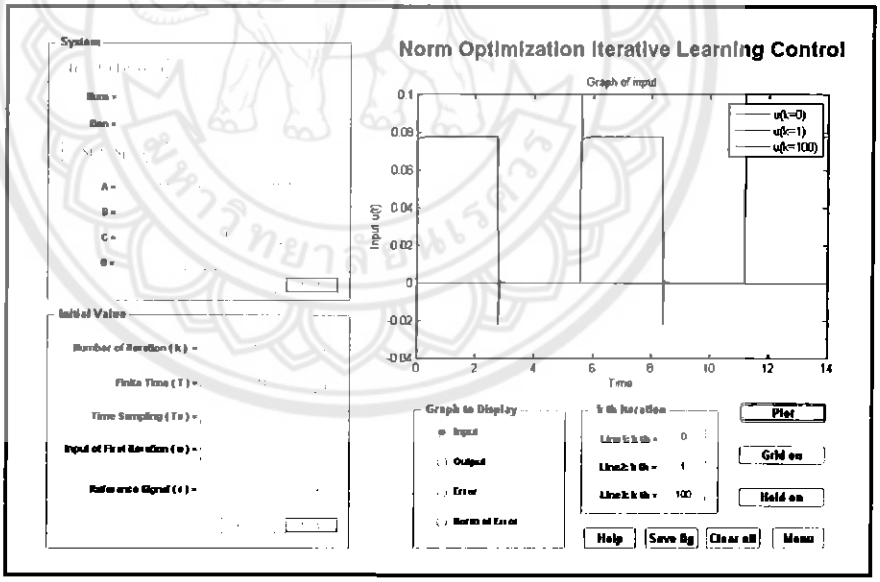
ระบบที่เลือกนำมาทดสอบเป็นระบบของตัวกระทำการของหุ่นยนต์พูม่า 560 (PUMA-560) สามารถเขียนในรูปปริภูมิสถานะได้ดังนี้

$$A = \begin{bmatrix} -6.6526 & 1.2125 \\ -7.5782 & -51.6671 \end{bmatrix}, B = \begin{bmatrix} -0.9616 \\ 679.3 \end{bmatrix}, C = [0 \quad 1], D = [0]$$

กำหนดให้เวลาในการชักตัวอย่าง $Ts = 0.028$ วินาที ทำงานในช่วงเวลา $T = [0,14]$ มีค่าตัวแปรค่าถ่วงน้ำหนักแบบคงที่ $w = 10^{-6}$ จำนวนพจน์ของพหุนามเท่ากับ $M = 3$ และมีการวนรอบจำนวน $k = 100$ ครั้ง โดยเลือกสัญญาณอ้างอิง $r =$ square wave จะได้ผลการทดสอบดังนี้

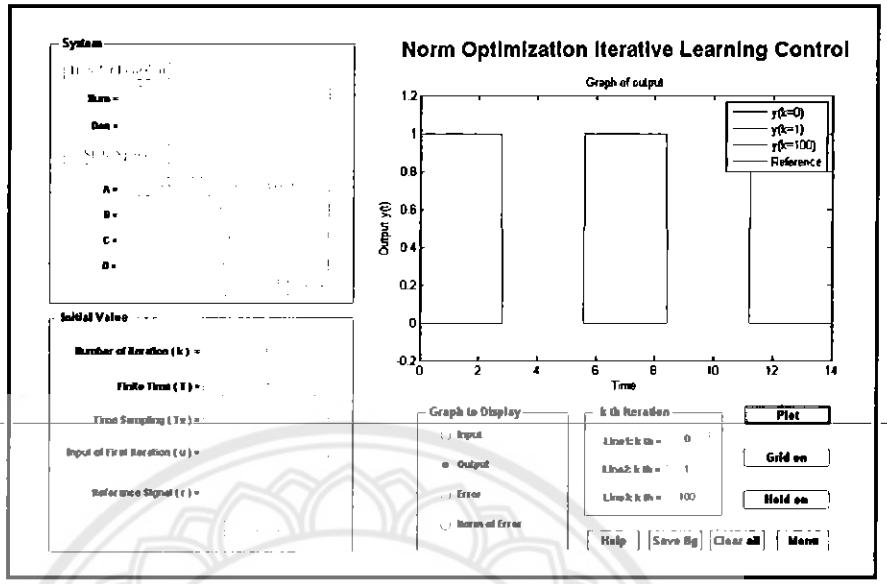
ก. วิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้ค่าประจำนอร์มที่เหมาะสมที่สุด (Norm optimization iterative learning control)

1. จากระบบใช้งานจริง จะได้กราฟแสดงค่าสัญญาณขาเข้าดังรูปที่ 4.43



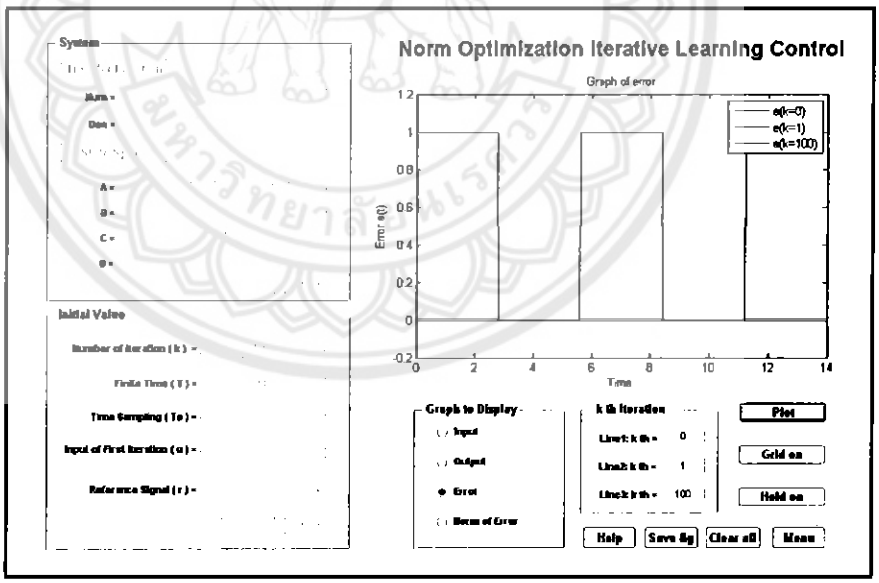
รูปที่ 4.43 กราฟแสดงค่าสัญญาณขาเข้าของระบบใช้งานจริงที่ได้จากวิธี NOILC

2. จากระบบใช้งานจริง จะได้กราฟแสดงค่าสัญญาณขาออกดังรูปที่ 4.44



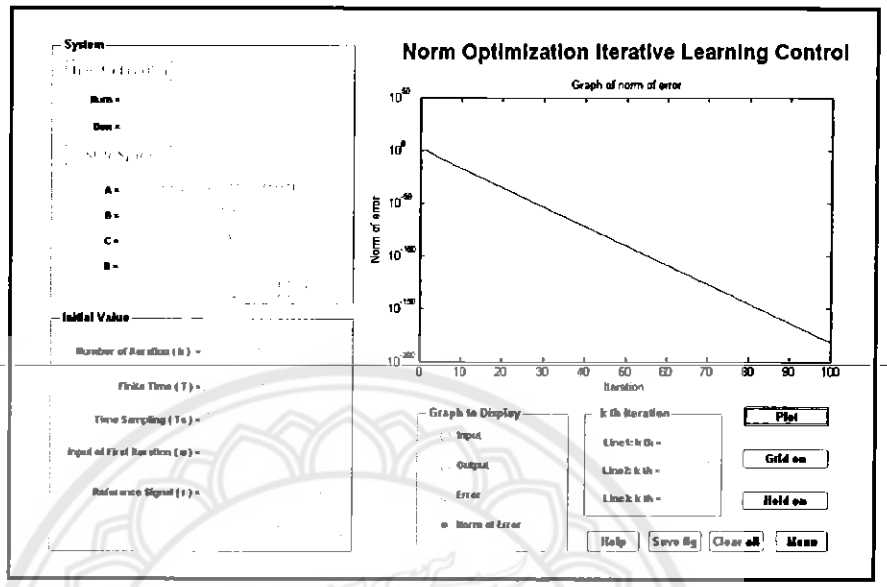
รูปที่ 4.44 กราฟแสดงค่าสัญญาณขาออกของระบบใช้งานจริงที่ได้จากวิธี NOILC

3. จากระบบใช้งานจริง จะได้กราฟแสดงค่าสัญญาณความผิดพลาดดังรูปที่ 4.45



รูปที่ 4.45 กราฟแสดงค่าสัญญาณความผิดพลาดของระบบใช้งานจริงที่ได้จากวิธี NOILC

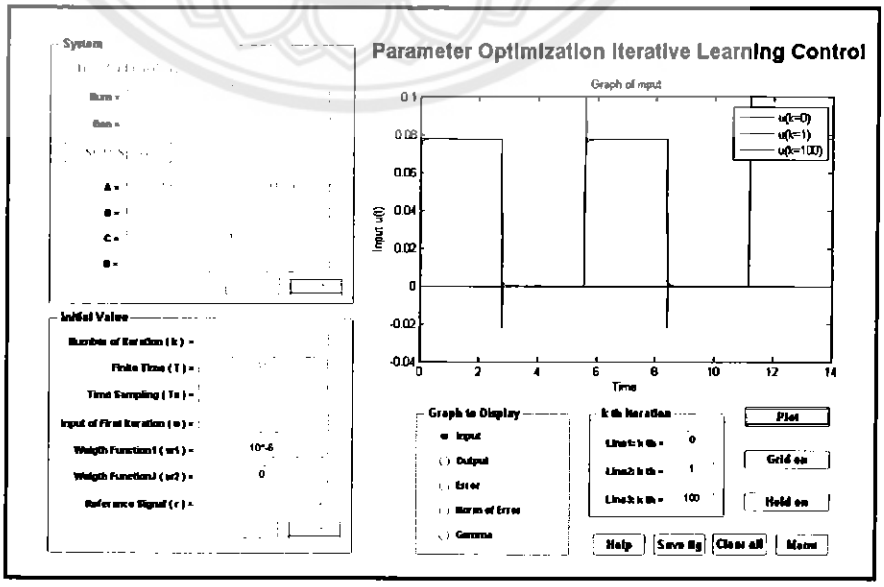
4. จากระบบใช้งานจริง จะได้กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดดังรูปที่ 4.46



รูปที่ 4.46 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบใช้งานจริงที่ได้จากวิธี NOILC

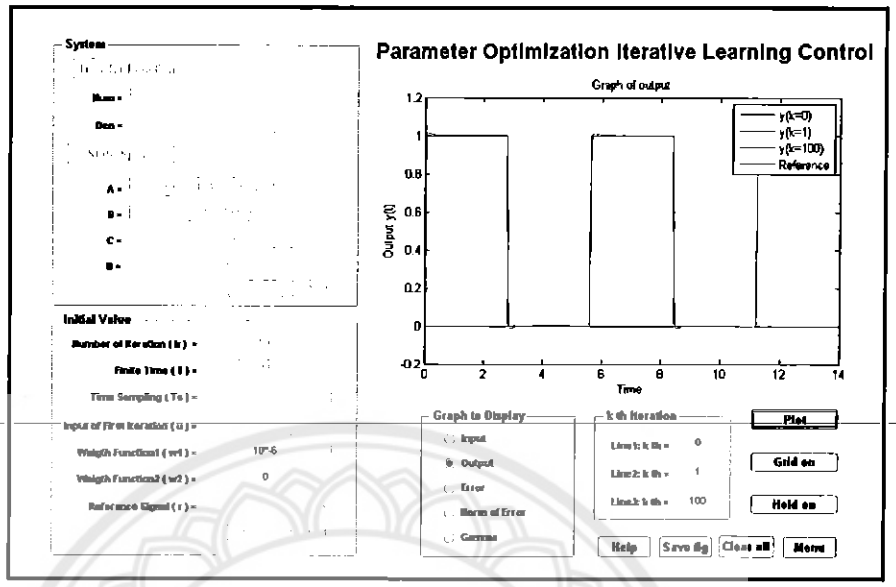
ข. วิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้ตัวแปรที่เหมาะสมที่สุด (Parameter optimization iterative learning control)

1. จากระบบใช้งานจริง จะได้กราฟแสดงค่าสัญญาณขาเข้าดังรูปที่ 4.47



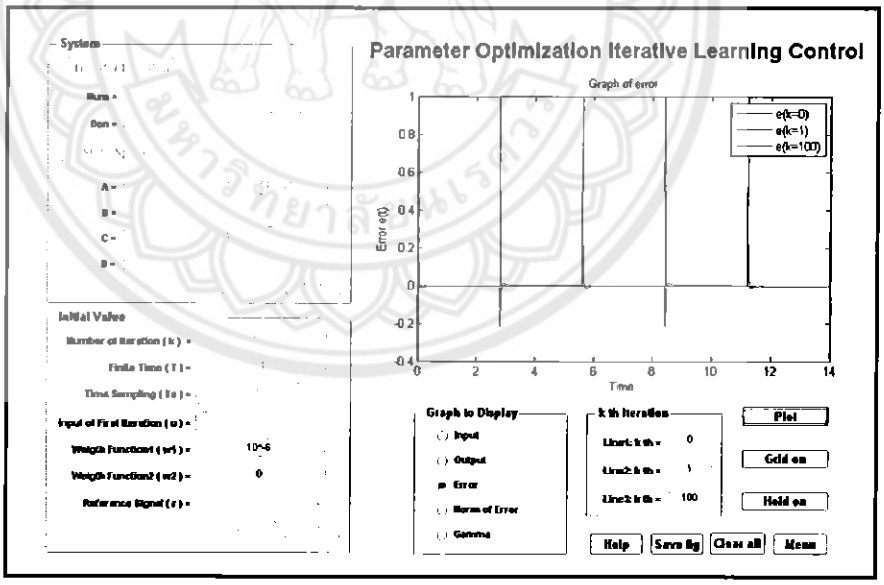
รูปที่ 4.47 กราฟแสดงค่าสัญญาณขาเข้าของระบบใช้งานจริงที่ได้จากวิธี POILC

2. จากระบบใช้งานจริง จะได้กราฟแสดงค่าสัญญาณขาออก ดังรูปที่ 4.48



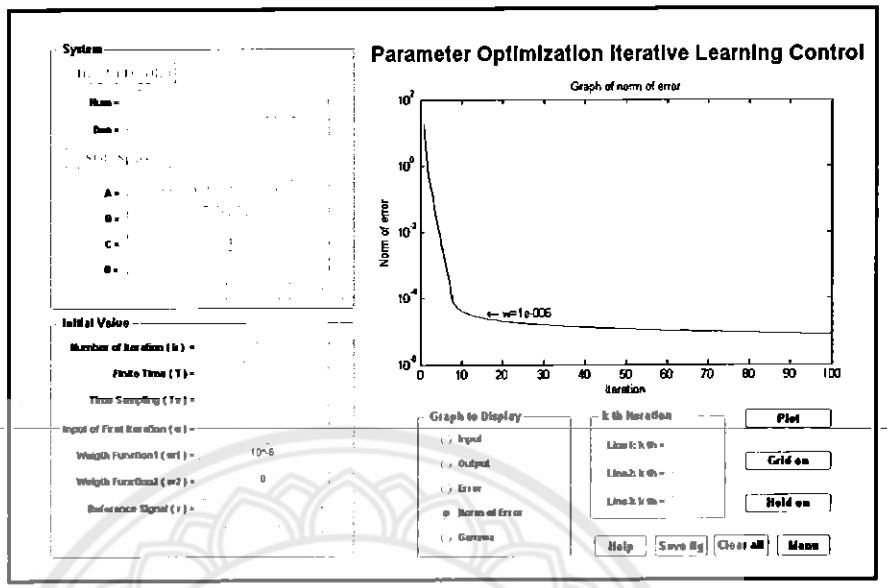
รูปที่ 4.48 กราฟแสดงค่าสัญญาณขาออกของระบบใช้งานจริงที่ได้จากวิธี POILC

3. จากระบบใช้งานจริง จะได้กราฟแสดงค่าสัญญาณความผิดพลาดดังรูปที่ 4.49



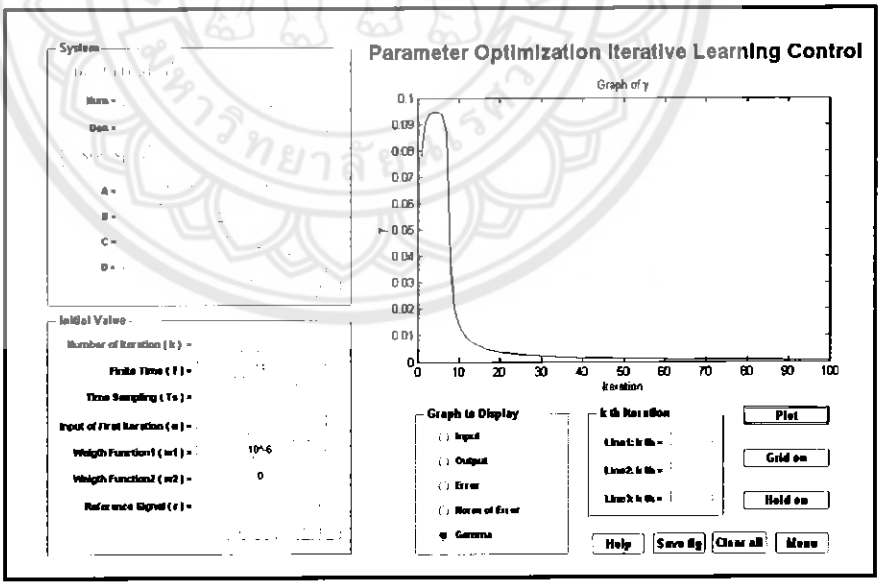
รูปที่ 4.49 กราฟแสดงค่าสัญญาณความผิดพลาดของระบบใช้งานจริงที่ได้จากวิธี POILC

4. จากระบบใช้งานจริง จะได้กราฟแสดงค่าสัญญาณความผิดพลาดดังรูปที่ 4.50



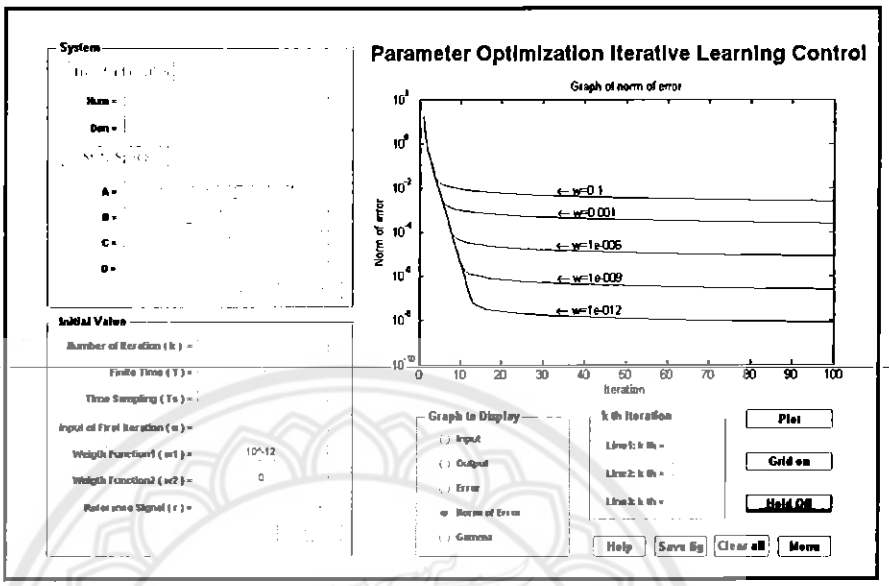
รูปที่ 4.50 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบใช้งานจริงที่ได้จากวิธี POILC

5. จากระบบใช้งานจริง จะได้กราฟแสดงค่าตัวแปรอิสระดังรูปที่ 4.51



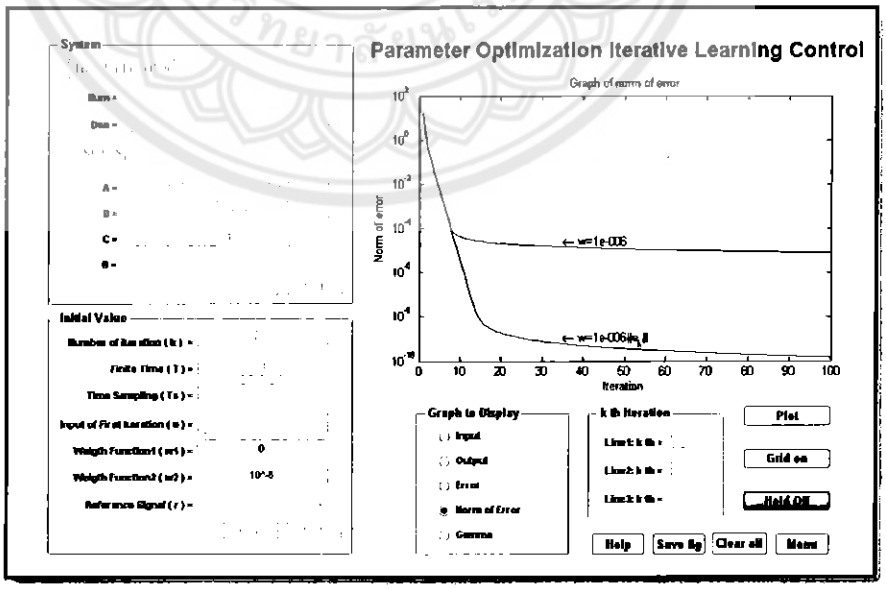
รูปที่ 4.51 กราฟแสดงค่าตัวแปรอิสระ (Gamma) ของระบบใช้งานจริงที่ได้จากวิธี POILC

6. จากระบบใช้งานจริง จะได้กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดในกรณีที่มีการแปรเปลี่ยนตัวแปรค่าถ่วงน้ำหนักแบบคงที่ดังรูปที่ 4.52



รูปที่ 4.52 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบใช้งานจริงในกรณีที่มีการแปรเปลี่ยนตัวแปรค่าถ่วงน้ำหนักแบบคงที่ที่ได้จากวิธี POILC

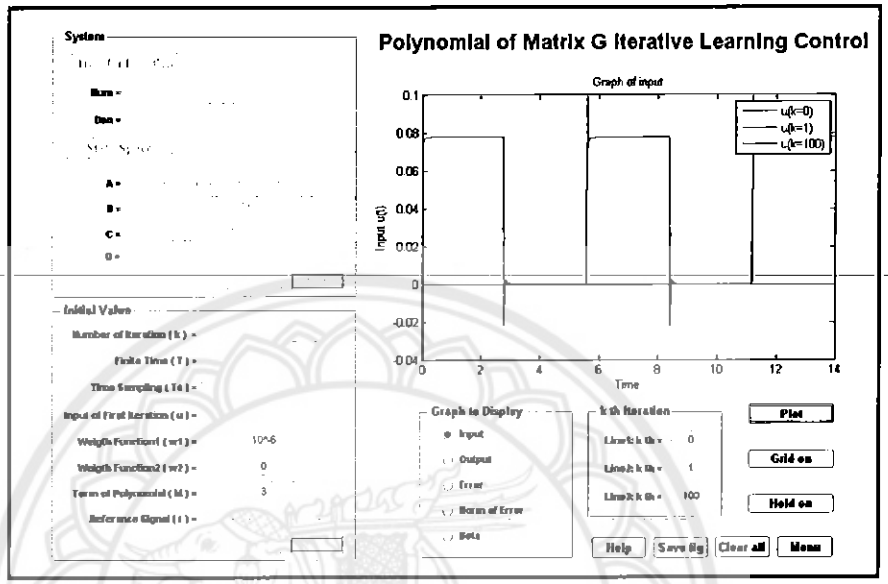
7. จากระบบใช้งานจริง จะได้กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดในกรณีที่มีตัวแปรค่าถ่วงน้ำหนักแบบคงที่และแบบปรับค่าได้ดังรูปที่ 4.53



รูปที่ 4.53 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบใช้งานจริงในกรณีที่มีตัวแปรค่าถ่วงน้ำหนักแบบคงที่และแบบปรับค่าได้ที่ได้จากวิธี POILC

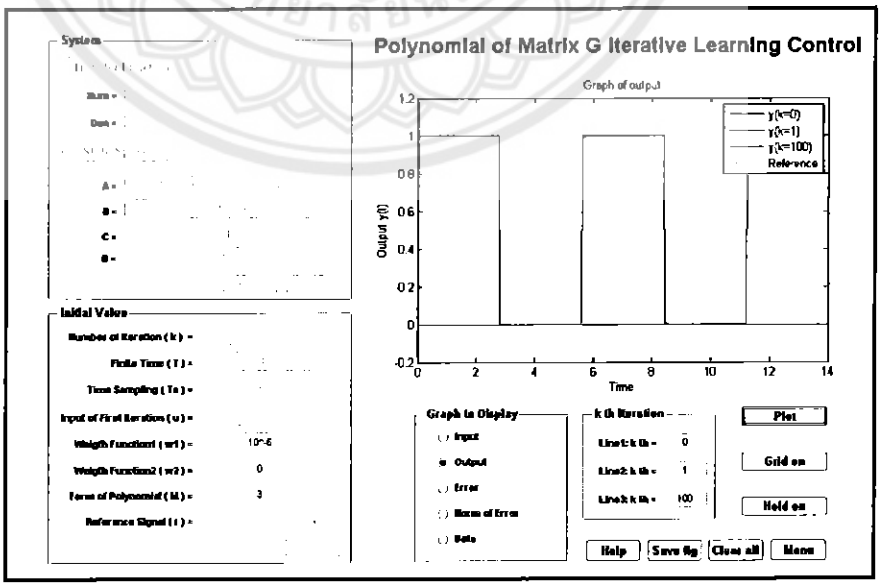
ค. วิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้พหุนามของเมทริกซ์ G (Polynomial of matrix G iterative learning control)

1. จากระบบใช้งานจริง จะได้กราฟแสดงค่าสัญญาณขาเข้าดังรูปที่ 4.54



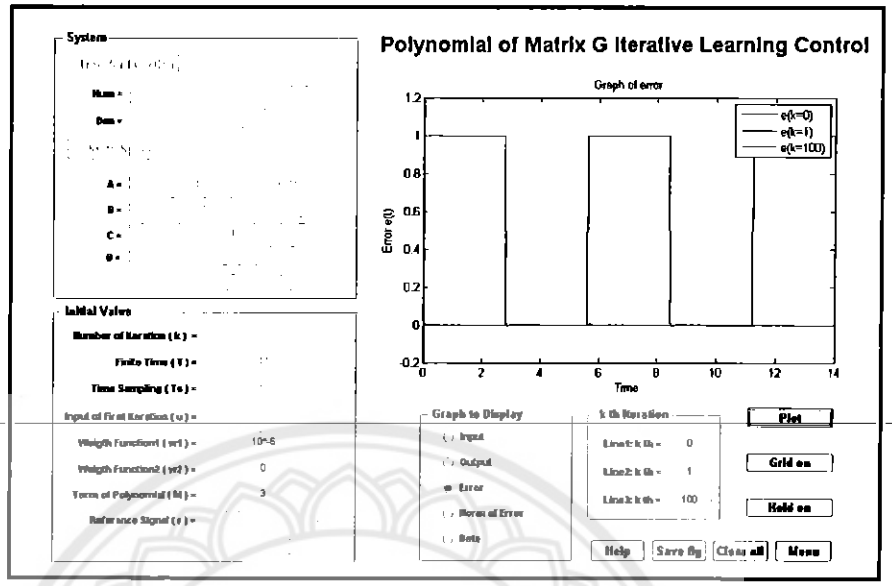
รูปที่ 4.54 กราฟแสดงค่าสัญญาณขาเข้าของระบบใช้งานจริงที่ได้จากวิธี Polynomial of matrix G ILC

2. จากระบบใช้งานจริง จะได้กราฟแสดงค่าสัญญาณขาออกดังรูปที่ 4.55



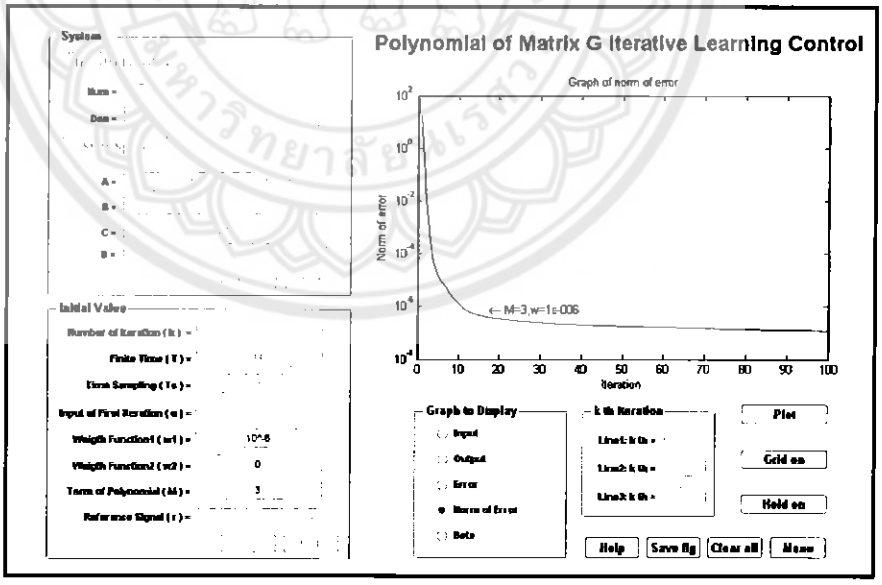
รูปที่ 4.55 กราฟแสดงค่าสัญญาณขาออกของระบบใช้งานจริงที่ได้จากวิธี Polynomial of matrix G ILC

3. จากระบบใช้งานจริง จะได้กราฟแสดงค่าสัญญาณความผิดพลาดดังรูปที่ 4.56



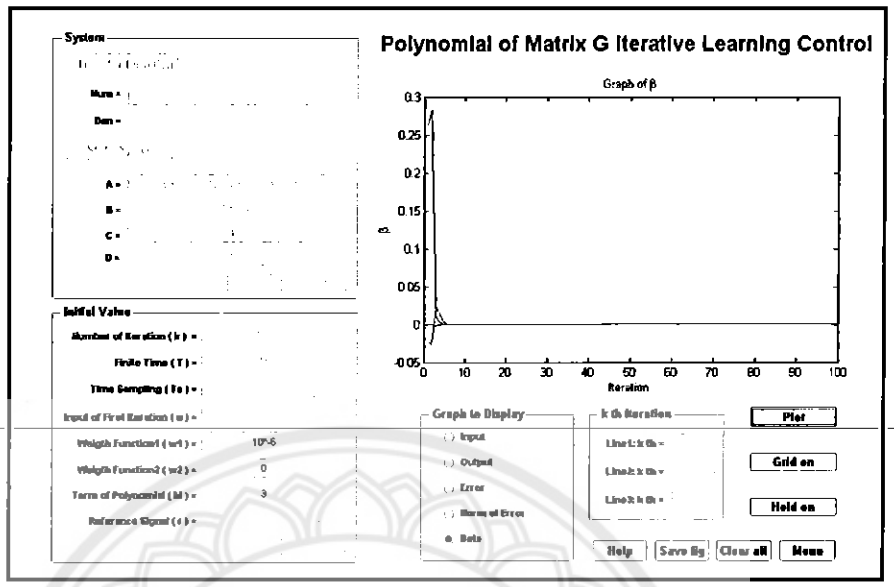
รูปที่ 4.56 กราฟแสดงค่าสัญญาณความผิดพลาดของระบบใช้งานจริงที่ได้จากวิธี Polynomial of matrix G ILC

4. จากระบบใช้งานจริง จะได้กราฟแสดงค่าสัญญาณความผิดพลาดดังรูปที่ 4.57



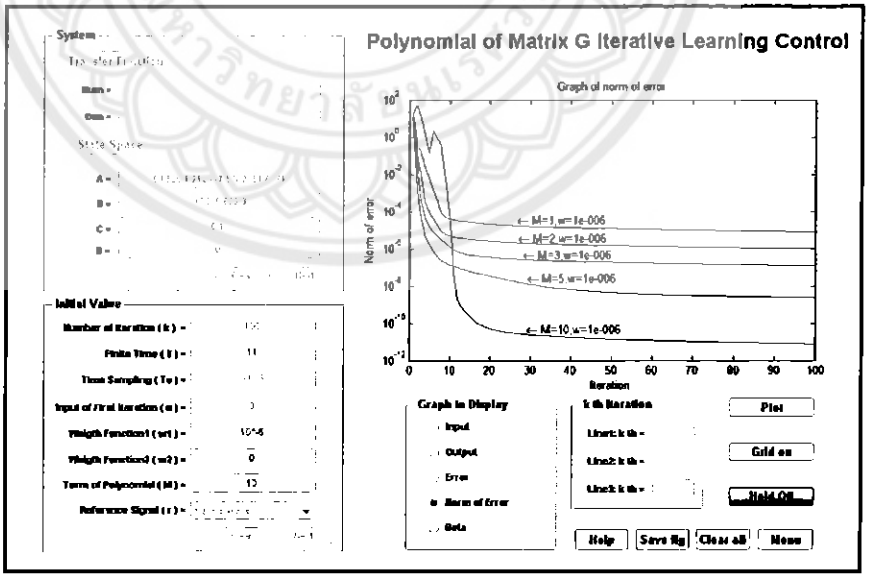
รูปที่ 4.57 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบใช้งานจริงที่ได้จากวิธี Polynomial of matrix G ILC

5. จากระบบใช้งานจริง จะได้กราฟแสดงค่าตัวแปรอิสระดังรูปที่ 4.58



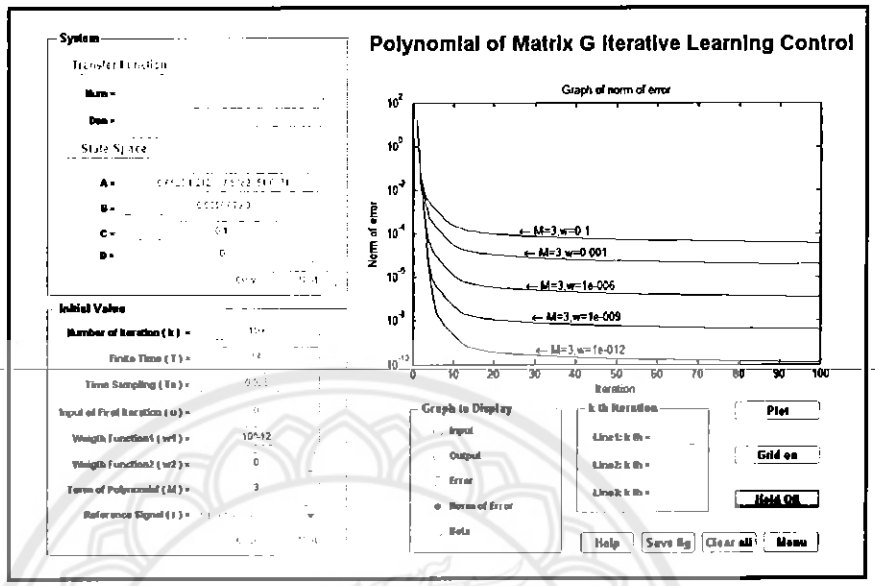
รูปที่ 4.58 กราฟแสดงค่าตัวแปรอิสระ (Beta) ของระบบใช้งานจริงที่ได้จากวิธี Polynomial of matrix G ILC

6. จากระบบใช้งานจริง จะได้กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดในกรณีที่มีการแปรเปลี่ยนจำนวนพจน์ของพหุนามดังรูปที่ 4.59



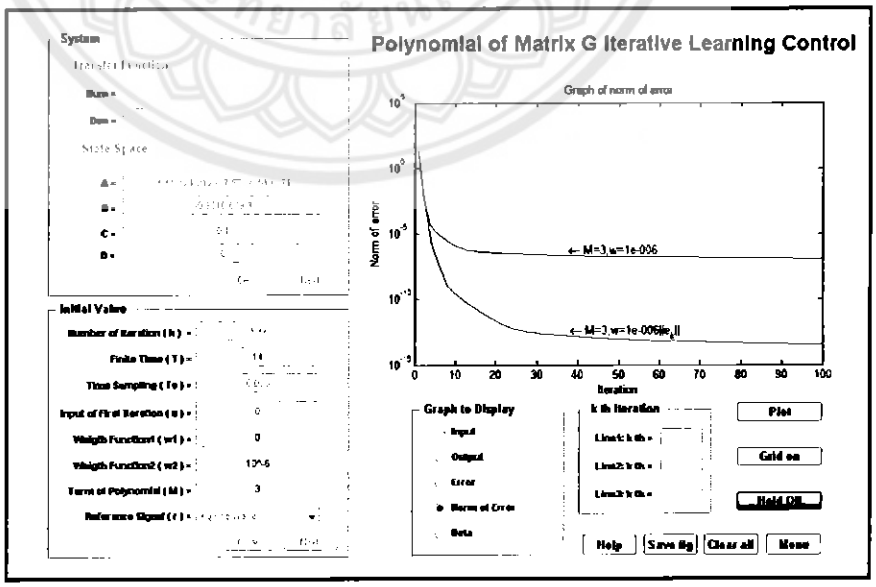
รูปที่ 4.59 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบใช้งานจริงในกรณีที่มีการแปรเปลี่ยนจำนวนพจน์ของพหุนามที่ได้จากวิธี Polynomial of matrix G ILC

7. จากระบบใช้งานจริง จะได้กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดในกรณีที่มีการแปรเปลี่ยนตัวแปรค่าถ่วงน้ำหนักแบบคงที่ดังรูปที่ 4.60



รูปที่ 4.60 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบใช้งานจริงในกรณีที่มีการแปรเปลี่ยนตัวแปรค่าถ่วงน้ำหนักแบบคงที่ที่ได้จากวิธี Polynomial of matrix G ILC

8. จากระบบใช้งานจริง จะได้กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดในกรณีที่มีตัวแปรค่าถ่วงน้ำหนักแบบคงที่และแบบปรับค่าได้ดังรูปที่ 4.61



รูปที่ 4.61 กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบใช้งานจริงในกรณีที่มีตัวแปรค่าถ่วงน้ำหนักแบบคงที่และแบบปรับค่าได้ที่ได้จากวิธี Polynomial of matrix G ILC

บทที่ 5
สรุปผลการดำเนินโครงงาน

5.1 สรุปและวิเคราะห์ผล

โครงงานนี้เป็นการนำเสนอการใช้โปรแกรม MATLAB ในสร้างโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้สำหรับวิธีการควบคุมแบบเรียนรู้ซ้ำ เพื่ออำนวยความสะดวกแก่ผู้ที่สนใจและต้องการศึกษาวิธีการควบคุมแบบเรียนรู้ซ้ำ โดยโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้สำหรับวิธีการควบคุมแบบเรียนรู้ซ้ำนี้จะมีการรับค่าในรูปแบบเมทริกซ์หรือสเกลาร์และมีการแสดงผลตอบสนองในรูปแบบของกราฟต่างๆ ดังนี้

1. Norm Optimization Iterative Learning Control (วิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้ค่าประจำนอร์มที่เหมาะสมที่สุด)

   - กราฟแสดงค่าสัญญาณขาเข้าของระบบ
   - กราฟแสดงค่าสัญญาณขาออกของระบบ
   - กราฟแสดงค่าสัญญาณความผิดพลาดของระบบ
   - กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบ

2. Parameter Optimization Iterative Learning Control (วิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้ตัวแปรที่เหมาะสมที่สุด)

   - กราฟแสดงค่าสัญญาณขาเข้าของระบบ
   - กราฟแสดงค่าสัญญาณขาออกของระบบ
   - กราฟแสดงค่าสัญญาณความผิดพลาดของระบบ
   - กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบ
   - กราฟแสดงค่าตัวแปรอิสระ (Gamma) ของระบบ

3. Polynomial of Matrix G Iterative Learning Control (วิธีการควบคุมแบบเรียนรู้ซ้ำโดยใช้พหุนามของเมตริกซ์ G)

   - กราฟแสดงค่าสัญญาณขาเข้าของระบบ
   - กราฟแสดงค่าสัญญาณขาออกของระบบ
   - กราฟแสดงค่าสัญญาณความผิดพลาดของระบบ
   - กราฟแสดงค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบ
   - กราฟแสดงค่าตัวแปรอิสระ (Beta) ของระบบ

จากการทดสอบการใช้งานของโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้สำหรับวิธีการควบคุมแบบเรียนรู้ซ้ำนี้ ด้วยระบบ 3 ระบบคือ ระบบอับดับหนึ่ง (First order system) ระบบอันดับสอง (Second order system) และระบบที่ใช้งานจริง พบว่าการแสดงผลตอบสนองในรูปแบบต่างๆของแต่ละวิธี มีความถูกต้องและแม่นยำ เมื่อเทียบกับการเขียนโค้ดคำนวณแบบปกติในโปรแกรม MATLAB ดังนั้นจึงสามารถนำโปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้สำหรับวิธีการควบคุมแบบเรียนรู้ซ้ำไปใช้งานจริงได้

5.2 ปัญหาที่พบ

1) ในกรณีที่มีการปรับค่าจำนวนพจน์ของพหุนามหรือตัวแปรค่าถ่วงน้ำหนักเพื่อเปรียบเทียบค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบ จำเป็นต้องกดปุ่ม hold on ก่อน จึงจะสามารถเก็บกราฟเดิมเพื่อนำมาเปรียบเทียบได้
2) ในกรณีที่กราฟของค่าประจำนอร์มของค่าสัญญาณความผิดพลาดมีค่าใกล้เคียงกัน อาจจะทำให้ตัวอักษรระบุค่าจำนวนพจน์ของพหุนามหรือตัวแปรค่าถ่วงน้ำหนักของเส้นกราฟมีการทับซ้อนกันได้
3) ในการเลือกจำนวนพจน์ของพหุนามสูงๆจะส่งผลให้การทำงานของโปรแกรมใช้เวลานาน

5.3 ข้อเสนอแนะ

1) ผู้ที่ใช้โปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้สำหรับวิธีการควบคุมแบบเรียนรู้ซ้ำนี้ ควรศึกษาคู่มือแนะนำการใช้งานโปรแกรมให้เข้าใจเสียก่อน เพื่อที่จะแสดงกราฟเปรียบเทียบค่าประจำนอร์มของค่าสัญญาณความผิดพลาดของระบบได้อย่างถูกต้อง
2) ในกรณีที่มีการเปรียบเทียบกราฟของค่าประจำนอร์มของค่าสัญญาณความผิดพลาด ผู้ใช้ควรเลือกระบุตำแหน่งของตัวอักษรจำนวนพจน์ของพหุนามหรือตัวแปรค่าถ่วงน้ำหนักให้ห่างกันพอสมควร เพื่อป้องกันการทับซ้อนกัน
3) ผู้ใช้โปรแกรมส่วนต่อประสานกราฟิกกับผู้ใช้สำหรับวิธีการควบคุมแบบเรียนรู้ซ้ำนี้ ควรเลือกจำนวนพจน์ของพหุนามที่ไม่สูงมากนัก

เอกสารอ้างอิง

[1] D. H. Owen and K. Feng. (2003). **Parameter optimization in iterative learning control.** International Journal of Control. 76(11).

[2] David Owen and Jari Hatonen. **Iterative learning control, delays and repetitive control.** University of Sheffield.

[3] R. W. Longman. (2000). **Iterative learning control and repetitive control for engineering practice.** International Journal of Control. 73(10):930-954.

[4] Scott T. Smith. (2006). **MATLAB Advance GUI Development.** United States of America: Dog Ear Publishing.

[5] Stephen J. Chapman. (2008). **MATLAB Programming for Engineers (Fourth Edition).** Australia: Thomson Learning.

[6] ดร.ปริญญา สงวนสัตย์. (2553). **คู่มือ MATLAB ฉบับสมบูรณ์.** นนทบุรี: บริษัท ไอดีซี พรีเมียร์ จำกัด.

[7] ดร.มุทิตา สงฆ์จันทร์. (2552). **รายงานวิจัยฉบับสมบูรณ์ เรื่องการเปรียบเทียบสมรรถนะของระบบควบคุมโดยใช้วิธีการควบคุมแบบเรียนรู้ซ้ำพหุนามเมตริกซ์ G อันดับหนึ่งและอันดับสูงกว่าเมื่อใช้ค่าถ่วงน้ำหนักแบบปรับค่าได้.** พิษณุโลก: มหาวิทยาลัยนเรศวร.

[8] โศรฎา แข็งการและกนต์ธร ชำนิประศาสน์. **การใช้ MATLAB สำหรับงานทางวิศวกรรม 6.** นครราชสีมา: มหาวิทยาลัยเทคโนโลยีสุรนารี.

ประวัติผู้ดำเนินโครงงาน

ชื่อ นายการุณย์ มะทิมา
ภูมิลำเนา 40/1 หมู่ 4 ต.ไตรตรึงษ์ อ.เมือง จ.กำแพงเพชร
ประวัติการศึกษา

- จบมัธยมศึกษาจากโรงเรียนนครไตรตรึงษ์
- ปัจจุบันกำลังศึกษาในระดับปริญญาตรี ชั้นปีที่ 4 สาขาวิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail: Krun_mathima@hotmail.com

ชื่อ นายวิโรจน์ พวงมั่ง
ภูมิลำเนา 122 หมู่ 20 ต.ร่มเย็น อ.เชียงคำ จ.พะเยา
ประวัติการศึกษา

- จบมัธยมศึกษาจากโรงเรียนเชียงคำวิทยาคม
- ปัจจุบันกำลังศึกษาในระดับปริญญาตรี ชั้นปีที่ 4 สาขาวิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail: Wiroj.pm@gmail.com