รายการอ้างอิง

Bluckley, P.S., "Recent Advances in Averaging Level Control," *in Productivity through control Technology,* April, pp. 93-98, 1980.

Bluckley, P.S., "Nonlinear Averaging Level Control with Digital Controller," *Texas A&M Instrumentation Symposium*, Jan.pp 37-44, 1986.

Bluckley, P.S., "Technique of process control,"Wiley & Sons, New York, N.Y.,1964

Cheung, T.F. "*Liquid Level Control*," Dissertation Lehigh University, PH.D.,1978.

Cheung & Luyben, "Liquid-Level Control in Single Tank and Cascades of Tanks With Proportional-Only and Proportional-Integral Feedback Controllers," *Ind.Eng.Chem. Fund.*,Vol. 18, pp.15-21,1979.

_____., "Nonlinear and Nonconventional Liquid-Level Controllers," *Ind Eng. Chem.Fund.*,Vol. 19 , pp.93-97, 1980.

_____., "A Furture study of the PL Level Controllers," *ISA Transcation.*,Vol. 18 No.2, pp.73-77, 1980.

Coughanowr, "*Process Systems Analysis and Control,*" Singapore: McGraw-Hill book Company, 1991.

Cutler C.M., "Dynamic Matrix Control of Imbalanced Systems," *ISA Transcation* Vol. 21, No. 1, pp. 1-6, 1982.

Korchinski W.J., "Advanced Level Control: Explained and analyzed," *Hydrocarbon processing*, June, pp 65-70, 1995.

Kuo & Hanselman, "*MATLAB Tools for control System Analysis and Design*," Prentice Hall International, Inc 1994.

Luyben,"*ProcessModeling, Simulation and Control for Chemical Engineers,*" Singapore McGraw-Hill book Company, 1990.

Luyben &Buckley, "A Proportional-Lag Controllers," *Instrument.Tech.*,Vol. 65, Dec, pp 65-68, 1977.

Marlin, "*Process Control : Designing Processes and Control Systems for Dynamic Performance,*" Singapore : McGraw-Hill book Company, 1995.

McDonald, McAvoy & A.Tits,"Optimal Averaging Level Control," *AIChE Journal*, Vol. 32, No.1, pp 75-86, 1986.

Roy & Smith, "Better than Averag[ing] Level Control," *Intech*, July, pp 50-55, 1995.

_____., "New Tunning Equations for Optimum Performance of Linear and Nonlinear Averaging Level Controllers", *ISA Conference Toronto, Canada*, April 25, 1995.

Saadat, "*Computational Aid in Control Systems Using Matlab*," McGraw-Hill book Company, 1993.

Seborg, Edgar & Mellichamp,"*Process Dynamic and Control*", Singapore: John Wiley & Sons, Inc.1989.

Shunda & Fehervari, "Nonlinear Control of Liquid Level," *Instrument.Tech.*,Vol. 43 Jan, pp 43-48, 1976.

The MathWorks, Inc.,"*MATLAB User's Guide*," Concord, MA, 1991.

ภาคผนวก ก

การเขียนโปรแกรมการควบคุมระดับของเหลวแบบเฉลี่ยโดยใช้โปรแกรมคอมพิวเตอร์แม็ทแลบ (MATLAB)

จากความเจริญก้าวหน้าทางด้านซอฟแวร์รวมทั้งอุตสาหกรรมไมโครโปรเซสเซอร์ของคอมพิวเตอร์ในยุคปัจจุบัน ทำให้การนำคอมพิวเตอร์มาใช้ในการเขียนโปรแกรมการควบคุมระดับแบบเฉลี่ย เพื่อทำการศึกษาเปรียบเทียบและประเมินสมรรถนะของตัวควบคุมในระบบควบคุมจึงมีความเหมาะสม ให้ผลการประเมินที่ถูกต้องแม่นยำและประหยัดเวลาในการทำงาน โปรแกรมคอมพิวเตอร์แม็ทแลบ (MATLAB) จัดเป็นโปรแกรมที่นิยมนำมาใช้อย่างแพร่หลายในทางด้านวิศวกรรม เนื่องจากเป็นโปรแกรมที่สามารถทำการคำนวนทางคณิตศาสตร์ขั้นสูง สามารถแก้ปัญหาเกี่ยวกับแมทริกซ์และเวกเตอร์ได้เป็นอย่างดี นอกจากนั้นยังประกอบไปด้วยฟังก์ชันจำนวนมากที่ช่วยให้ผู้ใช้สามารถเรียกนำมาใช้งานได้สะดวกและรวดเร็ว รวมทั้งมีการแสดงผลทางด้านกราฟิกทำให้สามารถแสดงผลการคำนวนออกมาในรูปของกราฟได้ในหลายลักษณะ มีความยืดหยุ่นในการใช้งานสูง สามารถนำมาใช้งานร่วมกับโปรแกรมอื่น ๆ เช่น โปรแกรมภาษาซี, โปรแกรมภาษาฟอร์แทรนได้เป็นอย่างดี

ก.1 โปรแกรมคอมพิวเตอร์แม็ทแลบ (MATLAB)

โปรแกรมคอมพิวเตอร์แม็ทแลบถูกพัฒนาขึ้นครั้งแรกที่มหาวิทยาลัยนิวแม็กซิโกและมหาลัยสแตนฟอร์ดในช่วงปลายปี 1970 เพื่อใช้ในการเรียนการสอนทางด้านทฤษฎีเกี่ยวกับแมทริกซ์ (Matrix labotatory), ฟังก์ชันพีชคณิตเชิงเส้น (Linear algebra function) และการวิเคราะห์ในเชิงตัวเลข (Numerical analysis) โดยได้รวมเอาการพัฒนาโปรแกรมลินแพค (LINPACK) และอีสแพค (EISPACK) ซึ่งเป็นแพคเกจที่ประกอบด้วยรูทีนย่อย (subroutine) ในภาษาฟอร์แทนสำหรับการคำนวนทางด้านแมทริกซ์เข้ามาด้วย การพัฒนาโปรแกรมแม็ทแลบในช่วงแรกเพื่อให้นักศึกษาสามารถใช้โปรแกรมสำเร็จรูปเกี่ยวกับการคำนวนทางด้านแมทริกซ์ได้โดยไม่ต้องเขียนโปรแกรมในภาษาฟอร์แทน

ในปัจจุบันความสามารถของโปรแกรมแม็ทแลบได้ถูกพัฒนาขึ้นจากเดิมเป็นอย่างมาก ภายใต้ชื่อ “Matrix Laboratory” โปรแกรมแม็ทแลบถูกนำมาใช้งานอย่างกว้างขวางทั้งในทางด้านวิศวกรรมศาสตร์และวิทยาศาสตร์ โปรแกรมแม็ทแลบจัดเป็นโปรแกรมแบบปฏิสัมพันธ์ (Interactive program) ผู้ใช้สามารถเขียนโปรแกรมเพื่อสร้างฟังก์ชันหรือโปรแกรมพิเศษขึ้นมาเพื่อใช้งานโดยเฉพาะในรูปของเอ็มไฟล์ (M-files) นอกจากนั้นผู้ใช้ยังสามารถเรียกใช้งานฟังก์ชันของคำสั่งสำเร็จรูปที่ถูกพัฒนาขึ้นแล้วและบรรจุอยู่ในส่วนหนึ่งของโปรแกรมแม็ทแลบซึ่งเราเรียกว่า กล่องเครื่องมือหรือทูลบ็อกซ์ "Toolboxes"

โปรแกรมแม็ทแลบประกอบด้วยคำสั่งที่มีประโยชน์มากมาย สามารถนำมาใช้งานทางด้านต่าง ๆ ดังต่อไปนี้

1. การคำนวนเกี่ยวกับแมทริกซ์

โปรแกรมแม็ทแลบสามารถทำการทรานสโพสแมทริกซ์ การคูณแมทริกซ์ การหาดีเทอร์มิแนนท์ การทำอินเวอร์สแมทริกซ์ นอกจากนั้นยังสามารถทำการแก้สมการเชิงเส้น การประมาณค่าพารามิเตอร์ต่าง ๆ เป็นต้น

2. การคำนวนโพลีโนเมียล

โปรแกรมแม็ทแลบสามารถทำการคำนวนทางด้านโพลีโนเมียล เช่น การหารากสมการโพลีโนเมียล การหาคอนโวลูชัน (Convolution) และดีคอนโวลูชัน (Deconvolution) การหาสมการโพลีโนเมียลและการหาสมการถดถอยแบบโพลีโนเมียล เป็นต้น

3. การจัดการเกี่ยวกับเวคเตอร์และการวิเคราะห์ข้อมูล

โปรแกรมแม็ทแลบสามารถนำมาคำนวนผลรวมแบบเวกเตอร์ การหาค่าเฉลี่ย การหาค่าเบี่ยงเบนมาตรฐาน และการหาค่าต่ำสุดและสูงสุดของข้อมูล เป็นต้น

4. การจัดการเกี่ยวกับการแสดงผลกราฟ

โปรแกรมแม็ทแลบสามารถทำการแสดงผลในรูปของกราฟได้ทั้งหมด 7 ประเภท ประกอบด้วย

- การพล็อทค่า x-y บนกราฟสเกลเส้นตรง
- การพล็อทค่า x-y บนกราฟสเกลล็อค-ล็อค
- การพล็อทค่า x-y บนกราฟสเกลกึ่งล็อคบนแกน x
- การพล็อทค่า x-y บนกราฟสเกลกึ่งล็อคบนแกน y

- การพล็อทกราฟแบบโพลาร์
- การพล็อทกราฟแบบคอนทัวร์
- การพล็อทกราฟแบบตะแกรงสามมิติ

5. การจัดการต่าง ๆ ของกระบวนการสัญญาณ

6. การวิเคราะห์ไม่เชิงเส้น

7. การประเมินและวิเคราะห์ผลทางสถิติ

ทูลบ็อกซ์ (Toolboxes) ในโปรแกรมแม็ทแลบได้ถูกพัฒนาสำหรับการใช้งานทางด้านต่าง ๆ มากมายประกอบไปด้วย

- Control System Toolbox
- Frequency-Domain System Identification Toolbox
- Fuzzy Logic Toolbox
- Higher-Order Spectral Analysis Toolbox
- Image Processing Toolbox
- Model Predictive Control Toolbox
- Mu-Analysis and Synthesis Toolbox
- NAG Foundation Toolbox
- Neural Network Toolbox
- Nonlinear Control Design Toolbox

- Optimization Toolbox
- Quantitative Feedback Theory Toolbox
- Robust Control Toolbox
- Signal Processing Toolbox
- SIMULINK
- SIMULINK Real-Time Workshop
- Spline Toolbox
- Statistics Toolbox
- Symbolic Math Toolbox
- System Identification Toolbox

โดยทูลบ็อกซ์ (Toolbox) ที่สำคัญและนิยมใช้กันอย่างกว้างขวางในโปรแกรมแม็ทแลบอันหนึ่ง คือ ซิมูลิงค์ (SIMULINK) ซึ่งสามารถช่วยในการวิเคราะห์ ออกแบบ ทดสอบและเลียนแบบระบบต่าง ๆ ทั้งในแบบเชิงเส้นและไม่เชิงเส้น แบบดีสครีต หรือแบบผสมต่าง ๆ ที่เปลี่ยนแปลงตามเวลาได้เป็นอย่างดี การใช้งานซิมูลิงค์สามารถทำได้ง่ายสะดวกและรวดเร็ว มีความแม่นยำถูกต้อง ผู้ใช้สามารถสร้างและทำการเลียนแบบกระบวนการต่าง ๆ ได้โดยเลือกบล็อคไดอะแกรมย่อยต่าง ๆ ที่มีอยู่ในทูลบ็อกซ์ (Toolbox) มาสร้างเป็นบล็อคไดอะแกรมที่ต้องการลงในแผ่นบันทึก (Notepad) โดยใช้เมาส์ควบคุม โดยบล็อคไดอะแกรมดังกล่าวสามารถเปลี่ยนทำการเปลี่ยนค่าพารามิเตอร์ต่าง ๆ ได้ตามความต้องการของผู้ใช้ การแสดงผลจะแสดง

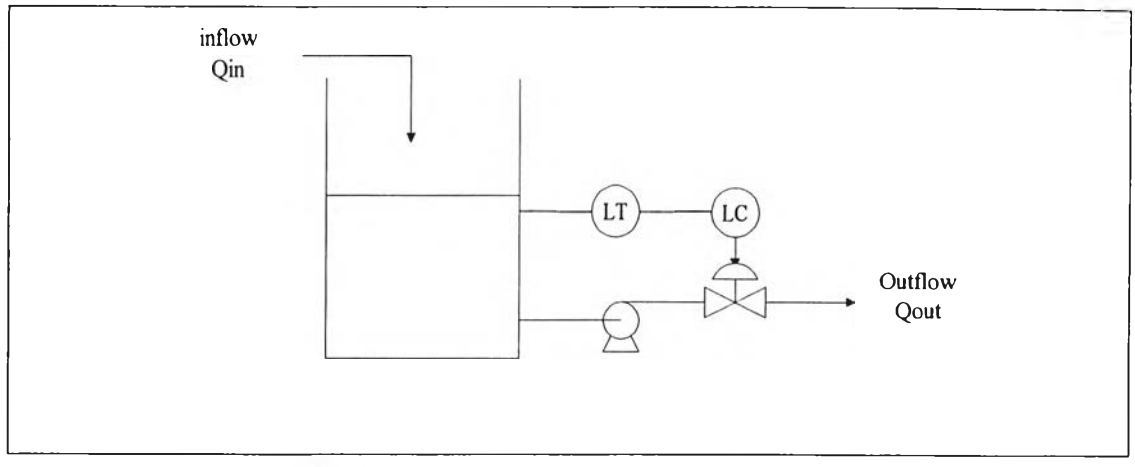
ออกมาในรูปกราฟิคสามารถมองเห็นผลการเปลี่ยนแปลงได้ชัดเจนและรวดเร็ว

วิธีการเขียนและใช้งานโปรแกรมคอมพิวเตอร์แม็ทแลบ (MATLAB)

เราสามารถทำการเขียนโปรแกรมสำหรับทำการเลียนแบบระบบใด ๆ ลงในโปรแกรมแม็ทแลบได้เช่นเดียวกับโปรแกรมภาษาอื่น ๆ โดยทำการเรียกแผ่นบันทึก (Notepad) ในโปรแกรมแม็ทแลบแล้วทำการเขียนคำสั่งต่าง ๆ ที่เลือกใช้ลงในแผ่นบันทึก (Notepad) แล้วทำการเก็บ (Save) โปรแกรมที่เขียนขึ้นในไฟล์ที่มีนามสกุล .m (M files) เมื่อทำการเขียนโปรแกรมที่ต้องการเสร็จเรียบร้อยแล้วสามารถทำการทดสอบและเรียกใช้งานโปรแกรมที่เขียนขึ้นได้ โดยเข้าไปยังพื้นที่หน้าจอโปรแกรมแม็ทแลบแล้วทำการเขียนคำสั่งโดยใช้ชื่อไฟล์(นามสกุล.m) ที่เราทำการเก็บข้อมูลโปรแกรมที่ได้เขียนขึ้นไว้ หลังจากนั้นทำการกดแป้น "Enter" ของแป้นพิมพ์ โปรแกรมแม็ทแลบจะรับคำสั่งโดยทำการคำนวนวิเคราะห์และแสดงผลตามลำดับคำสั่งที่ได้เขียนขึ้นในโปรแกรม

ก.2 ตัวอย่างการเขียนโปรแกรมการควบคุมระดับแบบเฉลี่ยโดยใช้โปรแกรมคอมพิวเตอร์แม็ทแลบ

งานวิจัยชิ้นนี้เป็นการควบคุมระดับแบบเฉลี่ยในถังพัก โดยระดับจะถูกควบคุมโดยการปรับอัตราการไหลขาเข้าและขาออกจากถังพักดังแสดงรูปที่ ก.1 ซึ่งระบบดังกล่าวสามารถอธิบายได้ดังสมการต่อไปนี้



รูปที่ ก.1 ระบบการควบคุมระดับของเหลวในถังพัก

$$\frac{d(\rho V)}{dt} = \rho Q_i - \rho Q_o$$

หรือ $$A\frac{dH}{dt} = Q_i - Q_o \qquad (ก\text{-}1)$$

จากวิธีการเชิงตัวเลข (Numerical) โดยใช้วิธีของออย์เลอร์ (Euler Method) ตัวแปรที่เราต้องการทราบค่าในรูปฟังก์ชันของเวลาจากสมการที่ (ก-1) คือ ระดับความสูง H เราจะแทนตัวแปรระดับความสูงในรูปฟังก์ชันของเวลาในโปรแกรมด้วยตัวแปร HDOT ที่สเต็พเวลาใดๆ (*n*th step in time) สามารถอธิบายสมการ ก.1 ได้ดังนี้

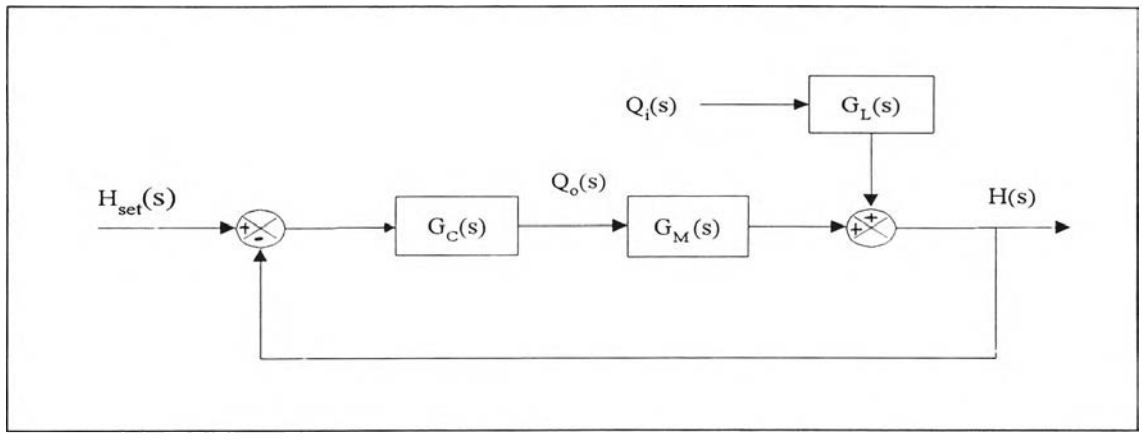
$$(HDOT)_n = [(Qin)_n - (Qout)_n] / A \qquad (ก\text{-}2)$$

ค่าใหม่ของระดับความสูงที่เวลาสเต็พเวลา n +1 สามารถคำนวนได้จากการใช้สมการของออย์เลอร์ (Euler algorithm) โดยแทนตัวแปรช่วงสเต็พของเวลาด้วย TDELTA ดังแสดงในสมการต่อไปนี้

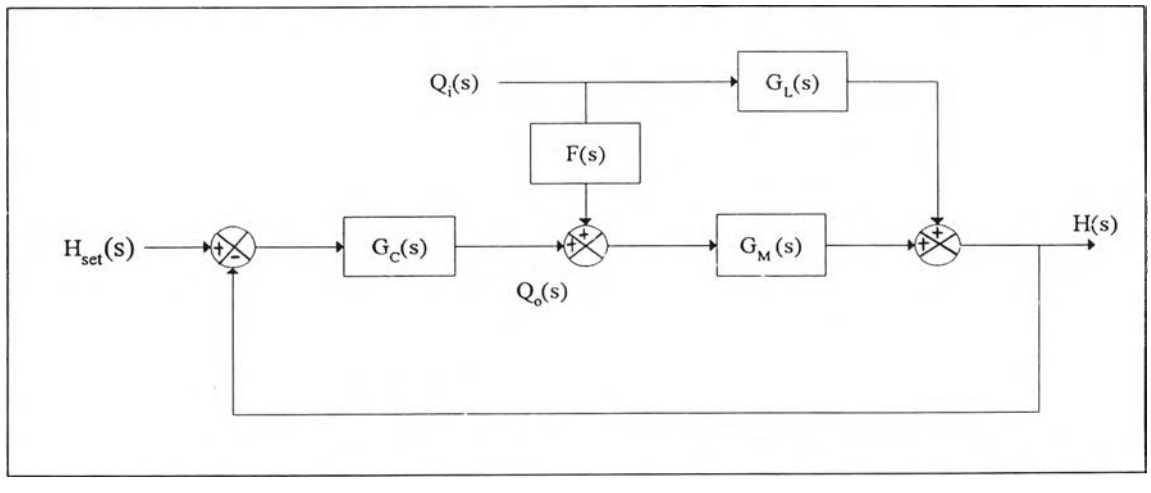
$$(H)_{n+1} = (H)_n + TDELTA\ (HDOT)_n \qquad (ก\text{-}3)$$

ในที่นี้ระบบเป็นแบบลูพปิด (Closed loop system) โดยกำหนดให้ตัวควบคุมที่ใช้เป็น

ตัวควบคุมป้อนกลับหรือตัวควบคุมแบบป้อนหน้า/ป้อนกลับ สามารถอธิบายระบบลูพปิดดังกล่าวในรูปของบล็อคไดอะแกรมได้ดังต่อไปนี้



รูปที่ ก.2 บล็อคไดอะแกรมของระบบลูพปิดสำหรับการควบคุมระดับของเหลวแบบป้อนกลับ



รูปที่ ก.3 บล็อคไดอะแกรมของระบบลูพปิดสำหรับการควบคุมระดับของเหลวแบบป้อนหน้า/ป้อนกลับ

โดย G_L คือ ทรานสเฟอร์ฟังก์ชันของตัวแปรโหลด (Load variable transfer function)

G_M คือ ทรานสเฟอร์ฟังก์ชันของตัวแปรปรับค่า (Manipulated variable transfer function)

G_c คือ ทรานสเฟอร์ฟังก์ชันของตัวควบคุมแบบป้อนกลับ (Feedback controller transfer funtion)

F(s) คือ ทรานสเฟอร์ฟังก์ชันของตัวควบคุมแบบป้อนหน้า (Feedforward controller transfer funtion)

ในที่นี้ขอยกตัวอย่างการควบคุมแบบป้อนกลับโดยใช้ตัวควบคุมแบบพีไอ สามารถหาค่าอัตราการไหลขาออกที่ถูกปรับค่าโดยตัวควบคุมแบบพีไอได้ดังสมการต่อไปนี้

$$(Qout)_n = (Qout)_{n-1} + K_C((E)_n + \frac{1}{\tau_I}\int_0^t E(t)dt) \qquad (ก\text{-}4)$$

โดย $(Qout)_n$ คือ ค่าอัตราการไหลขาออกในรอบนี้

$(Qout)_{n-1}$ คือ ค่าอัตราการไหลขาออกในรอบก่อนหน้านี้

$(E)_n$ คือ ค่าความผิดพลาดในรอบนี้มีค่าเท่ากับ $H(s)- H_{set}(s)$

K_C คือ ค่าเกนของตัวควบคุมแบบพีไอ (ไร้หน่วย)

τ_I คือ ค่าเวลารีเซ็ทหรือค่าคงที่ของเวลาอินทิกรัลของตัวควบคุมแบบพีไอ (หน่วยนาที)

เพื่อความสะดวกในการเขียนโปรแกรมนิยมใช้สมการของตัวควบคุมแบบพีไอในรูปของ velocity form ดังนี้

$$(Qout)_n = (Qout)_{n-1} + K_C((E)_n - (E)_{n-1}) + ((TDELTA / \tau_I)(E)_n) \qquad (ก\text{-}5)$$

โดย TDELTA คือ ช่วงสเต็พของเวลาที่ใช้ในการคำนวน

$(E)_{n-1}$ คือ ค่าความผิดพลาดในรอบก่อนหน้านี้

กำหนดให้ค่าพารามิเตอร์ของระบบการควบคุมระดับแบบเฉลี่ยในถังพักที่ทำการเลียนแบบเป็นดังนี้

พื้นที่หน้าตัดของถัง A = 1 ลูกบาศก์เมตร

ค่าระดับสูงสุดในถังพัก = 4 เมตร

ค่าระดับต่ำสุดในถังพัก = 0 เมตร

อัตราการไหลขาออกสูงสุด $Q_{o,max}$ = 4 ลูกบาศก์เมตร/นาที

ค่าเริ่มต้นที่สถานะคงตัวของระดับ = 50% full scale = 2 เมตร

ค่าเซ็ทพอยน์ของระดับที่ต้องการ = 2 เมตร

ค่าเริ่มต้นที่สถานะคงตัวของอัตราการไหล = 1 ลูกบาศก์เมตร/นาที

ค่าการรบกวนโหลด (Load disturbance) เท่ากับ 100 % สเต็พ โดยเพิ่มค่าอัตราการไหลขาเข้าจาก 1 m^3/min เป็น 2 m^3/min

เนื่องจากช่วงสเต็พของเวลา (TDELTA) ที่กำหนดขึ้นในโปรแกรมมีผลต่อค่าอัตราการเปลี่ยนแปลงการไหลขาออกสูงสุด (MRCO) และค่าระดับของเหลวสูงสุด (MPH) ที่คำนวนได้ เพื่อให้ผลการทดลองที่ได้มีความถูกต้องแม่นยำ ในงานวิจัยชิ้นนี้ได้กำหนดให้ช่วงสเต็พของเวลาที่ใช้ในการคำนวน = 0.005 นาที

เมื่อทำการปรับจูนค่าพารามิเตอร์ของตัวควบคุมแบบพีไอโดยใช้วิธีที่กล่าวไว้ดังบทที่ 3 จะได้

ค่าเกนของตัวควบคุมแบบพีไอ = 1.25

ค่าเวลารีเซ็ทของตัวควบคุมแบบพีไอเท่ากับ = 4 นาที

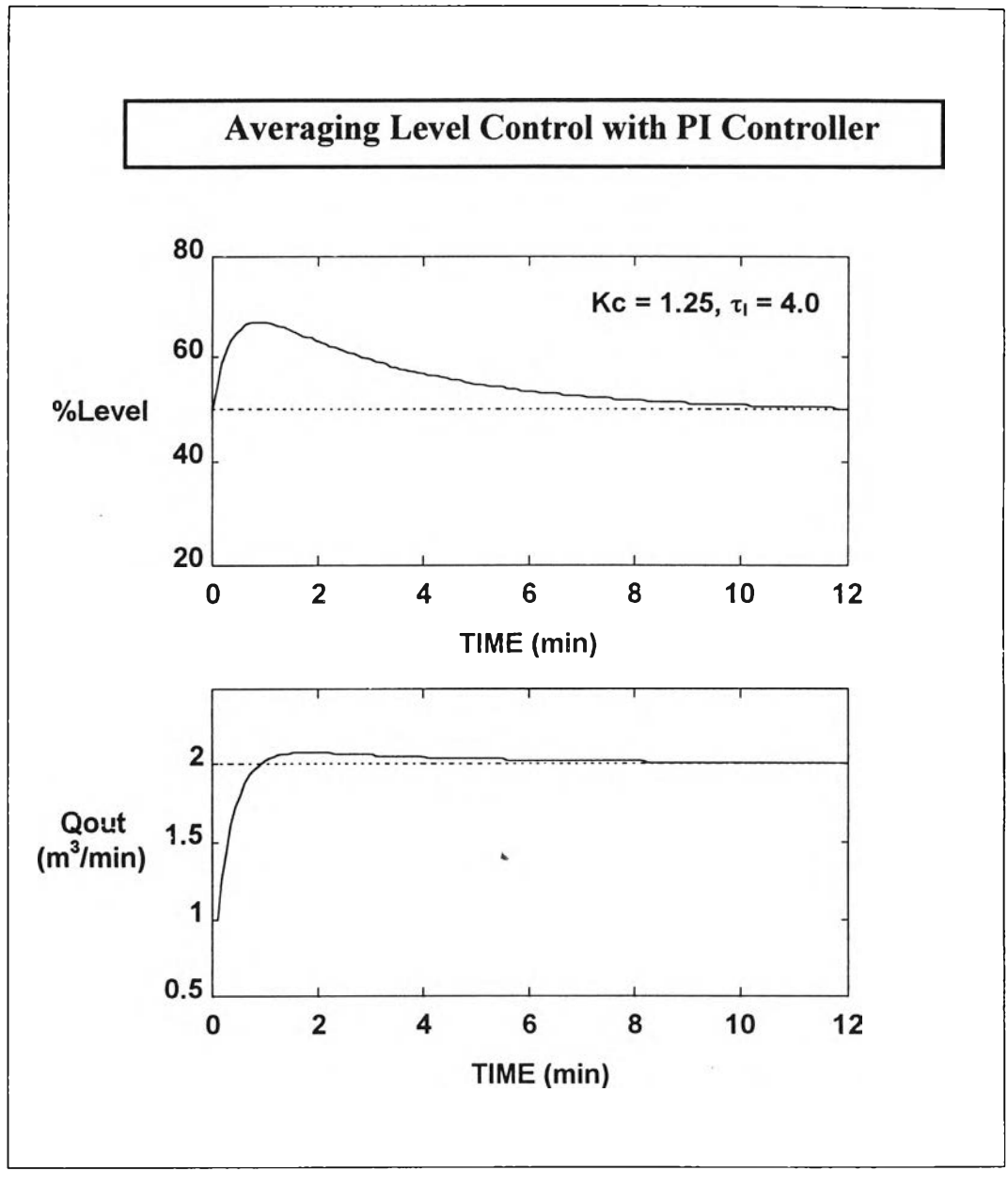
จากระบบและพารามิเตอร์ดังกล่าวมาแสดงตัวอย่างการเขียนโปรแกรมการควบคุมระดับแบบเฉลี่ยในถังพักโดยใช้โปรแกรมคอมพิวเตอร์แม็ทแลบได้ดังต่อไปนี้

```
% Averaging level control with PI controller
clear all
% initial conditions:
Time = 0  ;
Hmax = 2 ;
Hmin = 0 ;
DH = Hmax- Hmin ;
H = Hmax / 2 ;
HSET = Hmax / 2 ;
Qout = 1 ;
QSET = 1 ;
Qomax = 4 ;
E0 = 0 ;

% Disturbance:
Qin = 2.0 ;

% Parameter Values
A  = 1.0 ;
KC = 1.25 ;
KL = (KC*Qomax) / DH ;
```

```
TAUI = 4 ;
TDELTA = 0.005;
simtime = 20;

j = 0 ;
time(1) = Time ;
h(1) = H ;
hset(1) = HSET ;
QOUT(1) = Qout ;
Qset(1) = QSET ;

while Time<simtime
  j=j+1;

% feedback controller
E = H - HSET ;
Qout = Qout + KL*( (E-E0)+ (DELTA/TAUI)*E ) ;
E0 = E ;

% evaluate derivatives
HDOT = (Qin-Qout) / A ;
H = H + HDOT*TDELTA ;
Time = Time + TDELTA ;
time(j+1) = Time ;
h(j+1) = H ;
hset(j+1) = HSET ;
QOUT(j+1) = Qout ;
```

```
 Qset(j+1) = QSET ;
 end

% create percent Hs
for i = 1: j+1
        PHs(i) = ( 100 / (Hmax-Hmin) ) * (h(i)-Hmin) ;
        PHset(i) = ( 100/ (Hmax-Hmin) ) * (hset(i) -Hmin ) ;
end

% calculate MRCO and MPH
 dQ = diff(QOUT)./diff(time);
 MPH = max(PHs)
 MRCO = max(dQ)

% plot graph
 subplot(2,1,1)
 plot(time,PHs,time,PHset,':')
 axis( [0  12  20  80] )
 xlabel('TIME (min)')
 ylabel('%Level')
 title('Averaging level control with PI controller' )

 subplot(2,1,2)
 plot(time,QOUT,time,Qset,':')
 axis( [0  12  0.5  2.5] )
 xlabel('TIME (min)')
 ylabel('Qout (m3/min)')
```

จากการเขียนโปรแกรมการควบคุมระดับแบบเฉลี่ยในถังพักโดยใช้ตัวอย่างโปรแกรมแม็ทแลบดังกล่าวข้างต้น สามารถแสดงผลที่ได้ดังรูป ก.3



รูปที่ ก.4 ผลที่ได้จากการเขียนโปรแกรมระบบการควบคุมระดับแบบเฉลี่ยในถังพัก โดยใช้ตัวอย่างโปรแกรมข้างต้น

ก.3 ขั้นตอนการเขียนโปรแกรมการควบคุมระดับแบบเฉลี่ยโดยใช้ตัวควบคุมประเภทต่าง ๆ

จากตัวอย่างการเขียนโปรแกรมการควบคุมระดับแบบเฉลี่ยในหัวข้อ ก.2 สามารถแยกออกเป็นขั้นตอนพร้อมทั้งแสดงการเขียนโปรแกรมการควบคุมระดับแบบเฉลี่ยโดยใช้ตัวควบคุมประเภทต่าง ๆ ที่ได้ทำการศึกษาดังนี้

ก.3.1 การกำหนดค่าเริ่มต้นของระบบการควบคุมระดับแบบเฉลี่ยในถังพัก

- **ตัวควบคุมแบบป้อนกลับ** ได้แก่ ตัวควบคุมแบบทั่วไป และตัวควบคุมแบบไม่เชิงเส้น

```
clear all
% initial conditions:
Time = 0;
Hmax = 2;
Hmin = 0 ;
DH = Hmax-Hmin ;
H = Hmax / 2;
HSET = Hmax /2;
QSET = 2 ;
Qomax = 4 ;
Qout = 1 ;
E0 = 0 ;
```

- ตัวควบคุมแบบป้อนหน้า/ป้อนกลับ ได้แก่ ตัวควบคุมแบบพีแอล

```
% initial conditions:
Time = 0 ;
Hmax = 2 ;
Hmin = 0 ;
DH = Hmax-Hmin;
H = Hmax / 2 ;
HSET = Hmax /2 ;
QSET = 2 ;
Qomax = 4 ;
Qout = 1;
Qin0 = 1 ;
CO0 = 1 ;
E0 = 0 ;
```

ก.3.2 การกำหนดค่าการรบกวนโหลดในอัตราการไหลขาเข้า

```
% Disturbance:
Qin = 2.0;
```

ก.3.3 การกำหนดค่าพารามิเตอร์ของตัวควบคุมและการเขียนสมการตัวควบคุมชนิดต่าง ๆ

- **ตัวควบคุมแบบพี**

```
% Parameter Values
A = 1.0;
KC = 1.25;
KL = (KC*Qomax)/DH ;
TDELTA = 0.1;
simtime = 20;
j = 0;
time(1) = Time;
hset(1) = HSET;
h(1) = H;
QOUT(1) = Qout ;
Qset (1) = QSET;
while Time<simtime
j=j+1;
% P controller
E = H - HSET;
Qout = Qout + KL*(E-E0) ;
E0 = E;
```

- ตัวควบคุมแบบพีไอ

```
% Parameter Values
 A = 1.0;
 KC = 1.0;
 TAUI = 3.2;
 KL = (KC*Qomax)/DH;
 DELTA = 0.1;
 simtime = 20;
 j=0 ;
 time(1) = Time ;
 h(1) = H ;
 hset(1) = HSET ;
 QOUT(1) = Qout ;
 Qset(1) = QSET ;

while Time<simtime
 j=j+1;
 % PI controller
 E = H - HSET ;
 Qout=Qout + KL*( (E-E0)+ (DELTA/TAUI)*E );
 E0 = E ;
```

- **ตัวควบคุมแบบพีแอล**

```
% Parameter Values
A = 1.0;
KC = 0.8;
KL = (KC*Qomax)/DH;
TDELTA = 0.1;
simtime = 20;
TAUF = 1.0;
j=0;
time(1) = Time;
hset(1) = HSET ;
h(1) = H;
QOUT(1) = Qout ;
Qset(1) = QSET;
while Time<simtime
j=j+1;
% PL controller
E = H - HSET  ;
CO1 = KL*E ;
EXTF = exp(-TDELTA / TAUF) ;
CO2 = (EXTF*CO0) + (1-EXTF)*Qin0 ;
CO0 = CO2 ;
Qin0 = Qin ;
Qout =  (CO1+CO2 ) ;
```

- ตัวควบคุมแบบช่วงกว้าง

```
% Parameter Values
 A = 1.0 ;            % Tank cross-sectional area (ft3)
 K =  1.0 ;           % Nonlinearity factor
 KC0 = 0.25 ;         % Zero-error gain
 TAUI0 = 8.3 ;        % Zero-error integral time
 TDELTA = 0.1;
 simtime = 20;
 j = 0;
 time(1) = Time;
 h(1) = H ;
 QOUT(1) = Qout;
 Qset(1) = QSET;
 hset(1) = HSET;
 while Time<simtime
 j=j+1;
% NL-wide range feedback controller
 E = H - HSET ;
 KCINT = (1+ abs(E)*K*log(25) )* (25^(abs(E)*K)) ;
 KC = KCINT* KC0 ;
 KL = (KC*Qomax)/DH ;
 TAUI = TAUI0 / KCINT ;
 Qout = Qout + KL*( (E-E0) +(TDELTA /TAUI)*E) ;
 E0 = E;
```

- ตัวควบคุมแบบพีไอพี

```
% Parameter Values
 A = 1.0 ;
 KC = 1.25 ;
 KL= (KC*Qomax)/DH;
 TAUI = 3.50 ;
 Eb = 0.2 ;
 TDELTA = 0.1 ;
 simtime = 20 ;
 j = 0;
 time(1) = Time ;
 h(1) = H ;
 hset(1) = HSET ;
 qset(1) = QSET ;
 QOUT(1) = Qout ;
 while Time<simtime
 j = j+1;
% PIP controller
 E = H-HSET ;
 if abs(E) <= Eb
        Qout = Qout +KL* (E-E0) ;
        E0 = E ;
 else
 if  E > 0
        Qout = Qout +KL*( (E-E0)+ (TDELTA/TAUI)*(E-Eb));
```

```
        E0 = E ;
        else
        Qout = Qout +KL*( (E-E0)+ (TDELTA/TAUI)*(E+Eb));
        E0 = E ;
        end
end
```

- **ตัวควบคุมแบบดีอาร์ไอพี**

```
% Parameter Value
 A = 1.0 ;
 KC = 1.25 ;
 KL = (KC*Qomax)/DH ;
 TAUI1 = 3.7 ;
 TAUI2 =17.00;
 Eb = 0.25 ;
 TDELTA = 0.1 ;
 simtime = 20 ;
 j=0;
 time(1) = Time ;
 h(1) = H ;
 QOUT(1) = Qout ;
 hset(1) = HSET ;
 qset(1) = QSET ;
 while Time<simtime
 j = j+1;
```

```
% DRIP controller
 E = H - HSET ;
 if  abs(E) < Eb
        Qout = Qout + KL*( (E-E0)+ (TDELTA/TAUI2)*E ) ;
        E0 = E ;
 else
        Qout = Qout + KL*( (E-E0)+ (TDELTA/TAUI1)*E );
        E0 = E ;
 end
```

- **ตัวควบคุมแบบพีโดยจำกัดการเปลี่ยนแปลงขาออก**

```
% Parameter Values
 A = 1.0;
 KC1 = 1.25;
 KL1 = (KC1*Qomax)/DH;
 MRCO1 = 2.5;
 TDELTA = 0.1;
 COmax1 = MRCO1*TDELTA;
 simtime = 20;
 j=0 ;
 time(1) = Time ;
 h1(1) = H1 ;
 hset(1) = HSET ;
 QOUT1(1)= CO1 ;
 Qset(1) = QSET ;
```

```
while Time<simtime
j = j+1;

% LOC-P controller
E1 = H1 - HSET ;
COold1 = CO1 ;
CO1 = CO1 + KL1*(E1-E01) ;
COnew1 = CO1 ;
DELCO1 = COnew1 -COold1 ;
if  DELCO1 <= Comax1
CO1 = COnew1 ;
else
        if  E1 < 0
        CO1 = COold1 + COmax1 ;
        else
        CO1 = COold1 - COmax1 ;
        end
end
E01 =E1 ;
```

- **ตัวควบคุมแบบพีไอโดยจำกัดการเปลี่ยนแปลงขาออก**

```
% Parameter Values
A = 1.0;
KC1=1.3;
KL1=(KC1*Qomax)/DH;
```

```
TAUI1 = 12.0;
MRCO = 2.50;
TDELTA = 0.1;
COmax = MRCO*TDELTA;
simtime = 20;
j=0 ;
time(1) = Time ;
h1(1) = H1 ;
hset(1) = HSET ;
QOUT1(1)= CO1 ;
Qset(1) = QSET ;
while Time<simtime
j=j+1;
% LOC-PI controller
E1 =  H1 - HSET ;
COold1 = CO1 ,
CO1 = CO1+ KL1*( (E1-E01)+ (TDELTA/TAUI1)*E1);
COnew1 = CO1 ;
DELCO1 = COnew1 -COold1 ;
if  DELCO1 <= COmax
CO1 = COnew1 ;
else
        if  E1 < 0
        CO1 = COold1 + COmax ;
        else
        CO1 = COold1 - COmax ;
        end
```

```
end
E01 =E1 ;
```

- **ตัวควบคุมพีแบบไม่เชิงเส้นโดยจำกัดการเปลี่ยนแปลงขาออก**

```
% Parameter Values
A = 1.0;
MRCOs1 = 2.50;
K1=1.25;            % Nonlinearity factor
KC01=0.29;          % Zero-error gain
TDELTA = 0.1;
COmax1 = MRCOs1*TDELTA;
simtime = 20;
j=0 ;
time(1) = Time ;
hset(1) = HSET ;
Qset(1) = QSET ;
h1(1)=H1;
QOUT1(1)=CO1 ;
while Time<simtime
j=j+1;
% LOC-NL/P controller
E1 = H1 - HSET;
COold1 = CO1 ;
KCINT1 = (1+ abs(E1)*K1*log(25) )* (25^(abs(E1)*K1)) ;
KC1 = KCINT1* KC01 ;
KL1 = ( KC1*Qomax)/DH ;
```

```
CO1 = CO1+ KL1* (E1-E01) ;
COnew1 = CO1 ;
DELCO1 = COnew1 -COold1 ;
if  DELCO1 <= COmax1
    CO1 = COnew1 ;
else
    if  E1 < 0
        CO1 = COold1 + COmax1 ;
    else
        CO1 = COold1 - COmax1 ;
    end
end
E01 = E1 ;
```

- **ตัวควบคุมแบบพีไอแบบไม่เชิงเส้นโดยจำกัดการเปลี่ยนแปลงขาออก**

```
% Parameter Values
A = 1.0;
MRCOs1 = 2.50;
K1=1.10;            % Nonlinearity factor
KC01=0.20;          % Zero-error gain
TAUI01= 10.95;    % Zero-error integral time
TDELTA = 0.1;
COmax1 = MRCOs1*TDELTA;
simtime = 20;
j=0 ;
```

```
time(1) = Time ;
hset(1) = HSET ;
Qset(1) = QSET ;
h1(1)=H1;
QOUT1(1)=CO1 ;
while Time<simtime
j=j+1;
% LOC-NL/PI controller
E1 =  H1 - HSET ;
COold1 = CO1 ;
KCINT1 = (1+ abs(E1)*K1*log(25) )* (25^(abs(E1)*K1)) ;
KC1 = KCINT1* KC01 ;
KL1 = ( KC1*Qomax)/DH ;
TAUI1 = TAUI01 / KCINT1 ;
CO1 = CO1 + KL1*( (E1-E01) +(TDELTA /TAUI1)*E1) ;
COnew1 = CO1 ;
DELCO1 = COnew1 -COold1 ;
if  DELCO1 <= COmax1
   CO1 = COnew1 ;
else
    if  E1 < 0
        CO1 = COold1 + COmax1 ;
    else
        CO1 = COold1 - COmax1 ;
            end
  end
  E01 =E1 ;
```

ก.3.4 การคำนวนหาค่าระดับความสูงโดยใช้วิธีออยเลอร์ (Euler Method) และการคำนวนหาค่าเปอร์เซ็นต์ความสูง

```
%evaluate derivatives
HDOT1 = (Qin-CO1)/A ;
H1 = H1 + HDOT1*TDELTA;
Time=Time+TDELTA;


time(j+1) = Time;
hset(j+1) = HSET ;
Qset(j+1) = QSET;
h1(j+1)=H1;
QOUT1(j+1)=CO1;
end
% create percent Hs
for i= 1: j+1
      PHset(i) = ( 100/ DH ) * (hset(i) -Hmin);
      PHs1(i) = ( 100/ DH ) * (h1(i)-Hmin) ;
end
```

ก.3.5 การแสดงผลการตอบสนองการควบคุมระดับแบบเฉลี่ยที่ได้ในรูปของกราฟ

```
% plot graph
subplot(2,1,1)
plot(time,PHset,':',time,PHs1)
axis( [0  12  20  80] )
```

```
xlabel('TIME(min)')
ylabel('% Level ')
title('Averaging Level Control with LOC-NL/PI Controller')

subplot(2,1,2)
plot( time,Qset,':', time,QOUT1)
axis( [0  12  0.5  2.5] )
xlabel('TIME(min)')
ylabel('Qout (ft3/min)')
```

ก.3.6 การคำนวนค่าอัตราการเปลี่ยนแปลงการไหลขาออกสูงสุด, ค่าระดับของเหลวสูงสุดและค่าดัชนีสมรรถนะ

```
% calculate MRCO and MPH
dQ1 = diff(QOUT1)./diff(time);
MRCO1 = max(dQ1)
MPH1 = max(PHs1)-50

% Calculate Performance Index
W1 = 1;
W2 = 2;
PI1 = W1*(MPHs1/MPHs1 - MPH1/MPHs1) + W2*(MRCOs1/MRCOs1-
      MRCO1/MRCOs1) ;
```

ก.3.8 การคำนวนหาค่าเปอร์เซ็นต์โอเวอร์ชูทและค่าเปอร์เซ็นต์อ็อฟเซ็ท

```
% Calculate Percent Overshoot
OS1 =  100*((max(QOUT1)-2)/1);


% Calculate Offset & Percent offset
OFS1 = PHs1(simtime/TDELTA)-50;
Poff =  (OFS1/MPH1)*100;
```

ก.3.9 การแสดงผลการคำนวนที่ได้ในรูปของตาราง

```
% display result
data = [1  ;KC1  ;TAUI1  ; MPH1 ; MRCO1  ; PI1 ;OS1 ;  Poff;  ];
disp('-------------------------------------------------------------------------------------------')
disp('                Result  of  comparison  performance of  PI controller                 ')
disp('-------------------------------------------------------------------------------------------')
disp  ('Case      Kc      TAUI      MPH      MRCO      PI      %OS      %offset')
disp  ('                                ( % )    (m3/min)/min                          ')
fprintf ('%2.0f   %6.2f    %6.2f    %6.2f    %6.2f    %6.2f    %6.2f    %6.2f /n',data)
```

ก.3.10 การสร้างสัญญาณรบกวนขึ้นในระบบควบคุม

โดยให้สัญญาณรบกวนมีขนาด 10% ของระดับของเหลวที่สภาวะคงตัว

```
%evaluate derivatives
noise = rand(1)*0.10;
HDOT1=(Qin-Qout)/A ;
H1 = H1 + HDOT1*TDELTA + noise*TDELTA;
```

ประวัติผู้เขียน

นางสาวสุชาดา พันธุ์พฤกษ์ เกิดวันที่ 28 กันยายน พ.ศ. 2515 ที่จังหวัดนราธิวาส สำเร็จการศึกษาระดับปริญญาตรีวิทยาศาสตรบัณฑิต สาขาเคมี คณะวิทยาศาสตร์ มหาวิทยาลัยสงขลานครินทร์ ในปีการศึกษา 2536 และศึกษาต่อในหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต สาขาวิศวกรรมเคมี คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2537