



รายงานวิจัยฉบับสมบูรณ์

โครงการควบคุมแบบพีซีพีไอดีด้วยวิธีพันธุศาสตร์สำหรับหุ่นยนต์

โดย ผศ. ดร. ปิติเชต สู้รักษาและคณะ

มิถุนายน 2545



รายงานวิจัยฉบับสมบูรณ์

โครงการควบคุมแบบพีซีพีไอดีด้วยวิธีพันธุศาสตร์สำหรับหุ่นยนต์

โดย ผศ. ดร. ปิติเขต สุรักษาและคณะ

มิถุนายน 2545

รายงานวิจัยฉบับสมบูรณ์

โครงการควบคุมแบบพีซีพีไอดีด้วยวิธีพันธุศาสตร์สำหรับหุ่นยนต์

คณะผู้วิจัย

สังกัด

1. ผศ. ดร. ปิติเขต สุรักษา

สถาบันเทคโนโลยีพระจอมเกล้า
เจ้าคุณทหารลาดกระบัง

2. Prof. Dr. Guanrong R. Chen

University of Houston, Texas, USA

สนับสนุนโดยสำนักงานกองทุนสนับสนุนการวิจัย

(ความเห็นในรายงานนี้เป็นของผู้วิจัย สกว. ไม่จำเป็นต้องเห็นด้วยเสมอไป)

Abstract

Project Code: PDF/95/44

Project Title: A Geno-fuzzy Proportional-Integral-Derivative (PID) Controller
for Robots

Investigator: (1) Asst. Prof. Dr. Pitikhate Sooraksa
King Mongkut's Institute of Technology Ladkrabang
(2) Prof. Dr. Guanrong R. Chen
University of Houston, Texas, USA

E-mail Address: kspitikh@kmitl.ac.th

Project Period: 1 years (July 2001 – June 2002)

In this research, a geno-fuzzy proportional-integral-derivative (PID) controller for robots is designed and implemented. The term "geno" comes from the "genetic algorithms or GA" employing to optimize the fuzzy-based controller. Utilizing GA, the optimal set of the controller gains is obtained. Simulation and experimental results yield a promising future for the findings in robotic application.

Keywords: PID control, GA, robot

บทคัดย่อ

รหัสโครงการ: PDF/95/44

ชื่อโครงการ: ตัวควบคุมแบบฟัซซีพีไอดีด้วยวิธีพันธุศาสตร์สำหรับหุ่นยนต์

ชื่อนักวิจัย: (1) ผศ. ดร. ปิติเขต สุรักษา
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
(3) ศาสตราจารย์ ดร. กวนรง อาริ เซน
มหาวิทยาลัยฮุสตัน, รัฐเทกซัส, สหรัฐอเมริกา

E-mail Address: kspitikh@kmitl.ac.th

ระยะเวลาโครงการ: 1 ปี (กรกฎาคม 2544 – มิถุนายน 2545)

งานวิจัยนี้ออกแบบและสร้างตัวควบคุมแบบฟัซซีพีไอดีด้วยวิธีพันธุศาสตร์สำหรับหุ่นยนต์ โดยอาศัยวิธีทางพันธุศาสตร์ซึ่งเป็นการหาค่าที่ดีที่สุดไปปรับค่าเกณฑ์ต่าง ๆ ของระบบควบคุมแบบฟัซซีพีไอดี ผลการจำลองและการทดลองมีนัยสำคัญแห่งความสำเร็จในการประยุกต์ใช้กับการควบคุมหุ่นยนต์

คำหลัก: ระบบควบคุมพีไอดี, วิธีพันธุศาสตร์, หุ่นยนต์

กิตติกรรมประกาศ

ขอขอบคุณสำนักงานกองทุนสนับสนุนการวิจัยที่ได้ให้โอกาสและสนับสนุนงานวิจัยในครั้งนี้

หากปราศจาก "แรงทุนส่งเสริม" สนับสนุนงานวิจัยในครั้งนี้ ผู้วิจัยหลักคง "หลุดไปจากวงโคจร" ของการทำวิจัยในประเทศไทย

ปิติเขต สุรักษา และคณะ

บทที่ 1

บทนำ

ระบบควบคุมแบบพีไอดี (Proportional-Integral-Derivative หรือ PID) เป็นระบบควบคุมที่ใช้กันแพร่หลายในโรงงานอุตสาหกรรม มีรายงานว่ามากกว่า 90% ของการควบคุมกระบวนการในประเทศญี่ปุ่นใช้ระบบควบคุมแบบนี้ และน่าจะเป็นในทำนองเดียวกันในประเทศอื่น ๆ ทั่วทุกมุมโลก [1] งานวิจัยในการปรับแต่งพารามิเตอร์ในการควบคุมระบบพีไอดีนี้ ยังคงเป็นหัวข้อที่สนใจของนักวิจัยต่าง ๆ ในปัจจุบัน [1-4]

หนึ่งในวิธีการปรับปรุงการทำงานของระบบควบคุมแบบพีไอดีแบบดั้งเดิม (conventional PID) ให้ดีขึ้น คือ การนำเอาตรรกศาสตร์ฟัซซี (fuzzy logic) เข้ามาประยุกต์ใช้โดยอาจใช้ในระบบผู้เชี่ยวชาญ หรือในรูปทำงานร่วมกับระบบพีไอดีแบบดั้งเดิม ตัวอย่างของงานวิจัยเหล่านี้ได้แก่ [1, 5, 6, 8-12] เนื่องจากในยุคปัจจุบันนี้ การสืบค้นข้อมูลเป็นไปได้โดยง่ายโดยอาศัยฐานข้อมูลต่าง ๆ ดังนั้นงานวิจัยที่ไม่เกี่ยวข้องโดยตรงกับรายงานฉบับนี้ จะไม่ขออ้างถึง ผู้ที่สนใจสามารถสืบค้นจากฐานข้อมูล [29] ผ่านระบบเครือข่ายอินเทอร์เน็ต ซึ่งเป็นฐานข้อมูลทางด้านวิศวกรรมไฟฟ้าที่ได้รับความเชื่อถือจากนักวิจัยในวงการวิศวกรรมไฟฟ้า

ในบรรดาระบบฟัซซีพีไอดีในรูปแบบต่าง ๆ [6-9] ระบบควบคุมที่ออกแบบโดยทีมวิจัยของมหาวิทยาลัยฮุสตัน (University of Houston, Texas, USA) ที่ได้ตีพิมพ์เผยแพร่ผลงานดังปรากฏใน [5] เป็นระบบควบคุมแบบไม่ต้องอาศัยการอ่านค่าจากตารางเกณฑ์ปฏิบัติงาน (look-up table) ผลการจำลองและประยุกต์ใช้ระบบควบคุมนี้พบว่า ได้ผลสัมฤทธิ์เลิศ (superior performance) กว่าระบบควบคุมแบบพีไอดีแบบดั้งเดิม [5, 8-11] แม้ว่าการควบคุมแบบฟัซซีพีไอดีที่ออกแบบใน [5] จะสามารถจัดกระบวนการทางการควบคุมแบบฟัซซีมาตรฐานให้ลดทอนอยู่ในรูปแบบง่าย ๆ ประหยัดเวลาดำเนินการของคอมพิวเตอร์อันเป็นจุดเด่นของระบบนี้ ถึงกระนั้นก็ตาม การจะหาค่าที่ดีที่สุดในการควบคุม ยังต้องใช้การลองผิดลองถูกแม้ว่าระบบจะฉลาดในเรื่องความทนทานต่อสัญญาณรบกวนจากภายนอกและ/หรือสัญญาณรบกวนด้านอินพุต

ด้วยเหตุนี้ งานวิจัยนี้ จึงเป็นการขยายผลการศึกษาในส่วนของภาคควบคุมแบบ Fuzzy PID ที่เป็นส่วนหนึ่งของระบบควบคุมแบบอัจฉริยะ (Intelligent control) ที่ผู้วิจัยและนักวิจัยที่เล็งได้ออกแบบและสร้างจริงสำเร็จมาแล้วในอดีต โดยงานวิจัยนี้จะใช้วิธีการทางพันธุศาสตร์ (Genetic Algorithms, GA) มาปรับเปลี่ยนค่าพารามิเตอร์ในการควบคุมดังกล่าวให้

ได้ค่าอัตโนมัติที่ดีที่สุดเพื่อใช้สำหรับหุ่นยนต์ นับแต่นี้จะเรียกวิธีการนี้ว่า “จีโน ฟัชซี พีไอดี” ที่จะออกแบบและสร้างต้นแบบจริงโดยผู้วิจัย งานวิจัยนี้ ออกแบบและสร้างตัวควบคุมจีโน ฟัชซีพีไอดี (Geno-fuzzy PID controller) หรือ GFPID สำหรับหุ่นยนต์

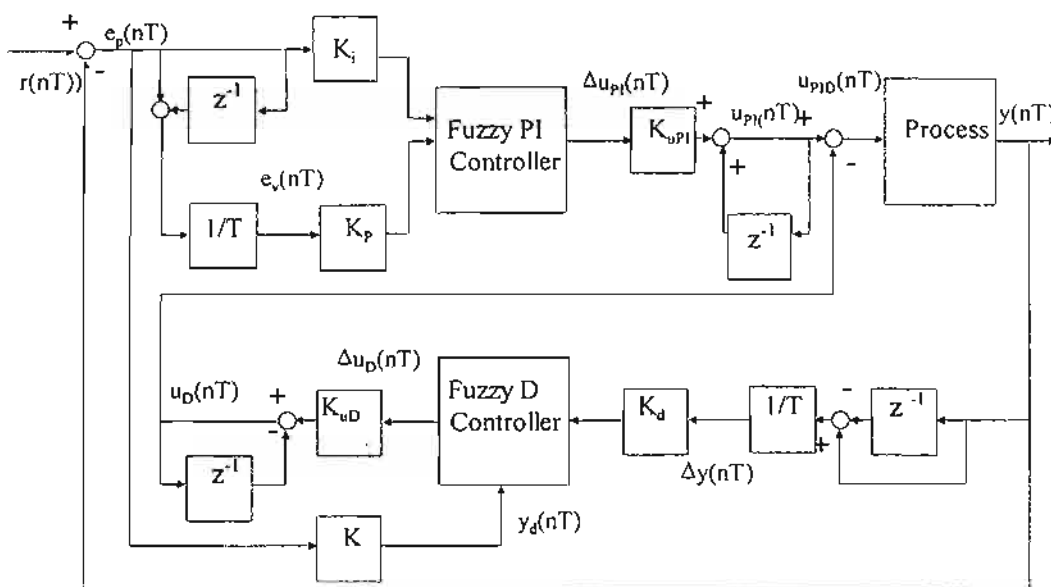
รายงานการวิจัย ฉบับนี้ เรียงลำดับการนำเสนอ ดังนี้ บทที่ 2 กล่าวถึง ระบบควบคุมแบบฟัชซีพีไอดีโดยย่อ บทที่ 3 กล่าวถึงการออกแบบตัวควบคุมแบบจีโนฟัชซี บทที่ 4 นำเสนอวิธีการหาค่าพารามิเตอร์ของระบบ (Parameter Identification) ในที่นี้ใช้วิธีการแบบ Least Square Method เพื่อหาค่าพารามิเตอร์จากมอเตอร์มือสองที่ไม่มีข้อมูลที่ใช้นำมาสร้างหุ่นยนต์ และนำพารามิเตอร์ที่ได้ไปใช้ในการจำลองผลในบทที่ 5 บทที่ 6 กล่าวถึงการประยุกต์ใช้ตัวควบคุมที่ได้ กับหุ่นยนต์ข้อต่ออ่อนตัว (flexible joint robot arm) บทสรุปและข้อเสนอแนะนำเสนอในบทที่ 7

บทที่ 2

ระบบควบคุมแบบฟัซซีพีไอดี

ในบทนี้กล่าวถึงระบบควบคุมแบบฟัซซีพีไอดี (fuzzy PI+D or fuzzy Proportional-Integral plus Derivative) ซึ่งเป็นตัวควบคุมที่พัฒนาขึ้นโดยทีมงานของนักวิจัย University of Houston โดย Chen และ คณะ [5] ซึ่งมีลักษณะเด่นที่แตกต่างกว่าระบบควบคุมแบบฟัซซีของงานวิจัยอื่น ๆ ดังนี้ คือ

1. รูปของความเป็นสมาชิกเป็นรูปง่าย ๆ
2. ขั้นตอนของการทำฟัซซีฟิเคชัน (fuzzification) การอนุมานฟัซซี (fuzzy inference) และการตัดฟัซซีฟิเคชัน (defuzzification) ทำโดยการวิเคราะห์แบบโยงเหตุผล (analytic) โดยผู้ออกแบบ โดยใช้วิธีการทางคณิตศาสตร์
3. ผลของการปรับค่าเกณฑ์การควบคุมจะออกมาเป็นสูตรสำหรับควบคุมง่ายเพียง 18 สูตร โดยครอบคลุมถึงทุกสถานะการณ์ ไม่เป็นตารางแสดงค่า
4. การออกแบบระบบฟัซซี ไม่ต้องไปออกแบบระบบพีไอดีแบบกระแทกกระเทือนโครงสร้าง (redesign) เพียงเพิ่มหน่วยควบคุมฟัซซี (fuzzy control block) อีกหนึ่งหน่วย ไปต่ออนุกรมกับผลที่ได้จากหน่วยพีไอดีกับหน่วยดีเพื่อช่วยในการตัดสินใจเท่านั้น (ดูรูป 2.1)



รูปที่ 2.1 ระบบควบคุมแบบฟัซซีพีไอดี แบบดิจิทัล

2.1 หลักการออกแบบตัวควบคุมแบบฟัซซีพีไอดี

(2.1- ก) การออกแบบหน่วยควบคุมฟัซซีพีไอ

จากรูปที่ 2.1 จะได้

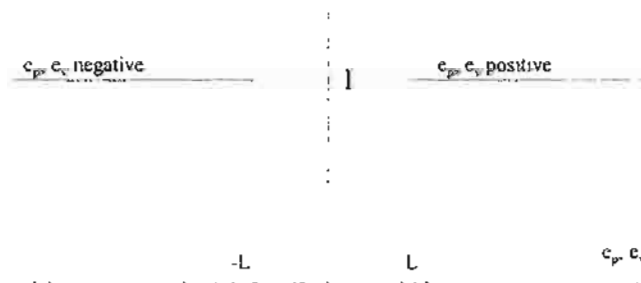
$$\Delta u_{PI}(nT) = K_p e_v(nT) + K_i e_p(nT), \quad (1)$$

$$u_{PI}(nT) = u_{PI}(nT - T) + K_{uPI} \Delta u_{PI}(nT), \quad (2)$$

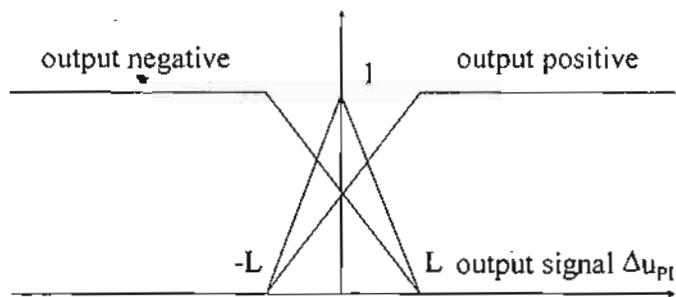
โดยที่ $\Delta u_{PI}(nT)$ เป็นการเพิ่มค่าในแต่ละรอบการควบคุมจากหน่วยฟัซซีพีไอ (the fuzzy control action of the PI controller), $e_p(nT)$ เป็นสัญญาณความผิดพลาดของสัญญาณ (the error signal), $e_v(nT)$ เป็นอัตราการเปลี่ยนแปลงของสัญญาณความผิดพลาด (the rate of change of the error signal), and K_p , K_{uPI} and K_i เป็นค่าเกณฑ์การควบคุมหาได้จากการออกแบบ การออกแบบการควบคุมแบบฟัซซีพีไอทำโดยอาศัยกระบวนการออกแบบระบบควบคุมฟัซซีตามวิธีมาตรฐาน กล่าวคือ ใช้การทำฟัซซีฟิเคชัน การอนุมานทางฟัซซีและการทำดีฟัซซีฟิเคชัน

(2.1- ข) การทำฟัซซีฟิเคชันของฟัซซีพีไอ

ดังจะเห็นได้จากรูป 2.1 ว่าตัวควบคุมแบบฟัซซีพีไอมีการเพิ่มค่าสัญญาณในรอบควบคุมนั้นเป็น, $\Delta u_{PI}(nT)$ และมีสัญญาณอินพุตสองสัญญาณ คือ $e_p(nT)$ สัญญาณความผิดพลาด และสัญญาณอัตราเปลี่ยนสัญญาณความผิดพลาด $e_v(nT)$ อินพุตและเอาต์พุตบอกระดับความเป็นสมาชิกแสดงดังรูป 2.2 และ 2.3 ตามลำดับ โดยค่าคงที่ที่ปรับค่าได้ $L > 0$ สามารถเลือกค่าจากการออกแบบ



รูปที่ 2.2 ระดับความเป็นสมาชิกของอินพุตของฟัซซีพีไอ



รูปที่ 2.3 ระดับความเป็นสมาชิกของเอาต์พุตของฟัซซี่พีไอ

(2.1- ค) กฎการอนุมานแบบฟัซซี่สำหรับหน่วยฟัซซี่พีไอ

โดยอาศัยระดับความเป็นสมาชิกในรูปที่ 2.2 และ 2.3 เราสามารถสร้างกฎการควบคุมสำหรับฟัซซี่พีไอ จากการวิเคราะห์กระบวนการติดตามเป้าหมายจากสมรรถนะ (a natural tracking control performance analysis) [8] โดยสามารถสรุปได้ดังตาราง 2.1 นี้

ตารางที่ 2.1 กฎการควบคุมของฟัซซี่พีไอ

กฎข้อที่	สถานการณ์	คำแนะนำ สำหรับการติดตามเป้าหมาย ในการควบคุม	ถ้า e_p	และ e_v	แล้ว PI-Output
1	เอาต์พุตอยู่เหนือ จุดตั้งค่าในทิศทาง ออกจากเป้าหมาย	นำเอาต์พุต กลับสู่เป้าหมาย	negative	negative	Negative
2	เอาต์พุตอยู่เหนือ จุดตั้งค่าในทิศลงสู่ เป้าหมาย	ปล่อยให้เอาต์ พุตกลับลงมา เอง	negative	positive	Zero
3	เอาต์พุตอยู่ต่ำกว่า จุดตั้งค่าในทิศมุ่งสู่ เป้าหมาย	ปล่อยให้ตามธรรมชาติ	positive	negative	Zero
4	เอาต์พุตอยู่ต่ำกว่า จุดตั้งค่าในทิศทาง ออกไปจากเป้าหมาย	กลับทิศเอาต์ พุตสู่เป้าหมาย	positive	positive	Negative

หมายเหตุ ในกฎข้างต้นนั้น $e_p := r - y$ เป็นสัญญาณความผิดพลาด (จุดตั้งค่าลบด้วยเอาต์พุต) และ $e_v = \dot{e}_p = 0 - \dot{y} = -\dot{y}$ เป็นอัตราเปลี่ยนแปลงสัญญาณผิดพลาด

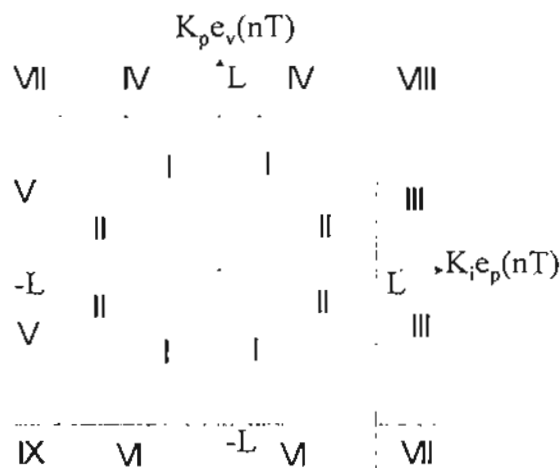
(2.1- ง). การดีฟัซซิฟิเคชันสำหรับฟัซซิฟิไ

จากกฎควบคุมข้างต้น เราจะสามารถหาการเปลี่ยนแปลงของอินพุตในการควบคุมได้โดยอาศัยการทำดีฟัซซิฟิเคชัน ซึ่งในที่นี้เลือกใช้วิธีการหา "จุดศูนย์กลางมวล" โดย

$$\Delta u_{pi}(nT) = \frac{\sum_i \mu_i u_i}{\sum_i \mu_i}, \quad i=1,2,3,4, \quad (3)$$

โดยที่ μ_i เป็นค่าระดับความเป็นสมาชิกที่ "i" และ u_i เป็นเอาต์พุตที่สอดคล้องกับค่าระดับความเป็นสมาชิกดังกล่าว

จากกระบวนการทั้งหมดในข้างต้น เราจะสามารถจำแนกการควบคุมการเปลี่ยนอินพุตได้ตามอาณาเขตต่าง ๆ ออกเป็น 9 มณฑลด้วยกัน โดยอาศัยการพิจารณาตามการจัดหมู่ของสเกลความผิดพลาดทางตำแหน่งและสเกลความผิดพลาดเชิงอัตราเร็ว ดังรูปที่ 2.4



รูปที่ 2.4 การแบ่งมณฑลการจัดหมู่อินพุตของฟัซซิฟิไ

จะได้ การเปลี่ยนหรือเพิ่มค่าอินพุตควบคุม เป็นสูตรสำเร็จดังต่อไปนี้ คือ

$$\begin{aligned}
 \Delta u_{PI}(nT) &= \frac{L[K_i e_p(nT) + K_p e_v(nT)]}{2[2L - K_i |e_p(nT)|]} && \text{มณฑลที่ I,} \\
 &= \frac{L[K_i e_p(nT) + K_p e_v(nT)]}{2[2L - K_i |e_v(nT)|]} && \text{มณฑลที่ II,} \\
 &= \frac{1}{2}[K_p e_v(nT) + L] && \text{มณฑลที่ III,} \\
 &= \frac{1}{2}[K_i e_p(nT) + L] && \text{มณฑลที่ IV,} \\
 &= \frac{1}{2}[K_p e_v(nT) - L] && \text{มณฑลที่ V,} \\
 &= \frac{1}{2}[K_i e_p(nT) - L] && \text{มณฑลที่ VI,} \\
 &= 0 && \text{มณฑลที่ VII,} \\
 &= L && \text{มณฑลที่ VIII,} \\
 &= -L && \text{มณฑลที่ IX.}
 \end{aligned}
 \tag{4}$$

2.2 ตัวควบคุมฟัซซีดี

จากรูปที่ 2.1 จะได้

$$\Delta u_D(nT) = K_d \Delta y(nT) + K y_d(nT) \tag{5}$$

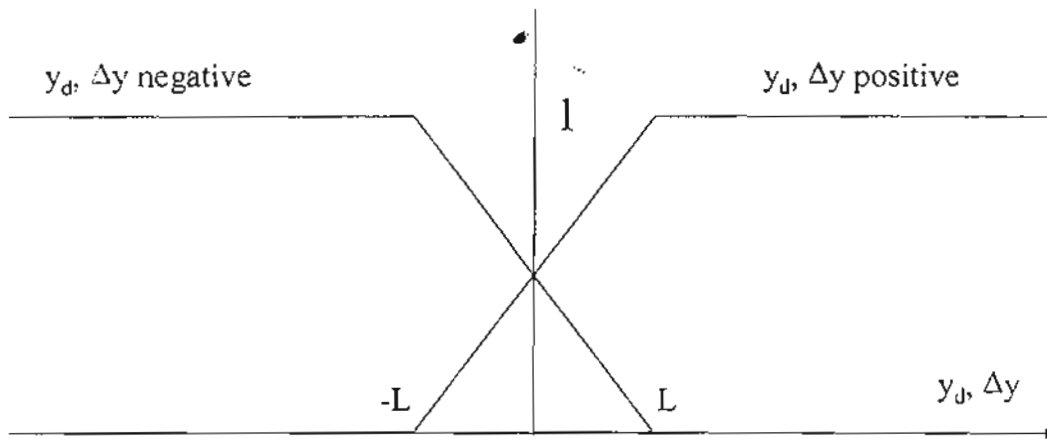
$$u_D(nT) = -u_D(nT - T) + K_{uD} \Delta u_D(nT), \tag{6}$$

โดยที่ $\Delta u_D(nT)$ เป็นอัตราการเปลี่ยนสัญญาณควบคุมฟัซซีดี (the fuzzy control action of the D controller), $y_d(nT)$ เป็นค่าลบของสัญญาณผิดพลาด (the negative error signal), $\Delta y(nT)$ เป็นอัตราการเปลี่ยนแปลงเอาต์พุต (the rate of change of the output), K_D , K_{uD} และ K เป็นค่าเกนคงที่ ที่หาได้จากการออกแบบ

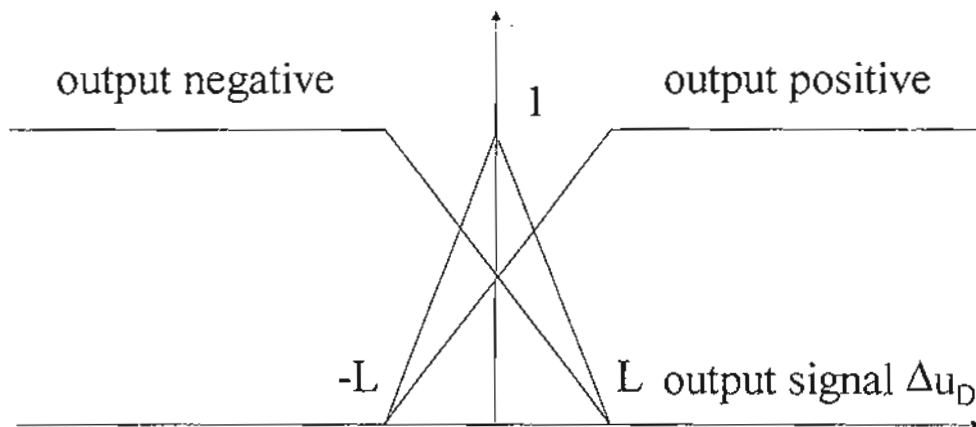
อนึ่ง การออกแบบตัวควบคุมแบบฟัซซีดีนั้น ทำเหมือนกับการออกแบบฟัซซีพีไอในข้างต้น คือ ใช้วิธีมาตรฐานโดยมีกระบวนการเป็น ฟัซซีฟิเคชัน การอนุมานฟัซซี และดีฟัซซีฟิเคชัน ตามลำดับ

(2.2 - ก) การทำฟัซซีฟิเคชันสำหรับหน่วยฟัซซีดี

จากรูป 2.1 จะเห็นได้ว่าบล็อกนี้มีหนึ่งเอาต์พุต คือ อัตราการเปลี่ยนแปลงอินพุตดี $\Delta u_D(nT)$ และสองอินพุตคือ ค่าสัญญาณผิดพลาดที่กลับเครื่องหมาย (negative error signal), $y_d(nT) = -e(nT)$, และค่าอัตราการเปลี่ยนไปของเอาต์พุตของระบบ $\Delta y(nT)$ ค่าระดับความเป็นสมาชิกของอินพุตและเอาต์พุตของฟัซซีดีแสดงดังรูปที่ 2.5 และ 2.6 ตามลำดับ โดยที่ค่าคงที่ $L > 0$ เป็นค่าเดียวกับที่หาได้จากกรณีฟัซซีพีโอ



รูปที่ 2.5 ระดับความเป็นสมาชิกของอินพุตฟัซซีดี



รูปที่ 2.6 ระดับความเป็นสมาชิกของเอาต์พุตฟัซซีดี

(2.2 - ข) การอนุมานฟัซซีดี

ในการอนุมานฟัซซีดีใช้ระดับความเป็นสมาชิกของหน่วยฟัซซีดี ในรูปที่ 2.5 และ 2.6 กฎการอนุมานฟัซซีดีหาได้จากการวิเคราะห์แบบเดียวกับกรณีของฟัซซีพีไอ โดยแสดงได้ตามตาราง 2.2

ตารางที่ 2.2 กฎการควบคุมของฟัซซีพีไอ

กฎข้อที่	สถานการณ์	คำแนะนำ สำหรับการติด ตามเป้าหมาย ในการควบคุม	ถ้า y_d	และ Δy	แล้ว PI-Output
5	เอาร์ทพุทอยู่เหนือ จุดตั้งค่าในทิศทาง ออกจากเป้าหมาย	นำเอาร์ทพุท กลับสู่เป้า หมาย	negative	negative	Zero
6	เอาร์ทพุทอยู่เหนือ จุดตั้งค่าในทิศลงสู่ เป้าหมาย	ปล่อยให้เอาร์ท พุทกลับลงมา เอง	negative	positive	Negative
7	เอาร์ทพุทอยู่ต่ำกว่า จุดตั้งค่าในทิศมุ่งสู่ เป้าหมาย	ปล่อยให้ตามธรรมชาติ	positive	negative	Positive
8	เอาร์ทพุทอยู่ต่ำกว่า จุดตั้งค่าในทิศห่าง ออกไปจากเป้า หมาย	กลับทิศเอาร์ท พุทสู่เป้าหมาย	positive	positive	Zero

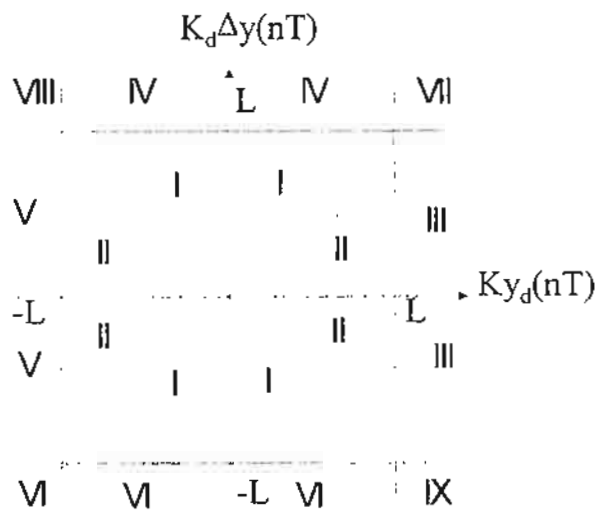
หมายเหตุ จากกฎดังกล่าวข้างต้น $y_d(nT) := y(nT) - r(nT) = -e(nT)$ ซึ่งก็คือ สัญญาณความผิดพลาดที่กลับเครื่องหมาย และ $\Delta y(nT) = \frac{y(nT) - y(nT - T)}{T}$ ก็คือ อัตราการเปลี่ยนแปลงของเอาร์ทพุท y จากตารางดังกล่าวข้างต้นจะเห็นได้ว่าเราออกแบบกฎให้การทำงานของฟัซซีดีและฟัซซีพีไอ ไม่ซ้ำซ้อนกันแต่จะทำงานช่วยเสริมกันในแต่ละช่วง

(2.2 - ค) การตีพัชชีพีเคชันสำหรับพัชชีดี

การตีพัชชีพีเคชันพัชชีดีก็ใช้วิธีเช่นเดียวกันกับของพัชชีพีไอ โดยการเปลี่ยนค่าอินพุทของพัชชีดี $\Delta u_D(nT)$ ที่สอดคล้องกับอินพุท การเปลี่ยนอินพุทได้ตามอาณาเขตต่าง ๆ ออกเป็น 9 มณฑลด้วยกัน โดยอาศัยการพิจารณาตามการจัดหมู่ของสเกลความผิดพลาดทางตำแหน่งและสเกลความผิดพลาดเชิงอัตราเร็ว ดังรูปที่ 2.7 โดยได้สูตรการเปลี่ยนค่าอินพุทดังกล่าวดังนี้

$$\begin{aligned}\Delta u_D(nT) &= \frac{L[Ky_d(nT) + K_d\Delta y(nT)]}{2[2L - K|y_d(nT)|]} && \text{มณฑลที่ I,} \\ &= \frac{L[Ky_d(nT) + K_d\Delta y(nT)]}{2[2L - K|\Delta y(nT)|]} && \text{มณฑลที่ II,} \\ &= \frac{1}{2}[-K_d\Delta y(nT) + L] && \text{มณฑลที่ III,} \\ &= \frac{1}{2}[Ky_d(nT) - L] && \text{มณฑลที่ IV,} \\ &= \frac{1}{2}[-K_d\Delta y(nT) - L] && \text{มณฑลที่ V,} \\ &= \frac{1}{2}[Ky_d(nT) + L] && \text{มณฑลที่ VI,} \\ &= 0 && \text{มณฑลที่ VII,} \\ &= -L && \text{มณฑลที่ VIII,} \\ &= L && \text{มณฑลที่ IX.}\end{aligned}$$

(7)



รูปที่ 2.7 การแบ่งมณฑลการจัดหมู่อินพุทของพัชชีดี

2.3 ตัวควบคุมพีซีพีโอพลัสดี

จากตัวควบคุมแบบพีซีพีโอและพีซีดี ในข้างต้น เมื่อเรานำทั้งสองสัญญาณควบคุมมารวมกันแบบพีชคณิต จะได้กฎการควบคุมแบบพีซีพีโอพลัสดี ในรูป

$$u_{PID}(nT) = u_{PI}(nT) - u_D(nT) \quad (8)$$

แม้ว่าระบบควบคุมที่ได้ออกแบบในบทนี้จะสามารถนำไปประยุกต์ใช้ได้ผลดีใน [8, 10, 17] และแสดงถึงความสามารถในการควบคุมระบบที่ไม่เป็นเชิงเส้น ระบบหน่วงเวลา และระบบที่มีความซับซ้อนได้ผลสัมฤทธิ์ทางการควบคุมที่ดีก็ตาม แต่ค่าเกนหรือพารามิเตอร์ในการควบคุมที่ได้จากการลองผิดลองถูกไม่อาจจะบ่งบอกได้ว่า เป็นค่าที่ดีที่สุด ดังนั้น เพื่อเป็นการขยายสมรรถนะของระบบควบคุมนี้ จึงต้องขยายผล โดยการนำเอาวิธีทางการคำนวณหรือประมวลผลแบบพันธุศาสตร์ (GA) มาใช้ ดังแสดงแนวคิดดังกล่าวในบทต่อไป

บทที่ 3

การออกแบบจีโนฟิซซีฟิไอดี

ดังจะเห็นได้จากรูปที่ 2.1 ในบทที่ 2 ว่า ค่าพารามิเตอร์ในการควบคุมสำหรับระบบฟิซซีฟิไอนั้น ประกอบด้วย $K_p, K_i, K_d, K, K_{uPI}, K_{uPD}$, และ L ดังนั้น พารามิเตอร์เหล่านี้จะใช้เป็นส่วนประกอบของโครโมโซม (Chromosome) ในวิธี GA ซึ่งขณะนี้เป็นการหาปรับค่าที่ดีที่สุด (the optimal values) สุดอันเป็นวิธีมาตรฐานวิธีหนึ่ง ที่ได้เขียนไว้อย่างละเอียดในเอกสารอ้างอิงหมายเลข [25-28, 30-32] ในบทนี้จะกล่าวเฉพาะส่วนที่เกี่ยวข้องกับงานวิจัยนี้เท่านั้น

ด้วยวิธีการทางพันธุศาสตร์จาก [31, 32] เรานิยามโครโมโซม χ ในรูป

$$\chi = \{K_p, K_i, K_d, K, K_{uPI}, K_{uPD}, L\} \quad (9)$$

จากนั้น เราจะใช้ค่าโครโมโซมเหล่านี้โดยอาศัยวิธีการดังนี้คือ

ขั้นตอนที่ 1 เริ่ม โดยกำหนด เวลาตั้งต้น (initial time) $t:=0$

ขั้นตอนที่ 2 กำหนดค่าเริ่มต้นในการสุ่มประชากร (Population) $P(t)$

ขั้นตอนที่ 3 ประเมินผลความลงตัว (fitness) สำหรับแต่ละกลุ่มประชากรในแต่ละตอนที่ t

ขั้นตอนที่ 4 ทดสอบเกณฑ์ประเมิน (test for termination criterion) เช่น เวลา หรือ ความลงตัว เป็นต้น

หากทั้ง 4 ขั้นตอนนี้ยังไม่ให้ผลสัมฤทธิ์ตามที่ต้องการ ให้ทำกระบวนการต่อไปนี้

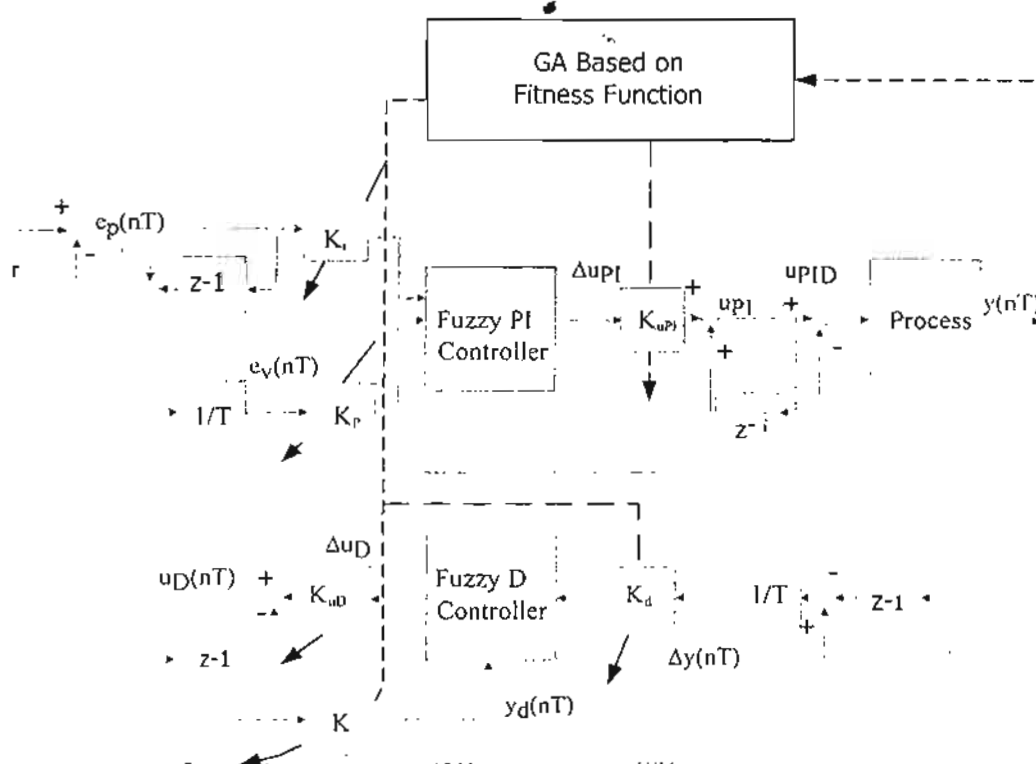
- เพิ่มเวลาของตัวนับ (increase the time counter)
 $t := t + 1$
- เลือกกลุ่มย่อยของประชากร สำหรับเป็นต้นแบบการสร้างรุ่นลูก (offspring production)
 $P' = \text{Select_parents } P(t)$
- จับคู่ "ยีนส์ (genes)" ใหม่จากพ่อแม่ที่เลือกมาได้
Recombine $P'(t)$
- กลายพันธุ์ (mutate) คู่ที่ได้ของประชากรที่คัดสรรไว้ ด้วยวิธีแบบสุ่ม
Mutate $P'(t)$
- ประเมินความลงตัว อีกครั้ง
Evaluate $P'(t)$

- เลือกประชากรที่ผ่านเกณฑ์ประเมิน

$P := \text{Survive } P, P'(t)$

หมายเหตุ หากยังไม่ได้ตรงกับคุณสมบัติที่ต้องการ ทำกระบวนการนี้ซ้ำใหม่อีกครั้ง

ในงานวิจัยนี้ กระบวนการข้างต้นนี้ จะนำมาใช้กับระบบควบคุมฟัซซี่พีไอดี ที่ได้ออกแบบไว้ในบทที่ 2 โดยผลที่ได้จะเป็นระบบควบคุมแบบจีโนฟัซซี่พีไอดี ดังรูปที่ 3.1



รูปที่ 3.1 ระบบควบคุมแบบจีโนฟัซซี่พีไอดี

โดยมีหน่วยปรับเปลี่ยนค่าพารามิเตอร์ของการควบคุมโดยอาศัยวิธีการแบบ GA ดังกล่าวข้างต้น ซึ่งกระบวนการคำนวณที่ใช้ จะอาศัยซอฟต์แวร์พาณิชย์ (commercial software) MATLAB ในการคำนวณ และใช้ขั้นตอนที่สำคัญดังนี้คือ

จะเห็นได้ว่าการให้ได้ค่าความลงตัว Ξ มากที่สุด ก็คือ การทำให้ Ψ มีค่าน้อยที่สุดนั่นเอง เนื่องจากว่าในระบบหุ่นยนต์ที่เราต้องการนำเอาตัวควบคุมแบบจิ้นพัชชีไปใช้นั้น โดยระบบหุ่นยนต์นั้น โดยทั่วไปเป็นหุ่นยนต์ที่ขับเคลื่อนด้วยมอเตอร์ไฟฟ้า ดังนั้น ระบบชนิดนี้อาศัยมอเตอร์ไฟฟ้าเป็นตัวขับเคลื่อน การใช้กำหนดตรรกะนี้สมรรถนะโดยอาศัยค่าต่าง ๆ ดังที่กล่าวมาแล้วในข้างต้น จัดได้ว่าเป็นที่สามารถชี้ผลสัมฤทธิ์ได้ดีดัชนีหนึ่งและสามารถตรวจวัดค่าดัชนีเหล่านี้ได้โดยไม่ยากนัก

อาศัยหลักการ และรหัสเทียม (Pseudo code) ในบทนี้ เราสามารถนำไปจำลองผล (simulation) และทดลอง (experiment) กับระบบหุ่นยนต์ในบทที่ 5 และ 6 ต่อไป และสำหรับในบทที่ 4 ซึ่งเป็นบทถัดไปนั้นเป็นการหาค่าพารามิเตอร์ของมอเตอร์มือสองที่ไม่ทราบค่ามาใช้หาแบบจำลองคณิตศาสตร์ ไว้สำหรับการจำลองผลโดยคอมพิวเตอร์ในบทที่ 5

3.1 ปฏิบัติการทางพันธุศาสตร์ (Genetic operations)

อาศัยวิธีการตาม [31] โดยกำหนดให้โครโมโซมแทนด้วยเลขจำนวนจริง เราจะได้ว่าค่าโครโมโซมที่ ζ สำหรับลูกหลานรุ่น χ' สามารถหาได้จากสมการ

$$\chi'_\zeta = \Phi \chi_\zeta^{(\alpha)} + (1 - \Phi) \chi_\zeta^{(\beta)} \quad (10)$$

โดยที่ $\Phi \in [0, 1]$ และเป็นค่าที่มีการกระจายแบบเอกพันธ์ (uniformly distributed random number) ส่วน $\chi^{(\alpha)}$ และ $\chi^{(\beta)}$ เป็น คู่โครโมโซมของพ่อแม่

3.2 ฟังก์ชันวัตถุประสงค์ (Objective functions)

การกำหนดฟังก์ชันวัตถุประสงค์นี้ ก็เพื่อให้ได้ ค่าเหมาะสมที่สุดตามวัตถุประสงค์ที่วางไว้ การที่จะกำหนดรู้ได้ว่าผลลัพธ์ที่ได้บรรลุถึงวัตถุประสงค์แล้วหรือยัง สามารถตรวจดูได้จากเป้าหมายอ้างอิง $r(t)$ และ เอาท์พุท $y(t)$ สำหรับในงานวิจัยนี้ วัตถุประสงค์ที่ต้องการคือ

- (ก) ลดค่าพุ่งเกินสูงสุดของเอาท์พุท, OS, (Overshoot) ให้ได้ค่าน้อยที่สุด
- (ข) ลดค่าเวลาได้ขึ้นของเอาท์พุท, t_r , (rise time) ให้ได้ค่าน้อยที่สุด
- (ค) ลดค่าเวลาเข้าสู่ภาวะเข้าที่, t_s , (setting time) ให้ได้ค่าน้อยที่สุด
- (ง) ลดค่าผลรวมกำลังสองของความผิดพลาด (Integrated Square Error, ISE) ให้ได้ค่าน้อยที่สุด

3.3 การกำหนดค่าความลงตัว (Fitness assignment)

ค่าฟังก์ชันความลงตัว Ξ (Fitness function) นิยามโดย

$$\Xi = \frac{1}{1 + \Psi} \quad (11)$$

โดย ดัชนีสมรรถนะ (performance index)

$$\Psi = OS + t_r + t_s + ISE \quad (12)$$

บทที่ 4

การระบุค่าพารามิเตอร์ของกระบวนการ

ในบทนี้จะเป็นวิธีการระบุค่าพารามิเตอร์ของกระบวนการ (parameter identification) เพื่อนำไปใช้กับมอเตอร์มือสองที่นำมาใช้ในการทดลองในบทที่ 6 โดยที่ แบบจำลองที่ได้จากวิธีการที่กล่าวถึงในบทนี้ จะนำมาใช้ในบทที่ 5 วิธีการการระบุค่าพารามิเตอร์นั้นมีอยู่หลากหลายวิธีการด้วยกัน ในที่นี้จะกล่าวถึงวิธีการที่ใช้ในที่นี้เท่านั้นคือ การประมาณแบบวิธีการกำลังสองน้อยที่สุด (Least square method) ในกรณีที่มีสัญญาณรบกวนอื่นเจือปน เนื่องจากวิธีการควบคุมที่ใช้ เป็นวิธีการแบบฟัซซี่ตรรกศาสตร์ฟัซซี่ (fuzzy logic based method) ดังนั้นจึงไม่มีความจำเป็นที่จะต้องให้ได้ค่าการประมาณที่แม่นยำสูงมากและใช้วิธีการที่มีความซับซ้อนมาก

1.1 กระบวนการระบุค่าพารามิเตอร์แบบออฟไลน์ (off-line Identification) โดยไม่มีสัญญาณรบกวน

ในส่วนนี้เราจะกล่าวถึง การประมาณค่าพารามิเตอร์ของแบบจำลองเชิงเส้น (linear model) จากข้อมูลอินพุต-เอาต์พุต ซึ่งเป็นพื้นฐานของวิธีการกำลังสองน้อยที่สุดที่จะกล่าวถึงต่อไป วิธีนี้เป็นการสมมติลำดับของแบบจำลอง จากข้อมูลจากอินพุต-เอาต์พุตที่มีอยู่ ในส่วนแรกเราจะพิจารณาถึงวิธีการที่แตกต่างกันไป ในการประมาณค่าพารามิเตอร์ของเวลาที่ไม่ต่อเนื่อง (discrete time) ในส่วนของปัญหาของการประมาณค่าพารามิเตอร์ที่สอดคล้องกับเวลาที่ต่อเนื่อง (continuous time) จะได้กล่าวถึงต่อไป

พิจารณาระบบอินพุต-เอาต์พุตจะแสดงอยู่ในรูป 4.1 โดยใช้วิธีการแปลงแบบซี (Z-transform) โดยที่อินพุต-เอาต์พุต จะมีความสัมพันธ์กันโดย

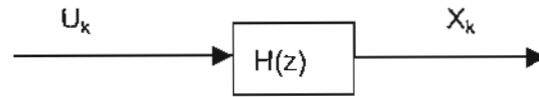
$$\frac{X(z)}{U(z)} = H(z) = \frac{a_0 + a_1 z^{-1} + \dots + a_m z^{-m}}{1 + b_1 z^{-1} + \dots + b_n z^{-n}} \quad (13)$$

โดย $z = e^{st}$ และ T จะเป็นช่วงเวลาของการสุ่ม (sampling time)

สมการที่ (13) อาจจะเขียนได้อีกแบบดังสมการข้างล่างนี้

$$x_k = \sum_{i=0}^m a_i u_{k-i} - \sum_{i=1}^n b_i x_{k-i} \quad (14)$$

โดยที่ $x_l \equiv x(lT)$
 $u_l \equiv x(lT)^*$ $l = 1, 2, \dots$



รูปที่ 4.1 แสดงกระบวนการอย่างง่าย

ดังนั้นอีกปัญหาจะเป็นการหาค่าพารามิเตอร์ $a_0, a_1, \dots, a_m, b_1, \dots, b_n$ จากข้อมูลของ อินพุต-เอาต์พุต จะเป็นการนำค่าของ x_l และ u_l ที่ต่างกันนำมา รวมเข้าไว้ด้วยกันสมการที่ 2 นั้นอาจจะนำมาแสดงให้อยู่ในรูปของสมการเมตริกซ์

$$\begin{bmatrix}
 u_k & u_{k-1} & \dots u_{k-m} & -x_{k-1} & -x_{k-2} & \dots -x_{k-n} \\
 u_{k+1} & u_k & \dots u_{k-m+1} & -x_k & -x_{k-1} & \dots -x_{k-n+1} \\
 \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
 u_{k+p-1} & u_{k+p-2} & \dots u_{k+p-m-1} & -x_{k+p-2} & -x_{k+p-3} & \dots -x_{k+p-n-1}
 \end{bmatrix}
 \begin{bmatrix}
 a_0 \\
 a_1 \\
 \vdots \\
 a_m \\
 b_1 \\
 b_2 \\
 \vdots \\
 b_n
 \end{bmatrix}
 =
 \begin{bmatrix}
 x_k \\
 x_{k+1} \\
 \vdots \\
 x_{k+p-1}
 \end{bmatrix}
 \quad (15)$$

หรือ

$$A'_k \underline{\theta} = \underline{x}_k \quad (16)$$

ซึ่ง

$$A'_k = \begin{bmatrix}
 u_k & u_{k-1} & \dots u_{k-m} & -x_{k-1} & -x_{k-2} & \dots -x_{k-n} \\
 u_{k+1} & u_k & \dots u_{k-m+1} & -x_k & -x_{k-1} & \dots -x_{k-n+1} \\
 \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
 u_{k+p-1} & u_{k+p-2} & \dots u_{k+p-m-1} & -x_{k+p-2} & -x_{k+p-3} & \dots -x_{k+p-n-1}
 \end{bmatrix} \quad (17)$$

$$\underline{\theta} = \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_m \\ b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix} \equiv \text{พารามิเตอร์เวกเตอร์} \quad (18)$$

$$\underline{x}_k = \begin{bmatrix} x_k \\ x_{k+1} \\ \vdots \\ x_{k+p-1} \end{bmatrix} \equiv \text{เอาต์พุตเวกเตอร์} \quad (19)$$

ถ้า A'_k เป็นเมทริกซ์ไม่เป็นเอกฐาน (nonsingular) ($p=m+n+1$) และ $\det$ ของ $A'_k \neq 0$ อาจหาพารามิเตอร์เวกเตอร์ได้จาก

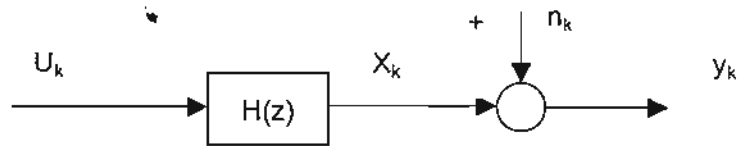
$$\underline{\theta} = (A'_k)^{-1} \underline{x}_k \quad (20)$$

4.2 การประมาณค่าพารามิเตอร์ด้วยวิธีถ่วงน้ำหนักแบบกำลังสองน้อยที่สุด (Weighted least-squares) โดยมีการปนมาของสัญญาณรบกวน (noise-contaminated)

ผลที่ได้จากในหัวข้อ 4.1 ที่ผ่านมาจะเป็นไปได้ในทางทฤษฎีเท่านั้น เนื่องจากการวัดนั้น จะมีสัญญาณรบกวนปนมาด้วยเสมอ ดังนั้นในสถานการณ์ที่เหมาะสมนั้น เราอาจจะใช้แบบจำลองของระบบดังแสดงในรูปที่ 4.2 ซึ่งการวัดที่เอาต์พุตนั้นจะแสดงตามสมการที่ (21)

$$y_i = x_i + n_i \quad (21)$$

โดยที่ n_i จะแทนสัญญาณรบกวนที่วัดได้ที่เอาต์พุต



รูปที่ 4.2 ระบบที่มีสัญญาณรบกวน

เราอาจจะทราบหรือไม่ทราบถึงลำดับของสัญญาณรบกวน (n_i) เราอาจจะสมมติให้ n_i เป็น สัญญาณรบกวนแบบสุ่ม (white Gaussian noise sequence) ถ้าการวัดสัญญาณรบกวน นั้นไม่บริสุทธิ์ บ่อยครั้งที่เราอาจจะพิจารณาที่ เอาท์พุทของการกรองแบบเชิงเส้นด้วยอินพุทของ สัญญาณรบกวนแบบความแปรปรวนหนึ่งหน่วย (unit-variance white Gaussian noise) แทน สมการที่ (21) ในสมการที่ (14) จะได้

$$y_k = \sum_{i=0}^m a_i u_{k-i} - \sum_{i=1}^n b_i y_{k-i} + v_k = \underline{\phi}_k^T \underline{\theta} + v_k \quad (22)$$

ซึ่ง

$$\underline{\phi}_k^T = [u_k u_{k-1} \dots u_{k-m} - y_{k-1} - y_{k-2} \dots - y_{k-n}] \quad (23)$$

และ

$$v_k = n_k + \sum_{i=1}^n b_i n_{k-i} \quad (24)$$

เราจะเรียกว่าเอาท์พุท หรือ สมการความผิดพลาด (equation errors)

สมการที่ (23) อาจจะรวมให้อยู่ในสมการเดียวกันโดยให้

$$A_p \underline{\theta} = \underline{y}_p - \underline{v}_p \quad (25)$$

โดยที่

$$A_p = \begin{bmatrix} u_k & u_{k-1} & \dots & u_{k-m} & -y_{k-1} & -y_{k-2} & \dots & -y_{k-n} \\ u_{k+1} & u_k & \dots & u_{k-m+1} & -y_k & -y_{k-1} & \dots & -y_{k-n+1} \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ u_{k+p-1} & u_{k+p-2} & \dots & u_{k+p-m-1} & -y_{k+p-2} & -y_{k+p-3} & \dots & -y_{k+p-n-1} \end{bmatrix} \quad (26)$$

≡ เมตริกซ์สังเกตการณ์ (observation matrix)

และ

$$y_p = \begin{bmatrix} y_k \\ y_{k+1} \\ \vdots \\ y_{k+p-1} \end{bmatrix} \equiv \text{เวกเตอร์ที่ได้จากการวัด} \quad (27)$$

เพราะการที่มีสัญญาณรบกวนอยู่นั้น เราต้องการมากกว่า $(n+m+1)$ สมการกับการประมาณค่าพารามิเตอร์เวกเตอร์ จากสมการที่ (25) จะได้เงื่อนไขบังคับเป็น

$$p > m + n + 1 \quad (28)$$

ในการแสดงถึงการประมาณค่าของ $\underline{\theta}$ จะอยู่บนพื้นฐานของ เซต p คือเซตของอินพุต-เอาต์พุต เช่นเดียวกับ $\underline{\theta}_p$ ถ้าเราสมมติให้ $\underline{\theta}_p$ เป็นการประมาณค่าของพารามิเตอร์เวกเตอร์ที่เหมาะสม ต่อจากนั้นการประมาณค่าทางเอาต์พุตเวกเตอร์ (y_p) จะแสดงโดย

$$\underline{y}_p = A_p \underline{\theta}_p \quad (29)$$

การประมาณค่าพารามิเตอร์ของเวกเตอร์ความคลาดเคลื่อน (error vector)

$$\tilde{\underline{\theta}}_p = \underline{\theta} - \underline{\theta}_p \quad (30)$$

และเอาต์พุตของเวกเตอร์ความคลาดเคลื่อน

$$\tilde{\underline{y}}_p = \underline{y}_p - \underline{y}_p \quad (31)$$

แทนสมการที่ (25) และ (29) ในสมการที่ (31) จะได้

$$\tilde{\underline{y}}_p = A_p \tilde{\underline{\theta}}_p + \underline{v}_p \quad (32)$$

$\tilde{\underline{y}}_p$ เรียกว่า สมการเวกเตอร์ความคลาดเคลื่อน (vector equation error)

พิจารณาโดยให้ $\underline{\theta}_p$ เป็นการวัดของ สมการเวกเตอร์ความคลาดเคลื่อน ตัวอย่างของกรณีนี้เราจะทำให้น้อยลงโดยใช้วิธีของค่ากำลังสองของนอร์ม (norm-squared) ของ $\tilde{\underline{y}}_p$

$$J = \tilde{\underline{y}}_p^T \tilde{\underline{y}}_p \quad (33)$$

$$\tilde{\underline{y}}_p = \underline{y}_p - \underline{\hat{y}}_p = \underline{y}_p - A_p \underline{\hat{\theta}}_p \quad (34)$$

ดังนั้น

$$J = \tilde{\underline{y}}_p^T \tilde{\underline{y}}_p = \tilde{\underline{y}}_p^T A_p \underline{\hat{\theta}}_p - \underline{\hat{\theta}}_p^T A_p^T \underline{y}_p + \underline{\hat{\theta}}_p^T A_p^T A_p \underline{\hat{\theta}}_p \quad (35)$$

ทำการเลือกค่า $\underline{\hat{\theta}}_p$ ที่ทำให้ J น้อยที่สุด ต่อจากนั้นทำการหาอนุพันธ์ของ J เทียบกับ $\underline{\hat{\theta}}_p$ และได้เท่ากับ 0 ดังนั้น จะได้

$$\underline{\hat{\theta}}_p = (A_p^T A_p)^{-1} A_p^T \underline{y}_p \quad (36)$$

สมการที่ (36) จะได้มาจากการหาอนุพันธ์ สมการที่ (35) เทียบกับ $\underline{\hat{\theta}}_p$

$$\frac{dJ}{d\underline{\hat{\theta}}_p} = 0 - \underline{y}_p^T A_p - [A_p^T \underline{y}_p]^T + 2\underline{\hat{\theta}}_p^T A_p^T A_p = 0$$

หรือ

$$2A_p^T \underline{y}_p = 2A_p^T A_p \underline{\hat{\theta}}_p$$

ให้

$$\underline{\hat{\theta}}_p = (A_p^T A_p)^{-1} A_p^T \underline{y}_p$$

ดังนั้น

$$\frac{d^2 J}{d\underline{\hat{\theta}}_p^2} = 2A_p^T A_p$$

ซึ่งจะสมมาตรและเป็นนิยามเชิงบวก (positive definite) ถ้า A_p มีพิสัยเต็ม (full rank)

หรือ

$$\underline{\theta}_p = A_p^+ \cdot y_p \quad (37)$$

ซึ่ง A_p^+ นั้นจะแทน $(A_p^T A_p)^{-1} A_p^T$ เป็น อินเวอร์สเทียม (pseudoinverse) ของ A_p เมื่อ A_p มีจำนวนแถวมากกว่าจำนวนหลัก และจะเป็นเอกลักษณ์ถ้า A_p มีพิสัยเต็มและมีความเป็นอิสระเชิงเส้น

การคำนวณจะใช้สมการที่ (36) ซึ่งเป็นการประมาณค่าโดยใช้วิธีกำลังสองน้อยที่สุด square วิธีนี้จะมีขนาดเล็ก โดยผลบวกกำลังสองขององค์ประกอบในสมการเวกเตอร์ความคลาดเคลื่อน $\underline{\tilde{y}}_p$

หลักเกณฑ์ของความผิดพลาด โดยทั่วไปจะเกิดขึ้นเมื่อผลบวกกำลังสองขององค์ประกอบของ $\underline{\tilde{y}}_p$ จะมีขนาดเล็กคือ

$$J = \underline{\tilde{y}}_p^T w \underline{\tilde{y}}_p \quad (38)$$

ซึ่ง w จะเป็น เมตริกซ์สมมาตรนิยามบวก (positive definite symmetric matrix) และตัวอย่างในกรณีนี้จะเป็นเมตริกซ์ทแยงมุม (diagonal matrix) คือ

$$w = \text{diag}(w_1, w_2, \dots, w_p) \quad (39)$$

ในกรณีนี้จะเขียนแบบง่ายได้ดังนี้

$$J = \underline{y}_p^T w \underline{y}_p - \underline{y}_p^T w A_p \underline{\theta} - \underline{\theta}^T A_p^T w \underline{y}_p + \underline{\theta}^T A_p^T w A_p \underline{\theta} \quad (40)$$

และก่อนหน้านี้นี้ถ้า J มีค่าน้อยจะได้

$$\underline{\theta}_p = (A_p^T w A_p)^{-1} A_p^T w \underline{y}_p \quad (41)$$

โดย w สมมติให้มีนิยามบวก (positive definite) และสมมาตร โดย A_p เป็นเมตริกซ์ที่มีพิสัยเต็ม (full rank)

หมายเหตุ นำ $w = 1$ จากสมการที่ (41) ไปใส่ในสมการที่ (39) ทำให้วิธีของวิธีแบบกำลังสองน้อยที่สุดนี้ เรียกอีกอย่างหนึ่งว่า ผลเฉลยโดยวิธีการถ่วงน้ำหนักกำลังสองน้อยที่สุด (weighted least-square solution) ซึ่งจะต้องพิจารณาถึงคำถามสำหรับใช้กับสมการที่ (41) ในบทต่อไป จะนำวิธีการที่บรรยายในบทนี้ ไปหาค่าพารามิเตอร์ของมอเตอร์ที่ใช้ในการสร้างหุ่นยนต์ เพื่อให้ได้แบบจำลองทางคณิตศาสตร์ที่สามารถนำไปใช้ได้ในการจำลองผลในบทที่ 5

บทที่ 5

การจำลองผลการควบคุม

โดยอาศัยองค์ความรู้ในบทที่ 2 – 4 บทนี้แสดงผลการจำลองด้วยคอมพิวเตอร์ ในบทนี้จะกล่าวถึง ผลที่ได้จากการจำลองเท่านั้น ส่วนต้นรหัสของซอฟต์แวร์ (Source code) เป็นลิขสิทธิ์ในส่วนของนักวิจัยที่เลี้ยงที่สร้างกล่องเครื่องมือ (tool box) สำหรับวิธีทางพันธุศาสตร์ ณ ที่นี้ จึงแสดงเฉพาะผลการวิจัยเท่านั้น

มอเตอร์ที่นำมาทดลองเป็นมอเตอร์มือสองที่ไม่ทราบค่าพารามิเตอร์ เพื่อเป็นการจำลองผลที่ได้ จึงนำเอาค่าอินพุตและเอาต์พุตที่ได้มาทำการระบุค่าพารามิเตอร์ตามวิธีดังกล่าวข้างต้น ในบทที่ 4 ได้ค่าพารามิเตอร์ตามแบบจำลอง ฟังก์ชันถ่ายโอนในโดเมนซี้ $H(z)$ และแปลงกลับโดยอาศัย การแปลงอินเวอร์สซี้ เป็นโดเมนความถี่เอส $H(s)$ (Transfer function in continuos domain) ได้ค่าพารามิเตอร์สำหรับ AC motor ที่ใช้เป็น

$$H(s) = \frac{0.6347}{1.03851s + 0.1541} \quad (42)$$

หรือ ในรูปสมการปริภูมิสถานะในโดเมนเวลาแบบไม่ต่อเนื่องเป็น

$$x(k+1) = G * x(k) + H * u(k) \quad (43)$$

$$y(k) = C * x(k) \quad (44)$$

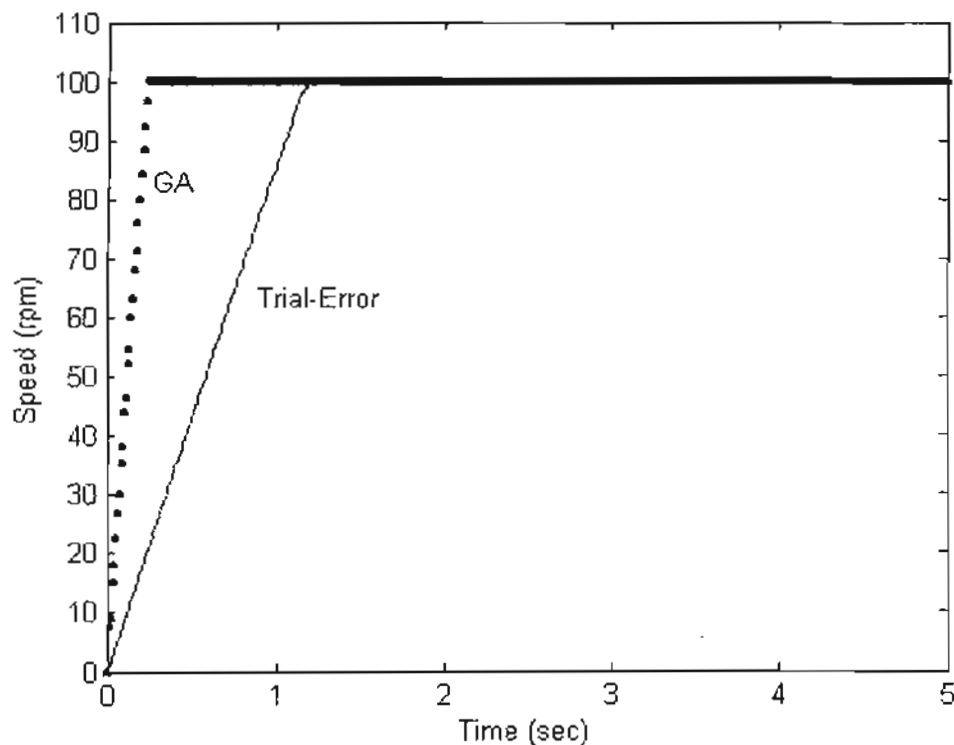
โดยที่

เมตริกซ์ระบบ $G = 0.99846018673084$

เวกเตอร์อินพุต $H = 0.00999229895628$

เมตริกเอาต์พุต $C = 15.31070475970381$

อนึ่ง ค่าพารามิเตอร์ที่ระบุได้นี้ เป็นค่าพารามิเตอร์ที่ได้จากนำอินพุตเป็นแรงดัน (V) กับ ค่าเอาต์พุตเป็นรอบของมอเตอร์วัดโดยใช้เอนโคเดอร์ เมื่อนำค่าฟังก์ชันถ่ายโอนที่ได้ไปใช้ เพื่อจำลองผลตรวจสอบการออกแบบในบทที่ 2 และ 3 โดยปรับค่าพารามิเตอร์โดยการลองผิดลองถูก และวิธีของ GFPID โดยกำหนดให้เงื่อนไขการหยุดจับคู่ยีนส์หรือหยุดการกลายพันธุ์ที่ ฟังก์ชันความลงตัว $E > 0.001$ ได้ผลตามรูป 5.1 ในรูปนี้เป็นการจำลองสถานการณ์ที่ความเร็ว รอบของมอเตอร์เปลี่ยนจาก 1000 รอบต่อนาที ไปเป็น 1,100 รอบต่อนาที ตาราง 5.1 แสดง สมรรถนะของการติดตามเป้าหมายในรูปที่ 5.1 โดยเปรียบเทียบการวิธีกำหนดค่าพารามิเตอร์ที่ ใช้ในการควบคุมด้วยการลองผิดลองถูกโดยพยายามปรับค่าให้ได้ OS น้อยที่สุดหรือไม่มีเลย กับแบบ GFPID จะเห็นได้ว่าค่าที่ได้จากวิธี GFPID ดีกว่ามาก



รูปที่ 5.1 เปรียบเทียบผลการจำลองระบบควบคุมแบบฟัซซีฟัซซีไอดีโดยวิธีลองผิดลองถูกกับวิธีทางพันธุศาสตร์

ตาราง 5.1 เปรียบเทียบผลการจำลองระบบควบคุมแบบฟัซซีพีไอดีโดยวิธีลองผิดลองถูกกับ
วิธีทางพันธุศาสตร์

	โดยวิธีลองผิดลองถูก	โดยวิธีการทางพันธุศาสตร์
$K_p, K_b, K_D, K,$ K_{uph}, K_{wd}, L	2, 0.6, 1, 1.6 1.5, 0.2, 350	1.26, 1.015, 0.885, 0.989 2.06, 0.12, 1145.63
$t_r(s)$	1.12	0.19
$t_s(s)$	0.93	0.24
$OS(\%)$	0	0.14281
ISE	3997.89071	916.70472
Ψ	3999.94071	917.27753
Ξ	0.00025	0.00109

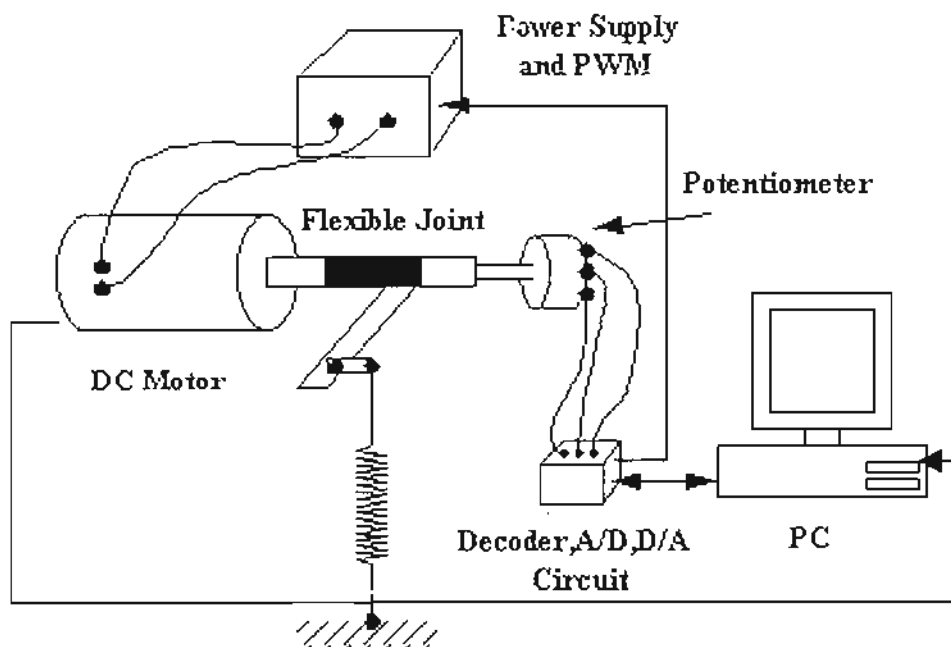
แม้ว่าผลที่ได้ในรูปที่ 5.1 จะแสดงให้เห็นว่า วิธีการปรับค่าแบบ GA ให้ผลที่ดีกว่าการปรับค่าแบบลองผิดลองถูก อย่างไรก็ตาม ในด้านของฮาร์ดแวร์โดยเฉพาะอย่างยิ่งในเรื่องของค่าเวลาได้ขึ้นนั้น ในทางปฏิบัติแม้จะใช้เกนขยายเพียงใดก็ตาม แต่ขีดจำกัดของฮาร์ดแวร์ทำให้ไม่สามารถที่จะเร่งค่าเวลาได้ขึ้นให้สูงกว่าขีดจำกัดสูงสุดได้ การทำปรับค่าทางวิธีการทางพันธุศาสตร์จึงควรกำหนดเงื่อนไขรวมเข้าไปในวิธีการประเมินผลด้วย ในกรณีที่ค่าพุ่งเกินหรือพารามิเตอร์ของฟังก์ชันวัตถุประสงค์สำหรับการประเมินผลใด ๆ มีผลต่อการประยุกต์ใช้งาน ผู้ออกแบบควรใช้วิธีการถ่วงน้ำหนักมากกว่าให้ค่าเท่ากันดังที่ผ่านมา

ในบทที่ 6 จะเป็นการประยุกต์ใช้งานตัวควบคุมแบบจีโนฟัซซีพีไอดี

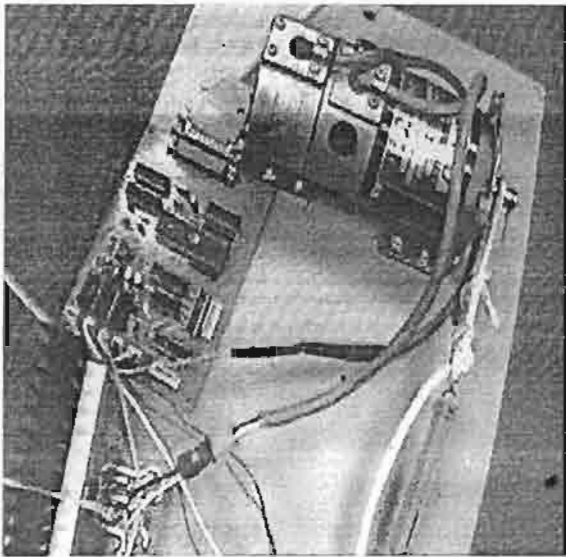
บทที่ 6

การประยุกต์ระบบควบคุมกับระบบหุ่นยนต์

ในบทนี้ เป็นการประยุกต์ใช้งานตัวควบคุมที่ได้ออกแบบไว้ในข้างต้นกับระบบหุ่นยนต์ โดยระบบดังกล่าวเป็นระบบของแขนกลข้อต่ออ่อนตัว (Flexible joint robot arm) [13, 14] โดยมีสภาวะรบกวนจากภาระที่เปลี่ยนไปตลอดเวลา ในการทดลองนี้ใช้สปริงที่ไม่ทราบค่าพารามิเตอร์เชิงรบกวน โดยในการทดลองดังกล่าวนี้เป็นการทดลองที่ขนานกับการจำลองผล กล่าวคือ ใช้วิธีการควบคุมความเร็วรอบของมอเตอร์ และเพื่อตรวจสอบถึงความครอบคลุมของการออกแบบในการทดลองนี้จึงลองเปลี่ยนจากมอเตอร์กระแสสลับ เป็นมอเตอร์กระแสตรงที่ไม่ทราบค่าพารามิเตอร์ เพื่อที่จะทดสอบสมรรถนะของระบบควบคุมที่สามารถควบคุมได้แม้เปลี่ยนชนิดของมอเตอร์ และไม่ทราบค่าพารามิเตอร์ดังกล่าวเลย อย่างไรก็ตาม หากทราบค่าพารามิเตอร์โดยหาได้จากวิธีการระบุค่าพารามิเตอร์ดังแสดงไว้ในบทที่ 4 ก็จะทำให้การออกแบบระบบได้ค่าพารามิเตอร์ตั้งต้นใกล้เคียงกับค่าจริงมากยิ่งขึ้นอีกด้วย การติดตั้งการทดลองแสดงในรูป 6.1 ในรูปที่ 6.2 เป็นภาพถ่ายอุปกรณ์จริงที่ใช้ในการทดลอง



รูปที่ 6.1 แสดงการติดตั้งอุปกรณ์ในการทดลอง



รูปที่ 6.2 อุปกรณ์การทดลองก่อนติดตั้งแขนระยางจากท่อ พีวีซี ที่ปลายมอเตอร์กับภาระ

สัญญาณป้อนกลับผ่าน ISA Slot [15] บันทึกทุก 0.4 วินาที ได้ผลการทดลองดังรูปที่ 6.3 และรูปที่ 6.4 รูปที่ 6.3 แสดงผลการทดลองเปรียบเทียบสมรรถนะ ระหว่างการควบคุมแบบ GFPID ที่นำเสนอและอีกสองวิธีการควบคุมความเร็วมาตรฐานที่ใช้กันอยู่ในภาคอุตสาหกรรมทั่วไป คือการควบคุมแบบ เฟสล็อกกลูป (Phase-locked loop, pll) และวิธีแบบพีไอดีดั้งเดิม (PID) ค่าพารามิเตอร์ที่ใช้ในที่นี้คือ

สำหรับการควบคุมแบบพีไอดี

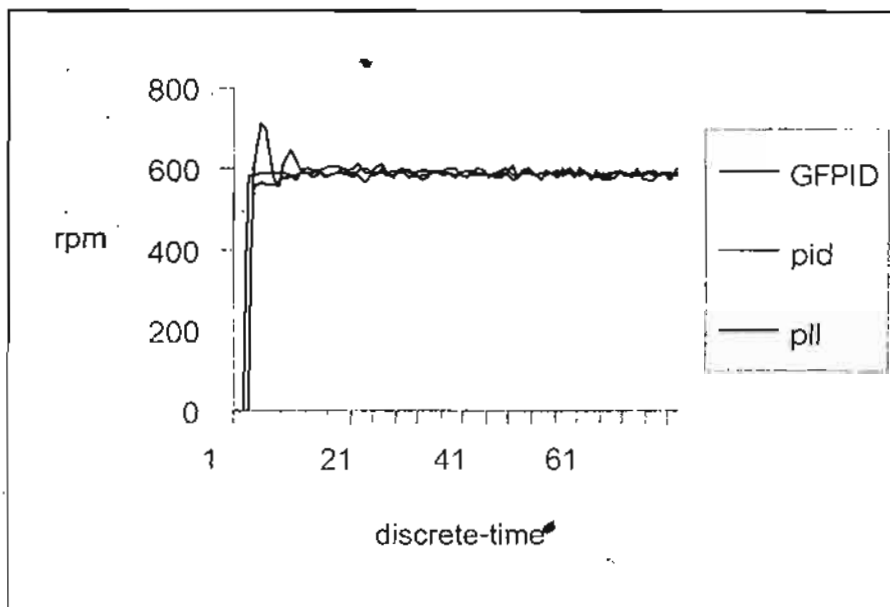
$$K_p = 0.5, K_i = 0.1, K_d = 0.1, T = 0.01 \text{ sec}$$

สำหรับ GFPID

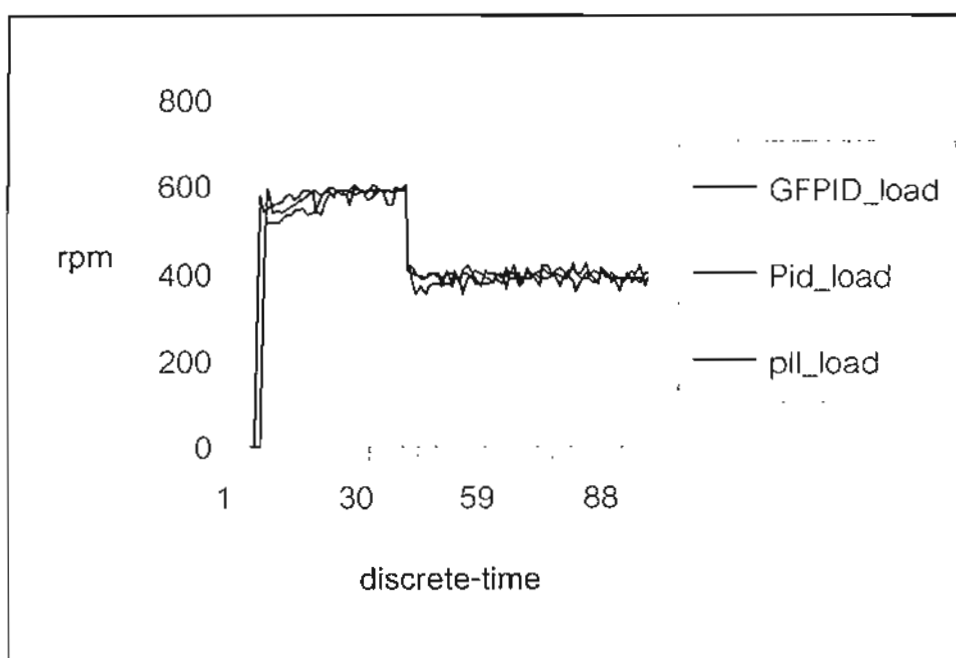
$$K_p = 0.1, K_i = 0.1, K_d = 0.1, K = 1, K_{uPI} = 0.1, K_{uPD} = 0.2, L = 5, T = 0.01 \text{ sec}$$

สำหรับวงจรเฟสล็อกกลูป (เป็นวงจรอนาล็อก)

$K_\phi = 0.111 \text{ V/rad}$ (gain constant for the MC4044 phase detector), $R_1 = 10 \text{ M}\Omega$, $R_2 = 1.5 \text{ M}\Omega$, $C = 0.1 \text{ }\mu\text{F}$ (parameter for passive loop filter), $K_1 = 1.2 \text{ V/V}$ (motor drive gain), $K_2 = 159.236 \text{ (pulse/rad)}$, $K_3 = 0.1$ (the counter modulus 100)



รูปที่ 6.3 เปรียบเทียบวิธีการแบบ GFPID กับวิธีมาตรฐาน PID และ PLL ที่ใช้ในปัจจุบัน



รูปที่ 6.4 เป็นการเพิ่มภาระ 2 kg เข้าไปในระบบ ในขณะที่ยังคงการรบกวนจากสปริงที่ไม่ทราบค่า

จากผลการทดลองทั้งสองรูปจะเห็นได้ว่า GFPID ไม่มีค่าพุ่งเกินเหมือนกับวิธี PID ในขณะเดียวกันให้ผลตอบสนองดีกว่าวิธีแบบ PLL ซึ่งวิธีแบบ PLL นี้เป็นวิธีที่ให้ผลตอบสนองที่ดีในช่วงสภาวะ steady state อย่างไรก็ตาม เนื่องจากการรบกวนของสปริงซึ่งเป็นภาระแปรผันตามเวลา ทำให้เกิดการสั่นรบกวน ในกรณีนี้เป็นการรบกวนอย่างมากจึงทำให้ในช่วงสภาวะคงตัวเกิดการกระเพื่อมของสัญญาณ แม้ว่า GFPID จะให้ค่าพารามิเตอร์การควบคุมที่เหมาะสมสำหรับระบบ แต่ในกรณีที่เกิดการรบกวนอย่างรุนแรง (strong disturbance, not perturbation) พารามิเตอร์ในการควบคุมเหล่านี้ไม่สามารถปรับเปลี่ยนตัวเองได้ (self-tuning) เนื่องจากวิธีการแบบ GA ที่ใช้เป็นวิธีการแบบ “off-line” optimization ไม่ใช่ แบบ “on-line” ข้อเสนอแนะ เพื่อแก้ปัญหาต่อไปจากการทดลองนี้เสนอไว้ในบทที่ 7

บทที่ 7

บทสรุป

ในงานวิจัยนี้ การปรับปรุงการปรับเปลี่ยนค่าพารามิเตอร์การควบคุมสามารถทำได้ด้วยวิธีการพันธุศาสตร์ การจำลองผลและการทดลองซึ่งให้เห็นผลสัมฤทธิ์ของการประยุกต์ใช้งานทางด้านการควบคุมความเร็วในการเคลื่อนที่ของหุ่นยนต์ข้อต่ออ่อนตัว บ่งชี้ถึงความสามารถในการประยุกต์ใช้งานกับการขับเคลื่อนหุ่นยนต์เคลื่อนที่อื่น ๆ โดยใช้ควบคุมความเร็วทั้งในระบบที่ใช้ตัวขับเคลื่อน (actuator) ด้วยกำลังไฟฟ้าได้ทั้งแบบกระแสสลับและกระแสตรง

สำหรับการควบคุมตำแหน่งนั้น ไม่ได้แสดงให้เห็นในการจำลองผลและการทดลองนี้ เนื่องจากว่า [8, 10, 17, 22] ได้แสดงความสามารถในการควบคุมตำแหน่งของระบบควบคุมแบบ FPID นี้แล้วแม้ว่าค่าพารามิเตอร์ที่ใช้ในการควบคุมดังกล่าวจะใช้วิธีแบบลองผิดลองถูกก็ตาม แต่วิธีการปรับค่าเกณฑ์ทำได้ในทำนองเดียวกันโดยไม่ยาก

จากการจำลองผลจะเห็นได้ว่าผลที่ได้จากการประยุกต์ใช้ GA ในการปรับเปลี่ยนค่าพารามิเตอร์ในการควบคุมดีกว่าวิธีลองผิดลองถูก แต่อย่างไรก็ตาม จากบทที่ 6 ในการประยุกต์ใช้งานเราจะเห็นได้ว่า ค่าเวลาได้ขึ้นของวิธีการแบบ GA และ แบบลองผิดลองถูกในการทดลองนั้นจะใกล้เคียงกันเนื่องจากข้อจำกัดทางด้านฮาร์ดแวร์ ดังนั้น ในการใช้งานจริงค่าที่ควรถ่วงน้ำหนักให้มากในการใช้งานจริงก็คือ ค่าของความคงทนต่อสัญญาณรบกวนและภาระที่เปลี่ยนแปลง (loads)

จากการที่การปรับค่าพารามิเตอร์แบบ GA เป็นการปรับแบบ "off-line" ดังนั้น เมื่อมีสัญญาณรบกวนหรืออินพุตรบกวนที่รุนแรง ระบบไม่อาจที่จะควบคุมโดยการปรับพารามิเตอร์เอง (self-tune) ให้มีภาวะคงตัว (steady state) ที่ราบเรียบ ซึ่งหากการรบกวนรุนแรงเกินกว่าสภาวะคงทน ระบบอาจเกิดสภาวะไร้เสถียรภาพ ซึ่งค่าพารามิเตอร์ต่าง ๆ ควรได้รับการออกแบบให้อยู่ภายใต้เงื่อนไขเสถียรภาพ [5, 8, 9, 11, 16] เพื่อเป็นการปรับปรุงให้ระบบมีความสามารถในการปรับตัวแบบ "on-line" วิธีการใน [18-20] อาจสามารถนำมาดัดแปลงประยุกต์ใช้กับ FPID ให้มีความสามารถปรับค่าพารามิเตอร์ควบคุมได้ทันทีทันใด

อนึ่ง วิธีการหาอนุमानแบบเลือกค่าน้อย (MIN) ในการออกแบบ FPID อาจสามารถใช้วิธีอื่นได้ ดังแสดงใน [22] นอกจากนี้ตัวตรวจวัด (sensor) ที่ใช้ ควรได้รับการปรับปรุงให้มีขีดความสามารถเชิงปัญญา [23-24] ซึ่งจะเป็นหัวข้อและทิศทางการวิจัย อันมีแนวโน้มควรค่าต่อการติดตามเพื่อดำรงไว้ซึ่งภูมิปัญญาไทยในการพัฒนางานวิจัยทางด้านวิศวกรรมต่อไป หรืออย่างน้อยก็เพื่อที่จะเป็น "ผู้ฉลาดซื้อ" ณ ภาวะปัจจุบัน

เอกสารอ้างอิง

1. G. Chen, "Conventional and fuzzy PID controller: an overview", *Int. J. Intelligent Control System*, pp. 235-246, 1996.
2. G. J. Silva, A. Datta, and S. P. Bhattacharyya, "New results on the synthesis of PID controllers," *IEEE Transactions on Automatic Control*, vol.47, iss. 2, pp.241-252, 2002.
3. A. Visioli, "Tuning of PID controllers with fuzzy logic," *IEE Proc. Control Theory and Applications*, vol. 148, iss 1, pp. 1-8, 2001.
4. A. Isaksson, and T. Hagglund, "Editorial - pid control," *IEE Proc. Control Theory and Applications*, vol. 149, iss 1, pp. 1-2, 2002.
5. D. Misir, H. A. Malki, and G. Chen, "Design and Analysis a Fuzzy Proportional Derivative Controller", *Fuzzy Set and System*, pp.297-314, 1996.
6. G.K.I. Mann, Bao-Gang Hu; and R. G. Gosine, "Two-level tuning of fuzzy PID controllers," *IEEE Transactions on Systems, Man and Cybernetics, Part B*, vol.31, iss. 2, pp. 263 –269, 2001.
7. M. Uragami, M. Mizumoto, K. Tanaka, "Fuzzy robot controls", *J. Cybernet.*, pp. 39-64, 1976.
8. P. Sooraksa, and G. Chen, "Mathematical modeling and fuzzy control for flexible-link robots," *Journal of Mathematical Computing and Modeling*, vol. 27, no. 6, pp. 73-93, 1998.
9. G. Chen and H. Ying, "BIBO stability of nonlinear fuzzy PI control systems," *J. Intel. Fuzzy Syst.*, vol. 5, pp. 3-21, 1997.
10. H. A. Malki, D. Feigenspan, D. Misir, and G. Chen, "Fuzzy PID control of a flexible joint robot arm with uncertainties for time-varying load", *IEEE Trans. Control System Technology*, pp. 371-378, 1997.
11. H. A. Malki, H. Li, and G. Chen, "New design and stability analysis of fuzzy proportional-derivative control system", *IEEE Trans. Fuzzy System*, pp.245-254, 1994.
12. T. Brehm, and K. S. Rattan, "Hybrid fuzzy logic PID controller," *Proc. of the IEEE* 1993
National Aerospace and Electronics Conference, vol.2, pp. 807 – 813, 1993.
13. M. Spong, "Modeling and control of elastic joint robot", *ASME J. Dyn. System, Meas. Control* 109, pp. 146-155, 1987.

14. M. C. Readman, "Flexible joint robots", CRC Press, 1994.
15. M. A. Mazadi, J. G. Mazadi, "The 80x86 IBM PC & Compatible Computer Volume II: Design and Interfacing of the IBM PC, PS and Compatible", Prentice-Hall. 1995.
16. H. Ying, W. Siler, J.J. Buckley, Fuzzy control theory: a nonlinear case, Automatica, pp. 513-520, 1990.
17. W. Tang, G. Chen, and R. Lu, "A modified fuzzy PI controller for a flexible-joint robot arm with uncertainties," Fuzzy Sets and Systems, vol. 118, pp. 109-119, 2000.
18. Implementation of self-tuning controllers, Edited by K. Warwick, Peter Peregrinus Ltd., IEE., 1988.
19. D. Popovic, "Intelligent control with neural network," in Soft Computing & Intelligent Systems: Theory & Applications, Edited by N. K. Sinha, and M. M. Gupta, Academic Press: 1999.
20. C-C Lee, Intelligent control based on fuzzy logic and neural network theory, Ph.D. Dissertation, University of California, Berkeley, 1990.
21. B. M. Mohan, and A. V. Patel, "Analytical structures and analysis of the simplest fuzzy PD controllers," IEEE Trans. on Systems, Man, and Cybernetics, Part B, vol. 32, no. 2, pp. 239-248, 2002.
22. T. Pattaradej, P. Chandang, M. Lertwatechakul, P. Sooraksa, and G. Chen, "A speed evaluation for conventional PID and fuzzy controllers," Proceedings of the 2nd International Conference on Intelligent Technologies, pp. 91-96, 2001.
23. J. V. Spiegel, "Computational Sensors: The basis for truly intelligent machines," in "Handbook of Sensors and Actuators," edited by H. Yamasaki, Vol 3., Elsevier, Amsterdam, pp. 19-37, 1996.
24. H. Yamasaki, "What are the intelligent sensors," "____", pp. 1-17, 1996.
25. Practical Handbook of Genetic Algorithms: *Applications*, Edited by Lance Chamberts, Vol 1., CRC Press., New York, 1995.
26. Practical Handbook of Genetic Algorithms: *New Frontiers*, Edited by Lance Chamberts, Vol 2., CRC Press., New York, 1995.
27. Practical Handbook of Genetic Algorithms: *Complex Coding Systems*, Edited by Lance Chamberts, Vol 3., CRC Press., New York, 1995.
28. B. Soucek, and the IRIS Group, Dynamics, Genetic, and Chaotic Programming: The Sixth Generation, John Wiley & Sons, Inc., New York, 1992.
29. <http://ieeexplore.ieee.org/Xref/DynWel.jsp>

30. J. H. Holland, "Adaptation in Natural and artificial Systems," MIT Press, Cambridge, MA, 1975.
31. Z. Michalewicz, "Genetic Algorithms + Data Structures = Evolution Program," 3rd., Springer-Verlag, New York, 1996.
32. K. F. Man, K. S. Tang, and S. Kwong, Genetic Algorithms: Concepts and Designs, Springer-Verlag, London, 1999.

ภาคผนวก

ผลงานตีพิมพ์ และ ผลพลอยได้

----- Original Message -----

From: Prof. Hojjat Adeli

To: P. Sooraksa

Sent: Sunday, April 14, 2002 3:54 AM

Subject: Submission to ICAE

Dear Dr. Sooraksa:

The attached letter and accompanying reviews will be sent to you by airmail on Monday.

Does your library subscribe to ICAE?

At 10:18 PM 3/21/2002 +0800, you wrote:

Dr. Pitikhate Sooraksa

Dept. of Electronic Engineering,

City University of Hong Kong,

83 Tat Chee Ave.,

Kowloon, Hong Kong.

Regards,

Hojjat Adeli

Professor of Biomedical Informatics

Professor of Civil and Environmental Engineering and Geodetic Science

Director, Knowledge Engineering Lab

Editor-in-Chief, Integrated Computer-Aided Engineering - An International Journal

College of Engineering

The Ohio State University

470 Hitchcock Hall, 2070 Neil Avenue

Columbus, OH 43210 U.S.A.

Voice and Fax: 614-292-7929

Email: Adeli.1@osu.edu

Integrated Computer-Aided Engineering (<http://www.iospress.nl>)

April 13, 2002

Dr. Pitikhate Sooraksa
Dept. of Electronic Engineering,
City University of Hong Kong,
83 Tat Chee Ave.,
Kowloon, HONG KONG

Re: Design of Fuzzy P²ID Controller for Handlebar of a Bicycle Robot
P. Sooraksa, T. Pattaradej, and G. Chen

Dear Dr. Sooraksa:

Enclosed please find the comments of three reviewers. I am pleased to inform you that your manuscript will be accepted for publication in **Integrated Computer-Aided Engineering** provided that you revise it according to the enclosed comments of the reviewers. Only unpublished original manuscripts are considered for possible publication in ICAE. Submission of a manuscript implies that your manuscript or any variation of it has not been submitted elsewhere. If this manuscript is extension of your previously-published paper we ask you to send us a copy of that paper promptly so that our reviewers can determine the archival significance and scientific originality of the current submission compared with your previous publication. In revising your manuscript you should please concentrate on your own original contribution with archival values such as computational models, theories, and algorithms and creative research discoveries and leave out materials that can be found elsewhere.

Please send me four copies of the revised manuscript typed double-spaced and in single-column, prepared according to the format of the journal (see <http://www.iospress.nl> for format instructions), a set of high quality camera-ready black&white figures (each printed on a separate sheet), an electronic copy on diskette, and an explanation of what changes you have made and how you have satisfied the reviewers concerns. The length of each manuscript should be limited to 40 double-spaced pages including tables and figures. Non-adherence to the formatting style of the journal will delay the publication process. Thank you for your cooperation and helping us maintain a high standard of publication.

Sincerely yours,

Hojjat Adeli
Professor

----- Original Message -----

From: Prof. Hojjat Adeli

To: P. Sooraksa

Sent: Thursday, May 16, 2002 4:33 AM

Subject: ICAE

Dear Dr. Sooraksa:

Thank you for revising "the manuscript carefully based upon the comments to produce a good paper." Please send the manuscript and disk to me by Express Mail ASAP. If I receive your properly revised manuscript soon I may be able to include it in issue 9:4, the last issue of 2002.

At 04:21 PM 5/9/2002 +0800, you wrote:

Dear Prof. Adeli,

Many thanks for sending me the comments from the reviewers. I just received the letter today. My colleagues and I will revise the manuscript carefully based upon the comments to produce a good paper.

Best Regards,

P. Sooraksa

Regards,

Hojjat Adeli

Professor of Biomedical Informatics

Professor of Civil and Environmental Engineering and Geodetic Science

Director, Knowledge Engineering Lab

Editor-in-Chief, Integrated Computer-Aided Engineering - An International Journal

College of Engineering

The Ohio State University

470 Hitchcock Hall, 2070 Neil Avenue

Columbus, OH 43210 U.S.A.

Voice and Fax: 614-292-7929

Email: Adeli.1@osu.edu

Integrated Computer-Aided Engineering (<http://www.iospress.nl>)

Dr. Pitikhate Sooraksa
Dept. of Electronic Engineering,
City University of Hong Kong,
83 Tat Chee Ave.,
Kowloon, HONG KONG

Phone: 852-2788-9938
Email: sooraksa@ee.cityu.edu.hk

May 17, 2002

Prof. Hojjat Adeli, Editor-in-Chief
College of Engineering
The Ohio State University
470 Hitchcock Hall
2070 Neil Avenue
Columbus, Ohio 43210-1275
U.S.A.

**Re: Design and Implementation of Fuzzy P²ID Controller for Handlebar
Control of a Bicycle Robot**

Dear Prof. Adeli, Editor-in-Chief, and the reviewers:

Enclosed please find four copies of our revised manuscript typed in double-space and in the single-column format, revised according to the comments of the reviewers and the format of the journal.

In the revised paper, the following items have been changed for improvement and enhancement of the presentation, according to the reviewers' suggestions:

1. Abstract: The abstract has been reduced to fit the required length of no longer than 200 words.
2. Introduction:
 - 2.1 The first paragraph has been modified to make a clear statement about the design of the fuzzy logic controllers in correspondence with the comments of Reviewer 3. Knowledge of effects and calculation of poles and zeroes of the controller and the plant are not required and are not major concerned of our design of the proposed fuzzy controller. Review 3 may bare

in mind the conventional linear control theory but not the aspect of a practical approximate reasoning using fuzzy logic for nonlinear systems.

- 2.2 The fourth paragraph is modified to concisely explain about the criteria of acceptable performance. In fact, when one rides a bicycle, he does not expect a perfect tracking for zero steady-state error, but usually within a small tolerance, which would be acceptable as long as one can control the bicycle body to not to fall down to the ground and can go along in the desired direction. We followed this principle in our design.

Body of Text:

- 2.3 Sections 2 and 3: All subsections are now condensed. Figures and tables have been double-checked. The "typo" in formulas as mentioned by Reviewer 1 is now corrected. We have checked the computer code for the controller from the previous version and the code was correct—only the formulas in the text of the previous manuscript were "typos", not the code, fortunately.
- 2.4 Section 4: We have now provided more insights for the derivation of the proposed controller by illustrating and providing explanations for the missing Fig. 9 as mentioned by Review 1. The missing figure, Fig. 9, has been fixed. The switching criterion is depicted graphically and used the theoretical analysis of the local stability criteria of fuzzy PI and fuzzy PD controllers from [4] and [8], to intuitively establish a new criterion based on BIBO stability at the switching time by the selector.
- 2.5 Section 5: We have contributed more results based on different sets of data, which have more perturbations than the previous one, to illustrate the effectiveness of the proposed control scheme, as compared to the conventional PID controller. Error signals and control input signals are also presented to convince the bounded input signals produced at the transition point between the two controllers. We have also re-plotted the graph by using MS Excel instead of the rough scale of our own software using a C compiler. The results look

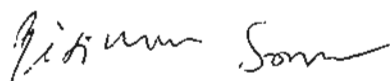
much clearer and nicer than the previous ones. As a result, it is now convinced that our proposed control scheme is distinguishably different than those in comparison, as depicted by Figs. 12 - 15.

2.6 Improvement of the body text has been done corresponding to the constructive comments of the reviewers.

3. Conclusion: There is no essential change in the conclusion.
4. References: The italic character used to indicate names of journal are corrected to be normal. The format is used based on the provided information by the publisher. Ref [18] has been added to support the claim in Section 4. Ref [4] has been updated.
5. Tables: Tables 1 and 2 associated with the references, which could confuse the readers as mention by Reviewer 1, have been modified.
6. Figure captions and Figures: We have added more figures in Section 5 to enhance and convince the results obtained from our experiments, as suggested by Reviewer 1, and to make them clearer and more readable than the ubiquitous ones presented previously, following the suggestions of Reviewer 3. Figure 10 is also updated.

Once again, we would like to thank all reviewers and the editor for their helpful comments and suggestions.

Sincerely yours,



Pitikhate Sooraksa, Ph.D.

Cc: Mr. T. Pattaradej and Prof. G. Chen

Design and Implementation of Fuzzy P²ID Controller for Handlebar Control of a Bicycle Robot

Pitikhate Sooraksa^{*}, Thana Pattaradej^a and Guanrong Chen^b

^aDepartment of Information Engineering,

Faculty of Engineering, King Mongkut's Institute of Technology,

Ladkrabang, Chalongkrung Rd., Ladkrabang, Bangkok, 10520, Thailand

Email: kspitikh@kmitl.ac.th

^bDepartment of Electronic Engineering

City University of Hong Kong, Kowloon, Hong Kong SAR, P. R. China

Email: gchen@ee.cityu.edu.hk

Abstract

This paper presents the design and implementation of a new fuzzy proportional-integration plus fuzzy proportional-derivative control scheme, named the fuzzy P²ID controller hereafter, for controlling the handlebar of a bicycle robot with time-varying load (disturbance). The fuzzy P²ID controller combines some advantages of the fuzzy PI controller and the ones of the fuzzy PD controller, which can compensate their individual shortcomings. Detailed design is described, along with the implementation of the integrated computer-aided designed bicycle control system. Functionally, the controller is a nonlinear controller with variable gains, providing the expected fast response and robustness. Structurally, it is an embedded version of a conventional PI controller and a PD controller, which has an additional fuzzy control block, where a switching rule of control signal is also implemented. Experimental results show the effectiveness of the new control scheme for controlling the handlebar of a bicycle robot, built from "off-the-shelf" parts with completely unknown hardware parameters. Comparison tests among several existing fuzzy controllers as well as the conventional PID controller have been conducted, so as to evaluate the superior performance of the new scheme. This new control scheme can be directly applied to general robotic systems and other industrial processes under similar physical conditions.

^{*} Corresponding author.

1. Introduction

In the last three decades, fuzzy logic control has evolved as an alternative or complementary to the conventional control strategies in various engineering areas [15], particularly in robotics [5,11,14]. Fuzzy control theory usually provides nonlinear controllers that are capable of performing different complex nonlinear control actions [2,3], even for uncertain nonlinear systems [7]. Unlike conventional controls, designing a fuzzy logic controller does not require precise knowledge of the system model such as the poles and zeroes of the system transfer functions. Imitating the way of human learning, the trial error and the rate of the error are two crucial inputs for the design of such a fuzzy control system.

It was reported that PID controllers are still the major controllers used in the industry today [2]. On the other hand, it has also been shown that a hybrid scheme of fuzzy control and PID control has superior performance over individual conventional and fuzzy control algorithms [1,6,10,16]. Motivated by these facts, in this paper, we design and implement a fuzzy P²ID controller, based on the successful fuzzy PI controller and fuzzy PD controller developed earlier by Chen et al [3,4,8]. The fuzzy PI and fuzzy PD controllers are derived from a practical approach of a conventional digital PI and PD controllers, as shown in Fig 1.

[Place Fig. 1 near here]

To validate the efficiency of the new control scheme under a complex and significantly uncertain environment, we applied the new algorithm to the control of the handlebar of a bicycle robot driven by a DC motor, which has completely unknown parameters. The readers are referred to [12,13] for more detailed theory and design of the "in-house-built" bicycle robot used for the current study. In this paper, our objective is focused on the design and implementation of the new hybrid fuzzy controller. More specifically, a fuzzy P²ID controller is introduced, in which a computer-aided control system is used to track a reference input voltage. The aim of control is to achieve simultaneously a fast rise time and a minimum steady-state error with robustness within an acceptable tolerance, for which it is known that the fuzzy PD controller can handle the former while the fuzzy PI controller can deal with the latter. It is noticed that the fast rise time and robustness are crucial for this bicycle system because the robot can collide on obstacles if the control action is too slow; or even in the case of slight collision it should be able to self-adjust the system so as to continue the motion. Zero steady-state error may not be an

absolutely necessary criterion for this task if the tracking output can be guaranteed to be within an acceptable tolerance for the motion of the bicycle.

This paper is organized as follows. Section 2 describes the derivation of the fuzzy control law and the basic framework of the fuzzy PI controller, including fuzzification, control rule base establishment, and defuzzification. Section 3 provides similar discussion as in Section 2, but for the fuzzy PD controller. Section 4 then illustrates the proposed fuzzy P²ID control scheme, along with its switching criteria. Section 5 shows its corresponding hardware implementation and experimental results. Finally, Section 6 concludes the paper.

2. The Fuzzy PI Controller

In this section, we briefly describe the design of the fuzzy PI controller: its mathematical derivation and technical aspects. The fuzzy PI controller is actually a specially-designed conventional digital controller. Detailed analysis and derivation of this controller is referred to [4].

We first describe the design principle and basic structure of the fuzzy PI controller. The output of the digital fuzzy PI controller in the discrete time-domain, as can be easily verified from Fig. 2, is given by

$$u_n(nT) = u_n(nT - T) + K_{u_n} \Delta u_n(nT) \quad (1)$$

where $T > 0$ is the sampling time, K_p and K_i are the proportional and integral gains, respectively, and $K_{u_{PI}}$ is a fuzzy control gain to be determined later in the design. In Fig. 2, $e(nT)$ is the tracking error signal at the n -th sampling time. The fuzzy logic control part has a single output, called the incremental control output, and is given by

$$\Delta u_{PI}(nT) = K_p e_V(nT) + K_i e_P(nT) \quad (2)$$

where

$$\Delta u_{PI}(nT) = \frac{u_{PI}(nT) - u_{PI}(nT - T)}{T} \quad (3)$$

$$e_V(nT) = \frac{e(nT) - e(nT - T)}{T} \quad (4)$$

$$e_P(nT) = e(nT) \quad (5)$$

Here, $\Delta u_{PI}(nT)$ is the incremental control output, $e_P(nT)$ the error signal, and $e_V(nT)$ the rate of change of the error signal.

[Place Fig. 2 near here]

2.1 Fuzzification

The fuzzy PI control part employs two inputs: the tracking error signal (position), $e_p(nT)$, and the rate of change of the error signal (velocity), $e_v(nT)$. The fuzzy PI control part has a single output, called the incremental control output, and is denoted by $\Delta u_{PI}(nT)$. The input and output membership functions of the fuzzy PI controller (inside the box of Fig. 2) are shown in Figs. 3 and 4, respectively, where the constant $L > 0$ is chosen by the designer and is fixed after being determined.

[Place Figs. 3,4 near here]

2.2 Control Rule Base

Using the aforementioned membership functions, the following control rules (shown in Table 1) are established for the fuzzy PI control part. Note that we let the output membership functions be singletons here for the simplicity of design.

In Table 1, $e_p := r - y$ is the error; where if r is a constant then $e_v = \dot{e}_p = 0 - \dot{y} = -\dot{y}$ is the rate of the error. Here, "PI-output" is the control output $\Delta u(nT)$, " $e_p.p$ " means "position error positive" and " $o.p$ " means "output positive" etc. Also, "AND" is the Zadeh's logical "AND." These four rules together yield the control actions for this fuzzy PI control part (Table 1).

[Place Table 1. near here]

The formulation of these rules can be understood as follows [4].

Rule 1 (R1): From Table 1, we get $e_p.p$ (the error is positive) and $e_v.p$ (the rate of error is positive)

over the interval of $r > y$, which controls y to run on the positive direction toward r .

Rule 2 (R2): The system starts to work in the interval where y is uncontrolled climbing naturally.

Rule 3 (R3): Similar to Rule 2 (R2), when $e_p.n$ and $e_v.p$, the system runs freely without control,

where the system output is coming back by itself toward the setting point.

Rule 4 (R4): It is the inverse control under condition $e_p.n$ and $e_v.n$, when y passes through r ($y > r$)

on this interval, so we must control it to turn back toward r .

2.3. Defuzzification

In the defuzzification step, the commonly used "center of mass" formula is employed to defuzzify the incremental control of the fuzzy logic control part. The defuzzification formula is

$$\Delta u_{PI}(nT) = \frac{\sum \{\text{membership of input} \times \text{output corresponding to the membership of input}\}}{\sum \{\text{membership of input}\}} \quad (6)$$

For the fuzzy PI controller, the value-ranges of the two inputs, the errors, and the rates of change of the errors, are decomposed into twenty adjacent input-combination regions as shown in Fig. 5 (a). This figure is understood as follows: We put the membership function of the error signal over the horizontal $K_i e_p(nT)$ -axis, and put the membership function of the rate of change of the error signal over the vertical $K_p e_v(nT)$ -axis, on Fig. 5. These two membership functions then overlap and form the third-dimensional picture over the 2-dimensional regions as the top view shown in Fig. 5 (a). When we look at region 1, for example, if we look rightward to the $K_i e_p(nT)$ -axis, we see the domain $[0, L]$ and the membership function (in the third dimension) over $[0, L]$ of the error signal; if we look downward to the $K_p e_v(nT)$ -axis, we see the domain $[-L, 0]$ and the membership function over $[-L, 0]$ of the rate of change of the error signal.

[Place Fig. 5 near here]

The above control rules, (R1)-(R4), along with the chosen membership functions and the determined regions, are used to generate appropriate fuzzy control actions when the error and rate signals are located within a certain region.

In doing so, we consider the locations of the error $K_i e_p(nT)$ and the rate $K_p e_v(nT)$ in the regions. Let us look at region 1, for example, where $e_p(nT)$ is in the range $[0, L]$ and $e_v(nT)$ in $[L, 0]$. For these two signals, we have $e_v \cdot n > 0.5 > e_p \cdot n$. Figure 6 and (R1) together, where Zadeh's logical "AND" is used, lead to { error = $e_p \cdot n$ AND rate = $e_v \cdot n$ } = $\min \{ e_p \cdot n, e_v \cdot n \} = e_v \cdot n$. The rest of the cases are summarized in Table 2.

[Place Fig. 6 near here]

[Place Table 2 near here]

It can be verified that the above are true for the four input combinations labeled Regions 1. Thus, in Regions 1, it follows from the defuzzification formula (6) that

$$\Delta u_{PI} = \frac{\mu_1 U_1 + \mu_2 U_2 + \mu_3 U_3 + \mu_4 U_4}{\mu_1 + \mu_2 + \mu_3 + \mu_4}$$

$$\Delta u_{PI}(nT) = \frac{e_p \cdot n \times 0 \cdot n + e_p \cdot n \times 0 \cdot z + e_v \cdot n \times 0 \cdot z + e_v \cdot p \times 0 \cdot p}{e_p \cdot n + e_p \cdot n + e_v \cdot n + e_v \cdot p}$$

Here, it is important to note that if one follows the above procedure to work through the four segments of Region 1, he will find that all cases give the same result of $e_p \cdot n$. To this end, we apply $0 \cdot p = L$, $0 \cdot n = -L$, $0 \cdot z = 0$, and following the straight line formulas obtained from the geometry of Figs. 3 and 4:

$$e_p \cdot p = \frac{K_p e_p(n) + L}{2L}$$

$$e_p \cdot n = \frac{-K_p e_p(n) + L}{2L}$$

$$e_v \cdot p = \frac{K_p e_v(n) + L}{2L}$$

$$e_v \cdot n = \frac{-K_p e_v(n) + L}{2L}$$

where it should be noted that $e_p(nT) \leq 0$ in Regions I, then we arrive at the following result for the four segments of regions I:

$$\Delta u_{PI} = \frac{L(K_p e_p(nT) + K_p e_v(nT))}{2(2L - K_p |e_v(nT)|)}$$

Working through all regions in the same way, we obtain the following formulas for the twenty regions, which can then be reduced into nine formulas:

$$\begin{aligned} \Delta u_{PI}(nT) &= \frac{L(K_p e_p(nT) + K_p e_v(nT))}{2(2L - K_p |e_v(nT)|)} && \text{in region I} \\ &= \frac{L(K_p e_p(nT) + K_p e_v(nT))}{2(2L - K_p |e_p(nT)|)} && \text{in region II} \\ &= \frac{1}{2} [K_p e_v(nT) + L] && \text{in region III} \\ &= \frac{1}{2} [K_p e_p(nT) + L] && \text{in region IV} \\ &= \frac{1}{2} [K_p e_v(nT) - L] && \text{in region V} \\ &= \frac{1}{2} [K_p e_p(nT) - L] && \text{in region VI} \\ &= 0 && \text{in region VII} \\ &= +L && \text{in region VIII} \\ &= -L && \text{in region IX} \end{aligned} \quad (7)$$

Finally, it is worthwhile mentioning that the stability of this fuzzy PI controller has been analyzed, resulting in a practical and rigorous mathematical condition, in [4].

3. Fuzzy PD Controller

Similar to the derivation of the fuzzy PI controller discussed in the previous section, the fuzzy PD controller, shown in Fig. 7, is described by

$$u_{PD}(nT) = -u_{PD}(nT - T) + K_{uPD} \Delta u_{PD}(nT) \quad (8)$$

where K_{uPD} is a fuzzy control gain and the incremental control is

$$\Delta u_{PD}(nT) = K_p e_p(nT) + K_d e_v(nT) \quad (9)$$

Detail derivation and stability analysis of this controller is referred to [8].

[Place Fig. 7 here]

3.1. Fuzzification.

We fuzzify the P and D components of the PD controller individually, in a way similar to the fuzzification of the fuzzy PI controller discussed above. The input and output membership functions of the PD component are also chosen to be the same as that shown in Figs. 3 and 4, for simplicity of the design.

The fuzzy PD controller employs two inputs: the average error signal, e_p , and the rate of change of the error signal, e_v . The fuzzy PD controller has a single output, denoted by $\Delta u_{pd}(n)$.

3.2. Control Rule Base

Using the chosen membership functions, the following control rules are established for the fuzzy PD controller, as indicated in Table 3. Here, for simplicity, we use the singleton membership function for the outputs, as the one shown in Fig. 4. In Table 3, the rules can be understood as follows:

Rule 1 (R1): The system starts to work over the interval where y is uncontrolled, climbing naturally.

Rule 2 (R2): From Table 2, we get $e_{p,p}$ (the error is positive) and $e_{v,n}$ (the rate of error negative) on the interval where $r > y$, in which the rule controls y to run in the positive direction toward r .

Rule 3 (R3): Under the conditions $e_{p,n}$ and $e_{v,p}$, the output y passes through r ($y > r$) on this interval, so we must control it to turn back toward r .

Rule 4 (R4): It is similar to Rule 1 (R1), for $e_{p,n}$ and $e_{v,n}$, we let the system run freely without control. The output signal is coming back toward the set-point, so we let it run by itself.

3.3. Defuzzification

In this step, the “center of mass” formula is employed again to defuzzify the incremental control, as shown in (6).

Now, by applying the values $o.p = L$, $o.n = -L$, $o.z = 0$, and the following the straight line formulas obtained from the geometry of Figs. 3 and 4, we obtain the following nine formulas for the nine reduced input combination regions similar to Fig. 5(b):

$$\begin{aligned}\Delta u_{pd}(nT) &= \frac{L[K_p e_p(nT) - K_d e_v(nT)]}{2[2L - K_d |e_v(nT)|]} && \text{in region a} \\ &= \frac{L[K_p e_p(nT) - K_d e_v(nT)]}{2[2L - K_p |e_p(nT)|]} && \text{in region b} \\ &= \frac{1}{2}[-K_d e_v(nT) + L] && \text{in region c}\end{aligned}$$

$$\begin{aligned}
&= \frac{1}{2}[K_p e_p(nT) - L] && \text{in region d} \\
&= \frac{1}{2}[-K_d e_v(nT) - L] && \text{in region e} \\
&= \frac{1}{2}[K_p e_p(nT) + L] && \text{in region f} \\
&= 0 && \text{in region g} \\
&= -L && \text{in region h} \\
&= L && \text{in region i}
\end{aligned} \tag{10}$$

4. The Fuzzy P²ID Controller

The proposed Fuzzy P²ID controller is an integrated scheme that combines the fuzzy PI and fuzzy PD control actions, where the advantages of both control actions are utilized to compensate and complement each individual shortcoming.

Based on our long-term experience with these fuzzy PI and fuzzy PD controllers, we found that the fuzzy PD controller has faster rise time than the fuzzy PI controller [22], while the fuzzy PI controller possesses better steady-state accuracy than the other. In fact, this is clear by inspecting the design strategies employed in their control rule bases. Hence, it is expected that if these two controllers are combined together in a suitable way, to form a hybrid structure where an appropriate selecting rule is imposed by switching, then a superior performance over each individual controller may be achieved. With this concept, the architecture of the new control algorithm is developed, as depicted in Fig. 8.

[Place Fig. 8 near here]

To verify and validate the design, we separate the transient response plane into two shaded areas, which are the domains for launching the fuzzy PI control action and the fuzzy PD control action, respectively. Conceptually, we assign the boot-up task to the fuzzy PD controller, which is also responsible for bring the output back toward the set-point when the output is beyond the pre-set overshoot limit. The fuzzy PI control, on the other hand, is selected as a second pilot to drive the process output to be within the given tolerance limit, near a certain percentage of the desired steady-state value. The transition point depends on the designer and the process to be controlled.

Since these two individual controllers guarantee their local stability, by the small gain theorem as studied in [4] and [8], the operations are stable after choosing the gains that satisfy the stability criteria locally but for all times expect at the switching points. With this in mind, we consider only the stability at the switching points. Utilizing the error-phase plane shown in Figs. 5, we can visualize a principle

for designing the switching rule in the following way: if we consider the areas inside the two phase planes as the attraction regions in the sense of bounded input and bounded output (BIBO) stability [4,8], then at the switching point (see Fig. 9) the change of the control energy at the transition between the two situations must be bounded even though it is inherently discontinuous. Based upon this fact, we must enforce a bounded control transition between the two controllers. In other words, at a switching point, the control energy using by the fuzzy PI controller must be less than that by the fuzzy PD controller.

[Place Fig. 9 near here]

By inspection of (7) for region "I" and (10) for region "a" and with the visual aid of Fig. 9, we can see that if $K_p^{PD} > K_i^{PI}$, $K_p^{PI} > K_d^{PD}$, $K_d^{PD} |e_v(nT)| \ll 2L$, and $K_i^{PI} |e_v(nT)| \ll 2L$, then an energy criterion is met, though at a conservative level. Hereafter, we use superscript to indicate which gain belongs to which controller. Substituting these conditions into those equations in (7) and (10) in the specified regions "I" and "a," then with a little algebra we can show that $\|\Delta u_{PI}(nT)\|_2 < \|\Delta u_{PD}(nT)\|_2$. Similarly, by using the same equations and repeating the same on regions "II" and "b," we can also see that if $K_p^{PD} > K_i^{PI}$, $K_p^{PI} > K_d^{PD}$, $K_d^{PD} |e_p(nT)| \ll 2L$ and $K_i^{PI} |e_p(nT)| \ll 2L$, then a similar criterion for bounded control energy is obtained. We now summarize the above analysis and state the switching criteria as follows.

A selector is said to be sufficiently satisfied the bounded conditions in changing control inputs at the transition point or switching point between the fuzzy PD to fuzzy PI controllers, if the following criteria are met:

(a) All gains satisfy the small gain theorem, as specified in [4] and [8].

(b) $K_p^{PD} > K_i^{PI}$

(c) $K_p^{PI} > K_d^{PD}$

(d) $K_d^{PD} |e_v(nT)| \ll 2L$, and $K_i^{PI} |e_v(nT)| \ll 2L$

(f) $K_d^{PD} |e_p(nT)| \ll 2L$, and $K_i^{PI} |e_p(nT)| \ll 2L$.

If these five conditions are met, the bounded trajectory confined in the error-phase planes before/at/ after the transition period between the two planes (shown in Fig. 9) will be guaranteed. Note that

condition (a) guarantees the local stability during operation by individual controller, and conditions (b)-(f) are used to guaranteed the stability of the selector at the switching times. Note also that the inequalities in (d) and (f) can be used to calculate the switching times using the sampling outputs corresponding to the two sampling times. The error and the rate of error can be obtained by measuring data from the experiment and by using equations in (4) and (5).

We will use the above-derived criteria to design a set of control parameters for the proposed control scheme when it is applied to the handlebar of a bicycle robot in the next section.

5. Hardware Setup and Experiment

The apparatus used in this experiment, as shown in Figs. 10 and 11, consists of a DC motor connected to the turning unit of a bicycle robot [12,13]. The handlebar of the robot is hanging with a plastic bottle filled with water at one end emulating disturbance loads.

[Place Fig. 10 near here]

[Place Fig. 11 near here]

Since we use an "off-the-shelf" motor, all the parameters of the platform are completely unknown. This motor has a built-in plastic gear, which can be considered as a flexible joint. The sensing apparatus in the feedback loop is a potentiometer and an Analog-to-Digital (A/D) converter. The output apparatus consists of a Digital-to-Analog (D/A) converter, a Pulse-Width-Modulation (PWM) circuit, a power amplifier, and a DC servomotor. All electronic circuits and mechanical parts used are "in-house-building," prototyped by the authors.

Since the experiment is focused on the computer-aided control system, not on the robotic machine, the fuzzy P²ID controller is performed on computer via software implementation. The computer used in this experiment is a 500MHz AMD K6-2 computer running on the Windows ME operating system. The computer program is written in Turbo C++ version 3.0. The ISA slots No.0308h, No.0309h, No.030Ah [9] are used as ports.

This experiment is set up to partially simulate the situation when the robot bicycle is rolling along a rough road, with uncertain ground disturbances. These effects could disturb the whole dynamic system, including the handlebar of the robot bicycle. To simulate this environment, we used a plastic bottle filled with water at one end and hang it on the handlebar to generate the disturbance input during the

experiment (see Fig. 11). We also experimented the case of a bigger disturbance abruptly hitting the handlebar, through impulsive hits onto the handlebar by human hand.

Performance evaluation was carried out via a comparison study for the proposed control scheme and some other existing control schemes, such as the individual fuzzy PI control, fuzzy PD control, and conventional PID control algorithms. Figures 12 and 13 show the experimental results. In Fig. 12, the following control gains were determined by trial-and-error testing. For the conventional PID control, $K_p=20$, $K_i=0.1$, $K_d=5$. This set of PID gains was obtained by fine-tuning several times till we got the best possible results for a fair comparison. For fuzzy PI control, $K_p = 1$, $K_i=1$, $K_{upi}=1$, and $L=100$; whereas for the fuzzy PD control, $K_p=10$, $K_d=0.1$, $K_{upd}=1$, and $L=100$. The new fuzzy P^2ID control scheme has the same parameters as the cases of the two individual fuzzy PI and fuzzy PD controllers. We had no difficulty about tuning these fuzzy controllers because we selected the gains based upon the criteria given and discussed in the previous section. We set the sampling time to $T=0.1$ sec. for all cases. The switching instant between the fuzzy PI and fuzzy PD controllers in the fuzzy P^2ID scheme was set at the point of 5% off the set-point.

It is important to note that each step of the position was set according to the mapping voltage into the new positioning unit. The designer can change this as desired. In this experiment, we mapped 0-5 voltages into 0-255 steps corresponding to an 8-bit representation, according to the limitation of the physical structure of the handlebar. Note also that the initial position was not at 0 but was at around the 90th step, to simulate the real situation when the robot bicycle was in operation. The control signal herein is the one measured at the output from the computer before it was fetched to the PWM circuit, which may be alternatively implemented by using an 89C8051 microcontroller. Note moreover that the x-axes in Figs. 12-15 are recorded by using the sampling step of 10 msec for each step, which can be easily converted by using this scaling factor.

According to Fig. 12, the fuzzy P^2ID controller shows superior performance in many aspects over the other competitors in comparison. Figure 13 shows the tracking error signals corresponding to Fig. 12. The individual fuzzy PI controller ranks the third in performance for steady-state tracking but it has no overshoot, and the second is the individual fuzzy PD controller scheme, which however has some overshoot. For the rise time comparison, the fuzzy PD controller is the second, followed by the fuzzy PI controller. The conventional PID controller, however, is the worst one in both rise time and settling time. Note again that steady-state error-free is not a critical point for this test on bicycle motion since

for which the rise time and the settling time are more important. If the tracking is within the acceptable tolerance limits and within the required settling time for controlling the balance of the bicycle, then the control scheme can be used to serve the purpose.

[Place Fig. 12 near here]

[Place Fig. 13 near here]

Two remarks are in order: First, all control schemes can cope with external uncertainties (generated by the bottle filled with water hung at the handlebar) to some extent. Second, the conventional PID control scheme has the worst overshoot due to its linear control actions on this highly nonlinear and uncertain mechanical system. In addition, its initial position was set to start up around 10 steps higher than that of the other fuzzy control systems. Apparently, this means that the conventional PID has a rise-time response far behind the fuzzy controllers. However, the conventional PID controller can still be used to operate the control of the handlebar of the robot bicycle. This implies that the plant might actually have some significant linear features, at least within the bounded signaling ranges. Nevertheless, there is no guarantee for the strong disturbance cases where the linear PID controller can be tolerated to work as good as the other controllers. Note that the time for controlling the robot bicycle so that it does not fall onto the ground is only 1 sec.; otherwise the bicycle cannot be balanced. The controller's action must be fast enough to cope with this very critical situation.

Figure 14 shows the control input of the proposed fuzzy P^2ID control scheme, corresponding to Fig. 12. The solid line represents control input of the controller, in which there is one part generated by the fuzzy PD unit and the rest by the fuzzy PI unit, in accordance with the designed switching point. As can be seen from the figure, the control input of the proposed control scheme is bounded and less than the one of the individual fuzzy PD controller.

[Place Fig. 14 near here]

In Fig. 15, the control gains remain the same as those in Fig 12. In this test, we wanted to compare the robustness performance between the proposed control scheme and the conventional PID controller. This case is emulated when strong impulsive disturbance was applied to the handlebar of the robot bicycle from human-hand pushing. In the experiment, as shown in Fig. 15, we abruptly pushed the handlebar at some discrete time, roughly around the 800-th step, to move it away from the set-point. We found that the two control schemes are adequately robust to regulate the tracking, driving the handlebar back to the set-point in less than 1 sec. Still, the fuzzy controllers have better transient and

steady-state responses than the conventional controller. Notice also that the switching between the fuzzy PD and fuzzy PI controllers within the new scheme is smooth. The new control scheme is proved effective and also easy to design, in accordance with the given criteria in Section 4, and can be alternatively used to substitute the conventional PID control in systems like this robot bicycle that have highly non-linearity as well as uncertainties, once again verifying the reports of [7, 10].

[Place Fig. 15 near here]

6. Conclusions

We have described a new fuzzy P²ID controller designed by suitably integrated the individual fuzzy PI and fuzzy PD controllers with a switching rule. We have also reported our real performance evaluation and comparison, and summarized its merits and drawbacks as compared to the other comparable fuzzy and conventional PID-type controllers. It turns out that this new control scheme has superior performance over all its competitors, including the individual fuzzy PI and fuzzy PD controllers as well as the conventional PID controller. An additional finding is that although all these control schemes are robust to some extent in the presence of uncertainties, unknown parameters, and input disturbances, and yet the new scheme is overall the best among them.

Although our experimental results show the effectiveness and superior performance of the new controller, its online adaptive feature has not yet been carefully investigated. This powerful feature should be further evaluated and upgraded in the new control scheme, thereby making it an even more powerful hybrid controller for real engineering applications.

Acknowledgement

We would like to thank the anonymous reviewers for their valuable comments for the improvement of our original manuscript. This research was partially supported by the Thailand Government Research Funds under grant PDF/95/44 and the Hong Kong CityU Grant 9360051.

References

- [1] B.S. Butkiewicz, System with hybrid fuzzy - conventional PID controller, IEEE Intl Conf. on Systems, Man, and Cybernetics 2000, 5 (2000), pp. 3705 – 3709.

- [2] G. Chen, Conventional and fuzzy PIDcontroller: An overview, *Int. J., Intell. Control System.* 1 (1996), pp. 235-246.
- [3] G. Chen and T. T. Pham, *Introduction to Fuzzy Sets, Fuzzy Logic, and Fuzzy Control Systems*, CRC Press, 2000.
- [4] G. Chen and H. Ying, BIBO stability of nonlinear fuzzy PI control systems, *J. Intel. Fuzzy Syst.*, 5(1997), pp. 3-21.
- [5] Q.P. Ha, Q.H. Nguyen, D.C. Rye and H.F. Durrant-Whyte, Fuzzy sliding-mode controllers with applications, *IEEE Trans. on Industrial Electronics*, 48 (1) (2001), pp. 38 – 46.
- [6] K. Hirota, *Industrial Applications of Fuzzy Technology*, Springer-Verlag, Tokyo, 1993
- [7] H.A. Malki, D. Feigenpan, D. Misir and G. Chen, Fuzzy PID control of a flexible joint robot arm with uncertainties for time-varying load, *IEEE Trans. Control Syst. Technol.*, 5 (3) (1997), pp. 371-378.
- [8] H.A. Malki, H. Li and G. Chen, New design and stability analysis of fuzzy proportional-derivative control system, *IEEE Trans. Fuzzy Systems*, 2 (4) (1994), pp.345-354.
- [9] M.A. Mazadi and J.G. Mazadi, *The 80x86 IBM PC & Compatible Computer Volume II: Design and Interfacing of the IBM PC, PS and Compatible*, Prentice-Hall, 1995.
- [10] D. Misir, H.A. Malki and G. Chen, Design and analysis a fuzzy proportional-integral-derivative controller, *Fuzzy Set and System*, 79 (1996), pp. 297- 31
- [11] P. Sooraksa and G. Chen, Mathematical modeling and fuzzy control for flexible-link robots, *Comput. and Math Model.*, 27 (6) (1998), pp. 73-93.
- [12] P. Sooraksa and U. Sritheerawirojana, A bicycle robot: Part I, Modelling and control, *Proc. of ISARC 17th International Symposium on Automation and Robotic in Construction*, 2000, pp. 501-504
- [13] P. Sooraksa , U. Sritheerawirojana, S. Kaopratum, T. Uthairat and V. Monyakul, A bicycle robot: Part 2 System and Implementation, *Proc. of ISARC 17th International Symposium on Automation and Robotic in Construction*, 2000, pp. 525-528.
- [14] M. Uragami, M. Mizumoto and K. Tanaka, Fuzzy robot controls, *J. Cybernet*, 6 (1) (1976), pp. 39-64.
- [15] L.X. Wang, *A Course in Fuzzy Systems and Control*, Prentice-Hall, 1997.
- [16] Y. Wang and J.D. Birdwell, A nonlinear PID-type controller utilizing fuzzy logic, *Proc. of the IEEE/IFAC Joint Symposium on Computer-Aided Control System Design*, 1994, pp. 89-94.
- [17] W.A. Wolovich, *Automatic Control Systems*, Saunders College Pub., Orlando, FL, 1994.

- [18] T. Pattaradej, P. Chandang, M. Lertwatechakul, P. Sooraksa and G. Chen, A speed evaluation for conventional PID and fuzzy controllers, Proc. of the 2nd International Conference on Intelligent Technologies, 2001, pp. 91-96.

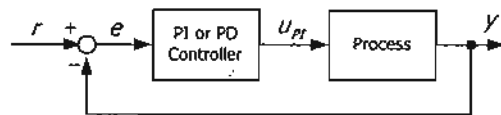


Fig 1. A conventional PI or PD continuous-time control system.

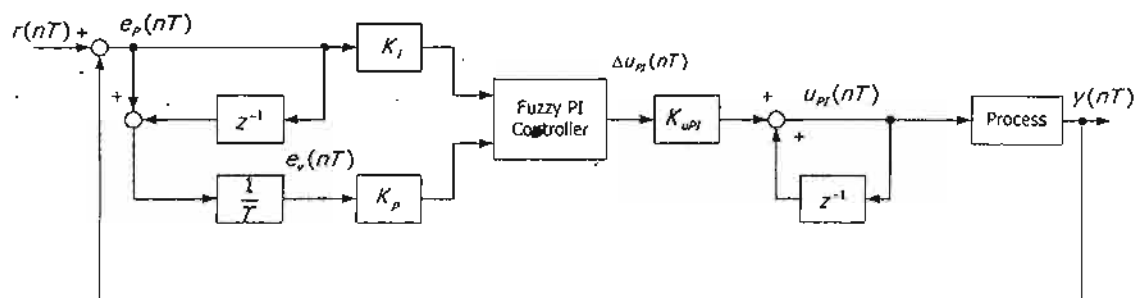


Fig 2. The fuzzy PI control system.

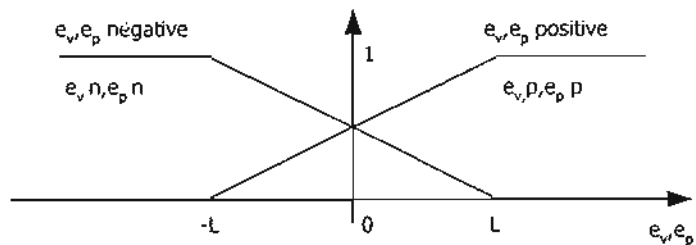


Fig 3. The input membership functions for the fuzzy logic control part.

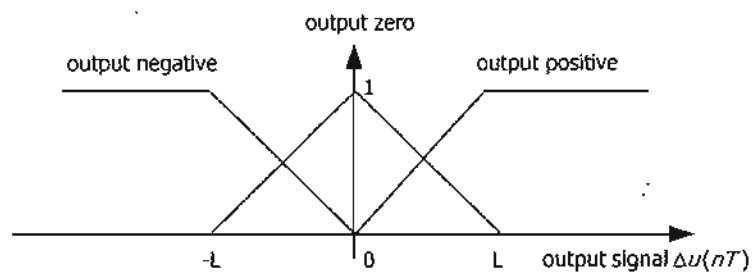


Fig 4. The output membership functions for the fuzzy logic.

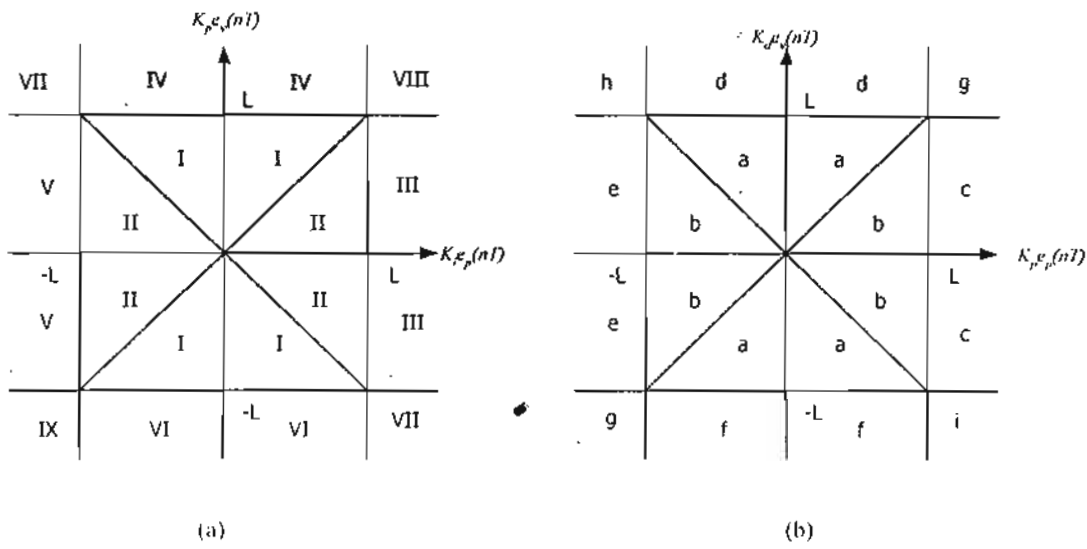


Fig 5. Decomposed regions of the input-combination values (a) for fuzzy PI control (b) for fuzzy PD control.

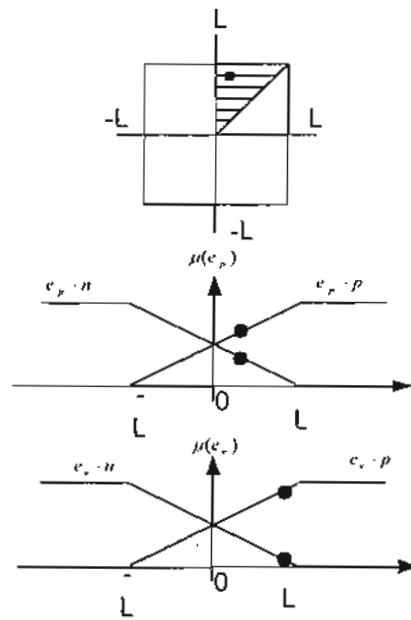


Fig 6. Marking sample for calculating an equation of region I

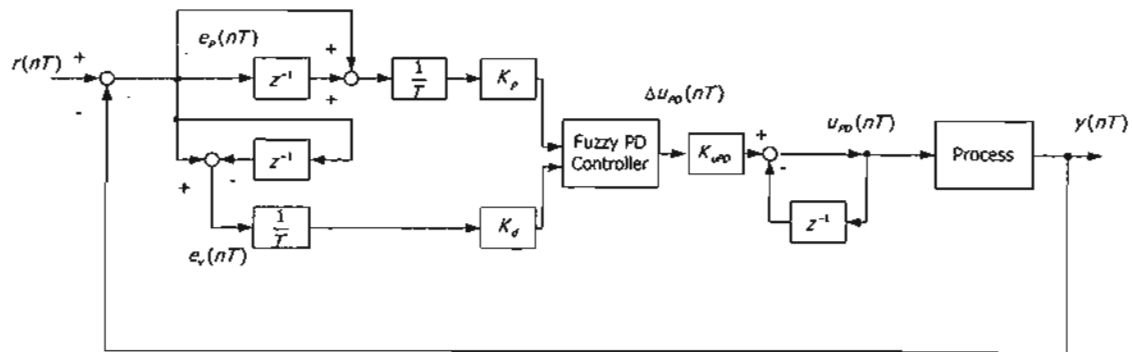


Fig. 7. The fuzzy PD control system.

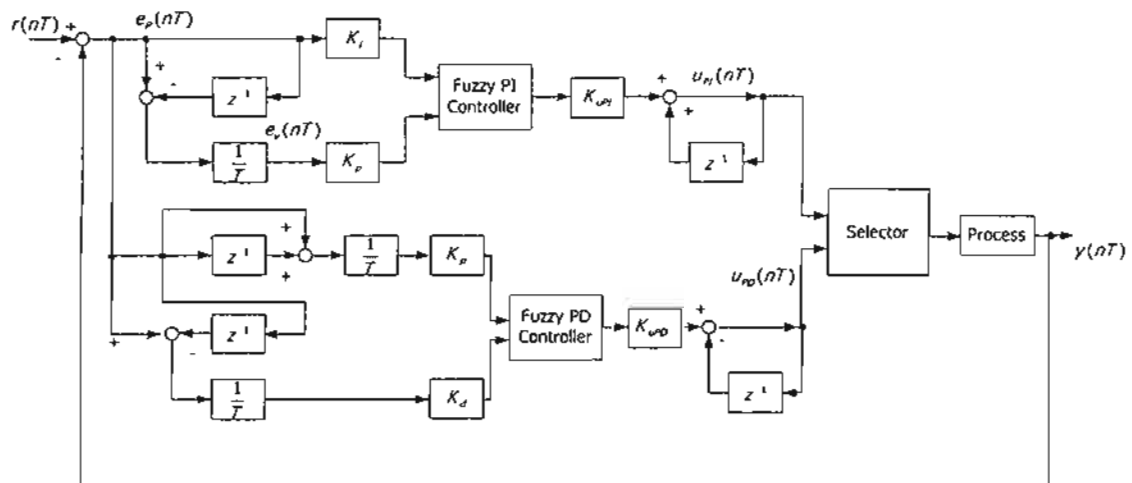


Fig. 8. The fuzzy P²ID control system.

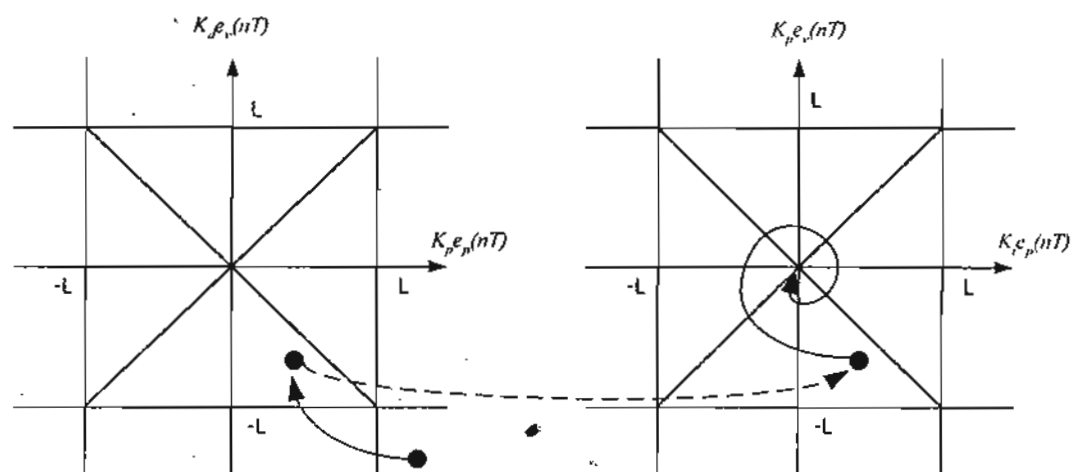


Fig. 9. At the switching point, the error trajectory is mapped from error-phase plane of fuzzy PD to that of fuzzy PI control.

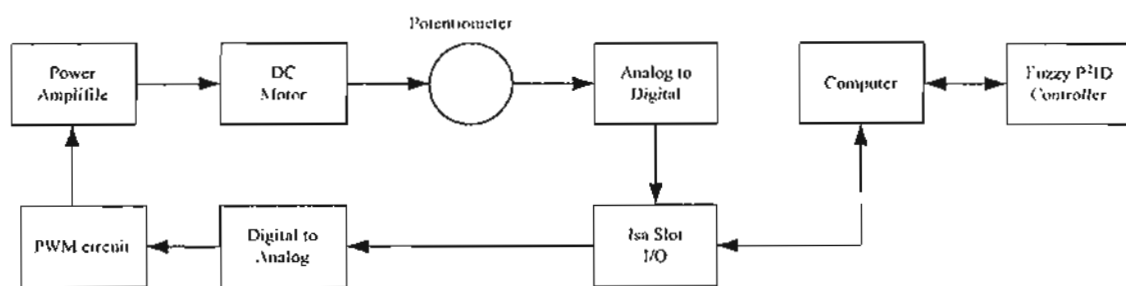


Fig. 10. Block diagram of the hardware setup

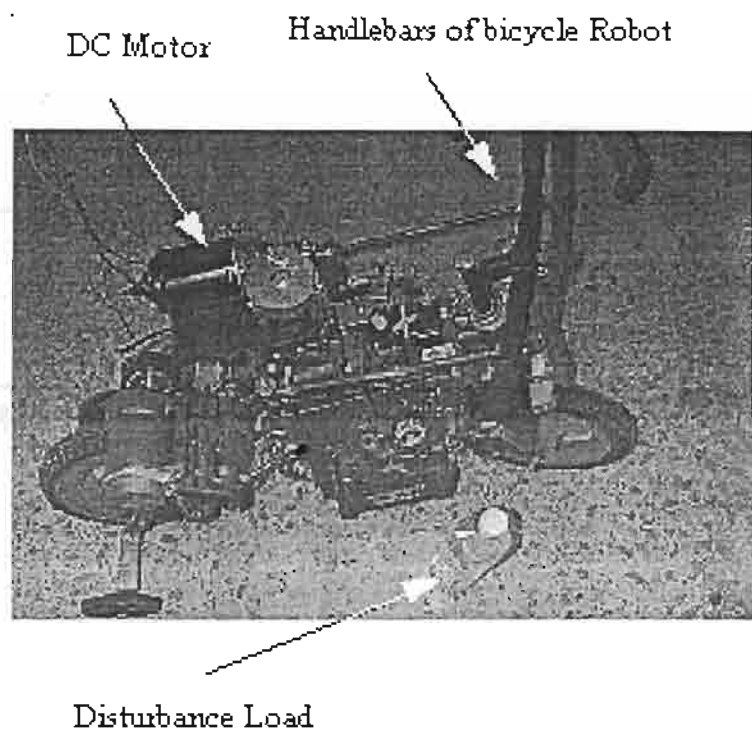


Fig. 11. A bicycle robot.

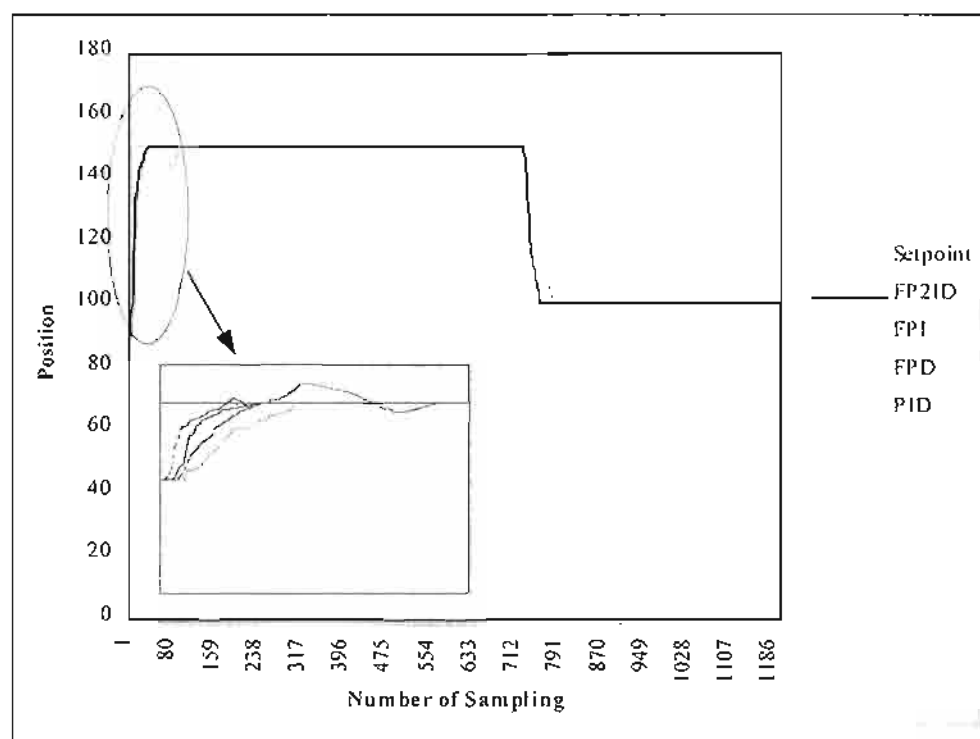


Fig. 12 Tracking performance comparison.

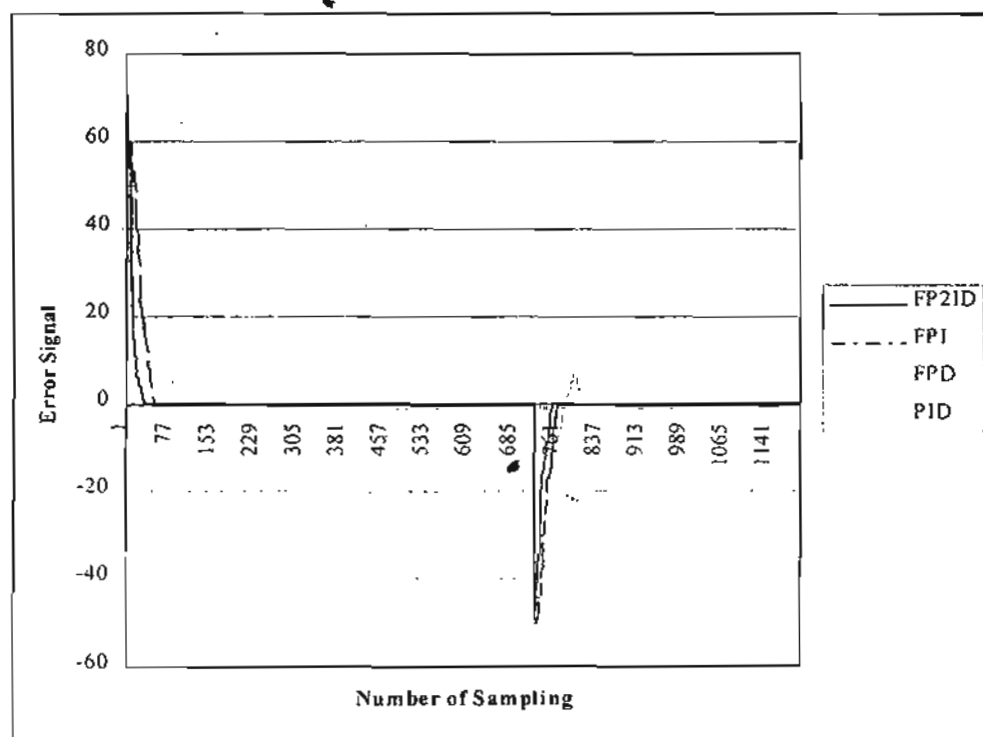


Fig. 13 Tracking error corresponding to Fig. 12.

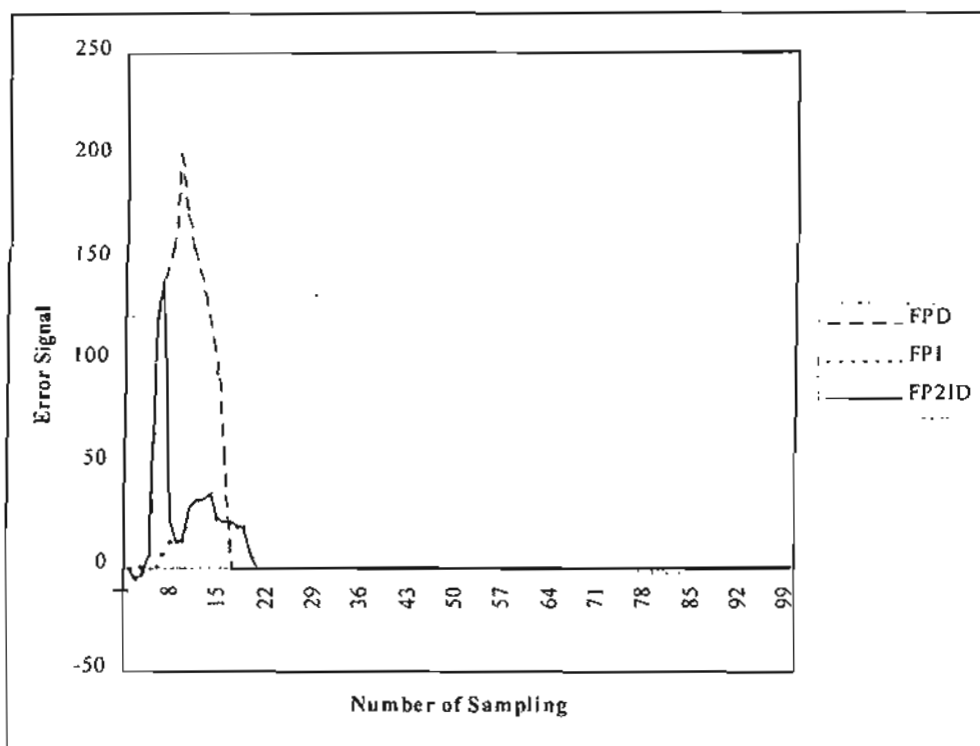


Fig. 14. Control input of the proposed scheme.

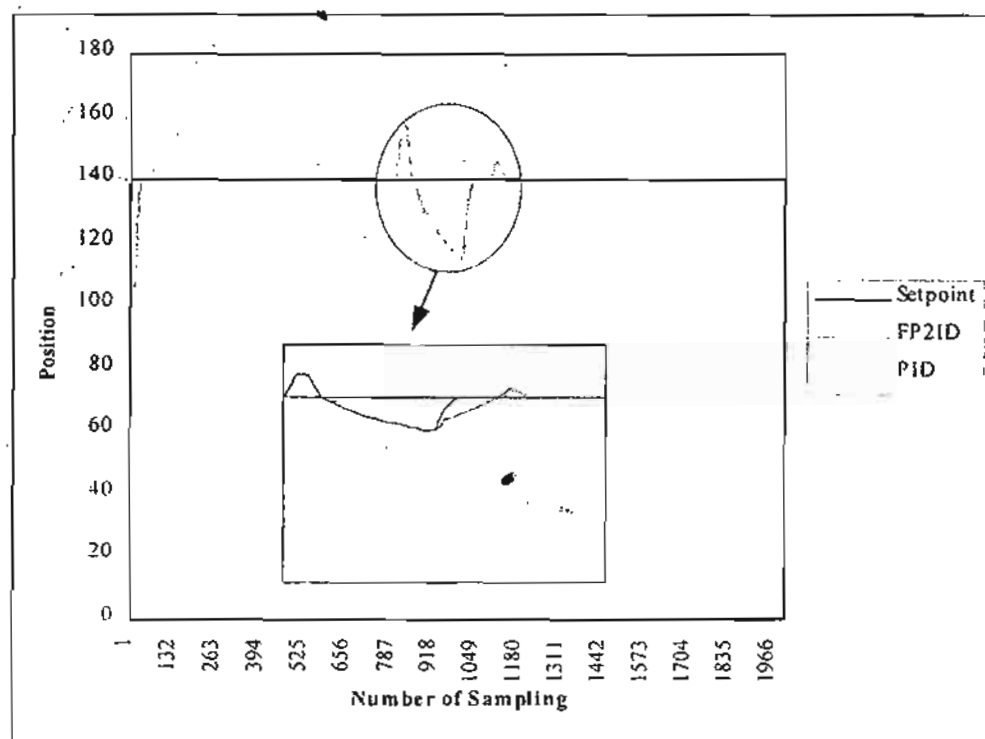
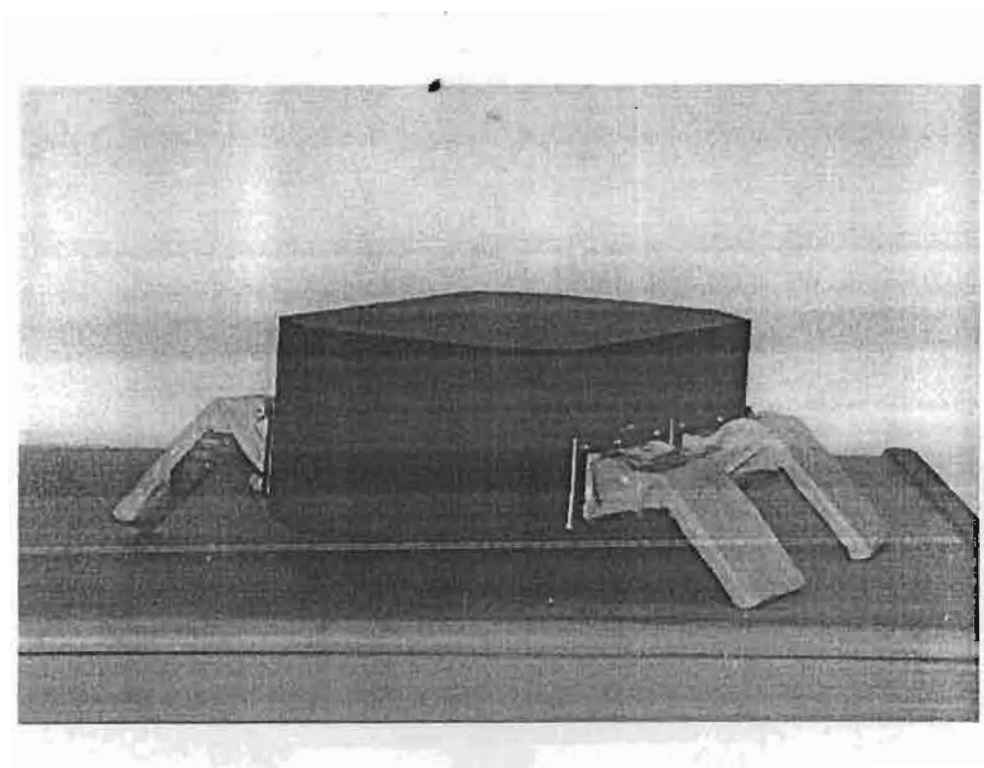


Fig.15. Impulse disturbance injecting testing for tracking robust performance.

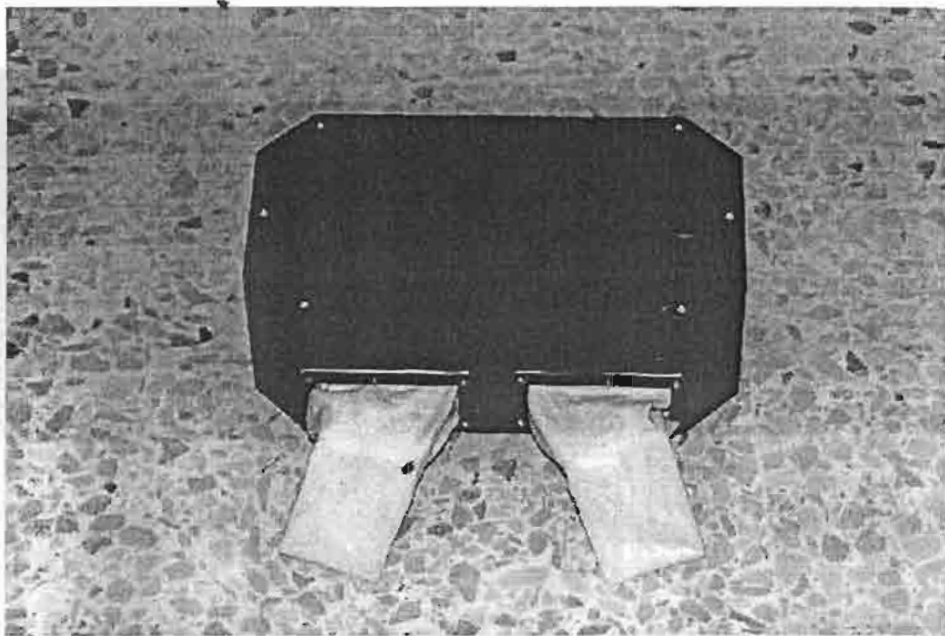
หุ่นยนต์ ต่อไปนี้ สร้างโดยอาศัยทุนในโครงการ เพื่อใช้ทดสอบระบบควบคุมซึ่งจะสามารถใช้ Plant ในการทดสอบระบบควบคุมที่จะต่อยอดได้ต่อไป

1. หุ่นยนต์เต่า (Turtle Robot)

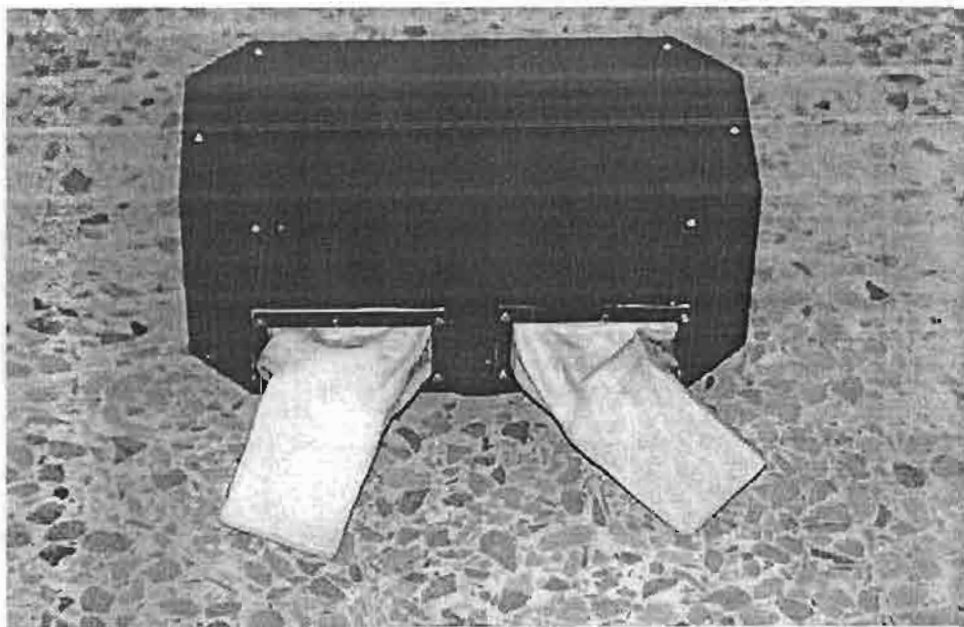
รูปภาพตัวหุ่นยนต์



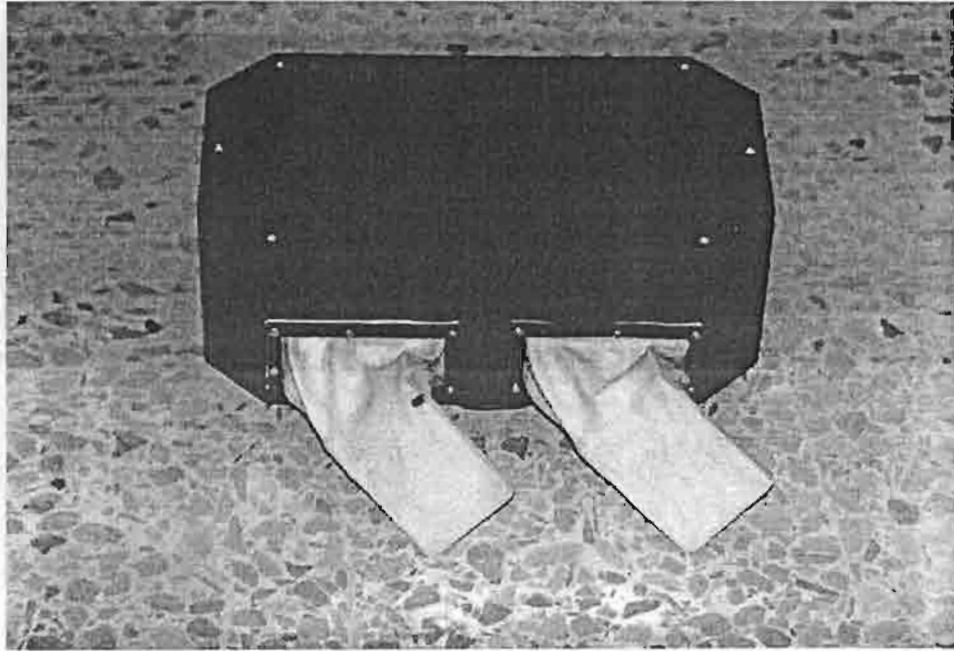
รูปที่ (ก) แสดงภาพเต็มตัวของหุ่นยนต์เต่า



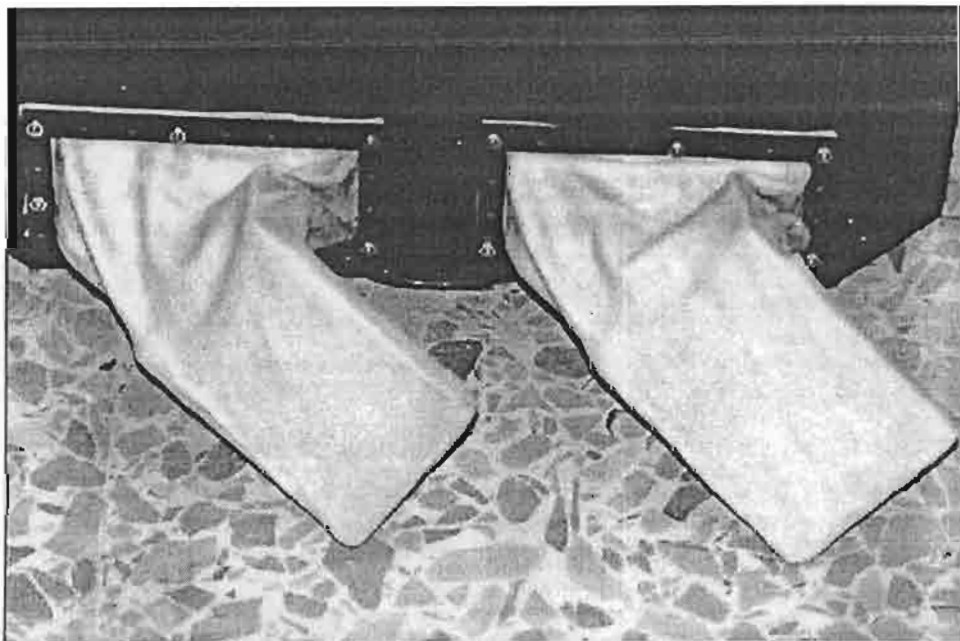
รูปที่ (ข) แสดงขณะหุ่นยนต์ดำเคลื่อนไหวนบนบก



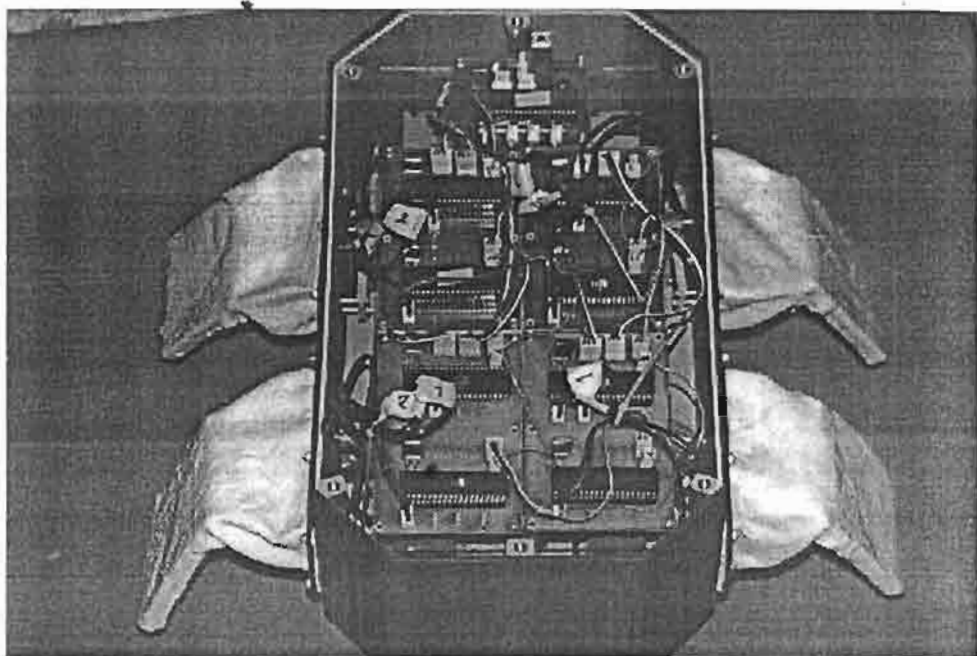
รูปที่ (ค) แสดงขณะหุ่นยนต์ดำเคลื่อนไหวนบนบก



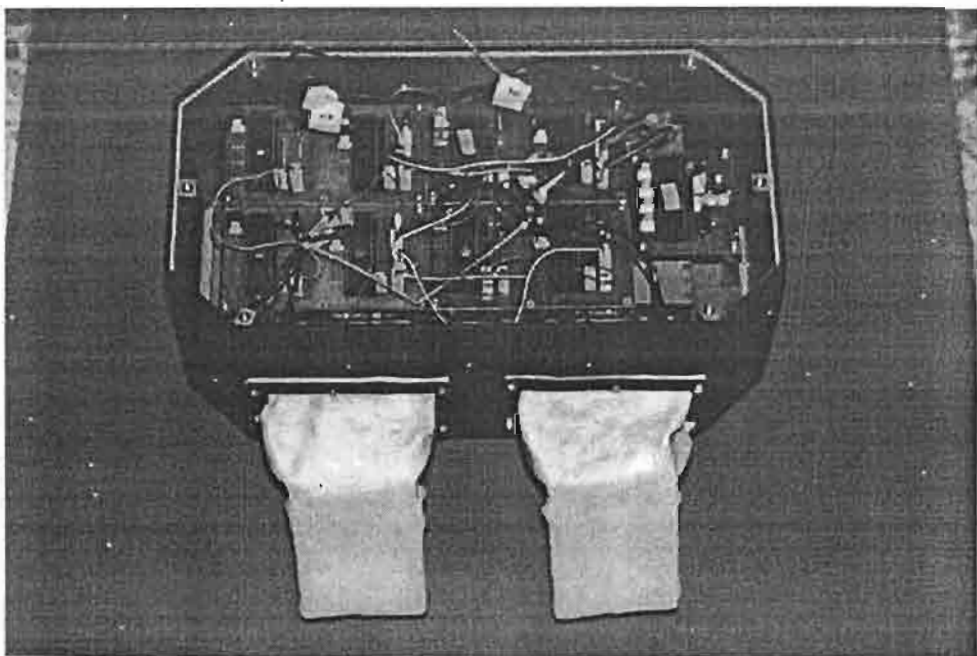
รูปที่ (ง) แสดงขณะหุ่นยนต์ดำเคลื่อนไหวในน้ำ



รูปที่ (จ) แสดงลักษณะขาของหุ่นยนต์ดำเมื่ออยู่ในน้ำ

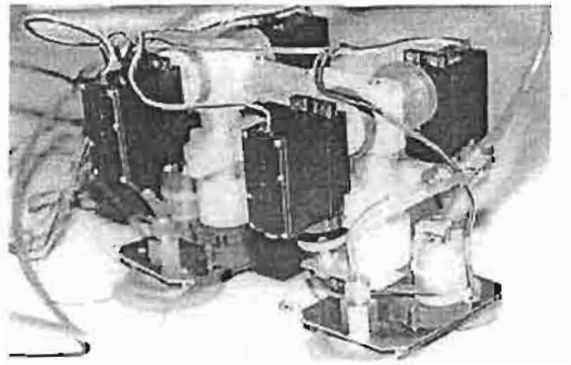


รูปที่ (ฉ) การติดตั้งบอร์ดในตัวหุ่นยนต์เต่าเมื่อมองจากด้านหน้า

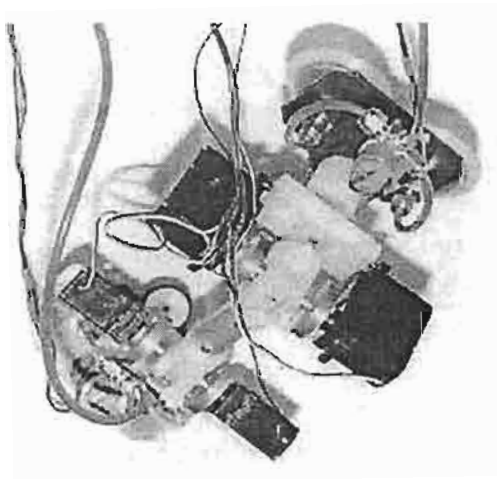


รูปที่ (ซ) การติดตั้งบอร์ดในตัวหุ่นยนต์เต่าเมื่อมองจากด้านข้าง

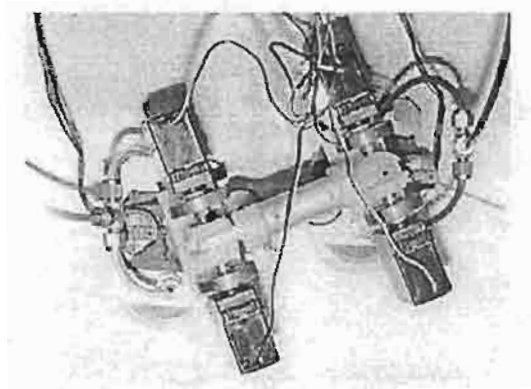
2. หุ่นยนต์หนอน



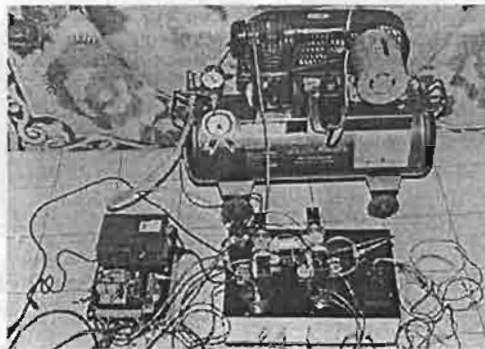
รูปที่ ก.1 ด้านหน้าหุ่นยนต์



รูปที่ ก.2 ขณะหุ่นยนต์เคลื่อนที่

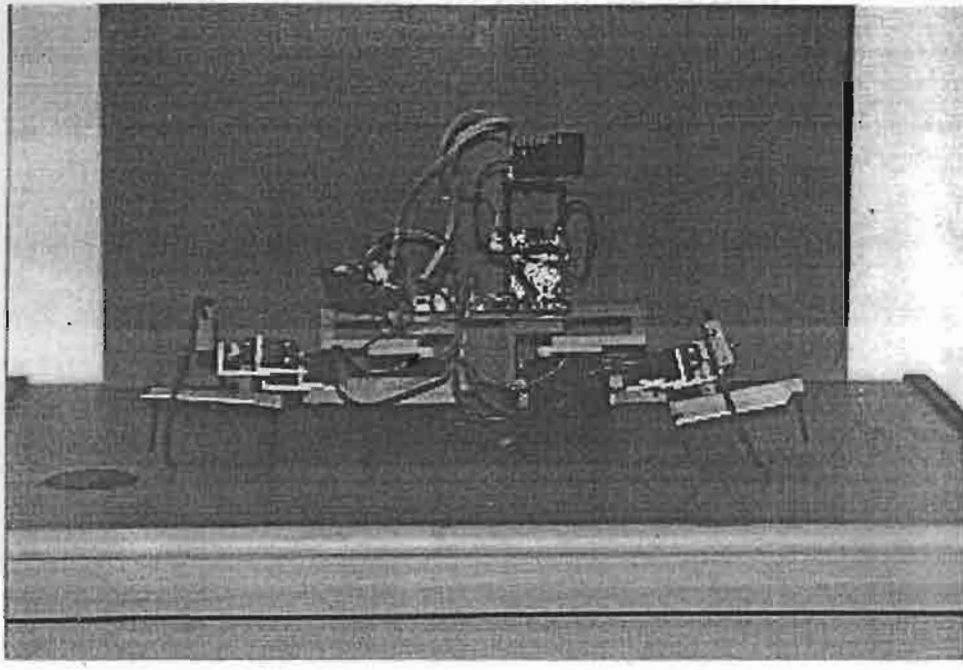


รูปที่ ก.3 หุ่นยนต์ขณะเกาะติดผนัง

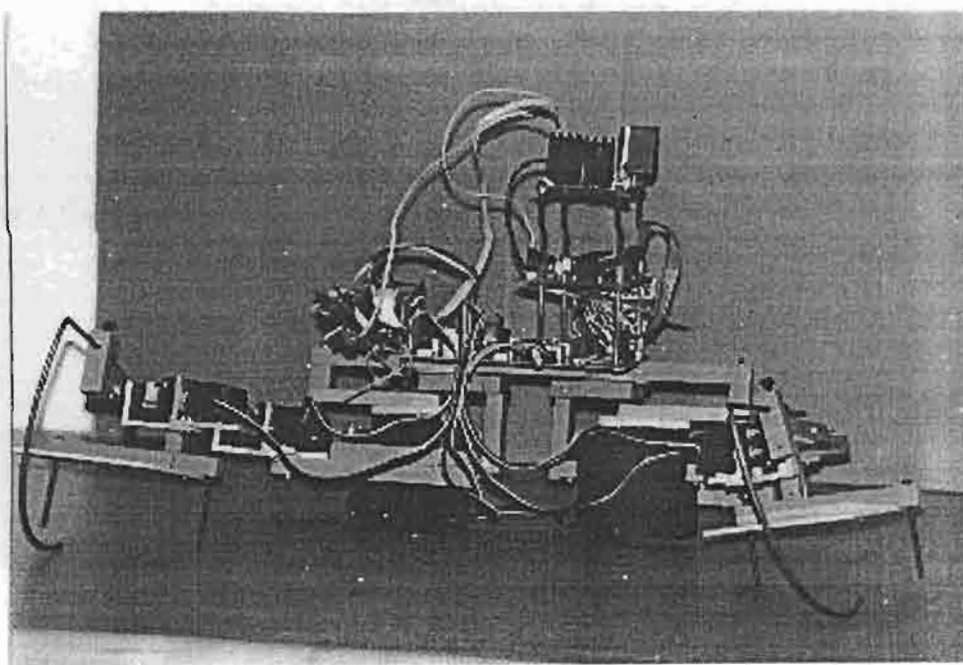


รูปที่ ก.4 ชุดอุปกรณ์หุ่นยนต์หนอนทั้งหมด

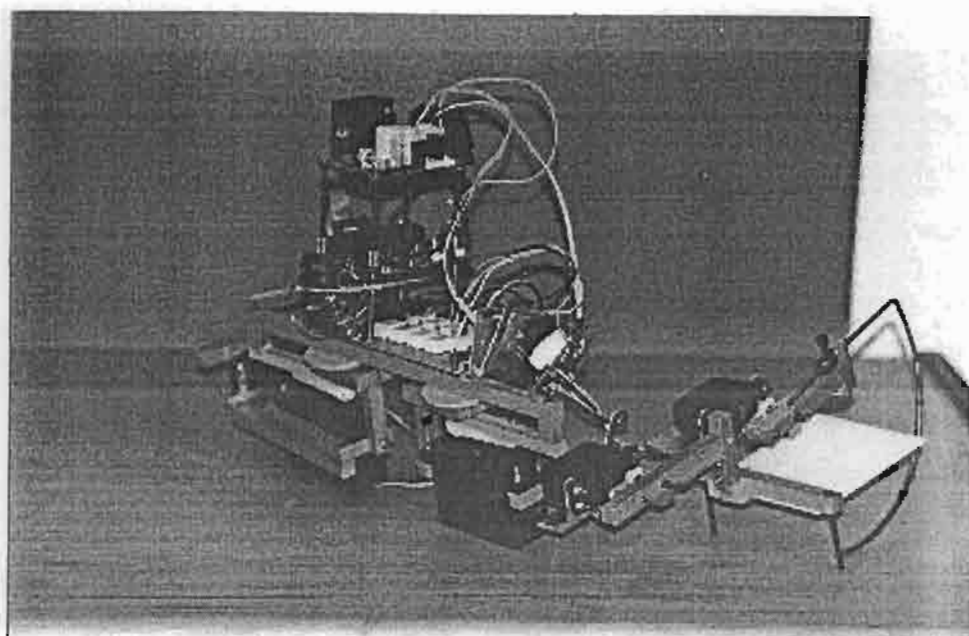
3. หุ่นยนต์ตกแตงไต้ผึ่ง



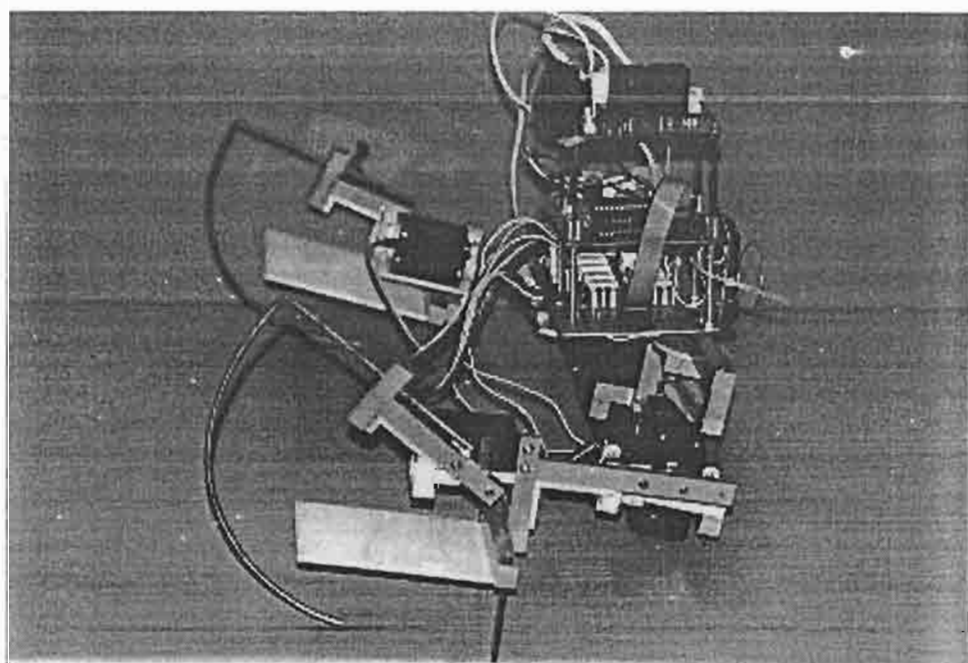
รูปที่ ก.1 ภาพถ่ายจากด้านหน้าของหุ่นยนต์



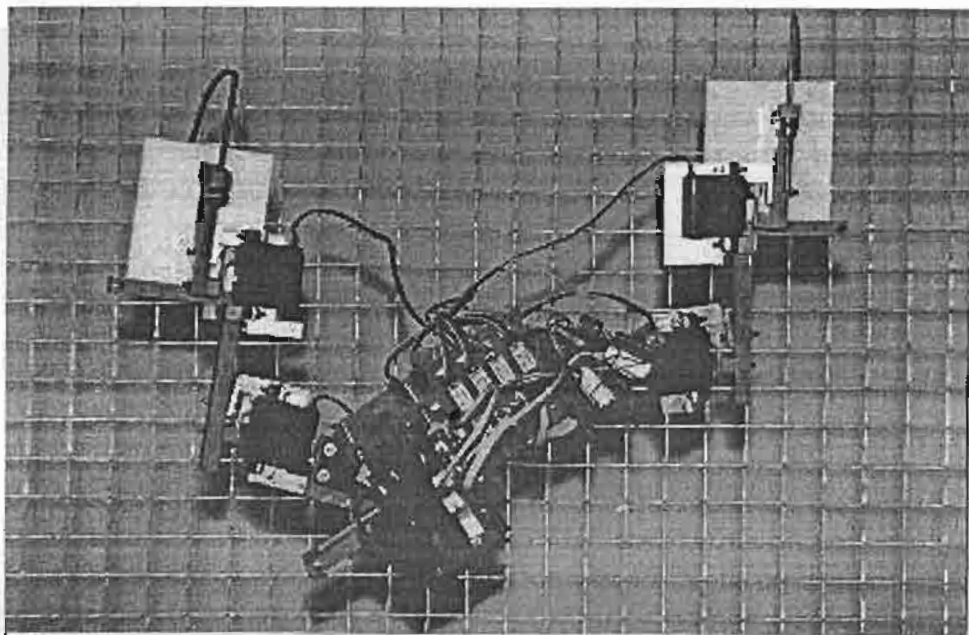
รูปที่ ก.2 ภาพถ่ายมุมเฉียงของหุ่นยนต์



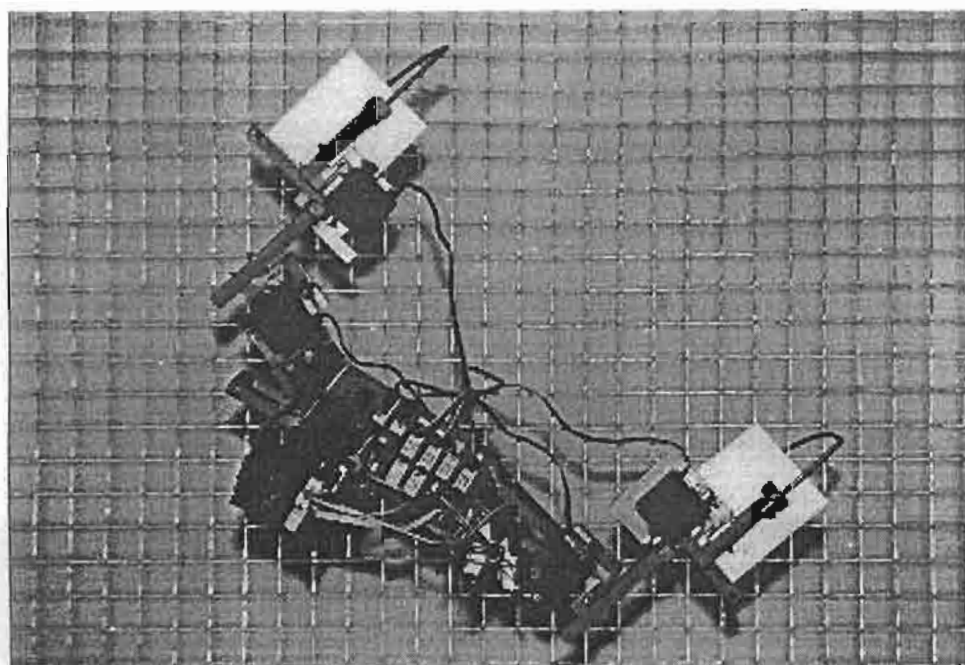
รูปที่ ก.3 ภาพถ่ายเฉียงด้านหลังของหุ่นยนต์



รูปที่ ก.4 ภาพถ่ายด้านข้างของหุ่นยนต์



รูปที่ ก.5 ภาพถ่ายขณะหุ่นยนต์กำลังไต่กำแพงตาข่าย



รูปที่ ก.6 ขณะหุ่นยนต์กำลังทำการเลี้ยว