

หลักเกณฑ์ที่เหลื่อมคางหมูเชิงประกอบสำหรับการแปลงลาปลาซผกผันเชิงตัวเลขของ ฟังก์ชันตรรกยะในระบบควบคุมแอลทีไอด้วยการโปรแกรมซีพลัสพลัส

Composite Trapezoidal Rule for Numerical Inverse Laplace Transform of Rational Functions in LTI Control Systems with C++ Programming

ชัยณรงค์ วิเศษศักดิ์วิชัย¹ และ ประเสริฐ เผ่าชู²

¹สาขาวิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีราชมงคลกรุงเทพ

²สาขาวิชาคณิตศาสตร์ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยเทคโนโลยีราชมงคลกรุงเทพ

โทร: 081-3437693 E-mail: chainarong.w@mail.rmutk.ac.th

บทคัดย่อ

บทความนี้นำเสนอการพัฒนาระเบียบวิธีเชิงตัวเลขสำหรับการแปลงลาปลาซผกผัน เพื่อใช้ในงานวิเคราะห์ระบบควบคุมแอลทีไอ (Linear Time Invariant; LTI) ด้วยคอมพิวเตอร์ โดยใช้ไลบรารีมาตรฐาน (Standard library) ของภาษาการโปรแกรมซีพลัสพลัส (C++) ที่เป็นแบบทรัพยากรเปิด (Open source) การใช้เทคนิคการโปรแกรมเชิงวัตถุ (Object-Oriented Programming; OOP) สร้างวัตถุเชิงวิธีดำเนินการแปลงลาปลาซผกผันเชิงตัวเลข ด้วยการทำปริพันธ์เชิงตัวเลขให้กับอินทิกรัลบรอมีวิช (Bromwich integral) ที่ดำเนินการลดรูปลงเป็นปริพันธ์ค่าจริงซึ่งมีเฉพาะตัวประกอบโคไซน์ และใช้หลักเกณฑ์ที่เหลื่อมคางหมูเชิงประกอบ สำหรับการแปรค่าช่วงกว้างของการปริพันธ์เชิงตัวเลข ให้ได้ผลความแม่นยำเป็นไปตามค่าคลาดเคลื่อนที่ยอมรับที่กำหนด โดยวัตถุเชิงโปรแกรมที่สร้างขึ้น ยังได้ถูกออกแบบให้ประกอบด้วยระเบียบวิธีเชิงตัวเลขอื่นที่เกี่ยวข้องได้แก่ ระเบียบวิธีของลาแกร์ (Laguerre's method) ขั้นตอนวิธีของฮอร์เนอร์ (Horner's algorithm) การหารสังเคราะห์ (Synthetic division) และการคูณซ้อนใน (nested multiplication) เพื่อใช้ในการหาศูนย์ของพหุนามสำหรับการเลือกคอนทัวร์ (Contour) ที่เหมาะสมในการหาค่าเชิงจุดเอกฐานของปริพันธ์ (Integrand) ของอินทิกรัลบรอมีวิช วัตถุเชิงวิธีดำเนินการแปลงลาปลาซผกผันเชิงตัวเลขด้วยหลักเกณฑ์ที่เหลื่อมคางหมู ของบทความนี้ ให้ผลเป็นที่น่าพอใจและมีค่าคลาดเคลื่อนในระดับต่ำ ทั้งในการหาผลการแปลงลาปลาซผกผันเชิงตัวเลขกับฟังก์ชันตรรกยะแบบเศษส่วนแท้ และผลตอบสนองขึ้นหนึ่งหน่วยของระบบควบคุมแอลทีไอ

คำสำคัญ : การโปรแกรมเชิงวัตถุ อินทิกรัลบรอมีวิช หลักเกณฑ์ที่เหลื่อมคางหมู ระเบียบวิธีของลาแกร์ ขั้นตอนวิธีของฮอร์เนอร์ การหารสังเคราะห์ การคูณซ้อนใน การแปลงลาปลาซผกผันเชิงตัวเลข

ABSTRACT

This paper presents a development of numerical methods for the inverse Laplace transform to analysis LTI (Linear Time Invariant) control systems by computer. The open source Standard C++ Library is used to create the programmatic objects by OOP (Object-Oriented Programming) for the numerical inverse Laplace transform object that uses numerical integration to the Bromwich integral. The composite trapezoidal rule is implemented on the simplified Bromwich integral which is real value integration that has only one cosine factor. The accuracy of numerical integration will be met by varying step size of the trapezoidal rule according to specified tolerance. The other numerical methods such that Laguerre's method, Horner's algorithm, synthetic division and nested multiplication will be included in the created programmatic objects for finding zeros of polynomial in order to avoid all singular points of integrand of Bromwich integral to make analytic integrand on contour of integration. The numerical inverse Laplace transform object of this article gives satisfactory results in both of the numerical inverse Laplace transform to the proper rational functions and unit step response of LTI control systems.

Keywords : Object-oriented programming, Bromwich integral, Trapezoidal rule, Laguerre's method, Horner's algorithm, Synthetic division, Nested multiplication, Numerical inverse Laplace transform

1. บทนำ

ความยากของการคำนวณผลเฉลยให้กับปัญหาทางคณิตศาสตร์ที่ประสบในงานวิศวกรรม กระตุ้นให้วิศวกรหรือนักวิจัย ทั้งในภาคอุตสาหกรรมและภาคการศึกษา ค้นหาขั้นตอนวิธี (Algorithms) สำหรับนำคอมพิวเตอร์เข้ามาเป็นเครื่องมือคำนวณหาผลเฉลยที่มีความยาก ให้ได้ผลเฉลยที่มีความแม่นยำและรวดเร็ว โดยเฉพาะกับงานวิเคราะห์สภาวะชั่วคราวของระบบทางวิศวกรรม การคำนวณผลการแปลงลาปลาซผกผันเชิงตัวเลข [1] จะเข้ามามีบทบาทสำคัญได้เช่นเดียวกับปัญหาการคำนวณสภาพการกระจายตัวของแรงดัน และกระแสบนสายส่งกำลังไฟฟ้า หรือ

แม้แต่ในขดลวดของหม้อแปลงไฟฟ้า [2-4] จะพบว่าระบบสมการเชิงอนุพันธ์ที่เกี่ยวข้อง ถูกทำให้อยู่ในรูปแบบที่ง่ายต่อการหาผลเฉลยด้วยผลการแปลงลาปลาซ ผลเฉลยที่คำนวณได้ในโดเมนของลาปลาซจะถูกแปลงกลับไปสู่โดเมนเวลาด้วยคอมพิวเตอร์ โดยใช้ผลการแปลงลาปลาซผกผันเชิงตัวเลข

ผลการแปลงลาปลาซผกผันเชิงตัวเลข ด้วยการหาปริพันธ์โดยตรง ได้ถูกนำเสนอโดย Dubner และ Abate [5-6] ซึ่งถูกเรียกว่าระเบียบวิธีอนุกรมฟูรีเยร์ (Fourier series method) ระเบียบวิธีนี้มี

ความง่ายในการนำไปดำเนินงานจริง แต่อาจจะประสบกับการเข้าสู่ผลเฉลยที่ช้า และมีลักษณะดังนี้

$$\tilde{f}(t) = \frac{e^{at}}{T} \sum_{n=0}^{\infty} \begin{pmatrix} \operatorname{Re} \left[F(a + j\frac{n\pi}{T}) \cos\left(\frac{n\pi t}{T}\right) \right] \\ - \operatorname{Im} \left[F(a + j\frac{n\pi}{T}) \sin\left(\frac{n\pi t}{T}\right) \right] \end{pmatrix} \quad (1)$$

เมื่อ $\tilde{f}(\cdot)$ คือการแปลงลาปลาซผกผันเชิงตัวเลขด้วยระเบียบวิธีอนุกรมฟูริเยร์ และ $F(\cdot)$ คือผลการแปลงลาปลาซสำหรับ $t \in [0, \frac{T}{2}]$ ในที่ซึ่ง $T > 0$ และ $a \in \mathbf{R}$ โดยที่ $a > \sigma_0$ ทั้งนี้พิกัดที่หนึ่ง (abscissa) ของบริเวณการลู่เข้ากำหนดด้วย σ_0 โดยความแม่นยำของระเบียบวิธีนี้ขึ้นกับพารามิเตอร์ a และจำนวนพจน์ที่ใช้ในการประเมินค่า [7]

เป้าหมายของงานในที่นี้คือ การนำเสนอระเบียบวิธีผลการแปลงลาปลาซผกผันเชิงตัวเลขสำหรับอินทิกรัลบรอมวิช โดยจะลดรูปลงให้เหลือเป็นปริพันธ์ค่าจริงซึ่งมีเฉพาะตัวประกอบโคไซน์สำหรับการหาปริพันธ์เชิงตัวเลขด้วยหลักเกณฑ์ที่เรียบง่ายทั้งนี้ค่าความแม่นยำของปริพันธ์เชิงตัวเลขจะเป็นไปตามค่าคลาดเคลื่อนยินยอม (Tolerance) ที่กำหนด สำหรับการวิเคราะห์ระบบควบคุมแอลทีโอ โดยใช้ไลบรารีมาตรฐานของภาษาการโปรแกรมซี พลัสพลัสซึ่งเป็นแบบทรัพยากรเปิด

2. หลักการที่เกี่ยวข้อง

ความสัมพันธ์แบบอินพุต-เอาต์พุตของระบบเชิงเส้นตัวแปรเดียวเป็นระบบเชิงเหตุ (Causal system) ไม่แปรตามเวลาและไม่มีพลังงานเก็บสะสมที่เวลาใดๆ คือ

$$y(t) = \int_{t_0}^t g(t-\tau)x(\tau)d\tau \quad (2)$$

โดยที่ $x(\cdot)$ คืออินพุต $g(\cdot)$ คือผลตอบสนองอิมพัลส์ของระบบ และ $y(\cdot)$ คือเอาต์พุต เมื่อทำการหาผลการแปลงลาปลาซของเอาต์พุตด้วย

$$Y(s) = \int_0^{\infty} y(t)e^{-st} dt \quad (3)$$

ทำให้เอาต์พุตของระบบไม่แปรตามเวลาเชิงเส้น ถูกแปลงจากโดเมนเวลา t ไปอยู่ในโดเมนความถี่เชิงซ้อน s ดังนี้

$$Y(s) = \int_{t=0}^{\infty} \left(\int_{\tau=0}^{\infty} g(t-\tau)x(\tau)d\tau \right) e^{-s(t-\tau)} e^{-s\tau} dt \quad (4)$$

เมื่อใช้ตัวแปรใหม่ $v = t - \tau$ จะจัดสมการได้เป็น

$$Y(s) = \int_{v=0}^{\infty} g(v)e^{-sv} dv \int_{\tau=0}^{\infty} x(\tau)e^{-s\tau} d\tau = G(s)X(s) \quad (5)$$

โดยฟังก์ชันถ่ายโอน (Transfer function) ของระบบแอลทีโอ $G(s)$ จะมีลักษณะดังต่อไปนี้

$$G(s) = \frac{c_1 s^q + c_2 s^{q-1} + c_3 s^{q-2} + \dots + c_q s + c_{q+1}}{s^r + b_1 s^{r-1} + b_2 s^{r-2} + \dots + b_{r-1} s + b_r} \quad (6)$$

ในที่ซึ่ง $c_i (i = 1, 2, 3, \dots, q+1)$ และ $b_j (j = 1, 2, 3, \dots, r)$ เป็นค่าคงที่ สำหรับระบบไม่แปรตามเวลาเชิงเส้น พหุนามตัวเศษของระบบจะมีระดับชั้น q และพหุนามตัวส่วนมีระดับชั้น r โดยให้ $q \leq r$ เพื่อให้ระบบ มีความสามารถเป็นได้จริงทางกายภาพ (Physically realizable system) ส่งผลให้เอาต์พุตของระบบแอลทีโอ ได้ผลการคำนวณอยู่ในรูปฟังก์ชันตรรกยะดังนี้

$$Y(s) = \frac{N(s)}{D(s)} \quad (7)$$

ฟังก์ชันตรรกยะ $Y(s)$ จะมี 2 ความเป็นไปได้เมื่อพิจารณาสัมพัทธ์กับระดับชั้นของ $N(s)$ และ $D(s)$ กล่าวคือ เมื่อกำหนดให้ระดับชั้นของ $N(s)$ คือ m และระดับชั้นของ $D(s)$ คือ n ฟังก์ชันตรรกยะ $Y(s)$ เป็นเศษส่วนแท้ (Proper fraction) เมื่อ $m < n$ แต่ถ้า $m \geq n$ แล้ว $Y(s)$ จะเป็นเศษเกิน (Improper fraction)

2.1 ปริพันธ์ตามเส้นเชิงซ้อนและการแปลงลาปลาซผกผัน

จากที่กำหนดให้พหุนาม $D(s)$ ของ $Y(s)$ มีศูนย์ทั้งหมด n ค่าในระนาบเชิงซ้อน และโดยการกำหนดให้จำนวนจริง σ มีค่ามากกว่า c ($\sigma > c$) เมื่อ c คือส่วนจริงของศูนย์ของพหุนาม $D(s)$ ที่อยู่ทางขวาสุดของระนาบเชิงซ้อน แล้วอินทิกรัลบรอมีวิช [8] สำหรับคำนวณเอาต์พุต $y(t)$ ในโดเมนเวลาจากเอาต์พุต $Y(s)$ ในโดเมนความถี่เชิงซ้อนคือ

$$y(t) = \frac{1}{2\pi j} \int_{\sigma-j\infty}^{\sigma+j\infty} Y(s)e^{st} ds \quad (8)$$

หรือ

$$\bar{y}(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \text{Re}[Y(\sigma + j\omega)] \cos(\omega t) d\omega - \frac{1}{2\pi} \int_{-\infty}^{\infty} \text{Im}[Y(\sigma + j\omega)] \sin(\omega t) d\omega$$

เมื่อ $\bar{y}(t) = y(t)e^{-\sigma t}$ และถ้า $y(t) = 0$ เมื่อ $t < 0$ ดังนั้น

$$\begin{aligned} \bar{y}(t) + \bar{y}(-t) &= \frac{1}{\pi} \int_{-\infty}^{\infty} \text{Re}[Y(\sigma + j\omega)] \cos(\omega t) d\omega \quad (9) \\ \bar{y}_e(t) &= \frac{2}{\pi} \int_0^{\infty} \text{Re}[Y(\sigma + j\omega)] \cos(\omega t) d\omega \end{aligned}$$

เมื่อ $\bar{y}_e(t)$ คือฟังก์ชันคู่และ $\bar{y}(t) = \bar{y}_e(t)u(t)$ โดย $u(t)$ คือฟังก์ชันขั้นหนึ่งหน่วย (Unit step function) ซึ่งทำให้ได้ว่า

$$y(t) = \frac{2e^{\sigma t}}{\pi} \int_0^{\infty} \text{Re}[Y(\sigma + j\omega)] \cos(\omega t) d\omega \quad (10)$$

เมื่อ $t > 0$ และเพื่อนำใช้หลักเกณฑ์ที่สลับคางหมูเชิงประกอบที่มีช่วงกว้างกำหนดโดยจุด $\omega_0 = 0, \omega_1, \omega_2, \dots, \omega_n$ และค่าคงที่ h ในลักษณะ $\omega_i = \omega_0 + ih$ เมื่อ $i = 1, 2, 3, \dots, n$ จึงจัดผลการแปลงลาปลาซผกผันใหม่ดังนี้

$$y(t) = \frac{2e^{\sigma t}}{\pi} \lim_{n \rightarrow \infty} \sum_{i=1}^n \int_{\omega_{i-1}}^{\omega_i} \text{Re}[Y(\sigma + j\omega)] \cos(\omega t) d\omega \quad (11)$$

2.2 การแปลงลาปลาซผกผันเชิงตัวเลข

อินทิกรัลบรอมีวิชของ $y(t)$ เมื่อเขียนใหม่โดยใช้หลักเกณฑ์ที่สลับคางหมูเชิงประกอบ จะได้ผลการแปลงลาปลาซผกผันเชิงตัวเลข $\bar{y}(t)$ ดังนี้

$$\begin{aligned} \bar{y}(t) &= \frac{2e^{\sigma t} h}{\pi} \lim_{n \rightarrow \infty} \left(\frac{\operatorname{Re}[Y(\sigma + j\omega_0)] \cos(\omega_0 t)}{2} + \right. \\ &\quad \left. \frac{\operatorname{Re}[Y(\sigma + j\omega_n)] \cos(\omega_n t)}{2} \right. \\ &\quad \left. + \sum_{i=1}^{n-1} \operatorname{Re}[Y(\sigma + j\omega_i)] \cos(\omega_i t) \right) \\ \bar{y}(t) &= \frac{2e^{\sigma t} h}{\pi} \left(\frac{\operatorname{Re}[Y(\sigma)]}{2} + \sum_{i=1}^{\infty} \operatorname{Re}[Y(\sigma + j\omega_i)] \cos(\omega_i t) \right) \end{aligned} \quad (12)$$

พารามิเตอร์ σ ของการแปลงลาปลาซ ผกผันเชิงตัวเลขเป็นผลจากอินทิกรัลบรอมีวีก และมีเกณฑ์การกำหนดค่าตามที่กล่าวแล้วข้างต้น โดยสามารถเจาะจงค่าที่แน่นอนได้ ด้วยการพิจารณาความจริงของตัวอย่างผลการแปลงลาปลาซผกผันต่อไปนี้ โดยการกำหนดให้ $\sigma = c + c'$ และ

$$\begin{aligned} Y(s) &= 1/(1+s) \\ y(t) &= \frac{e^{(-1+c')t}}{2\pi} \int_{-\infty}^{\infty} \frac{e^{j\omega t} d\omega}{1+[(1+c') + j\omega]} \\ &= \frac{e^{-t}}{2\pi} e^{c't} \int_{-\infty}^{\infty} \frac{e^{j\omega t} d\omega}{c't + j\omega} \end{aligned} \quad (13)$$

กรณีตัวอย่างนี้เห็นได้ชัดว่า $y(t) = e^{-t}$ การหาการแปลงลาปลาซผกผันเชิงตัวเลข ทำให้เกิดค่าคลาดเคลื่อนในการหาค่าปริพันธ์ จึงกำหนดให้ $\varepsilon(t)$ คือค่าคลาดเคลื่อนสัมบูรณ์ของการหาปริพันธ์เชิงตัวเลข และ $\Delta y(t)$ คือค่าคลาดเคลื่อนใน $y(t)$ ดังนั้น

$$\Delta y(t) = \frac{\varepsilon(t)e^{-t}}{2\pi} e^{c't} \quad (14)$$

หรือเขียนใหม่ในรูปค่าคลาดเคลื่อนสัมพัทธ์ (Relative error) คือ

$$\frac{\Delta y(t)}{y(t)} = \frac{\varepsilon(t)}{2\pi} e^{c't} \quad (15)$$

ที่แสดงให้เห็นว่าการเลือก c' ที่มีค่าโตมากเกินไปจะทำให้พจน์ $e^{c't}$ ของค่าคลาดเคลื่อนสัมพัทธ์เติบโตขึ้นอย่างรวดเร็วตามเวลา t แต่ในทางกลับกันคือเมื่อเลือก c' มีค่าน้อยจนเกินไปจะส่งผลให้คอนทัวร์วางตัวชิดติดกับจุดเอกฐานของปริพันธ์ในอินทิกรัลบรอมีวีก ดังนั้นจะทำการเลือกค่า c' ให้ขึ้นกับเวลา t [9] โดยให้ $c' = 1/t$ ทำให้พารามิเตอร์ σ ของการแปลงลาปลาซผกผันเชิงตัวเลขกำหนดได้ด้วย

$$\sigma = c + \frac{1}{t} \quad (16)$$

2.3 ระเบียบวิธีของลาแกร์

การคำนวณค่าส่วนจริง c จากศูนย์ของพหุนาม $D(s)$ สามารถทำได้ด้วยระเบียบวิธีของลาแกร์ที่ใช้ความจริงของอนุพันธ์บนพหุนาม $D(s)$ ดังนี้

$$D'(s) = \sum_{j=1}^n \frac{D(s)}{s - r_j} \quad (17)$$

สำหรับใช้ทำการนิยามค่า A ดังสมการ

$$A = -\frac{D'(s)}{D(s)} \quad (18)$$

แล้วทำการหาอนุพันธ์เข้าไปในผลหารพหุนามข้างต้นเพื่านิยามค่า B

$$B = -\frac{-D(s)D''(s) + [D'(s)]^2}{[D(s)]^2} \quad (19)$$

จึงทำให้เขียน B ในพจน์ของ A ได้ดังนี้

$$B = A^2 - \frac{D''(s)}{D(s)} \quad (20)$$

ส่งผลให้เมื่อทำการเลือกค่า C จากสมการ

$$C = n^{-1} \left[A \pm \sqrt{(n-1)(nB-A^2)} \right] \quad (21)$$

โดยเลือกเอาค่า C ที่มี $|C|$ มากที่สุด จะส่งผลให้พหุนาม $D(s)$ ระดับชั้น n มีศูนย์ในระนาบเชิงซ้อน สัมพันธ์กับ s ซึ่งเป็นจำนวนเชิงซ้อนใดๆ อยู่ในระยะห่าง $\sqrt{n}/C$ ของ s ในระนาบเชิงซ้อน [10] โดยจะคำนวณศูนย์ดังกล่าวด้วยสูตร

$$s_{new} = s + \frac{1}{C} \quad (22)$$

2.4 ค่าคลาดเคลื่อนของหลักเกณฑ์สี่เหลี่ยม

คางหมู

พิจารณาที่ช่วงกว้างใดช่วงกว้างหนึ่ง ของปริพันธ์เชิงตัวเลข ตามหลักเกณฑ์สี่เหลี่ยมคางหมูดังนี้

$$I = \int_{\omega_i}^{\omega_j} \mathcal{Y}(\omega) d\omega = \mathcal{Y}(\omega_i) - \mathcal{Y}(\omega_j) \quad (23)$$

เมื่อ

$$\mathcal{Y}(\omega) \propto \operatorname{Re} [Y(\sigma + j\omega)] \cos(\omega t) \quad (24)$$

และ $\mathcal{Y}'(\omega) = \mathcal{Y}(\omega)$ แต่จากที่ $h = \omega_j - \omega_i$ จะได้

$$\int_{\omega_i}^{\omega_j} \mathcal{Y}(\omega) d\omega \approx I_T = \frac{h}{2} [\mathcal{Y}(\omega_i) + \mathcal{Y}(\omega_j)] \quad (25)$$

เนื่องจาก $Y(\sigma + j\omega)$ เป็นฟังก์ชันวิเคราะห์ (Analytic function) บนคอนทัวร์ของอินทิกรัลบรอมีวีก $\mathcal{Y}(\omega)$ จึงขยายได้ด้วยอนุกรมเทย์เลอร์

$$\mathcal{Y}(\omega_j) = \mathcal{Y}(\omega_i) + h \mathcal{Y}'(\omega_i) + \frac{h^2}{2} \mathcal{Y}''(\xi) \quad (26)$$

เมื่อ $\omega_i < \xi < \omega_j$ ดังนั้นค่า I_T ในสมการข้างต้นเขียนใหม่ได้เป็น

$$I_T = \frac{h}{2} [2\mathcal{Y}(\omega_i) + h \mathcal{Y}'(\omega_i) + \frac{h^2}{2} \mathcal{Y}''(\xi)] \quad (27)$$

จากที่ $\mathcal{Y}(\omega)$ คืออนุพันธ์ของ $\mathcal{Y}(\omega)$ ทำให้การหาปริพันธ์เข้าไปในสมการของ $\mathcal{Y}(\omega_j)$ ได้

$$\mathcal{Y}(\omega_j) = h \mathcal{Y}(\omega_i) + \frac{h^2}{2} \mathcal{Y}'(\omega_i) + \frac{h^3}{3!} \mathcal{Y}''(\xi) + C \quad (28)$$

เมื่อ C คือค่าคงที่ของการหาปริพันธ์และสามารถคำนวณค่าได้เท่ากับ

$$C = \mathcal{Y}(\omega_i)$$

ดังนั้นได้

$$\mathcal{Y}(\omega_j) = \mathcal{Y}(\omega_i) + h \mathcal{Y}(\omega_i) + \frac{h^2}{2} \mathcal{Y}'(\omega_i) + \frac{h^3}{3!} \mathcal{Y}''(\xi) \quad (29)$$

หรือ

$$I = \mathcal{U}(\omega_i) - \mathcal{U}(\omega_j) = h\mathcal{U}(\omega_i) + \frac{h^2}{2}\mathcal{U}'(\omega_i) + \frac{h^3}{3!}\mathcal{U}''(\xi)$$

นำไปสู่การคำนวณผลต่างระหว่าง I_T กับ I สำหรับกำหนดค่าคลาดเคลื่อนได้ดังนี้

$$I_T - I = \frac{h^3}{12}\mathcal{U}''(\xi) \quad (30)$$

ซึ่งเห็นได้ชัดว่า หลักเกณฑ์ที่สี่เหลี่ยมคางหมุของการแปลงลาปลาซผกผันเชิงตัวเลข มีอันดับของการประมาณค่า (Order of approximation) อยู่ที่ $O(h^3)$ เพราะฉะนั้น การใช้หลักเกณฑ์สี่เหลี่ยมคางหมุเชิงประกอบ จะเปิดโอกาสให้สามารถเลือกค่า h ให้มีค่าน้อยเพียงพอที่จะทำให้การหาปริพันธ์เชิงตัวเลขมีค่าคลาดเคลื่อนยินยอมอยู่ภายใต้ความต้องการที่กำหนด

2.5 การโปรแกรมเชิงวัตถุของภาษาการโปรแกรมซีพลัสพลัส

คลาส (class) ของภาษาการโปรแกรมซีพลัสพลัส (C++ programming language) จะเป็นแม่แบบในการสร้างวัตถุของโอโอพีที่มีลักษณะดังภาพที่ 1

```
class ClassName
{
    public:
        //public member

    private:
        //private member
}
```

ภาพที่ 1 แม่แบบวัตถุของภาษาการโปรแกรมซีพลัสพลัส

ทั้งนี้โดยปกติ ภายในคลาสจะประกอบด้วยสมาชิก 2 แบบคือ สมาชิกข้อมูล (data members)

และสมาชิกฟังก์ชัน (function members) ซึ่งมีแม่แบบดังภาพที่ 2

```
TypeDefinition ClassName::FunctionName(arg1, ... )
{
    Declarations
    Statements
    return Value ;
}
```

ภาพที่ 2 แม่แบบสมาชิกฟังก์ชันภายในคลาสของภาษาการโปรแกรมซีพลัสพลัส

การสร้างคลาสให้กับการหาค่าการแปลงลาปลาซผกผันเชิงตัวเลข ในที่นี้จะทำการสร้างทั้งหมด 2 คลาสเพื่อให้ได้ 2 ตัวตั้งต้นแบบ (constructor) ตามหลักการของการโปรแกรมเชิงวัตถุ เพื่อใช้ในการคำนวณศูนย์ของพหุนาม $D(s)$ และการคำนวณค่าของอินทิกรัลบรอมวิช ด้วยหลักเกณฑ์สี่เหลี่ยมคางหมุเชิงประกอบ

3. วิธีการดำเนินงาน

เนื่องจากการคำนวณศูนย์ให้กับพหุนามด้วยระเบียบวิธีของลาแกร์ จะหาได้ครั้งละหนึ่งค่าถ้าเป็นศูนย์จำนวนจริง และครั้งละสองค่าหากเป็นศูนย์เชิงซ้อน ดังนั้นการคำนวณศูนย์ทั้งหมด n ค่าของ $D(s)$ ที่มีรูปแบบ

$$D(s) = a_n s^n + a_{n-1} s^{n-1} + a_{n-2} s^{n-2} + \dots + a_1 s + a_0 \quad (31)$$

จะต้องใช้ระเบียบวิธีเชิงตัวเลขเพิ่มเติมดังต่อไปนี้

3.1 ขั้นตอนวิธีของฮอร์เนอร์

การที่กำหนดให้ $D(s) = \sum_{k=0}^n a_k s^k$ มีระดับ

ขั้น n ขั้นตอนวิธีของฮอร์เนอร์จะคำนวณพหุนาม

$$d(s) = \sum_{k=0}^{n-1} b_k s^k \quad (32)$$

ที่มีระดับชั้นน้อยกว่า $D(s)$ อยู่ 1 ที่ทำให้สมการต่อไปนี้เป็นความจริง

$$d(s) = \frac{D(s) - r}{s - s_0} \quad (33)$$

เมื่อ r คือเศษเหลือและ s_0 คือค่าคงที่ โดยใช้วิธีทำซ้ำ (iteration) ต่อไปนี้

$$b_{k-1} = a_k + s_0 b_k ; n \geq k \geq 0 \quad (34)$$

เมื่อ $b_n = 0$ และ $r = b_{-1}$

3.2 การหารสังเคราะห์

สัมประสิทธิ์ของ $D(s)$ ประกอบด้วย a_k :

$$0 \leq k \leq n \quad \text{ดังนั้นหากกำหนดว่า } q(s) = \sum_{k=2}^n b_k s^{k-2}$$

คือผลหารของ $D(s)$ ด้วย $s^2 - us - v$ และมีเศษเหลือ $r(s)$ จะได้ว่า

$$\sum_{k=0}^n a_k s^k = \sum_{k=2}^n b_k s^{k-2} (s^2 - us - v) + r(s) \quad (35)$$

การคำนวณหา $q(s)$ ด้วยวิธีทำซ้ำ

$$b_k = a_k + ub_{k+1} + vb_{k+2} ; n \geq k \geq 2 \quad (36)$$

และ $r(s) = b_1(s - u) + b_0$ เมื่อ $b_{n+1} = b_{n+2} = 0$ คือการหารสังเคราะห์

3.3 การคูณซ้อนใน

เนื่องจากพหุนาม $P(s)$ ที่มีระดับชั้น n สามารถเขียนได้ ในลักษณะการคูณซ้อนใน

$$P(s) = (\cdots(((a_n)s + a_{n-1})s + a_{n-2})s + \cdots + a_1)s + a_0 \quad (37)$$

ดังนั้นการประเมินค่าพหุนามจะดำเนินการด้วยวิธีทำซ้ำต่อไปนี้

$$P_{k-1} = P_k s_0 + a_{k-1} ; n \geq k \geq 1 \quad (38)$$

เมื่อ $P_n = a_n$ และ $P(s_0) = P_0$ เพื่อใช้ในการประเมินค่าของ $Y(s)$ ซึ่งประกอบด้วย $N(s)$ และ $D(s)$ ที่ค่าคงที่จำนวนเชิงซ้อน s_0 ใดๆ ในการแปลงลาปลาซผกผันเชิงตัวเลข

3.4 การสร้างคลาสสำหรับคำนวณศูนย์ของพหุนาม

บทความนี้ได้ใช้ไลบรารีมาตรฐานซีพลัสพลัส สำหรับนาระเบียบวิธีเชิงตัวเลขที่ได้สร้างขึ้น ทั้งในส่วนของการหลักการที่เกี่ยวข้องก่อนหน้า และขั้นตอนวิธีต่างๆ ที่ได้นำเสนอดังนั้น คลาสที่ออกแบบสำหรับการคำนวณศูนย์ของพหุนาม [11] จึงมีลักษณะดังภาพที่ 3

```
class Poly
{ public:
  int PolyErr, degree;
  double *coeff;
  Poly(void) { }
  Poly(int, double *);
  Poly operator+(Poly);
  Poly operator*(Poly);
  Poly Horner_Divide(double, double, double );
  Poly Synthetic_Divide(double, double,
                        double, double);
  complex <double> Evaluate(complex <double>);
  int Laguerre_Roots(polyRoot*);
}
```

ภาพที่ 3 วัตถุเชิงพหุนามของการโปรแกรมซีพลัสพลัส

วัตถุเชิงพหุนามนี้ มีนามทางภาษาการโปรแกรมซีพลัสพลัสว่า Class Poly โดยมีสมาชิกฟังก์ชัน

Int Laguerre_Roots(polyRoot*);
ทำหน้าที่คำนวณศูนย์ของพหุนามด้วยระเบียบวิธีของลาแกร์ และเมื่อคำนวณค่าได้สำเร็จพหุนามจะถูกลดทอนระดับชั้นลง ด้วยขั้นตอนวิธีของฮอร์เนอร์ โดยสมาชิกฟังก์ชัน

Poly Horner_Divide(double, double, double);

เมื่อศูนย์ของพหุนามเป็นจำนวนจริง หรือลดทอนระดับชั้นลงด้วยการหารสังเคราะห์ด้วยฟังก์ชันสมาชิก

Poly Synthetic_Divide(double, double, double, double);

เมื่อศูนย์ของพหุนามเป็นจำนวนเชิงซ้อน

การสร้างวัตถุเชิงพหุนามตามที่วางแบบไว้ในภาพที่ 3 จะมีตัวตั้งต้นแบบดังนี้

Poly (int, double *);

ซึ่งจะต้องใช้ 2 อาร์กิวเมนต์ ได้แก่ ค่าจำนวนเต็มระดับชั้นของพหุนาม และเลขที่อยู่ (Address) ระบุค่าสัมประสิทธิ์ของพหุนาม การสร้างหน่วยวัตถุ (Instance) ของพหุนามใดๆ ที่ต้องการ เช่น

$$P_k(s) = c_k s^k + c_{k-1} s^{k-1} + c_{k-2} s^{k-2} + \dots + c_1 s + c_0$$

(39)

จะทำได้โดยใช้ข้อความสั่ง (Statement) ที่มีลักษณะดังต่อไปนี้

Poly Ps = Poly(Degree, Coefficient);

เมื่อ Ps คือชื่อที่ตั้งให้กับหน่วยวัตถุเชิงพหุนาม ส่วน Degree และ Coefficient คือตัวแปรหน่วยความจำแบบจำนวนเต็ม (ชนิด int) และตัวแปรหน่วยความจำชี้เลขที่อยู่ของแถวลำดับ แบบเลขทศนิยม (ชนิด Double) สำหรับการจัดเก็บข้อมูลระดับชั้นและสัมประสิทธิ์ของพหุนามไว้ ตามลำดับ เมื่อหน่วยวัตถุเชิงพหุนามถูกสร้างขึ้นแล้ว การคำนวณศูนย์ทั้งหมดของพหุนามด้วยระเบียบวิธีของลาแกร์ทำได้ด้วยข้อความสั่ง

Ps.Laguerre_Roots(Roots);

เมื่อ Roots คือชื่อที่ตั้งให้กับตัวแปรหน่วยความจำชี้เลขที่อยู่ของข้อมูลแบบโครงสร้าง (Structure) ที่สร้างขึ้นดังนี้

```
typedef struct polyRootTag {
    double real;
    double imag;
    Boolean isComplex;
}polyRoot;
```

สำหรับทำหน้าที่จัดเก็บข้อมูลศูนย์ของพหุนามที่คำนวณได้ในโปรแกรม

3.5 การสร้างคลาสสำหรับการแปลงลาปลาซผกผันเชิงตัวเลข

การสร้างตัวแบบที่มีลักษณะเป็นนามธรรมทางข้อมูลเชิงวิธีดำเนินการแปลงลาปลาซผกผันเชิงตัวเลขด้วยหลักเกณฑ์ที่เชื่อมโยงค่างหุ ได้ทำการจัดโครงสร้างของสมาชิกข้อมูล และสมาชิกฟังก์ชันในตัวแบบคลาสซึ่งได้กำหนดชื่อให้เป็น NumerInvLaplaceTrf ดังภาพที่ 4

```
class NumerInvLaplaceTrf
{ public:
    double AvrTolerance, SumTolerance;
    double AvrStep_h, SumStep_h;
    int NTimeCompute;
    Poly *Ns, *Ds;
    struct NILTsolution NILTanswer;
    float t; double Epsilon, Tolerance;
    NumerInvLaplaceTrf(void) { };
    NumerInvLaplaceTrf(Poly *, Poly *,
                        float, float);
    void Trapezoid(float);
    void Print(void);
};
```

ภาพที่ 4 วัตถุเชิงวิธีดำเนินการแปลงลาปลาซผกผันเชิงตัวเลขด้วยหลักเกณฑ์ที่เชื่อมโยงค่างหุ

จะเห็นได้ชัดในภาพที่ 4 ว่าวัตถุเชิงพหุนามถูกบรรจุให้เป็นสมาชิกข้อมูลของวัตถุเชิงวิธีดำเนินการแปลงลาปลาซผกผันเชิงตัวเลขด้วยหลักเกณฑ์ที่เชื่อมโยงค่างหุ โดยมีสองหน่วยวัตถุเชิงพหุนามคือ Ns กับ Ds ที่จัดให้เป็นตัวแปรหน่วยความจำชื่อที่

อยู่ สำหรับการจัดเก็บข้อมูลพหุนามเศษ $N(s)$ และพหุนามส่วน $D(s)$ ของฟังก์ชันตรรกยะ $Y(s)$ ที่จะทำให้การหาการแปลงลาปลาซผกผันเชิงตัวเลข $\tilde{y}(t)$ โดยทุกครั้งที่มีการหาผลบวกของพจน์ต่อไปนี้

$$\frac{Re[Y(\sigma)]}{2} + \sum_{i=1}^{\infty} Re[Y(\sigma + j\omega_i)] \cos(\omega_i t) \quad (40)$$

ให้กับ $\tilde{y}(t)$ การคำนวณค่า $Re[Y(\sigma + j\omega_i)]$ จะดำเนินงานด้วยการใช้สมาชิกฟังก์ชันของวัตถุเชิงพหุนามตามทีออกแบบไว้ในภาพที่ 3 คือ

```
complex<double>
    Evaluate(complex<double>);
```

สำหรับการคูณซ้อนในเพื่อหาค่า $Y(\sigma + j\omega_i)$ ด้วยข้อความสั่งดังต่อไปนี้

```
complex<double>ns = Ns->
    Evaluate(complex<double>(C,w));
complex<double>ds = Ds->
    Evaluate(complex<double>(C,w));
complex<double>Ys = ns/ds;
```

เมื่อ ns, ds, และ Ys คือตัวแปรหน่วยความจำชนิดเชิงซ้อนของไลบรารีมาตรฐานซีพลัสพลัส โดย C และ w คือตัวแปรหน่วยความจำที่จัดเก็บค่า σ และ ω_i ของ Y ตามลำดับ แต่จากที่กฎเกณฑ์ในการเลือกค่า σ จะต้องคำนวณศูนย์ของพหุนาม $D(s)$ ดังนั้นด้วยหน่วยวัตถุเชิงพหุนาม Ds ศูนย์ของพหุนามจะสามารถคำนวณได้ด้วยข้อความสั่ง

Ds->Laguerre_Roots(Roots);

ซึ่งแตกต่างไปจากแสดงไว้ก่อนหน้านี้เล็กน้อย เนื่องจาก Ds เป็นหน่วยวัดเชิงพหุนามแบบตัวแปร หน่วยความจำชี้เลขที่อยู่ โดยศูนย์ทั้งหมด n ค่าของ $D(s)$ ที่หาได้จะถูกนำมาเปรียบเทียบเพื่อหาศูนย์ของพหุนามที่มีส่วนจริงค่ามากที่สุด ด้วยข้อความสั่งต่อไปนี้

C = Roots[i].real > C ?

Roots[i].real : C;

เมื่อ i คือตัวแปรหน่วยความจำใช้เป็นดัชนีเพิ่มค่าจาก $i = 1, 2, 3, \dots, n-1$ ทั้งนี้จะต้องมีการกำหนดค่าให้กับตัวแปรหน่วยความจำ C ไว้ก่อนด้วย

C = Roots[0].real;

ทำให้เกณฑ์ $\sigma = c + 1/t$ ทำได้โดยการปรับปรุงค่าในตัวแปรหน่วยความจำ C ด้วยข้อความสั่ง

C+=1/t;

การคำนวณค่า $Y(s)$ ยังต้องใช้ค่า ω_i ที่จัดเก็บลงในตัวแปรหน่วยความจำ ดังนั้นเมื่อใช้ตัวแปรหน่วยความจำ w เป็นที่จัดเก็บค่าภายในตัวแปรหน่วยความจำนี้จึงควรมีค่าแปรจาก 0 ไปจนถึงอนันต์ เพราะการแปลงลาปลาซผกผันเชิงตัวเลข ต้องหาผลบวก $\sum_{i=1}^{\infty} Re[Y(\sigma + j\omega_i)] \cos(\omega_i t)$ ที่เป็นอนุกรมอนันต์ ซึ่งไม่สามารถดำเนินการได้บนคอมพิวเตอร์ เนื่องจากหน่วยความจำมีความจุที่จำกัด ดังนั้นด้วย

การเจาะจงหาผลการแปลงลาปลาซผกผันเฉพาะกับ $Y(s)$ ที่เป็นเศษส่วนแท้จึงทำให้

$$\lim_{\omega \rightarrow \infty} Y(\sigma + j\omega) = \lim_{\omega \rightarrow \infty} \frac{N(\sigma + j\omega)}{D(\sigma + j\omega)} = 0 \quad (41)$$

ด้วยเหตุนี้ วัตถุเชิงวิธีดำเนินการแปลงลาปลาซผกผันเชิงตัวเลขด้วยหลักเกณฑ์สี่เหลี่ยมคางหมู จะกำหนดจำนวนบวกลบค่า ε ขึ้นมาจัดเก็บในตัวแปรหน่วยความจำที่เป็นสมาชิกข้อมูลตามภาพที่ 4 ดังนี้

double Epsilon;

โดยจะทำการหยุดคำนวณหาผลบวกของอนุกรมอนันต์ที่ค่า $\omega_N = \omega_0 + Nh$ เมื่อ

$$Y(\sigma + j\omega_N) = \frac{N(\sigma + j\omega_N)}{D(\sigma + j\omega_N)} < \varepsilon \quad (42)$$

สิ่งนี้ทำให้ วัตถุเชิงวิธีดำเนินการแปลงลาปลาซผกผันเชิงตัวเลขด้วยหลักเกณฑ์สี่เหลี่ยมคางหมู ใช้ระเบียบวิธี $\tilde{y}_{TRP}(t)$ ต่อไปนี้

$$\tilde{y}_{TRP}(t) = \frac{2e^{\sigma t} h}{\pi} \left[\frac{Re[Y(\sigma)] + \sum_{i=1}^{N: Y(\sigma + j\omega_N) < \varepsilon} Re[Y(\sigma + j\omega_i)] \cos(\omega_i t)}{2} \right] \quad (45)$$

ดำเนินงานด้วยการใช้สมาชิกฟังก์ชัน ของวัตถุทางโปรแกรม (Programmatic object) แสดงไว้ในภาพที่ 4 คือ Void Trapezoid(float);

การหาผลการแปลงลาปลาซผกผันเชิงตัวเลข โดยฟังก์ชันสมาชิกนี้ ดำเนินโดยสร้างหน่วยวัตถุเชิงวิธีดำเนินการแปลงลาปลาซผกผันเชิงตัวเลข ด้วยหลักเกณฑ์ที่เปลี่ยนคางหมู ที่ต้องอาศัยตัวตั้งต้น แบบ

NumerInvLaplaceTrf(Poly *,
Poly *, float, float);

ประกอบด้วย 4 อาร์กิวเมนต์ (arguments) ในการสร้างหน่วยวัตถุดังกล่าวคือ วัตถุเชิงพหุนามสำหรับตัวเศษ $N(s)$ วัตถุเชิงพหุนามตัวส่วนสำหรับ $D(s)$ จำนวนบวกค่าน้อย ε และค่าคลาดเคลื่อนที่ยอมรับของการใช้หลักเกณฑ์ที่เปลี่ยนคางหมูเชิงประกอบตามลำดับ

3.6 การสร้างหน่วยวัตถุเชิงวิธีดำเนินการแปลงลาปลาซผกผันเชิงตัวเลขด้วยหลักเกณฑ์ที่เปลี่ยนคางหมู

เนื่องจากต้องอาศัยวัตถุเชิงพหุนาม จึงต้องทำการสร้างวัตถุเชิงพหุนาม ด้วยตัวตั้งต้นแบบ ดังที่ปรากฏในภาพที่ 3 ดังนี้

Poly Ns = Poly(DEG, COEFF);

เมื่อ DEG คือตัวแปรหน่วยความจำชนิดจำนวนเต็ม และ COEFF คือตัวแปรหน่วยความจำแบบตัวชี้ โดย Ns คือหน่วยวัตถุเชิงพหุนามสำหรับพหุนาม $N(s)$ ตามกำหนด ในทำนองเดียวกัน

Poly Ds = Poly(deg, coeff);

คือการสร้างหน่วยวัตถุเชิงพหุนาม สำหรับพหุนาม $D(s)$ ตามกำหนด วัตถุเชิงพหุนามทั้ง Ns จะ Ds ถูกนำไปสร้างหน่วยวัตถุเชิงวิธีดำเนินการแปลงลาปลาซผกผันเชิงตัวเลขด้วยหลักเกณฑ์ที่เปลี่ยนคางหมูดังข้อความส่งต่อไปนี้

NumerInvLaplaceTrf NILT =
NumerInvLaplaceTrf(&Ns, &Ds,
Eps, Tolerance);

เมื่อ NILT คือหน่วยวัตถุดังกล่าว ผลเฉลยเชิงตัวเลขของการแปลงลาปลาซผกผัน ณ เวลาใดเวลาหนึ่ง จะดำเนินการด้วยการใช้เพียง 2 ข้อความคำสั่ง

NILT.Trapezoid(t);
NILT.NILTanswer.Answer;

เมื่อ t คือตัวแปรหน่วยความจำสำหรับค่าเวลาที่ใช้หาผลการแปลงลาปลาซผกผันเชิงตัวเลข โดยแต่ละครั้งในการปฏิบัติข้อความสั่งนี้ ผลการคำนวณและข้อมูลเกี่ยวข้องประกอบการคำนวณ จะถูกบันทึกลงในตัวแปรหน่วยความจำข้อมูลแบบโครงสร้าง

struct NILTsolution NILTanswer;

ซึ่งเป็นสมาชิกข้อมูล ของวัตถุทางโปรแกรมเดียวกันนี้ และกำหนดตัวแบบโครงสร้างไว้แล้วให้ประกอบด้วย

```
struct NILTsolution {
    double Answer;
    float Time;
    long int Number_n;
```

```
double Step_h;
double RelatError;

};
```

การเข้าถึงผลเฉลยเชิงตัวเลข ที่ค่าเวลาในตัวแปร หน่วยความจำ t จึงต้องกระทำผ่านตัวแปรหน่วย ความจำ Answer ดังที่แสดงไว้ในข้อความข้างต้น

4. ผลการดำเนินงาน

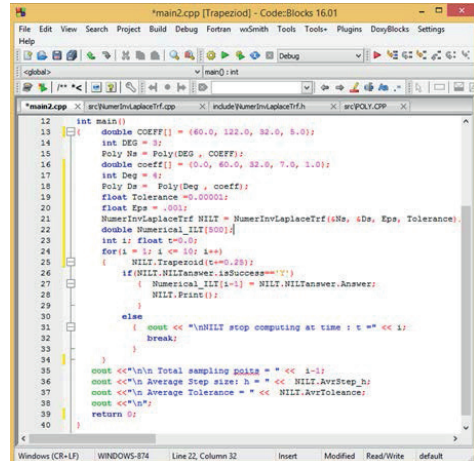
ด้วยวัตถุเชิงวิธีดำเนินการแปลงลาปลาซ ผกผันเชิงตัวเลขด้วยหลักเกณฑ์สี่เหลี่ยมคางหมูที่ สร้างขึ้น สามารถใช้ในการหาผลเฉลย

$$Y(s) = \frac{5s^3 + 32s^2 + 122s + 60}{s(s^3 + 7s^2 + 32s + 60)} \quad (46)$$

ซึ่งมีผลการแปลงลาปลาซผกผันคือ

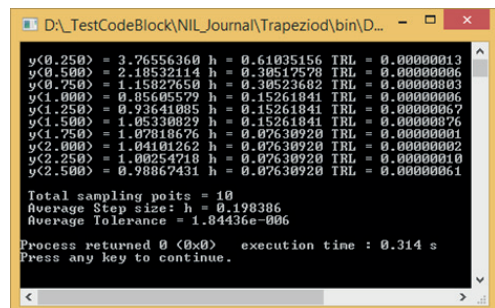
$$y(t) = \left(1 + 3e^{-3t} + e^{-2t}(\cos 4t + 2\sin 4t)\right)u(t) \quad (47)$$

โดยการใช้ภาษาการโปรแกรมซีพลัสพลัส สร้างหน่วย วัตถุของวัตถุเชิงพหุนามจำนวน 2 หน่วยคือ Ns และ Ds ที่แสดงอยู่ในภาพที่ 5 สำหรับนำไปสร้างหน่วย วัตถุเชิงวิธีดำเนินการแปลงลาปลาซผกผันเชิงตัวเลข ด้วยหลักเกณฑ์สี่เหลี่ยมคางหมูในนาม NILT



ภาพที่ 5 โปรแกรมคำนวณผลการแปลงลาปลาซ ผกผันเชิงตัวเลขด้วยหลักเกณฑ์สี่เหลี่ยมคางหมู

ซึ่งมีผลการทำงานของโปรแกรมแสดงดังต่อไปนี้



ภาพที่ 6 ผลการแปลงลาปลาซผกผันเชิงตัวเลข

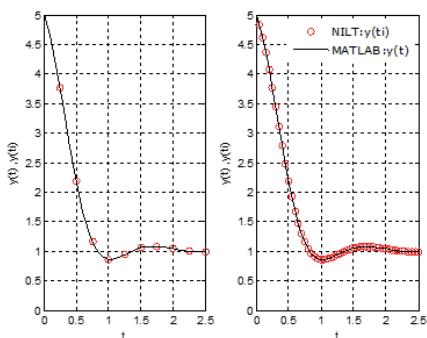
ด้วยหลักเกณฑ์สี่เหลี่ยมคางหมูของหน่วยวัตถุ NILT โปรแกรมในภาพที่ 5 ได้สร้างวัตถุเชิงพหุนาม 2 หน่วยวัตถุจาก $N(s) = 5s^3 + 32s^2 + 122s + 60$ และ $D(s) = s(s^3 + 7s^2 + 32s + 60)$ เพื่อหน่วยวัตถุ NILT จะทำการหาผลการแปลงลาปลาซผกผันเชิงตัวเลข $y(t_i)$ ทั้งหมด 10 ค่าตามกำหนดในโปรแกรม ได้แก่ $y(0.25), y(0.5), y(0.75), \dots, y(2.5)$ การคำนวณมีค่า เฉลี่ยของค่าคลาดเคลื่อนนิยมน 1.84436×10^{-6} และ ค่าเฉลี่ยช่วงกว้าง h ของหลักเกณฑ์สี่เหลี่ยมคางหมู เจริญประกอบคือ 0.198386 โดยการเปรียบเทียบผล

NILT แสดงดังตารางที่ 1

ดังภาพที่ 5

[illegible]

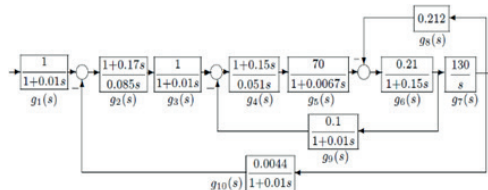
จำนวน 10 ค่าและ 50 ค่าแสดงดังภาพที่ 7



ผลเฉลี่ยแท้จริง

ผกผันเชิงตัวเลขด้วยหลักเกณฑ์สี่เหลี่ยมคางหมู ไปใช้

ไฟฟ้ากระแสตรง [12] ซึ่งมีแบบจำลองแสดงดังภาพที่ 8



เฟฟฟากระแสนตรง

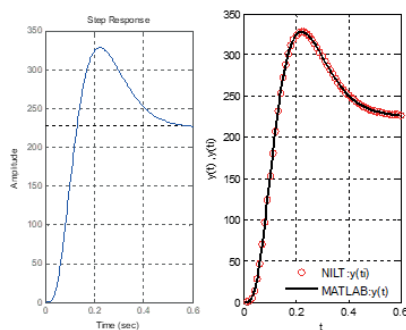
และฟังก์ชันถ่ายโอนรวมของระบบคือ

$$G(s) = \frac{1.1185 \times 10^{11} s^2 + 1.4036 \times 10^{12} s + 4.3864 \times 10^{12}}{1.0 s^7 + 355.9204 s^6 + 4.2218 \times 10^4 s^5 + \bar{D}(s)} \quad (48)$$

၁၆-

$$\begin{aligned}\bar{D}(s) &= 2.2019 \times 10^6 s^4 + 5.7376 \times 10^7 s^3 + \bar{\bar{D}}(s), \\ \bar{\bar{D}}(s) &= 8.3654 \times 10^8 s^2 + 6.1760 \times 10^9 s + 1.9300 \times 10^{10}\end{aligned}$$

MATLAB ที่แสดงดังภาพที่ 9 (ภาพซ้าย)



ภาพที่ 9 ผลตอบสนองขึ้นหนึ่งหน่วย ภาพซ้ายใช้โปรแกรม MATLAB ภาพขวาใช้หน่วยวัตถุเชิงโปรแกรม NILT

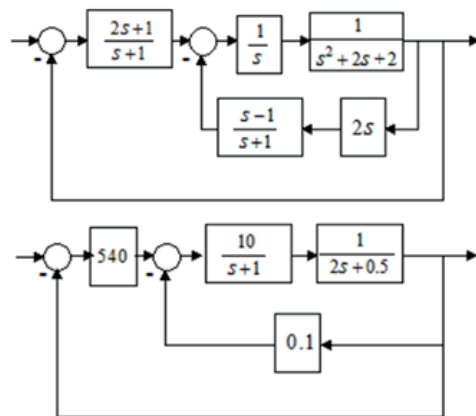
ผลตอบสนองขั้นหนึ่งหน่วยของระบบเดียวกันนี้ ยังสามารถหาได้ด้วยการใช้ ผลการแปลงลาปลาซ ผกผันเชิงตัวเลขของหน่วยวัตถุเชิงโปรแกรม NILT ซึ่งในภาพที่ 9 ทางขวา เป็นผลจากการปรับปรุงค่าของหน่วยวัตถุเชิงพหุนาม Ns และ Ds ของโปรแกรมเดิม ในภาพที่ 5 ให้เป็นไปตามฟังก์ชันถ่ายโอนรวม $G(s)$ ข้างต้นโดยใช้ค่าเวลา t จำนวน 60 ค่า ที่มีขั้นการเพิ่มค่าขึ้นละ 0.01 หน่วย

วัตถุเชิงวิธีดำเนินการแปลงลาปลาซผกผันเชิงตัวเลขด้วยหลักเกณฑ์สี่เหลี่ยมคางหมูที่สร้างขึ้นของบทความยังคงคำนวณได้ผลเร็ว และเป็นปกติเมื่อใช้จำนวนของค่าเวลา t ให้มีจำนวนมากเพิ่มขึ้นสามารถสังเกตได้จากผลตอบสนองขั้นหนึ่งหน่วยของแผนภาพกล่องระบบควบคุมเชิงเส้น 2 ระบบ ที่แสดงในภาพที่ 9 โดยแผนภาพด้านบนมีฟังก์ชันถ่ายโอนรวมคือ

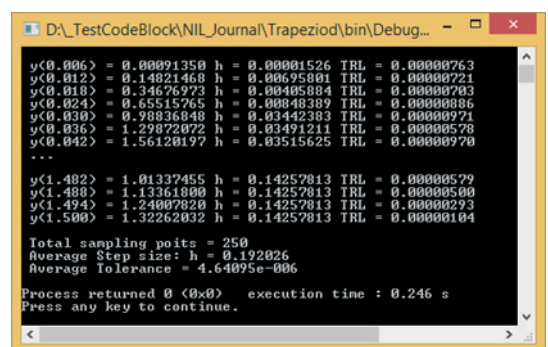
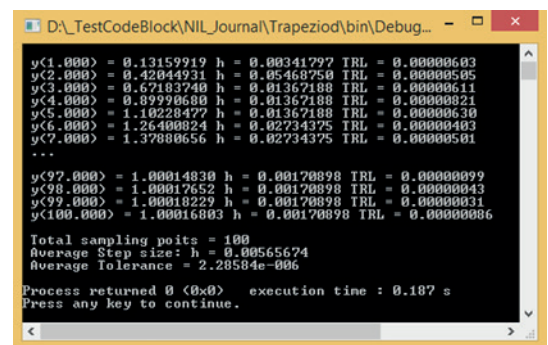
$$G_1(s) = \frac{2s^2 + 3s + 1}{s^5 + 5s^4 + 12s^3 + 14s^2 + 3s + 1} \quad (49)$$

และฟังก์ชันถ่ายโอนรวมของแผนภาพด้านล่างคือ

$$G_2(s) = \frac{5400}{2s^2 + 2.5s + 5402} \quad (50)$$

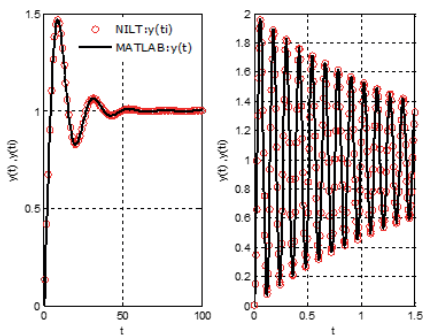


ภาพที่ 10 แผนภาพกล่องของระบบควบคุมเชิงเส้น 2 ระบบสำหรับใช้ทดสอบหาผลตอบสนองขั้นหนึ่งหน่วยด้วยหน่วยวัตถุเชิงโปรแกรม NILT



ภาพที่ 11 ผลการแปลงลาปลาซผกผันเชิงตัวเลขด้วยหลักเกณฑ์สี่เหลี่ยมคางหมู คำนวณผลตอบสนองขั้นหนึ่งหน่วยโดยหน่วยวัตถุ NILT

100 ค่า ขึ้นการเพิ่มค่าขึ้นละ 1 หน่วย ขณะที่ $G_2(s)$ ใช้ค่าเวลา t จำนวน 250 ค่า มีขึ้นการเพิ่มค่าขึ้นละ 0.006 หน่วย และใช้เวลาในการปฏิบัติโปรแกรม 0.187 และ 0.246 วินาทีตามลำดับ ทั้งนี้เป็นการพัฒนาโปรแกรมบนไอดีอี (Integrated Development Environment; IDE) ที่เป็นแบบทรัพยากรเปิดรู้จักในนาม “CODE::BLOCK” รุ่น 16.01 โดยใช้ตัวแปลโปรแกรม (compiler) “GNU GCC” บนระบบปฏิบัติการ “Windos 8.1” ตัวประมวลผล “Intel(R) Cor(TM) i7-4710HQ CPU@2.5 GHz” ผลการทำงานของหน่วยวัตถุเชิงโปรแกรม NILT ในการคำนวณผลตอบสนองขึ้นหนึ่งหน่วยของระบบควบคุมในภาพที่ 10 เมื่อเปรียบเทียบกับฟังก์ชัน STEP(SYS) ของซอฟต์แวร์สำเร็จรูป MATLAB แสดงดังภาพที่ 12



ภาพที่ 12 ผลตอบสนองขึ้นหนึ่งหน่วย ภาพซ้ายของระบบภาพขวาของระบบ

5. สรุปผลการทดลอง

วัตถุเชิงวิธีดำเนินการแปลงลาปลาซผกผันเชิงตัวเลขด้วยหลักเกณฑ์สี่เหลี่ยมคางหมูที่มีชื่อของคลาสในนาม NumerInvLaplaceTrf สำหรับการคำนวณผลการแปลงลาปลาซผกผัน โดยใช้ระเบียบวิธี $\tilde{y}_{TRP}(t)$ ที่สร้างขึ้น ถึงแม้ว่าในแต่ละช่วงกว้าง

h สูตรปริพันธ์เชิงตัวเลขด้วยหลักเกณฑ์สี่เหลี่ยมคางหมู จะมีค่าคลาดเคลื่อนอยู่ในระดับ $O(h^3)$ แต่ด้วยการกำหนดจำนวนบวกค่าน้อย ε ขึ้นมาใช้เป็นเกณฑ์ในการคำนวณภายใน $\tilde{y}_{TRP}(t)$ ทำให้สามารถแปรค่าช่วงกว้าง h ให้มีค่าน้อยลงจนกระทั่งการหาปริพันธ์เชิงตัวเลขมีค่าคลาดเคลื่อน ยินยอมตามกำหนด จึงเป็นผลให้การใช้หน่วยวัตถุเชิงวิธีดำเนินการแปลงลาปลาซผกผันเชิงตัวเลขด้วยหลักเกณฑ์สี่เหลี่ยมคางหมู ในนามหน่วยวัตถุ NILT ที่ผ่านมาประสบผลสำเร็จ มีค่าคลาดเคลื่อน และค่าคลาดเคลื่อนยินยอมในระดับต่ำเป็นที่น่าพอใจเมื่อเปรียบเทียบกับผลเฉลยจริง โดยเฉพาะเมื่อนำไปใช้หาผลตอบสนองขึ้นหนึ่งหน่วย ของระบบควบคุมแอลทีไอ และทำการเปรียบเทียบผลกับการใช้ฟังก์ชัน STEP(SYS) ของซอฟต์แวร์สำเร็จรูป MATLAB พบว่าผลตอบสนองขึ้นหนึ่งหน่วยเชิงตัวเลขของหน่วยวัตถุ NILT มีค่าตกลงบนกราฟผลตอบสนองขึ้นหนึ่งหน่วยของ MATLAB บนทุกๆ จุด ทั้งนี้ยังได้ทดลองเพิ่มจำนวนค่าเวลา t ในการหาผลการแปลงลาปลาซผกผันเชิงตัวเลขของหน่วยวัตถุ NILT พบว่าตามสภาพแวดล้อมของการพัฒนาโปรแกรมที่กำหนด ใช้เวลาปฏิบัติโปรแกรมต่ำกว่า 1 วินาที และยังเป็นการพัฒนาโปรแกรมบนไอดีอีและตัวแปลโปรแกรม ที่เป็นแบบแหล่งทรัพยากรเปิดด้วยภาษาการโปรแกรมซีพลัสพลัส

6. เอกสารอ้างอิง

- [1] Trinchero, R., Stievano, I. S. and Canavero, F. G. Simulation of buck converters via numerical inverse Laplace transform. In: Signal and Power

- [1] Integrity (SPI), 2017 IEEE 21st Workshop on. IEEE(2017), 1-4.
- [2] R-Smith, N. A. Z. and Brančík, L. Comparative study on one-dimensional numerical inverse Laplace transform methods for electrical engineering. Elektrovrevue-Electronic Journal, 18.1 (2016), 1-8.
- [3] Nuricumbo-Guillen, R., Gomez, P. and Espino-Cortes, F. P. Computation of transient voltage profiles along transmission lines by successive application of the numerical Laplace transform. In: North American Power Symposium (NAPS), 2013. IEEE(2013), 1-6.
- [4] Mikulović, J. Č. and Šekara, T. B. The numerical method of inverse Laplace transform for calculation of over voltages in power transformers and test results. Serbian Journal of Electrical Engineering, 11.2(2014), 243-256.
- [5] Dubner, H. and Abate, J. Numerical inversion of Laplace transforms by relating them to the finite Fourier cosine transform. Journal of the ACM (JACM), 15.1(1968), 115-123.
- [6] Durbin, F. Numerical inversion of Laplace transforms: an efficient improvement to Dubner and Abate's method. The Computer Journal, 17.4(1974), 371-376.
- [7] Lin, F. F. Numerical inversion of Laplace transforms by the trapezoidal-type methods. 2003. PhD Thesis.
- [8] Cohen, A. M. Numerical methods for Laplace transform inversion. Springer Science & Business Media (2007), 26-28.
- [9] Gustafson, S. Computing inverse Laplace Transforms using convergence acceleration. In: Computation and Control II. Birkhäuser Boston (1991), 151-160.
- [10] Kincaid and W.Cheney, Numerical Analysis. 2d ed., California:Brooks/Cole Publishing Company(1996), 100-103
- [11] ชัยณรงค์ วิเศษศักดิ์วิชัย, สายชล ชุดเจือจิน และคณะ. วัตถุเชิงพหุนามของการโปรแกรมซีพลัสพลัสสำหรับการประมวลผลฟังก์ชันถ่ายโอน. การประชุมวิชาการเครือข่ายวิศวกรรมไฟฟ้ามหาวิทยาลัยเทคโนโลยีราชมงคล. ครั้งที่ 7.ชลบุรี; 2558. หน้า 186-189.
- [12] D. Xue, D., Chen, Y. and Atherton, D. P. Linear Feedback Control Analysis and Design with MATLAB, Springer-Verlag (2002), 28-29.