รายการอ้างอิง

1. Combes, F., Boisse, P., Mazure, A. and Blanchard, A. (1995). *Galaxies and cosmology.* Translated by M. Seymour. Berlin: Springer-Verlag.
2. Toomre, A. and Toomre, J. 1972. Galactic bridges and tails. Astrophys. J. 178: 623-666.
3. Struck, C. Galaxy collisions [Online]. 1999. Available from: http://xxx.lanl.gov/html/ astro-ph/9908269 [2002, November 26]
4. Boulet, D.L. (1991). *Methods of orbit determination for the microcomputer.* Virginia: William-Bell, Inc.
5. Binney, J. and Tremaine, S. (1987). *Galactic dynamics.* New Jersey: Princeton University Press.
6. Frommert, H. and Kronberg, C. Messier objects [Online]. 2002. Available from: http:// www.seds.org/messier/objects.html#galaxy [2004, December 12]
7. Berman, S. Galaxies classification [Online]. 2004. Available from: http://hemes. physics. ox.ac.uk/users/Astrophysics/guides/galaxies/galtypes.shml [2004, December 14]
8. Murray, C.D. and Dermott, S.F. (1999). *Solar system dynamics.* Cambridge: Cambridge University Press.
9. Nemiroff, R. Hoag's object: A strange ring galaxy [Online]. 2002. Available from: http:// hubblesite.org/newscenter/newsdesk/archieve/releases/2002/21 [2003, November 7]
10. Turner, C. Galaxy interaction[Online]. Available from: http://gladstone.uoregon. edu/ ~cturner/ p401/collisions.html [2003, June 25]
11. Nemiroff, R. Polar ring galaxy NGC4650A[Online]. 1999. Available from: http:// antwrp.gsfc.nasa.gov/apod/ap990510.html [2004, January 6]
12. [Online]. Available from: http://www.ast.cam.ac.uk/AAO/images/general/ngc3923.html [2005, August 6]
13. Yabushita Shin. (1982). *Keisan butsuri I(Computational physics I).* Tokyo: Chijinshokan.
14. ฮิโรชิ คิโนะชิตะ. (2547). *กลศาสตร์ของวัตถุท้องฟ้าและวงโคจร.* แปลโดย พีรพัฒน์ ศิริสมบูรณ์ลาภ. กรุงเทพมหานคร: โรงพิมพ์แห่งจุฬาลงกรณ์มหาวิทยาลัย.
15. Onishi, H. 1989. *Galactic interaction seen by personal computer.* Chijinshokan henshubu (eds.), *Pasokon temmonkyoshitsu.* Tokyo: Chijinshokan.
16. Arp, H. (1966). *Atlas of peculiar galaxies.* Pasadena: California Institute of Technology.

17. Briggs, F.H.; Möller, O.; Higdon, J.L.; Trentham, N.; Ramirez-Ruiz, E. 2001. Did VV 29 collide with a dark matter halo? A&A. 380: 418-424

18. Lynds, R. and Toomre, A. 1976. On the interpretation of ring galaxies: The binary ring system II Hz 4. Astrophys J. 209: 382-388.

19. โก นางาซาวา. (2534). *กลศาสตร์ท้องฟ้าเบื้องต้น.* แปลโดย พีรพัฒน์ ศิริสมบูรณ์ลาภ. กรุงเทพมหานคร: โรงพิมพ์แห่งจุฬาลงกรณ์มหาวิทยาลัย.

20. พีรพัฒน์ ศิริสมบูรณ์ลาภ. (2533). *กลศาสตร์ท้องฟ้าทั่วไป.* กรุงเทพมหานคร: โรงพิมพ์แห่งจุฬาลงกรณ์มหาวิทยาลัย.

ภาคผนวก

ภาคผนวก ก

ปัญหาวัตถุ 2 ชิ้น

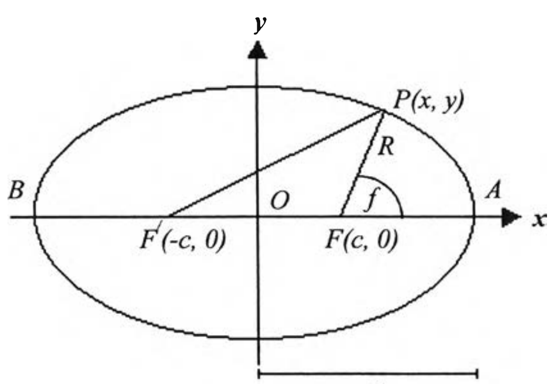
1. เรขาคณิตของวงโคจร

ในส่วนนี้จะแสดงลักษณะทางเรขาคณิตของรูปทรงภาคตัดกรวยซึ่งประกอบด้วย วงรี พาราโบลา และไฮเปอร์โบลา ซึ่งเป็นรูปแบบของวงโคจรของระบบที่อยู่ภายใต้ศักย์แบบเคปเลอร์

1.1 วงรี

นิยาม "วงรีเป็นเส้นทางของจุดที่มีผลบวกของระยะห่างจากจุดกำหนด 2 จุด เป็นค่าคงที่" สมการของวงรีในพิกัดคาร์ทีเซียนจะกำหนดจุด 2 จุด F และ F' อยู่บนแกน x ดังรูปที่ ก.1 และให้จุดกึ่งกลางระหว่าง FF' เป็นจุดกำเนิดแล้ว

$$PF = \sqrt{(x-c)^2 + y^2} \quad , PF' = \sqrt{(x+c)^2 + y^2}$$



รูปที่ ก.1 รูปวงรี

จากนิยามวงรี ผลรวมของ PF และ PF' จะเป็นค่าคงที่ดังนั้น

$$\sqrt{(x-c)^2 + y^2} + \sqrt{(x+c)^2 + y^2} = 2a \qquad (ก.1)$$

จากรูป a เท่ากับระยะ OA และ OB ย้ายเทอมที่ 2 ทางซ้ายของสมการข้างต้นไปทางขวาแล้วยกกำลังสองทั้งสองข้าง เมื่อจัดรูปแล้วได้เป็น

$$a\sqrt{(x+c)^2 + y^2} = a^2 + cx$$

ยกกำลังสองทั้งสองข้างอีกครั้งและจัดรูปใหม่ได้

$$\frac{x^2}{a^2} + \frac{y^2}{a^2 + c^2} = 1 \qquad (ก.2)$$

สมการนี้คือสมการแสดงรูปวงรีในพิกัดคาร์ทีเซียน เรียกจุด F และ F' ว่าจุดโฟกัส เรียก a ว่า ระยะครึ่งแกนเอก (semi-major axis) กำหนดให้

$$c = ae \qquad (ก.3)$$

e เป็นปริมาณแสดงการเลื่อนของจุดโฟกัสจากจุดกำเนิด เรียกปริมาณนี้ว่าค่าความรีหรือภาวะเยื้องศูนย์ (eccentricity) ถ้า $e = 0$ จุด F และ F' จะอยู่ที่จุดกำเนิดทำให้วงรีกลายเป็นวงกลม

จากรูปที่ ก.1 ความสัมพันธ์ของพิกัดคาร์ทีเซียนและพิกัดเชิงขั้วคือ

$$x = R\cos f + ae\ ,\ y = R\sin f \qquad (ก.4)$$

แทนสมการที่ ก.4 ลงใน ก.2 และจัดรูปได้

$$[\,R(1 + e\cos f) - a(1 - e^2)\,]\,[\,R(1 - e\cos f) + a(1 - e^2)\,] = 0 \qquad (ก.5)$$

เนื่องจาก $e < 1$ เทอมที่ 1 ทางซ้ายมือของสมการเท่านั้นที่ให้ R ที่มีค่าเป็นบวก จะได้ว่า

$$R = \frac{a(1-e^2)}{1 + e\cos f} \qquad (ก.6)$$

สมการที่ (ก.6) คือสมการวงรีในพิกัดเชิงขั้ว จากรูปที่ ก.1 ถ้าย้ายจุดกำเนิดมาอยู่ที่ F แล้ว จุด A เรียกว่าจุดใกล้ (pericenter) จุด B เรียกว่าจุดไกล (apocenter) เมื่อให้ระยะห่างระหว่างจุดโฟกัส F และจุดใกล้เป็น q แล้ว

$$q = a(1 - e) \qquad (ก.7)$$

แทนระยะจุดใกล้ลงในสมการที่ (ก.6) จะได้

$$R = \frac{q(1 + e)}{1 + e\cos f} \qquad (ก.8)$$

ในสมการที่ (ก.6) พารามิเตอร์เชิงเรขาคณิตของวงรีคือ a และ e ส่วนในสมการที่ (ก.8) จะเป็น q และ e

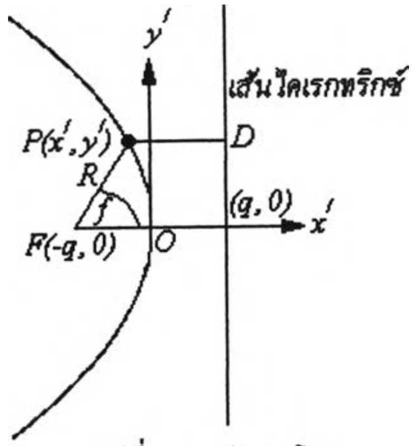
1.2 พาราโบลา

นิยาม “พาราโบลาเป็นเส้นทางของจุดบนระนาบ ที่ระยะทางจากจุดเหล่านี้ถึงจุดคงที่ (จุดโฟกัส) และระยะทางจากจุดเหล่านี้ถึงเส้นคงที่ (ไดเรคทริกซ์) มีค่าเท่ากัน”

จากรูปที่ ก.2 จุดโฟกัส F อยู่ที่ตำแหน่ง $(-q, 0)$ เส้นไดเรกทริกซ์ตัดกับแกน x' ที่ $x' = q$ จากนิยามพาราโบลา ระยะ FP และ PD มีค่าเท่ากัน ดังนั้น

$$(x' + q)^2 + y'^2 = (q - x')^2$$

$$y'^2 = -4qx' \qquad (ก.9)$$



รูปที่ ก.2 รูปพาราโบลา

สมการที่ (ก.9) เป็นสมการเส้นโค้งพาราโบลาในพิกัดคาร์ทีเชียนที่มีจุดกำเนิดอยู่ที่จุด *0* ถ้าย้ายจุดกำเนิดมาที่จุด *F* และ *(x , y)* เป็นพิกัดของจุด *P* เมื่อ *F* เป็นจุดกำเนิดแล้ว

$$x' = q - x,\ y' = y \qquad (ก.10)$$

และ

$$x = R\cos f,\ y = R\sin f \qquad (ก.11)$$

แทนสมการที่ (ก.10) และ (ก.11) ลงใน (ก.9)

$$R^2\sin^2 f = 4q(R\cos f - q)$$

$$(1 - \cos^2 f)R^2 - (4q\cos f)R + 4q^2 = 0$$

$$[R(1 - \cos f) - 2q][R(1 + \cos f) - 2q] = 0$$

$$\therefore R = \frac{2q}{1 + \cos f} \qquad (ก.12)$$

q คือระยะจุดใกล้เมื่อวัดจากจุด *F* ถึงเส้นโค้งพาราโบลา เมื่อเทียบสมการที่ (ก.12) กับ (ก.8) แล้ว จะเห็นว่าค่าความรีของเส้นโค้งพาราโบลาคือ $e = 1$

1.3 ไฮเปอร์โบลา

นิยาม "ไฮเปอร์โบลาเป็นเส้นทางของจุดที่ผลต่างของระยะห่างจากจุดคงที่ 2 จุดเป็นค่าคงที่" (รูปที่ ก.3)

จากนิยามจะได้ว่า

$$F'P - FP = 2a$$

$$\sqrt{(x-c)^2 + y^2} - \sqrt{(x+c)^2 + y^2} = 2a \qquad (ก.13)$$

ย้ายเทอมที่ 2 ทางซ้ายไปด้านขวาของสมการแล้วยกกำลังสองทั้งสองข้าง

$$-cx = a^2 + a\sqrt{(x+c)^2 + y^2}$$

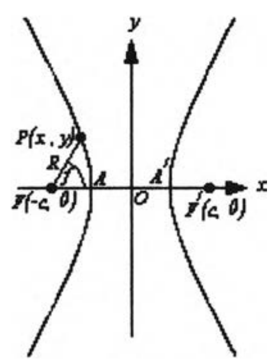
ยกกำลังสองทั้งสองข้างอีกครั้งจะได้

$$x^2(a^2 - c^2) + y^2 = a^2(a^2 - c^2)$$

จัดเทอมใหม่จะได้สมการไฮเปอร์โบลาในพิกัดคาร์ทีเซียนที่มีจุด O เป็นจุดกำเนิด

$$\left(\frac{x}{a}\right)^2 - \left(\frac{y}{b}\right)^2 = 1 \qquad (ก.14)$$

b มีค่าเท่ากับ $a\sqrt{e^2 - 1}$



รูปที่ ก.3 รูปไฮเปอร์โบลา

ถ้าให้ F เป็นจุดกำเนิด ความสัมพันธ์ระหว่างพิกัดคาร์ทีเซียนและเชิงขั้วคือ

$$x = R\cos f - ae\,,\ \ y = R\sin f \qquad (ก.15)$$

แทนสมการข้างต้นและ b ลงในสมการที่ (ก.14) ได้

$$[\,R(e\cos f - 1) - a(e^2 - 1)]\,[\,R(e\cos f + 1) - a(e^2 - 1)] = 0$$

เทอมที่สองเป็นจริงเพราะทำให้ $R > 0$ ดังนั้น

$$R = \frac{a(e^2 - 1)}{1 + e\cos f} \qquad (ก.16)$$

จากรูปที่ ก.3 a คือระยะ OA จะเห็นว่า $c > a$ แต่จากสมการที่ (ก.3) $c = ae$ เพราะฉะนั้น $e > 1$ สำหรับไฮเปอร์โบลา และระยะจุดใกล้

$$q = c - a$$

$$= a(e - 1) \quad (ก.17)$$

2. สมการการเคลื่อนที่

ให้เวกเตอร์ตำแหน่งในกรอบอ้างอิงเฉื่อย O ของจุดมวล m_1 และ m_2 คือ $\mathbf{R}_1'$ และ $\mathbf{R}_2'$ ให้ระยะห่างระหว่าง m_1 และ m_2 คือ $\mathbf{R}$ (รูปที่ ก.4) แรงที่มวล m_2 กระทำต่อ m_1 คือ

$$\mathbf{F} = \frac{Gm_1m_2\mathbf{R}}{R^3} \quad (ก.18)$$

จากกฎข้อที่ 2 ของนิวตัน จะเขียนสมการการเคลื่อนที่ของ m_1 เป็น

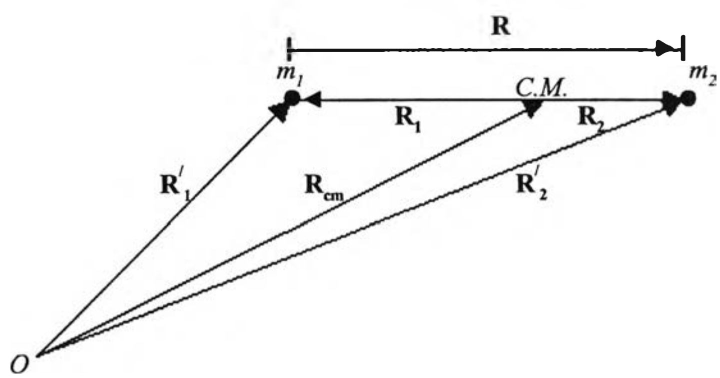
$$m_1\ddot{\mathbf{R}}_1' = \frac{Gm_1m_2\mathbf{R}}{R^3} \quad (ก.19)$$

จากกฎข้อที่ 3 ของนิวตัน สมการการเคลื่อนที่ของ m_2 คือ

$$m_2\ddot{\mathbf{R}}_2' = -\frac{Gm_1m_2\mathbf{R}}{R^3} \quad (ก.20)$$

ให้ $\mathbf{R}_{cm}$ คือเวกเตอร์ที่ชี้ไปยังจุดศูนย์กลางมวลของระบบ โดย

$$\mathbf{R}_{cm} = \frac{m_1\mathbf{R}_1' + m_2\mathbf{R}_2'}{m_1 + m_2} \quad (ก.21)$$



รูปที่ ก.4 ระบบวัตถุ 2 ชิ้น

อนุพันธ์อันดับสองของสมการที่ (ก.21) คือ

$$(m_1 + m_2)\,\ddot{\mathbf{R}}_{cm} = m_1\ddot{\mathbf{R}}'_1 + m_2\ddot{\mathbf{R}}'_2 \qquad \text{(ก.22)}$$

แทนค่าสมการที่ (ก.19) และ (ก.20) ลงในสมการที่ (ก.22)

$$(m_1 + m_2)\,\ddot{\mathbf{R}}_{cm} = 0 \qquad \text{(ก.23)}$$

สมการข้างต้นแสดงให้เห็นว่า จุดศูนย์กลางมวลของระบบเคลื่อนที่ด้วยความเร็วคงที่ถ้าไม่มีแรงภายนอกมากระทำต่อระบบ

สมการการเคลื่อนที่ของเวกเตอร์สัมพัทธ์ **R** หาได้โดยขจัด m_1 และ m_2 จากสมการที่ (ก.19) และ (ก.20) ตามลำดับแล้วผลต่างของทั้งสองสมการก็คือสมการการเคลื่อนที่ที่อ้างอิงกับจุดมวล m_1

$$\ddot{\mathbf{R}} = -\frac{\mu\mathbf{R}}{R^3} \qquad \text{(ก.24)}$$

เมื่อ $\mu = G(m_1 + m_2)$ โดยที่

$$\mathbf{R}_1 = -\frac{m_2}{(m_1 + m_2)}\mathbf{R} \qquad \text{(ก.25)}$$

$$\mathbf{R}_2 = \frac{m_1}{(m_1 + m_2)}\mathbf{R} \qquad \text{(ก.26)}$$

เป็นเวกเตอร์ตำแหน่งของ m_1 และ m_2 ในกรอบอ้างอิงของจุดศูนย์กลางมวล

3. การแก้สมการการเคลื่อนที่เพื่อหาตำแหน่งสัมพัทธ์ระหว่างวัตถุ

ผลคูณเชิงเวกเตอร์ของ **R** กับสมการที่ (ก.24) คือ

$$\mathbf{R} \times \ddot{\mathbf{R}} = 0$$

จาก

$$\frac{d}{dt}(\mathbf{R} \times \dot{\mathbf{R}}) = \mathbf{R} \times \ddot{\mathbf{R}} + \dot{\mathbf{R}} \times \dot{\mathbf{R}}$$

สังเกตว่าด้านขวาของสมการจะเท่ากับศูนย์ และ $\mathbf{R} \times \dot{\mathbf{R}}$ คือโมเมนตัมเชิงมุม ดังนั้น

$$\mathbf{R} \times \dot{\mathbf{R}} = \mathbf{h} = \text{เวกเตอร์คงที่} \qquad \text{(ก.27)}$$

h คือโมเมนตัมเชิงมุมซึ่งเป็นปริมาณที่อนุรักษ์ ผลคูณเชิงสเกลาร์ของ **R** และ **h** คือ

$$\mathbf{R} \cdot \mathbf{h} = \mathbf{R} \cdot (\mathbf{R} \times \dot{\mathbf{R}}) = \dot{\mathbf{R}} \cdot (\mathbf{R} \times \mathbf{R}) = 0$$

แสดงว่าการเคลื่อนที่ของวัตถุจะอยู่ในระนาบที่ตั้งฉากกับ **h** เรียกระนาบนี้ว่าระนาบวงโคจร ต่อไปจะหาผลคูณสเกลาร์ระหว่าง **R** ในรูปที่ ก.4 กับสมการที่ (ก.24) ได้

$$\dot{\mathbf{R}}\cdot\ddot{\mathbf{R}} = -\frac{\mu}{R^3}\ \mathbf{R}\cdot\dot{\mathbf{R}}$$

จัดเทอมใหม่แล้วอินทิเกรตจะได้

$$\frac{1}{2}|\dot{\mathbf{R}}|^2 - \frac{\mu}{R} = \text{ค่าคงที่} = E \qquad (\text{ก.}28)$$

เทอมแรกทางซ้ายมือคือพลังงานจลน์ เทอมที่สองคือศักย์ของแรงโน้มถ่วง กำหนดให้ $U = -\mu/R$ E คือพลังงานรวมของระบบซึ่งมีค่าคงที่ ดังนั้นสมการที่ (ก.28) แสดงถึงการอนุรักษ์พลังงานของระบบหากไม่มีแรงภายนอกมากระทำ

จะเห็นว่าการเคลื่อนที่สัมพัทธ์นี้จะเสมือนกับให้มวล $(m_1 + m_2)$ อยู่ที่จุดศูนย์กลางมวลและมีจุดอนุภาคสมมติเคลื่อนที่รอบจุดศูนย์กลางมวลนี้ด้วยระยะห่าง **R**

ถ้ากำหนดให้ระนาบวงโคจรคือระนาบ $x-y$ จะแสดงโมเมนตัมเชิงมุมให้อยู่ในพิกัดคาร์ทีเซียนได้เป็น

$$\mathbf{h} = (x\dot{y} - y\dot{x})\,\mathbf{k} \qquad (\text{ก.}29)$$

k คือเวกเตอร์หน่วยตั้งฉากกับระนาบวงโคจร และสมการที่ (ก.28) เขียนให้อยู่ในพิกัดคาร์ทีเซียนได้เป็น

$$\frac{1}{2}(\dot{x}^2 + \dot{y}^2) + U = E \qquad (\text{ก.}30)$$

จากรูปที่ ก.1 ให้จุด F คือจุดศูนย์กลางมวล ดังนั้นจะได้ความสัมพันธ์ระหว่างพิกัดคาร์ทีเซียนและพิกัดเชิงขั้ว (r, f) เป็น

$$x = R\cos f,\ \ y = R\sin f \qquad (\text{ก.}31)$$

$$\dot{x} = \dot{R}\cos f - R\dot{f}\sin f,\ \ \dot{y} = \dot{R}\sin f + R\dot{f}\cos f \qquad (\text{ก.}32)$$

แทนค่าสมการที่ (ก.31) และ (ก.32) ลงในสมการที่ (ก.29) และ (ก.30) จะได้

$$h = R^2\dot{f} \qquad (\text{ก.}33)$$

$$\frac{1}{2}(\dot{R}^2 + R^2\dot{f}^2) - \frac{\mu}{R} = E \qquad (\text{ก.}34)$$

จากสมการที่ (ก.33) สามารถแสดงความสัมพันธ์ระหว่างอนุพันธ์ของเวลาและอนุพันธ์ของมุมกวาดจริงได้เป็น

$$\frac{d}{dt} = \frac{h}{R^2}\frac{d}{df} \qquad (ก.35)$$

นิยามตัวแปรส่วนกลับของรัศมี $s = 1/R$ แล้วอนุพันธ์เวลาของ R คือ

$$\frac{dR}{dt} = hs^2 \frac{d}{df}\left(\frac{1}{s}\right) = -h\frac{ds}{df} \qquad (ก.36)$$

แทนค่า h และ $\dot{R}$ จากสมการที่ (ก.33) และ (ก.36) ลงใน (ก.34) และเปลี่ยนตัวแปรจาก R เป็น s จะได้

$$\frac{1}{2}h^2\left(\frac{ds}{df}\right)^2 + \frac{1}{2}h^2s^2 - \mu s = E \qquad (ก.37)$$

จัดรูปสมการที่ (ก.37) ใหม่จะได้ความสัมพันธ์ของการเปลี่ยนแปลงของมุมกวาดจริงและการเปลี่ยนแปลงของ s

$$df = \frac{ds}{\sqrt{2E/h^2 + 2\mu s/h^2 - s^2}} \qquad (ก.38)$$

อินทิเกรตได้

$$f = \omega - \cos^{-1}\left(\frac{sh^2/\mu - 1}{\sqrt{1 + 2Eh^2/\mu^2}}\right) \qquad (ก.39)$$

ω เป็นค่าคงที่การอินทิเกรต แทนค่า $s = 1/R$ ลงในสมการที่ (ก.39) แล้วจัดรูปใหม่จะได้สมการแสดงตำแหน่งสัมพัทธ์ R เป็น

$$R = -\frac{h^2/\mu}{1 + \sqrt{1 + (2Eh^2/\mu^2)}\cos(f - \omega)} \qquad (ก.40)$$

ที่ $f = \omega$ R มีค่าน้อยที่สุดซึ่งก็คือระยะจุดใกล้ q นั่นเอง เมื่อเปรียบเทียบสมการที่ (ก.40) กับสมการที่ (ก.6) ซึ่งเป็นสมการแสดงความสัมพันธ์ของ R กับพารามิเตอร์เชิงเรขาคณิตของวงโคจรจะได้ว่า

$$e = \sqrt{1 + \frac{2Eh^2}{\mu^2}} \qquad (ก.41)$$

จะเห็นว่าค่า e จะขึ้นกับ E จากการคำนวณในหัวข้อเรขาคณิตของวงโคจรเราทราบว่าค่า e จะสัมพันธ์กับรูปร่างของวงโคจร ดังนั้นจากสมการที่ (ก.40) และ (ก.41) ทำให้เราทราบความสัมพันธ์ระหว่างรูปร่างของวงโคจรกับพลังงานรวมและโมเมนตัมเชิงมุมของระบบดังตารางที่ ก.1

ตารางที่ ก.1 ความสัมพันธ์ของรูปร่างวงโคจรกับ E และ h

	วงรี	พาราโบลา	ไฮเปอร์โบลา
E	< 0	$= 0$	> 0
h	$\sqrt{\mu a(1-e^2)}$	$\sqrt{2\mu q}$	$\sqrt{\mu a(e^2-1)}$

4. สมการการเปลี่ยนแปลงตามเวลาของมุมกวาดจริง

การคำนวณในหัวข้อที่ผ่านมาทำให้เราทราบรูปร่างของวงโคจร ขั้นตอนต่อไปก็จะเป็นการหาตำแหน่งของวัตถุบนวงโคจร ณ เวลา t ใดๆ สิ่งแรกที่จะต้องทราบก็คือความสัมพันธ์ของมุมกวาดจริงกับเวลา จากสมการที่ (ก.33) เขียนใหม่ได้เป็น

$$\dot{f} = h/R^2$$

เมื่อทราบรูปร่างของวงโคจรแล้วก็เพียงแต่แทนค่า h จากตารางที่ ก.1 และ R จากสมการที่ (ก.8) (ก.12) หรือ (ก.16) ที่สัมพันธ์กับรูปร่างของวงโคจรลงในสมการข้างต้น ก็จะได้สมการแสดงการเปลี่ยนแปลงตามเวลาของมุมกวาดจริง

สมการอนุพันธ์ของมุมกวาดจริงสำหรับวงโคจรแต่ละแบบมีรูปสมการเป็นดังนี้

วงรี

$$\dot{f} = \left[\frac{m_1 + m_2}{q^3(e+1)^3}\right]^{1/2} (1+ e\cos f)^2 \qquad \text{(ก.42)}$$

พาราโบลา

$$\dot{f} = \left[\frac{m_1 + m_2}{8q^3}\right]^{1/2} (1+ \cos f)^2 \qquad \text{(ก.43)}$$

ไฮเปอร์โบลา

$$\dot{f} = \left[\frac{m_1 + m_2}{q^3(e-1)^3}\right]^{1/2} (1+ e\cos f)^2 \qquad \text{(ก.44)}$$

เมื่ออินทิเกรตสมการเหล่านี้ก็จะได้มุมกวาดจริง ณ เวลา t ใดๆ

5. ความเร็วของวงโคจร

ความเร็วในแนวเส้นสัมผัสวงโคจรคือ

$$v^2 = \dot{x}^2 + \dot{y}^2 = \dot{R}^2 + R^2\dot{f}^2 = 2(E + \frac{\mu}{R})$$

แทนค่า E จากสมการที่ (ก.41) ได้

$$v^2 = \frac{(e^2 - 1)\mu^2}{h^2} + \frac{2\mu}{R} \qquad (ก.45)$$

เมื่อทราบรูปแบบของวงโคจรก็แทนค่า h จากตารางที่ ก.1 ลงในสมการที่ (ก.45) สำหรับวงโคจรแต่ละรูปแบบ

สำหรับวงรี

$$v = \sqrt{\mu(\frac{2}{R} - \frac{1}{a})} \qquad (ก.46)$$

สำหรับวงโคจรวงกลมนั้น R จะเท่ากับ a เสมอ ดังนั้น

$$v = \sqrt{\mu/a} \qquad (ก.47)$$

วงโคจรพาราโบลา เนื่องจาก $e = 1$ ดังนั้นจากสมการที่ (ก.45) จะได้

$$v = \sqrt{2\mu/R} \qquad (ก.48)$$

วงโคจรไฮเปอร์โบลา

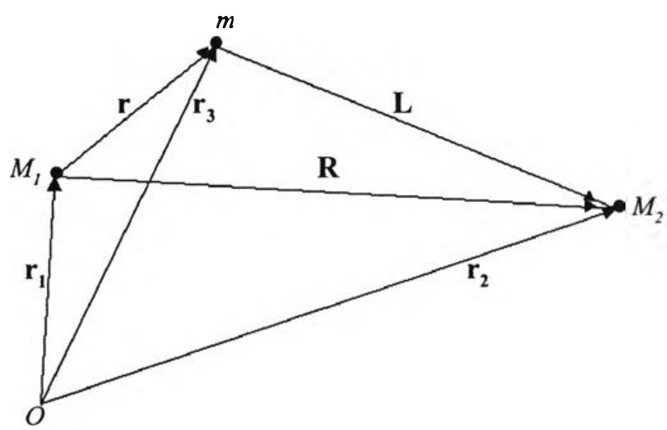
$$v = \sqrt{\mu(\frac{2}{R} + \frac{1}{a})} \qquad (ก.49)$$

ในกรณีของวงโคจรเปิดคือพาราโบลา และไฮเปอร์โบลา จะมีความแตกต่างกันตรงที่ ถ้า $R \rightarrow \infty$ กรณีวงโคจรพาราโบลาความเร็วจะเป็นศูนย์ แต่ถ้าเป็นไฮเปอร์โบลาความเร็วจะเท่ากับ $v = \sqrt{\mu/a}$

ภาคผนวก ข

สมการการเคลื่อนที่ของปัญหาวัตถุ 3 ชิ้น

พิจารณาระบบของวัตถุที่เป็นจุดมวล 3 ชิ้นคือ M_1 M_2 และ m ดังรูปที่ ข.1 $\mathbf{r}_1$ $\mathbf{r}_2$ และ $\mathbf{r}_3$ เป็นเวกเตอร์ตำแหน่งของ M_1 M_2 และ m ในกรอบอ้างอิงเฉื่อย O ตามลำดับ



รูปที่ ข.1 ระบบวัตถุ 3 ชิ้น

R และ **L** เป็นเวกเตอร์ที่ชี้จาก M_1 และ m ไปยัง M_2 วัตถุแต่ละชิ้นจะเคลื่อนที่ภายใต้สนามแรงโน้มถ่วงของวัตถุอีกสองชิ้น โดยวัตถุ M_1 M_2 และ m มีสมการการเคลื่อนที่เป็น

$$\ddot{\mathbf{r}}_1 = \frac{Gm\mathbf{r}}{r^3} + \frac{GM_2\mathbf{R}}{R^3} \qquad \text{(ข.1)}$$

$$\ddot{\mathbf{r}}_2 = -\frac{Gm\mathbf{L}}{L^3} - \frac{GM_1\mathbf{R}}{R^3} \qquad \text{(ข.2)}$$

$$\ddot{\mathbf{r}}_3 = -\frac{GM_1\mathbf{r}}{r^3} + \frac{GM_2\mathbf{L}}{L^3} \qquad \text{(ข.3)}$$

พิจารณาในแง่ของการรบกวน ถ้าแต่เดิม M_1 และ m เคลื่อนที่สัมพัทธ์กันอยู่แบบวัตถุ 2 ชิ้น หลังจากนั้นมีวัตถุ M_2 เคลื่อนที่เข้ามาใกล้แล้วทำให้เกิดการรบกวนต่อการเคลื่อนที่ของระบบวัตถุ M_1 และ m ถ้าจะพิจารณาการเคลื่อนที่ของ m สัมพัทธ์กับจุด M_1 หลังจากถูกรบกวน จากสมการที่ (ข.1) และ (ข.3) สมการการเคลื่อนที่ของ m จะเป็น

$$\ddot{\mathbf{r}} = -\frac{G(M_1 + m)\,\mathbf{r}}{r^3} - GM_2\left(\frac{\mathbf{R}}{R^3} - \frac{\mathbf{L}}{L^3}\right) \qquad \text{(ข.4)}$$

ภาคผนวก ค

ระเบียบวิธีรุงเกคุตตา อันดับ 5

ให้สมการอนุพันธ์อันดับหนึ่งของตำแหน่ง $\mathbf{x}$ เทียบกับเวลา t เป็น

$$\mathbf{x} = \mathrm{f}(\,t,\ \mathbf{x}\,) \tag{ค.1}$$

ถ้าการอินทิเกรตสมการที่ (4.1) เริ่มต้นเมื่อ $\mathbf{x} = \mathbf{x}_0$ เป็นตำแหน่งเริ่มต้น และ $t = t_0$ เป็นเวลาเริ่มต้น แล้ว ที่เวลา $t = t_0 + h$ จะได้ $\mathbf{x} = \mathbf{x}_0 + \delta\mathbf{x}$ เมื่อ h คือช่วงเวลา(step-size)ของการอินทิเกรต และ $\delta\mathbf{x}$ คือการเปลี่ยนแปลงของ $\mathbf{x}$ การประมาณด้วยระเบียบวิธีรุงเกคุตตาอันดับ 5 $\delta\mathbf{x}$ จะมีค่าเป็น

$$\delta\mathbf{x} = \frac{1}{90}\left(\,7\mathbf{F}_1 + 32\mathbf{F}_3 + 12\mathbf{F}_4 + 32\mathbf{F}_5 + 7\mathbf{F}_6\right) \tag{ค.2}$$

โดย

$$\left.\begin{aligned}
\mathbf{F}_1 &= h\mathrm{f}(\,t_0,\ \mathbf{x}_0) \\
\mathbf{F}_2 &= h\mathrm{f}(\,t_0 + \tfrac{1}{4}h,\ \mathbf{x}_0 + \tfrac{1}{4}\mathbf{F}_1) \\
\mathbf{F}_3 &= h\mathrm{f}(\,t_0 + \tfrac{1}{4}h,\ \mathbf{x}_0 + \tfrac{1}{8}\mathbf{F}_1 + \tfrac{1}{8}\mathbf{F}_2) \\
\mathbf{F}_4 &= h\mathrm{f}(\,t_0 + \tfrac{1}{2}h,\ \mathbf{x}_0 + \tfrac{1}{2}\mathbf{F}_2 + \mathbf{F}_3) \\
\mathbf{F}_5 &= h\mathrm{f}(\,t_0 + \tfrac{3}{4}h,\ \mathbf{x}_0 + \tfrac{3}{16}\mathbf{F}_1 + \tfrac{9}{16}\mathbf{F}_4) \\
\mathbf{F}_6 &= h\mathrm{f}(\,t_0 + h,\ \mathbf{x}_0 - \tfrac{3}{7}\mathbf{F}_1 + \tfrac{2}{7}\mathbf{F}_2 + \tfrac{12}{7}\mathbf{F}_3 - \tfrac{12}{7}\mathbf{F}_4 + \tfrac{8}{7}\mathbf{F}_5)
\end{aligned}\right\} \tag{ค.3}$$

ในกรณีสมการอนุพันธ์อันดับ 2 ซึ่งมีเงื่อนไขเริ่มต้นเป็น $\mathbf{x}_0$ และ $\mathbf{v}_0$ ที่ t_0 ถ้าให้ f($\mathbf{v}$) เป็นอนุพันธ์อันดับหนึ่งของ $\mathbf{x}$ และ g($\mathbf{x}$) เป็นอนุพันธ์อันดับสองของ $\mathbf{x}$ แล้ว สมการอนุพันธ์อันดับสองสามารถเขียนให้อยู่ในรูปของสมการอนุพันธ์อันดับหนึ่ง 2 สมการเป็น

$$\dot{\mathbf{x}} = f(\mathbf{v})$$

$$\dot{\mathbf{v}} = g(\mathbf{x})$$

จากนั้นก็ทำการอินทิเกรตเหมือนสมการที่ (ค.1) โดยใช้สมการที่ (ค.3) ที่เข้ากับแต่ละสมการอนุพันธ์ ได้

$$\left.\begin{aligned}
\mathbf{F}_1 &= hf(\mathbf{v}_0) \\
\mathbf{G}_1 &= hg(\mathbf{x}_0) \\
\mathbf{F}_2 &= hf(\mathbf{v}_0 + \tfrac{1}{4}\mathbf{G}_1) \\
\mathbf{G}_2 &= hg(\mathbf{x}_0 + \tfrac{1}{4}\mathbf{F}_1) \\
\mathbf{F}_3 &= hf(\mathbf{v}_0 + \tfrac{1}{8}\mathbf{G}_1 + \tfrac{1}{8}\mathbf{G}_2) \\
\mathbf{G}_3 &= hg(\mathbf{x}_0 + \tfrac{1}{8}\mathbf{F}_1 + \tfrac{1}{8}\mathbf{F}_2) \\
\mathbf{F}_4 &= hf(\mathbf{v}_0 - \tfrac{1}{2}\mathbf{G}_2 + \mathbf{G}_3) \\
\mathbf{G}_4 &= hg(\mathbf{x}_0 - \tfrac{1}{2}\mathbf{F}_2 + \mathbf{F}_3) \\
\mathbf{F}_5 &= hf(\mathbf{v}_0 + \tfrac{3}{16}\mathbf{G}_1 + \tfrac{9}{16}\mathbf{G}_4) \\
\mathbf{G}_5 &= hg(\mathbf{x}_0 + \tfrac{3}{16}\mathbf{F}_1 + \tfrac{9}{16}\mathbf{F}_4) \\
\mathbf{F}_6 &= hf(\mathbf{v}_0 - \tfrac{3}{7}\mathbf{G}_1 + \tfrac{2}{7}\mathbf{G}_2 + \tfrac{12}{7}\mathbf{G}_3 - \tfrac{12}{7}\mathbf{G}_4 + \tfrac{8}{7}\mathbf{G}_5) \\
\mathbf{G}_6 &= hg(\mathbf{x}_0 - \tfrac{3}{7}\mathbf{F}_1 + \tfrac{2}{7}\mathbf{F}_2 + \tfrac{12}{7}\mathbf{F}_3 - \tfrac{12}{7}\mathbf{F}_4 + \tfrac{8}{7}\mathbf{F}_5)
\end{aligned}\right\} \quad (ค.4)$$

จากสมการที่ (ค.2) สามารถเขียนสมการการเพิ่มขึ้นของ x และ v เป็น

$$\left.\begin{aligned} \delta \mathbf{x} &= \tfrac{1}{90}(7\mathbf{F}_1 + 32\mathbf{F}_3 + 12\mathbf{F}_4 + 32\mathbf{F}_5 + 7\mathbf{F}_6) \\ \delta \mathbf{v} &= \tfrac{1}{90}(7\mathbf{G}_1 + 32\mathbf{G}_3 + 12\mathbf{G}_4 + 32\mathbf{G}_5 + 7\mathbf{G}_6) \end{aligned}\right\} \qquad (ค.5)$$

ดังนั้นตำแหน่งและความเร็วที่เวลา $t = t_0 + h$ คือ

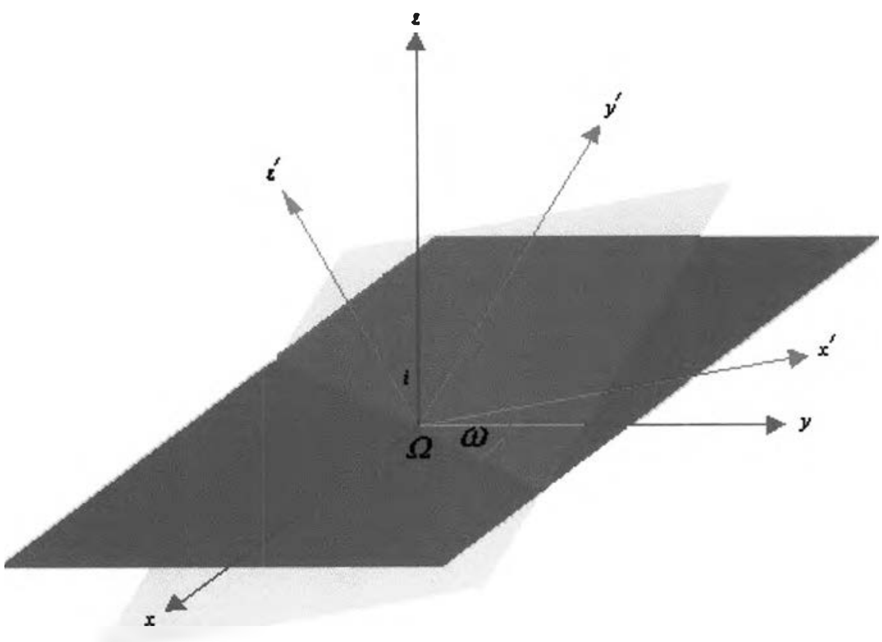
$$\left.\begin{aligned} \mathbf{x} &= \mathbf{x}_0 + \delta \mathbf{x} \\ \mathbf{v} &= \mathbf{v}_0 + \delta \mathbf{v} \end{aligned}\right\} \qquad (ค.6)$$

สมการที่ (ค.6) ก็จะกลายเป็นค่าเริ่มต้นสำหรับการอินทิเกรตของช่วงเวลาถัดไป

ภาคผนวก ง

การแปลงพิกัด

จากรูปที่ ง.1 มีระบบพิกัด (x, y, z) และ (x', y', z') ทั้ง 2 พิกัดมีจุดศูนย์กลางร่วมกัน แต่มีทิศทางการวางตัวที่แตกต่างกัน สามารถอธิบายทิศทางสัมพัทธ์ของพิกัดทั้ง 2 ด้วยมุม 3 มุมคือ i เป็นมุมที่บอกถึงความเอียงของระนาบ x'-y' กับระนาบ x - y Ω คือมุมที่วัดจากแกน x ถึงเส้นตัดของระนาบทั้งสอง ω คือมุมที่วัดจากเส้นตัดของระนาบจนถึงแกน x' การพิจารณาการเคลื่อนที่ของดาวโดยเฉพาะในระบบสุริยะซึ่งมีดวงอาทิตย์อยู่ที่ศูนย์กลางของระบบพิกัด หากมีดาวเคราะห์โคจรอยู่ในระนาบ x' - y' และหมุนทวนเข็มนาฬิการอบแกน z' ตำแหน่งของดาวเคราะห์เมื่อมันเคลื่อนมาอยู่ที่เส้นตัดของระนาบจะเรียกว่าจุดไต่ขึ้น Ω จะเรียกว่าระยะแวงของจุดไต่ขึ้น แกน x' จะกำหนดให้ชี้จุดใกล้ของวงโคจร ดังนั้น ω ก็คือระยะมุมของจุดใกล้



รูปที่ ง.1 แสดงทิศทางสัมพัทธ์ของระบบพิกัด (x, y, z) และ (x', y', z')

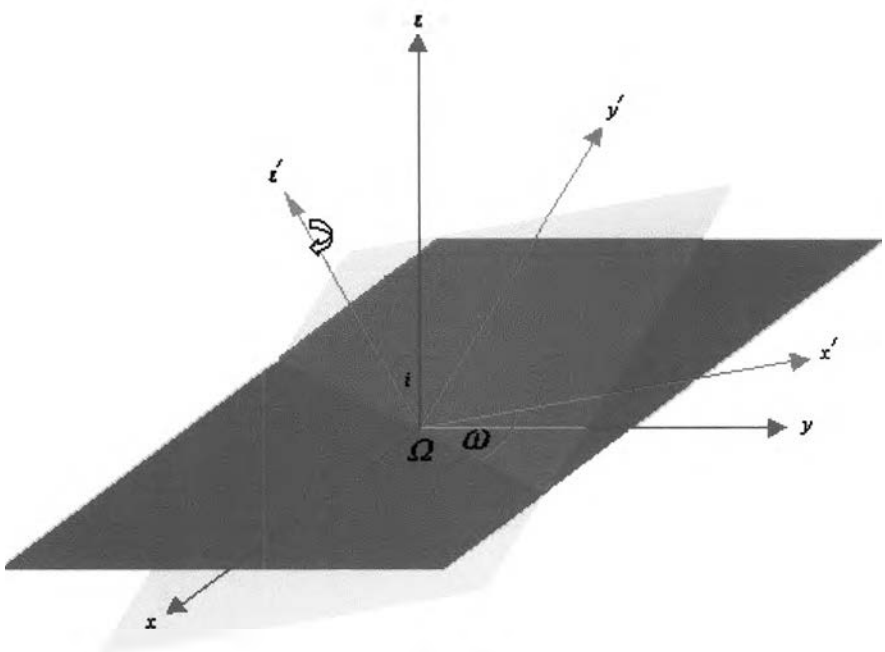
หากต้องการแปลงตำแหน่งของจุดในพิกัด (x', y', z') ไปสู่พิกัด (x, y, z) สามารถทำได้ตามขั้นตอนดังนี้

1. หมุนระนาบ $x' - y'$ ตามเข็มนาฬิการอบแกน z' เป็นมุม ω (รูปที่ ง.2) จะได้พิกัดใหม่ของจุดเป็น

$$x_1 = x' \cos \omega - y' \sin \omega$$

$$y_1 = x' \sin \omega + y' \cos \omega$$

$$z_1 = z'$$



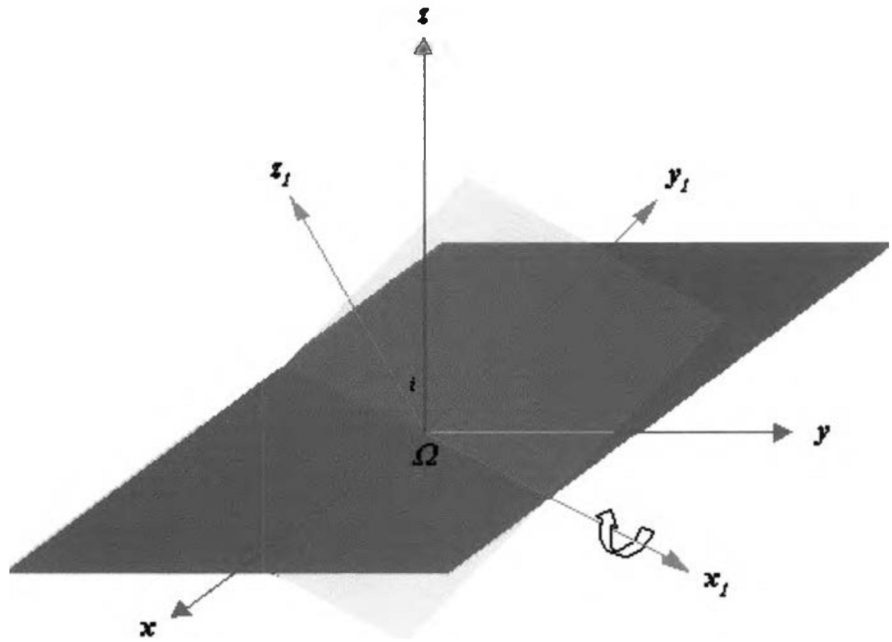
รูปที่ ง.2 การหมุนระนาบ $x' - y'$ ตามเข็มนาฬิการอบแกน z' เป็นมุม ω

2. หมุนระนาบ $z_1 - y_1$ ตามเข็มนาฬิการอบแกน x_1 เป็นมุม i (รูปที่ ง.3) จะได้พิกัดใหม่ของจุดเป็น

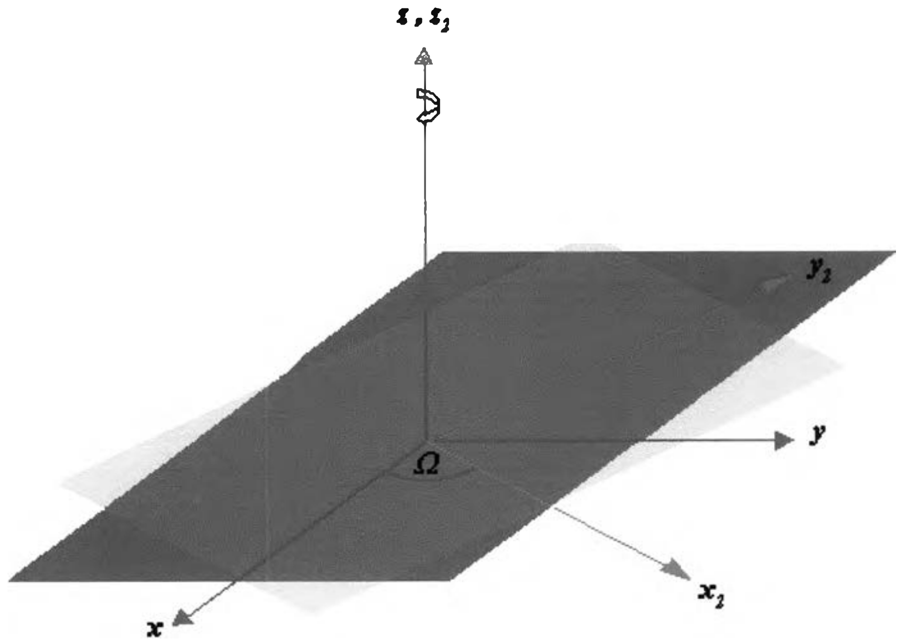
$$x_2 = x_1 = x' \cos \omega - y' \sin \omega$$

$$y_2 = y_1 \cos i - z_1 \sin i = x' \sin \omega \cos i + y' \cos \omega \cos i - z' \sin i$$

$$z_2 = y_1 \sin i + z_1 \cos i = x' \sin \omega \sin i + y' \cos \omega \sin i - z' \cos i$$



รูปที่ ง.3 หมุนระนาบ z_1 - y_1 ตามเข็มนาฬิการอบแกน x_1 เป็นมุม i



รูปที่ ง.4 การหมุนระนาบ x_2 - y_2 ตามเข็มนาฬิการอบแกน z_2 เป็นมุม Ω

3. หมุนระนาบ $x_2 - y_2$ ตามเข็มนาฬิการอบแกน z เป็นมุม Ω(รูปที่ ง.4) จะได้พิกัดของจุดใหม่ในระบบพิกัด (x, y, z) เป็น

$$x = x_2\cos\Omega - y_2\sin\Omega$$

$$y = x_2\sin\Omega + y_2\cos\Omega$$

$$z = z_2$$

แทนค่า x_2, y_2 และ z_2 ลงในสมการข้างต้นจะได้

$$\left.\begin{aligned}
x &= x'(\cos\omega\cos\Omega - \sin\omega\sin\Omega\cos i) \\
&\quad + y'(-\sin\omega\cos\Omega - \cos\omega\sin\Omega\cos i) \\
&\quad + z'\sin\Omega\sin i \\
y &= x'(\cos\omega\sin\Omega + \sin\omega\cos\Omega\cos i) \\
&\quad + y'(-\sin\omega\sin\Omega + \cos\omega\cos\Omega\cos i) \\
&\quad - z'\cos\Omega\sin i \\
z &= x'\sin\omega\sin i + y'\cos\omega\sin i + z'\cos i
\end{aligned}\right\} \qquad (\text{ง}.1)$$

ในกรณีที่ $\omega = 0$ หรือ แกน x' ชี้ไปยังจุดไต่ขึ้นแล้ว สมการที่ (ง.1) จะกลายเป็น

$$\left.\begin{aligned}
x &= x'\cos\Omega - y'\sin\Omega\cos i + z'\sin\Omega\sin i \\
y &= x'\sin\Omega + y'\cos\Omega\cos i - z'\cos\Omega\sin i \\
z &= y'\sin i + z'\cos i
\end{aligned}\right\} \qquad (\text{ง}.2)$$

ภาคผนวก จ

การใช้โปรแกรมจำลองภาพอันตรกิริยาระหว่างดาราจักร

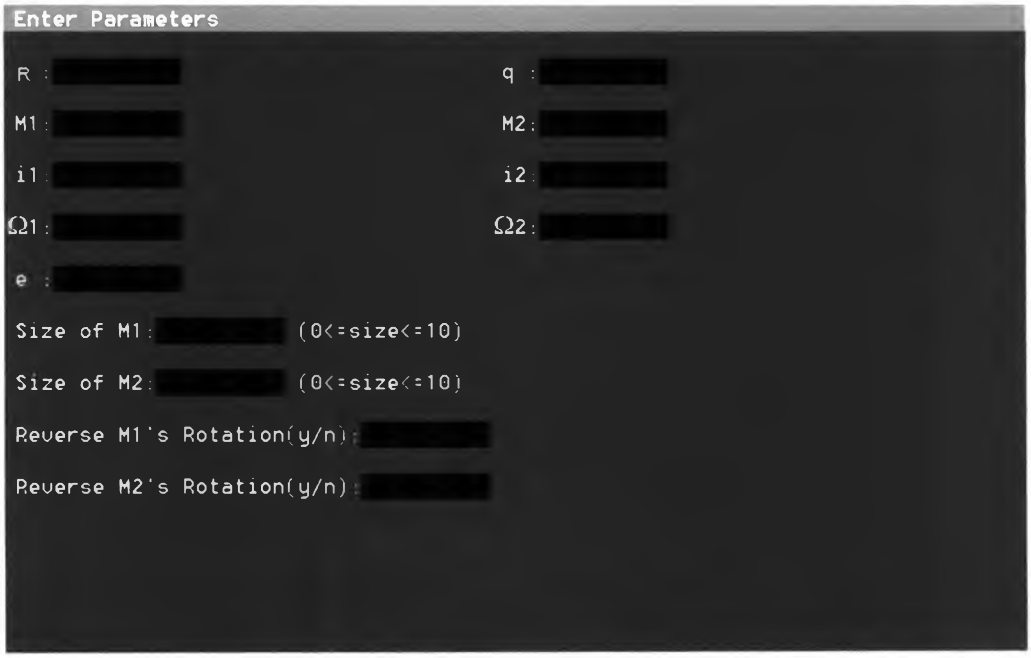
การเข้าโปรแกรม

โปรแกรมจำลองภาพอันตรกิริยาระหว่างดาราจักรนี้(โปรแกรม Galactic Interaction Version 1.26 บรรจุอยู่ในแผ่นซีดีที่แนบไว้ท้ายเล่ม ส่วนรายละเอียดของโปรแกรม(source code) อยู่ในภาคผนวก ช)จะทำงานบนระบบปฏิบัติการดอส การเข้าสู่โปรแกรมจะต้องไปที่บรรทัดคำสั่งของระบบปฏิบัติการดอส กำหนดไดเรคทอรีของโปรแกรมให้ถูกต้อง จากนั้นจึงพิมพ์คำสั่ง gal_in26.exe ก็จะเข้าสู่โปรแกรมซึ่งมีลักษณะดังรูปที่ จ.1

รูปที่ จ.1 หน้าแรกของโปรแกรม

การกำหนดค่าพารามิเตอร์

เมื่อเข้าสู่โปรแกรมแล้วดังรูปที่ จ.1 กดแป้น ← หรือ → เพื่อเลือก “CONTINUE” สำหรับการเข้าสู่หน้าต่อไปเพื่อกำหนดค่าพารามิเตอร์ดังรูปที่ จ.2 หรือเลือก “EXIT” เพื่อออกจากโปรแกรม จากนั้นกดแป้น Enter



รูปที่ จ.2 หลังจากเลือก “CONTINUE” โปรแกรมจะเข้าสู่หน้าจอสำหรับกำหนดค่าพารามิเตอร์

ค่าพารามิเตอร์ที่จะต้องกำหนดจะประกอบด้วย

R: ค่าระยะห่างเริ่มต้นของดาราจักร

q: ค่าระยะจุดใกล้ของวงโคจร

M_1: มวลของดาราจักร M_1

M_2: มวลของดาราจักร M_2

i_1: มุมเอียงของระนาบดาราจักร M_1

i_2: มุมเอียงของระนาบดาราจักร M_2

Ω_1: มุมเส้นตัดระนาบของดาราจักร M_1

Ω_2: มุมเส้นตัดระนาบของดาราจักร M_2

e: ค่าความรีของวงโคจร

Size of M_1: จำนวนวงโคจรของดาวในดาราจักร M_1 ให้ป้อนค่าเป็นจำนวนเต็มตั้งแต่ 0 ถึง 10

Size of M_2: จำนวนวงโคจรของดาวในดาราจักร M_2 ให้ป้อนค่าเป็นจำนวนเต็มตั้งแต่ 0 ถึง 10

Reverse M_1' Rotation: กลับทิศทางการหมุนของดาราจักร M_1 ป้อนค่าเป็น y หรือ n

Reverse M_2' Rotation: กลับทิศทางการหมุนของดาราจักร M_2 ป้อนค่าเป็น y หรือ n

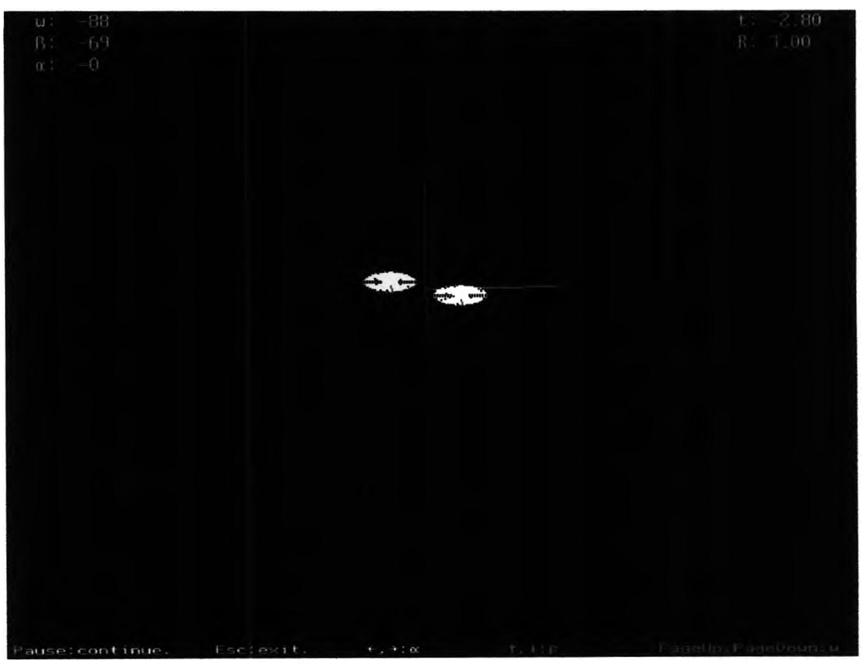
หมายเหตุ : การป้อนค่าพารามิเตอร์ที่เป็นตัวเลขจะต้องเป็นตัวเลขที่มีจุดทศนิยมด้วยเสมอ

: การเข้าสู่หน้าการแสดงผล เคอร์เซอร์จะต้องอยู่ที่ช่อง Reverse M_2' Rotation และกดแป้น Enter

: การย้ายเคอร์เซอร์ระหว่างช่องเติมค่าพารามิเตอร์ให้กดแป้น ← และ Enter หรือ →

: การลบอักขระที่อยู่ในช่องเติมค่าพารามิเตอร์ให้กดแป้น Backspace หรือ Delete

หลังจากกำหนดค่าพารามิเตอร์และเข้าสู่หน้าจอแสดงผลเรียบร้อยแล้ว โปรแกรมก็จะเริ่มทำการคำนวณและแสดงผลบนหน้าจอดังรูปที่ จ.3



รูปที่ จ.3 หน้าจอแสดงผล

บนหน้าจอแสดงผลนี้นอกจากจะแสดงภาพการจำลองแล้วยังมีส่วนที่แสดงข้อมูลที่สำคัญดังนี้

ที่มุมซ้ายด้านบน w : ค่ามุมระยะจุดใกล้ของวงโคจร วัดจากเส้นตัดระนาบของระนาบวงโคจรกับระนาบท้องฟ้าถึงจุดใกล้ของวงโคจร

β : มุมเอียงของระนาบวงโคจรเทียบกับระนาบท้องฟ้า

α : มุมเส้นตัดระนาบ เป็นมุมที่วัดจากแกน x_s ถึงเส้นตัดระนาบ

ที่มุมขวาด้านบน t : เวลา

R : ระยะห่างระหว่างดาราจักร

การควบคุมการทำงานของโปรแกรม

แป้นที่ใช้ควบคุมการทำงานของโปรแกรมจะประกอบด้วย

Esc : ออกจากโปรแกรม เมื่อกดแป้น Esc แล้วจะปรากฏหน้าจอในรูปที่ จ.1

Pause : หยุดการแสดงผล / ทำการแสดงผลต่อ ก่อนที่จะกดแป้นควบคุมการทำงานอื่นๆของโปรแกรมยกเว้น Esc ควรจะกดแป้น Pause ก่อนเสมอ

C : เปลี่ยนสีพื้นหลัง

H : ซ่อน / แสดงแกนอ้างอิงของจุดศูนย์กลางมวลของดาราจักร

G : ซ่อน / แสดงแกนอ้างอิงของระนาบดาราจักร

O : แสดงเส้นวงโคจรของดาราจักร

$\uparrow, \downarrow$: เพิ่มหรือลดค่า β

$\rightarrow, \leftarrow$: เพิ่มหรือลดค่า α

PageUp, PageDown : เพิ่มหรือลดค่า w

+ , - : ขยายหรือลดขนาดภาพที่แสดงบนหน้าจอ

1 : แสดงผลดาวเป็นจุด

2 : แสดงผลดาวเป็นกากบาท

3 : แสดงผลดาวเป็นวงกลม

F : เติม / ไม่เติมสีลงในวงกลมดาว

เนื่องจากโปรแกรมอาจจะยังไม่สมบูรณ์มากนัก หากเกิดข้อผิดพลาดในการทำงานผู้เขียนต้องขออภัยไว้ ณ ที่นี้ด้วย

ภาคผนวก ฉ

การจัดทำภาพการเกิดอันตรกิริยาระหว่างดาราจักรเป็นภาพเคลื่อนไหว

การจัดทำภาพเคลื่อนไหวจะเริ่มจากการจำลองภาพอันตรกิริยาระหว่างดาราจักรโดยโปรแกรม Galactic Interaction Version 1.26 จากนั้นจะทำสำเนาภาพ (copy) แต่ละเฟรมมาเรียงต่อกันและแปลงเป็นไฟล์ภาพเคลื่อนไหวโดยโปรแกรม Animation Shop Version 3.05 ของบริษัท Jasc Software

ไฟล์ภาพเคลื่อนไหวทั้งหมดจะอยู่ในแผ่นซีดีที่แนบไว้ท้ายเล่ม สำหรับรายละเอียดของภาพเคลื่อนไหวแต่ละภาพจะแสดงไว้ในตารางที่ ฉ.1

ตารางที่ ฉ.1

ดาราจักร	มุมมอง (β, λ) (องศา, องศา)	เวลาเริ่มต้น	เวลาสิ้นสุด	ช่วงเวลาทั้งหมด (ล้านปี)	จำนวนเฟรม
น้ำวน	*57, -2*	*-6.00*	*2.43*	*396*	*85*
หนู	*-90, -59*	*-6.00*	*3.50*	*446*	*96*
หนวดแมลง	*-83, -15*	*-6.00*	*2.20*	*385*	*83*
ลูกอ๊อด	*92, 123*	*-6.00*	*3.50*	*400*	*96*
ล้อเกวียน	*-93, -56*	*-0.50*	*0.63*	*53*	*58*

หลังจากได้ภาพดาราจักรในตารางที่ ฉ.1 แล้ว ก็จะทำการหมุนภาพโดยเพิ่มค่า β ครั้งละ 5° เพื่อให้สามารถมองเห็นระบบดาราจักรจากหลายๆมุมมอง จนกระทั่งมุมมองกลับมาอยู่ที่ตำแหน่งเดิมอีกครั้ง(เพิ่มมุมจนถึง $\beta + 360^\circ$) ซึ่งคิดเป็นจำนวนเฟรมในส่วนนี้ทั้งหมด *72* เฟรม

ภาคผนวก ช

โปรแกรมจำลองภาพอันตรกิริยาระหว่างดาราจักรในอวกาศ โดยวิธีปัญหาวัตถุ 3 ชิ้นกรณีจำกัด

```
#include<graphics.h>
#include<conio.h>
#include<math.h>
#include<stdio.h>
#include<stdlib.h>
#include<dos.h>
#define N1 103   // จำนวนดาวต่อหนึ่งวงโคจร
#define N2 10    // จำนวนวงโคจร
#define m 11
int Input();  // ฟังก์ชันรับค่าพารามิเตอร์
void Move(int, int*, int*);
void Getpress(int, int, char);
void G_Coor(float*, float*, float*, float*, float*, float*, float *, float *, float *, float, float);
                        // ฟังก์ชันกำหนดพิกัดแกนอ้างอิงของดาราจักร
void Rotate_Gco(float*, float*, float*, float*, float*, float*, float, float, float,  // ฟังก์ชันหมุน
          float, float, float, float, float, float, float, float, float);           // แกนอ้างอิงของ
                                                                                     //  ดาราจักร
void Rotate_Coor(float*, float*, float*, float*, float*, float*, float, float, float);  // ฟังก์ชันหมุน
                                                                 // แกนอ้างอิงของศูนย์กลางมวล
void Rotate(float*, float*, float, float, float, float, float, float);  // ฟังก์ชันแปลงพิกัดของจุดจาก
                                                     //กรอบอ้างอิงของศูนย์กลางมวลไปสู่ระนาบท้องฟ้า
float rkp(float, float, float, float); // ฟังก์ชันคำนวณ r-k 5th สำหรับวงโคจรพาราโบลา
float rk(float, float, float, float, float);  // ฟังก์ชันคำนวณ r-k 5th สำหรับวงโคจรไฮเปอร์โบลา
float rkell(float, float, float, float, float);  // ฟังก์ชันคำนวณ r-k 5th สำหรับวงโคจรวงรี
float L(float, float, float, float, float);  // ฟังก์ชันคำนวณระยะห่างระหว่างดาวกับ
                                       // ดาราจักรตัวรบกวน
```

```
float rs(float, float, float);  // ฟังก์ชันคำนวณระยะห่างระหว่างดาวกับดาราจักรเป้า
float gx(float, float, float, float, float, float, float, float);  // ฟังก์ชันคำนวณ r-k 5th เพื่อหาตำแหน่ง
                                                                   // ของดาวในแนวแกน x
float gy(float, float, float, float, float, float, float, float);  // ฟังก์ชันคำนวณ r-k 5th เพื่อหาตำแหน่ง
                                                                   // ของดาวในแนวแกน y
float gz(float, float, float, float, float, float, float);  // ฟังก์ชันคำนวณ r-k 5th เพื่อหาตำแหน่งของ
                                                            // ดาวในแนวแกน z

int count = 1;
char ce[m], cr[m], cq[m], cM1[m], cM2[m], ci1[m], ci2[m], co1[m], co2[m], cr1[m], cr2[m],
     cN12[m], cN22[m];
double e, r, q, i1, i2, M1, M2, o1, o2, r1, r2, N12, N22;

main()  // ฟังก์ชันการคำนวณหลัก
{int gdrive = DETECT, gmode = VGAHI;
 int  j, i, press, quit, ch, out = 1, countt, xx, yy, qdr, G_co;
 int  bk_num, co_num, size_num, draw_num, fill_num, orb_num;
 char  d2='n';
 float t, h, b, x, y, n1, n2, w, f, a, a2, M, aa, ww, ii, oo, u, tt, n;
 float c, d, g, l, o, p, s, ox, oy, ro, ow, roo, oox, ooy, oow, ot;
 float lx1, lx2, ly1, ly2, lz1, lz2;
 float lg1x1, lg1x2, lg1x3, lg1y1, lg1y2, lg1y3, lg1z1, lg1z2, lg1z3, lg2x1, lg2x2, lg2x3, lg2y1,
       lg2y2, lg2y3, lg2z1, lg2z2, lg2z3;
 float lg1x1p, lg1x2p, lg1y1p, lg1y2p, lg1z1p, lg1z2p, lg2x1p, lg2x2p, lg2y1p, lg2y2p, lg2z1p,
       lg2z2p;
 float f1, f2, f3, f4, f5, f6, g1, g2, g3, g4, g5, g6;
 float xgp1, ygp1, xgp2, ygp2;
 float vx10[N2][N1], vy10[N2][N1], vz10[N2][N1], vx1, vy1, vz1;
 float vx20[N2][N1], vy20[N2][N1], vz20[N2][N1], vx2, vy2, vz2;
 float xs10[N2][N1], ys10[N2][N1], zs10[N2][N1], xs1, ys1, zs1;
 float xs20[N2][N1], ys20[N2][N1], zs20[N2][N1], xs2, ys2, zs2;
 float xs1p, ys1p, xs2p, ys2p;
```

```
float xplot1, yplot1, xplot2, yplot2;


while(out!=2)
        {out = Input();
        if(out != 2)
                {initgraph(&gdrive, &gmode, "\\TC\\BGI");
                i1 = i1*M_PI/180.0;
                i2 = i2*M_PI/180.0;
                o1= o1*M_PI/180.0;
                o2 = o2*M_PI/180.0;
                t = 0.0;
                h = 0.1;  // กำหนด Step Size
                M = M1 + M2;
                if(e = = 1.0)  // กำหนดชนิดของวงโคจร
                        countt = 1;
                else if(e > 1.0)
                        countt = 2;
                else
                        countt = 3;
                switch(countt)  // คำนวณตำแหน่งเริ่มต้นของดาราจักร
                        {case 1: w = acos(2.0*q/r - 1.0);  // กรณีวงโคจรพาราโบลา
                                x = r*cos(w);
                                y = r*sin(w);
                                n = sqrt(M/(q*q*q));
                                u = r*sin(w)/(sqrt(2.0)*q);
                                tt= (u*u*u/6.0 + u)/n;
                        break;
                        case 2:  aa= q/(e - 1.0);  // กรณีวงโคจรไฮเปอร์โบลา
                                w = acos((aa*(e*e - 1.0) - r)/(e*r));
                                x = r*cos(w);
                                y = r*sin(w);
```

```
                n = sqrt(M/(aa*aa*aa));
                u = log(e - r*(cos(w) - sin(w)/sqrt(e*e - 1.0))/aa);
                tt= (e*sinh(u) - u)/n;
        break;
        case 3:  aa = q/(1.0 - e);  // กรณีวงโคจรวงรี
                w = acos((aa*(1.0 - e*e) - r)/(e*r));
                x = r*cos(w);
                y = r*sin(w);
                n = sqrt(M/(aa*aa*aa));
                u = acos((cos(w) + e)/(1.0 + e*cos(w)));
                tt= (u - e*sin(u))/n;
        break;
        }
oox = x; ooy = y; oow = w; roo = r;
G_Coor(&lg1x1, &lg1x2, &lg1x3, &lg1y1, &lg1y2, &lg1y3, &lg1z1, &lg1z2,
        &lg1z3, i1, o1);
G_Coor(&lg2x1, &lg2x2, &lg2x3, &lg2y1, &lg2y2, &lg2y3, &lg2z1, &lg2z2,
        &lg2z3, i2, o2);
ww = -15.0*M_PI/180.0, ii = -83.0*M_PI/180.0, oo = 35.0*M_PI/180.0;
Rotate_Coor(&lx1, &lx2, &ly1, &ly2, &lz1, &lz2, ww, ii, oo);
quit = 1; bk_num = 0; co_num = 0; size_num = 2;
draw_num = 1; fill_num = 0; orb_num = 0;
xx = 316; yy = 206; qdr = 0;
while(quit!= 0)  // ออกจากโปรแกรมเมื่อ quit = 0 (กดแป้น ESC.)
        {while(!kbhit())   // หยุดทำงานเมื่อมีการกดแป้นพิมพ์
                {cleardevice();
                switch(size_num)  // กำหนดขนาดขยายในการแสดงผล
                        {case 0: s = 0.25;
                        break;
                        case 1: s = 0.5;
                        break;
                        case 2: s = 1.0;
```

```
break;
case 3: s = 2.0;
break;
case 4: s = 3.0;
break;
case 5: s = 4.0;
break;
case 6: s = 5.0;
break;
case 7: s = 7.0;
break;
case 8: s = 9.0;
break;
case 9: s = 11.0;
break;
case 10: s = 13.0;
break;
case 11: s = 15.0;
break;
}
switch(qdr) // กำหนดโหมดในการแสดงผลว่าจะแสดงผลทุก
        จัตุภาค หรือเลือกแสดงบางจัตุภาค
{case 0: xx = 316; yy = 206;
break;
case 1: xx = 639; yy = 479;
break;
case 2: xx = 0;  yy = 479;
break;
case 3: xx = 639; yy = 0;
break;
case 4: xx = 0;  yy = 0;
break;
```

```
        }
    if((bk_num%2) = = 0)  // กำหนดสีพื้น
        setbkcolor(BLUE);
    else
        setbkcolor(BLACK);
```

/* ส่วนของการวาดแกนอ้างอิงและแสดงข้อมูลที่สำคัญบนหน้าจอ */

```
    if(qdr = = 0)
        {gotoxy(1, 1);
        printf("   ww: %.0f", ww*180.0/M_PI);
        gotoxy(1, 2);
        printf("   \xE1: %.0f", ii*180.0/M_PI);
        gotoxy(1, 3);
        printf("   \xE0: %.0f", oo*180.0/M_PI);
        setcolor(YELLOW);
        outtextxy(5, 470, "Pause:stop.   Esc: exit.");
        setcolor(WHITE);
        outtextxy(255, 470 ,"  \x1B,\x1A:\xE0");
        setcolor(RED);
        outtextxy(360, 470, "  \x18,\x19:\xE1");
        setcolor(MAGENTA);
        outtextxy(475, 470, "  PageUp,PageDown:ww");
        gotoxy(70, 1);
        printf("t: %.2f", (t-tt)*0.01118);
        gotoxy(70, 2);
        printf("r: %.2f", r/20.0);
        }
    if(t = = 0.0)
        {a = 2.0; a2 = 2.0;
        n1 = sqrt(M1/(a*a*a));
        n2 = sqrt(M2/(a2*a2*a2));
        }
```

```
if((co_num%2) = = 0)
        {setcolor(RED);
        line(xx, yy, xx + lx1*s, yy - lx2*s);
        setcolor(MAGENTA);
        line(xx, yy, xx + lz1*s, yy - lz2*s);
        setcolor(YELLOW);
        line(xx, yy, xx + ly1*s, yy - ly2*s);
        }
Rotate(&xgp1, &ygp1, ww, ii, oo, -M2*x/M, M2*y/M, 0.0);
Rotate(&xgp2, &ygp2, ww, ii, oo, M1*x/M, -M1*y/M, 0.0);
setcolor(LIGHTGREEN);
setfillstyle(SOLID_FILL, LIGHTGREEN);
circle(xx + xgp1*s, yy - ygp1*s, 2);
floodfill(xx + xgp1*s, yy - ygp1*s, LIGHTGREEN);
circle(xx + xgp2*s, yy - ygp2*s, 2);
floodfill(xx + xgp2*s, yy - ygp2*s, LIGHTGREEN);

/* ส่วนของการคำนวณตำแหน่งของดาวและการแสดงผลตำแหน่งของดาว*/
for(j = 0; j < N2 ; j++)
        {for(i = 0; i < N1; i++)
                {if(t = = 0.0)  // กำหนดค่าเริ่มต้นของดาว
                        {f = 2.0*M_PI*i/N1;
                        vx10[j][i] = - a*n1*sin(f)*cos(o1) –
                         a*n1*cos(f)*sin(o1)*cos(i1);
                        vy10[j][i] = -a*n1*sin(f)*sin(o1)
                         + a*n1*cos(f)*cos(o1)*cos(i1);
                        vz10[j][i] = a*n1*cos(f)*sin(i1);
                        xs10[j][i] = a*cos(f)*cos(o1)
                         - a*sin(f)*sin(o1)*cos(i1);
                        ys10[j][i] =  a*cos(f)*sin(o1)
                        + a*sin(f)*cos(o1)*cos(i1);
                        zs10[j][i] = a*sin(f)*sin(i1);
```

```
vx20[j][i] = - a2*n2*sin(f)*cos(o2)
- a2*n2*cos(f)*sin(o2)*cos(i2);
vy20[j][i] = -a2*n2*sin(f)*sin(o2)
+ a2*n2*cos(f)*cos(o2)*cos(i2);
vz20[j][i] =  a2*n2*cos(f)*sin(i2);
xs20[j][i] = a2*cos(f)*cos(o2)
- a2*sin(f)*sin(o2)*cos(i2);
ys20[j][i] =  a2*cos(f)*sin(o2)
+ a2*sin(f)*cos(o2)*cos(i2);
zs20[j][i] = a2*sin(f)*sin(i2);
if(r1 = = 'y')
        {vx10[j][i] = - vx10[j][i];
         vy10[j][i] = - vy10[j][i];
         vz10[j][i] = - vz10[j][i];
        }
if(r2 = = 'y')
        {vx20[j][i] = - vx20[j][i];
         vy20[j][i] = - vy20[j][i];
         vz20[j][i] = - vz20[j][i];
        }
}
if(j < N12) // แปลงพิกัดของดาวให้อยู่ในพิกัด
        ของระนาบท้องฟ้า
    Rotate(&xs1p, &ys1p, ww, ii, oo,
        xs10[j][i], ys10[j][i],
        zs10[j][i]);
if(j<N22)
    Rotate(&xs2p, &ys2p, ww, ii, oo,
        xs20[j][i], ys20[j][i],
        zs20[j][i]);
xplot1 = xgp1 + xs1p;
yplot1 = ygp1 + ys1p;
```

```
xplot2 = xgp2 + xs2p;
yplot2 = ygp2 + ys2p;
c = xs10[j][i];
d = ys10[j][i];
g = zs10[j][i];
l = xs20[j][i];
o = ys20[j][i];
p = zs20[j][i];
if(i = = 1) // แสดงผลตำแหน่งของดาว
        {if(j < N12)
                putpixel(xx + xplot1*s, yy –
                        yplot1*s, RED);
        if(j<N22)
                putpixel(xx + xplot2*s, yy –
                        yplot2*s, RED);
        }
else
        {if(j < N12)
                putpixel(xx + xplot1*s, yy –
                        yplot1*s, WHITE);
        if(j < N22)
                putpixel(xx + xplot2*s, yy –
                        yplot2*s,
                        YELLOW);
        }

/* ส่วนของการคำนวณ r-k 5th เพื่อหาตำแหน่งของดาวที่เวลา t ใดๆ */
        f1 = h*vx10[j][i];
        g1 = h*gx(x, - y, r, c, d, g, M1, M2);
        f2 = h*(vx10[j][i] + g1/4);
        g2 = h*gx(x, - y, r, c + f1/4, d, g, M1, M2);
        f3 = h*(vx10[j][i] + g1/8 + g2/8);
```

```
g3 = h*gx(x, - y, r, c + f1/8 + f2/8, d, g, M1,
          M2);
f4 = h*(vx10[j][i] - g2/2 + g3);
g4 = h*gx(x, - y, r, c - f2/2 + f3, d, g, M1,
          M2);
f5 = h*(vx10[j][i] + 3*g1/16 + 9*g4/16);
g5 = h*gx(x, - y ,r ,c + 3*f1/16 + 9*f4/16, d,
          g, M1, M2);
f6 = h*(vx10[j][i] - 3*g1/7 + 2*g2/7 +
        12*g3/7 - 12*g4/7 + 8*g5/7);
g6 = h*gx(x, - y, r, c - 3*f1/7 + 2*f2/7 +
          12*f3/7 - 12*f4/7 + 8*f5/7, d, g,
          M1, M2);
vx1 = vx10[j][i] + 0.0111111*(7*g1 + 32*g3
      + 12*g4 + 32*g5 + 7*g6);
xs1 = xs10[j][i] + 0.0111111*(7*f1 + 32*f3 +
      12*f4 + 32*f5 + 7*f6);
vx10[j][i] = vx1;
xs10[j][i] = xs1;


f1 = h*vx20[j][i];
g1 = h*gx(- x, y, r, l, o, p, M2, M1);
f2 = h*(vx20[j][i] + g1/4);
g2 = h*gx(- x, y, r, l + f1/4, o, p, M2, M1);
f3 = h*(vx20[j][i] + g1/8 + g2/8);
g3 = h*gx(- x, y, r, l + f1/8 + f2/8, o, p, M2,
          M1);
f4 = h*(vx20[j][i] - g2/2 + g3);
g4 = h*gx(- x, y, r, l - f2/2 + f3, o, p, M2,
          M1);
f5 = h*(vx20[j][i] + 3*g1/16 + 9*g4/16);
```

```
g5 = h*gx(- x, y, r, l + 3*f1/16 + 9*f4/17, o,
        p, M2, M1);
f6 = h*(vx20[j][i] - 3*g1/7 + 2*g2/7 +
      12*g3/7 - 12*g4/7 + 8*g5/7);
g6 = h*gx(- x, y, r, l - 3*f1/7 + 2*f2/7 +
        12*f3/7 - 12*f4/7 + 8*f5/7, o, p,
        M2, M1);
vx2 = vx20[j][i] + 0.0111111*(7*g1 + 32*g3
     + 12*g4 + 32*g5 + 7*g6);
xs2 = xs20[j][i] + 0.0111111*(7*f1 + 32*f3 +
     12*f4 + 32*f5 + 7*f6);
vx20[j][i] = vx2;
xs20[j][i] = xs2;


f1 = h*vy10[j][i];
g1 = h*gy(x, - y, r, c, d, g, M1, M2);
f2 = h*(vy10[j][i] + g1/4);
g2 = h*gy(x, - y, r, c, d + f1/4, g, M1, M2);
f3 = h*(vy10[j][i] + g1/8 + g2/8);
g3 = h*gy(x, - y, r, c, d + f1/8 + f2/8, g, M1,
        M2);
f4 = h*(vy10[j][i] - g2/2 + g3);
g4 = h*gy(x, - y, r, c, d - f2/2 + f3, g, M1,
        M2);
f5 = h*(vy10[j][i] + 3*g1/16 + 9*g4/16);
g5 = h*gy(x, - y, r, c, d + 3*f1/16 + 9*f4/16,
        g, M1, M2);
f6 = h*(vy10[j][i] - 3*g1/7 + 2*g2/7 +
      12*g3/7 - 12*g4/7 + 8*g5/7);
g6 = h*gy(x, - y, r, c, d - 3*f1/7 + 2*f2/7 +
        12*f3/7 - 12*f4/7 + 8*f5/7, g, M1,
```

```
            M2);
vy1 = vy10[j][i] + 0.0111111*(7*g1 + 32*g3
      + 12*g4 + 32*g5 + 7*g6);
ys1 = ys10[j][i] + 0.0111111*(7*f1 + 32*f3 +
      12*f4 + 32*f5 + 7*f6);
vy10[j][i] = vy1;
ys10[j][i] = ys1;


f1 = h*vy20[j][i];
g1 = h*gy(- x, y, r, l, o, p, M2, M1);
f2 = h*(vy20[j][i] + g1/4);
g2 = h*gy(- x, y, r, l, o + f1/4, p, M2, M1);
f3 = h*(vy20[j][i] + g1/8 + g2/8);
g3 = h*gy(- x, y, r, l, o + f1/8 + f2/8, p, M2,
            M1);
f4 = h*(vy20[j][i] - g2/2 + g3);
g4 = h*gy(- x, y, r, l, o - f2/2 + f3, p, M2,
            M1);
f5 = h*(vy20[j][i] + 3*g1/16 + 9*g4/16);
g5 = h*gy(- x, y, r, l, o + 3*f1/16 + 9*f4/16,
            p, M2, M1);
f6 = h*(vy20[j][i] - 3*g1/7 + 2*g2/7 +
        12*g3/7 - 12*g4/7 + 8*g5/7);
g6 = h*gy(- x, y, r, l, o - 3*f1/7 + 2*f2/7 +
            12*f3/7 - 12*f4/7 + 8*f5/7, p, M2,
            M1);
vy2 = vy20[j][i] + 0.0111111*(7*g1 + 32*g3
      + 12*g4 + 32*g5 + 7*g6);
ys2 = ys20[j][i] + 0.0111111*(7*f1 + 32*f3 +
      12*f4 + 32*f5 + 7*f6);
vy20[j][i] = vy2;
```

```
ys20[j][i] = ys2;


f1 = h*vz10[j][i];
g1 = h*gz(x, - y, c, d, g, M1, M2);
f2 = h*(vz10[j][i] + g1/4);
g2 = h*gz(x, - y, c, d, g + f1/4, M1, M2);
f3 = h*(vz10[j][i] + g1/8 + g2/8);
g3 = h*gz(x, - y, c, d, g + f1/8 + f2/8, M1,
        M2);
f4 = h*(vz10[j][i] - g2/2 + g3);
g4 = h*gz(x, - y, c, d, g - f2/2 + f3, M1, M2);
f5 = h*(vz10[j][i] + 3*g1/16 + 9*g4/16);
g5 = h*gz(x, - y, c, d, g + 3*f1/16 + 9*f4/16 ,
        M1, M2);
f6 = h*(vz10[j][i] - 3*g1/7 + 2*g2/7 +
       12*g3/7 - 12*g4/7 + 8*g5/7);
g6 = h*gz(x, - y, c, d, g - 3*f1/7 + 2*f2/7 +
        12*f3/7 - 12*f4/7 + 8*f5/7 , M1,
        M2);
vz1 = vz10[j][i] + 0.0111111*(7*g1 + 32*g3
      + 12*g4 + 32*g5 + 7*g6);
zs1 = zs10[j][i] + 0.0111111*(7*f1 + 32*f3 +
      12*f4 + 32*f5 + 7*f6);
vz10[j][i] = vz1;
zs10[j][i] = zs1;


f1 = h*vz20[j][i];
g1 = h*gz(- x, y, l, o, p, M2, M1);
f2 = h*(vz20[j][i] + g1/4);
g2 = h*gz(- x, y, l, o, p + f1/4, M2, M1);
```

```
                f3 = h*(vz20[j][i] + g1/8 + g2/8);
                g3 = h*gz(- x, y, l, o, p + f1/8 + f2/8, M2,
                        M1);
                f4 = h*(vz20[j][i] - g2/2 + g3);
                g4 = h*gz(- x, y, l, o, p - f2/2 + f3, M2, M1);
                f5 = h*(vz20[j][i] + 3*g1/16 + 9*g4/16);
                g5 = h*gz(- x, y, l, o, p + 3*f1/16 + 9*f4/16,
                        M2, M1);
                f6 = h*(vz20[j][i] - 3*g1/7 + 2*g2/7 +
                      12*g3/7 - 12*g4/7 + 8*g5/7);
                g6 = h*gz(- x, y, l, o, p - 3*f1/7 + 2*f2/7 +
                        12*f3/7 - 12*f4/7 + 8*f5/7, M2,
                        M1);
                vz2 = vz20[j][i] + 0.0111111*(7*g1 + 32*g3
                     + 12*g4 + 32*g5 + 7*g6);
                zs2 = zs20[j][i] + 0.0111111*(7*f1 + 32*f3 +
                     12*f4 + 32*f5 + 7*f6);
                vz20[j][i] = vz2;
                zs20[j][i] = zs2;
                }
        if(t = = 0.0)
                {a = a + 2.0; a2 = a2 + 2.0;
                n1 = sqrt(M1/(a*a*a));
                n2 = sqrt(M2/(a2*a2*a2));
                }
        }
t = t + h;
switch(countt) //คำนวณตำแหน่งของดาราจักรที่เวลา t = t + h
        {case 1: f1 = h*rkp(w, M1, M2, q); // กรณีพาราโบลา
                f2 = h*rkp(w + f1/4.0, M1, M2, q);
                f3 = h*rkp(w + f1/8.0 + f2/8.0, M1, M2, q);
                f4 = h*rkp(w - f2/2.0 + f3, M1, M2, q);
```

```
            f5 = h*rkp(w + 3.0*f1/16.0 + 9.0*f4/16.0,
                        M1, M2, q);
            f6 = h*rkp(w - 3.0*f1/7.0 + 2.0*f2/7.0 +
                        12.0*f3/7.0 - 12.0*f4/7.0 +
                        8.0*f5/7.0, M1, M2, q);
            w = w - (1.0/90.0)*(7.0*f1 + 32.0*f3 +
                  12.0*f4 + 32.0*f5 + 7.0*f6);
            r = 2.0*q/(1.0 + cos(w));
            x = r*cos(w);
            y = r*sin(w);
break;
case 2:  f1 = h*rk(w, M1, M2, aa, e);  // กรณีไฮเปอร์
                                          โบลา
            f2 = h*rk(w + f1/4.0, M1, M2, aa, e);
            f3 = h*rk(w + f1/8.0 + f2/8.0, M1, M2, aa, e);
            f4 = h*rk(w - f2/2.0 + f3, M1, M2, aa, e);
            f5 = h*rk(w + 3.0*f1/16.0 + 9.0*f4/16.0, M1,
                       M2, aa, e);
            f6 = h*rk(w - 3.0*f1/7.0 + 2.0*f2/7.0 +
                       12.0*f3/7.0 - 12.0*f4/7.0 +
                       8.0*f5/7.0, M1, M2, aa, e);
            w = w - (1.0/90.0)*(7.0*f1 + 32.0*f3 +
                  12.0*f4 + 32.0*f5 + 7.0*f6);
            r = aa*(e*e - 1.0)/(1.0 + e*cos(w));
            x = r*cos(w);
            y = r*sin(w);
break;
case 3:  f1 = h*rkell(w, M1, M2, aa, e);  // กรณีวงรี
            f2 = h*rkell(w + f1/4.0, M1, M2, aa, e);
            f3 = h*rkell(w + f1/8.0 + f2/8.0, M1, M2, aa,
                         e);
            f4 = h*rkell(w - f2/2.0 + f3, M1, M2, aa, e);
```

```
                    f5 = h*rkell(w + 3.0*f1/16.0 + 9.0*f4/16.0,
                                  M1, M2, aa, e);
                    f6 = h*rkell(w - 3.0*f1/7.0 + 2.0*f2/7.0 +
                                  12.0*f3/7.0 - 12.0*f4/7.0 +
                                  8.0*f5/7.0 , M1, M2, aa, e);
                    w = w - (1.0/90.0)*(7.0*f1 + 32.0*f3 +
                          12.0*f4 + 32.0*f5 + 7.0*f6);
                    r = aa*(1.0 - e*e)/(1.0 + e*cos(w));
                    x =  r*cos(w);
                    y =  r*sin(w);
               break;
               }
          } //while(!kbhit()), 168x13642
```

/* ส่วนของการเปลี่ยนแปลงการแสดงผลตามที่ป้อนค่าให้ทางแป้นพิมพ์ */

```
          ch = getch();
          if(ch = = 32)
               {if(d2 = = 'n')
                    {G_co = 0;
                    press = getch();
                    while(!((press = = 32)|(press = = 27)))  //Pause to
                                                              continue:
                                                              Esc to end
                                                              program.
                         {switch(press)
                              {case 75: oo = oo - M_PI/180.0;
                              break; // ลดค่า Ω
                              case 77: oo = oo + M_PI/180.0;
                              break;  // เพิ่มค่า Ω
                              case 72: ii = ii + M_PI/180.0;
                              break;  // เพิ่มค่า i
                              case 80: ii = ii - M_PI/180.0;
```

```
break;  // ลดค่า i
case 73: ww = ww + M_PI/180.0;
break;  // เพิ่มค่า ω
case 81: ww = ww - M_PI/180.0;
break;  // ลดค่า ω
case 99: bk_num = bk_num + 1;
break;  // เปลี่ยนสีพื้น
case 104: co_num = co_num + 1;
break; // แสดง/ซ่อนแกนอ้างอิง
case 43: if(size_num < 11)
                size_num + = 1;
break;  // ขยายขนาดการแสดงผล
case 45: if(size_num > 0)
                size_num - = 1;
break;  // ลดขนาดการแสดงผล
case 49: draw_num = 1;
break; // แสดงตำแหน่งดาวเป็นจุด
case 50: draw_num = 2;
break;  // แสดงตำแหน่งดาวเป็น
          กากบาท
case 51: draw_num = 3;
break;  // แสดงตำแหน่งดาวเป็น
          วงกลม
case 102: fill_num + = 1;
break; // เติมสีให้วงกลมของดาว
case 111: orb_num + = 1;
break; // แสดงรูปแนวทางโคจร
case 113: qdr = 0;
break; // แสดงผลทุกจัตุภาค
case 119: qdr = 1;
break;  // แสดงผลจัตุภาคที่ 1
case 101: qdr = 2;
```

```
		break;  // แสดงผลจัตุภาคที่ 2
		case 114: qdr = 3;
		break;  // แสดงผลจัตุภาคที่ 3
		case 116: qdr = 4;
		break;  // แสดงผลจัตุภาคที่ 4
		case 103: G_co + = 1;
		break;  // แสดง/ซ่อนแกนอ้างอิงของ
				ดาราจักร
		}
	switch(size_num)
		{case 0: s = 0.25;
		break;
		case 1: s = 0.5;
		break;
		case 2: s = 1.0;
		break;
		case 3: s = 2.0;
		break;
		case 4: s = 3.0;
		break;
		case 5: s = 4.0;
		break;
		case 6: s = 5.0;
		break;
		case 7: s = 7.0;
		break;
		case 8: s = 9.0;
		break;
		case 9: s = 11.0;
		break;
		case 10: s = 13.0;
		break;
```

```
        case 11: s = 15.0;
        break;
        }
switch(qdr)
        {case 0: xx = 316; yy = 206;
        break;
        case 1: xx = 639; yy = 479;
        break;
        case 2: xx = 0;  yy = 479;
        break;
        case 3: xx = 639; yy = 0;
        break;
        case 4: xx = 0;  yy = 0;
        break;
        }
cleardevice();
if((bk_num%2) = = 0)
        setbkcolor(BLUE);
else
        setbkcolor(BLACK);
Rotate_Coor(&lx1, &lx2, &ly1, &ly2, &lz1,
                &lz2, ww, ii, oo);
if((co_num%2) = = 0)
        {setcolor(MAGENTA);
        line(xx, yy, xx + lz1*s, yy – lz2*s);
        setcolor(YELLOW);
        line(xx, yy, xx + ly1*s, yy - ly2*s);
        setcolor(RED);
        line(xx, yy, xx + lx1*s, yy - lx2*s);
        }
if((orb_num%2) != 0)  // คำนวณวงโคจรของ
                                    ดาราจักรและแสดงผล
```

```
{ro = roo; ox = oox; oy = ooy;
 ow = oow; ot=0.0;
 while(((ot < =  0.95*
 sqrt(4.0*M_PI*M_PI*aa*aa*aa/M))
 && (countt= =3)) || (((countt= =1) ||
 (countt = = 2)) && (ro < = roo)))
         {Rotate(&xgp1, &ygp1, ww,
                ii, oo, -M2*ox/M,
                M2*oy/M, 0.0);
         Rotate(&xgp2, &ygp2, ww,
                ii, oo, M1*ox/M, -
                M1*oy/M, 0.0);
         putpixel(xx + xgp1*s, yy –
                   ygp1*s, BROWN);
         putpixel(xx + xgp2*s, yy –
                   ygp2*s, BROWN);
         switch(countt)
                {case 1: f1 =
              h*rkp(ow, M1, M2, q);
 f2 = h*rkp(ow + f1/4.0, M1, M2,q);
 f3 = h*rkp(ow + f1/8.0 + f2/8.0, M1, M2, q);

f4 = h*rkp(ow - f2/2.0 + f3, M1, M2, q);
f5 = h* rkp(ow + 3.0* f1/16.0 + 9.0*f4/16.0, M1, M2, q);
f6 = h* rkp(ow - 3.0*f1/7.0 + 2.0*f2/7.0 + 12.0*f3/7.0 -
        12.0*f4/7.0 + 8.0*f5/7.0 , M1, M2, q);
 ow = ow -(1.0/90.0)*(7.0*f1 + 32.0*f3 + 12.0*f4 +
       32.0*f5 + 7.0*f6);
 ro = 2.0*q/(1.0 + cos(ow));
 ox = ro*cos(ow);
 oy = ro*sin(ow);
                     break;
```

```
case 2:
f1 = h*rk(ow, M1, M2, aa, e);
f2 = h*rk(ow + f1/4.0, M1, M2, aa, e);
f3 = h*rk(ow + f1/8.0 + f2/8.0, M1, M2, aa, e);
f4 = h*rk(ow - f2/2.0 + f3, M1, M2, aa, e);
f5 = h*rk(ow + 3.0*f1/16.0 + 9.0*f4/16.0, M1, M2, aa,
        e);
f6 = h*rk(ow - 3.0*f1/7.0 + 2.0*f2/7.0 + 12.0*f3/7.0 –
        12.0*f4/7.0 + 8.0*f5/7.0, M1, M2, aa, e);
ow = ow - (1.0/90.0)*(7.0*f1 + 32.0*f3 + 12.0*f4 +
        32.0*f5 + 7.0*f6);
ro = aa*(e*e - 1.0)/(1.0 + e*cos(ow));
ox = ro*cos(ow);
oy = ro*sin(ow);
break;
case 3:
f1 = h*rkell(ow, M1, M2, aa, e);
f2 = h*rkell(ow + f1/4.0, M1, M2, aa, e);
f3 = h*rkell(ow + f1/8.0 + f2/8.0, M1, M2, aa, e);
f4 = h*rkell(ow - f2/2.0 + f3, M1, M2, aa, e);
f5 = h*rkell(ow + 3.0*f1/16.0 + 9.0*f4/16.0, M1, M2,
        aa, e);
f6 = h*rkell(ow - 3.0*f1/7.0 + 2.0*f2/7.0 + 12.0*f3/7.0
        – 12.0*f4/7.0 + 8.0*f5/7.0, M1, M2, aa, e);
ow = ow - (1.0/90.0)*(7.0*f1 + 32.0*f3 + 12.0*f4 +
        32.0*f5 + 7.0*f6);
ro = aa*(1.0 - e*e)/(1.0 + e*cos(ow));
ox = ro*cos(ow);
oy = ro*sin(ow);
break;
}
ot = ot + h;
```

```
                }
            }

/* ส่วนของการแสดงตำแหน่งของดาราจักรและดาว */
                Rotate(&xgp1, &ygp1, ww, ii, oo, - M2*x/M,
                    M2*y/M, 0.0);
                Rotate(&xgp2, &ygp2, ww, ii, oo, M1*x/M, -
                    M1*y/M, 0.0);
                setcolor(LIGHTGREEN);
                setfillstyle(SOLID_FILL, LIGHTGREEN);
                circle(xx + xgp1*s, yy - ygp1*s, 2);
                floodfill(xx + xgp1*s, yy - ygp1*s,
                    LIGHTGREEN);
                circle(xx + xgp2*s, yy - ygp2*s, 2);
                floodfill(xx + xgp2*s, yy - ygp2*s,
                    LIGHTGREEN);
                for(j = 0; j < N2; j++)
                    for(i = 0; i < N1; i++)
                        {if(j < N12)
    Rotate(&xs1p, &ys1p, ww, ii, oo, xs10[j][i], ys10[j][i], zs10[j][i]);
                        if(j < N22)
    Rotate(&xs2p, &ys2p, ww, ii, oo, xs20[j][i], ys20[j][i], zs20[j][i]);
                        xplot1 = xgp1 + xs1p;
                        yplot1 = ygp1 + ys1p;
                        xplot2 = xgp2 + xs2p;
                        yplot2 = ygp2 + ys2p;
                        switch(draw_num)
                            {case 1: if(i = = 1)
        {if(j < N12)
                putpixel(xx + xplot1*s, yy - yplot1*s, RED);
        if(j < N22)
                putpixel(xx + xplot2*s, yy - yplot2*s, RED);
```

```
}
                                        else
{if(j < N12)
        putpixel(xx + xplot1*s, yy - yplot1*s,
                 WHITE);
 if(j < N22)
        putpixel(xx + xplot2*s, yy - yplot2*s,
                 YELLOW);
}
                               break;
                               case 2: if(i = = 1)
{setcolor(RED);
 if(j < N12)
        circle(xx + xplot1*s, yy - yplot1*s, 1);
 if(j < N22)
        circle(xx + xplot2*s, yy - yplot2*s, 1);
}
                                        else
{if(j < N12)
        {setcolor(WHITE);
                circle(xx + xplot1*s, yy - yplot1*s, 1);
        }
  if(j < N22)
        {setcolor(YELLOW);
                circle(xx + xplot2*s, yy - yplot2*s, 1);
        }
}
                               break;
                               case 3: if(i = = 1)
{setcolor(RED);
 if(j < N12)
        {setfillstyle(SOLID_FILL, RED);
```

```
circle(xx + xplot1*s, yy - yplot1*s, 2);
if((fill_num%2) != 0)
        floodfill(xx + xplot1*s, yy - yplot1*s,
                RED);
}
if(j < N22)
        {setfillstyle(SOLID_FILL, RED);
         circle(xx + xplot2*s, yy - yplot2*s, 2);
         if((fill_num % 2) != 0)
                floodfill(xx + xplot2*s, yy - yplot2*s,
                        RED);
        }
}
                                        else
{if(j < N12)
        {setcolor(WHITE);
         setfillstyle(SOLID_FILL, WHITE);
         circle(xx + xplot1*s, yy-yplot1*s, 2);
         if((fill_num%2)! = 0)
                floodfill(xx + xplot1*s, yy - yplot1*s,
                        WHITE);
        }
 if(j < N22)
        {setcolor(YELLOW);
         setfillstyle(SOLID_FILL, YELLOW);
         circle(xx + xplot2*s, yy - yplot2*s, 2);
         if((fill_num % 2) != 0)
                floodfill(xx + xplot2*s, yy - yplot2*s,
                        YELLOW);
        }
}
                                break;
```

```
                    }
                }

/* ส่วนของการวาดแกนอ้างอิงและแสดงข้อมูลที่สำคัญบนหน้าจอ */
            if((G_co % 2) != 0)
                {Rotate_Gco(&lg1x1p, &lg1x2p,
    &lg1y1p, &lg1y2p, &lg1z1p, &lg1z2p, ww, ii, oo, lg1x1,
    lg1x2, lg1x3, lg1y1, lg1y2, lg1y3, lg1z1, lg1z2, lg1z3);
                Rotate_Gco(&lg2x1p, &lg2x2p,
    &lg2y1p, &lg2y2p, &lg2z1p, &lg2z2p, ww, ii, oo, lg2x1,
    lg2x2, lg2x3, lg2y1, lg2y2, lg2y3, lg2z1, lg2z2, lg2z3);
                setcolor(RED);
                line(xx + xgp1*s, yy - ygp1*s, xx +
    (xgp1 + lg1x1p)*s, yy - (ygp1 + lg1x2p)*s);
                line(xx + xgp2*s, yy - ygp2*s, xx +
    (xgp2 + lg2x1p)*s, yy - (ygp2 + lg2x2p)*s);
                setcolor(YELLOW);
                line(xx + xgp1*s, yy -ygp1*s, xx +
    (xgp1 + lg1y1p)*s, yy - (ygp1 + lg1y2p)*s);
                line(xx + xgp2*s, yy - ygp2*s, xx +
    (xgp2 + lg2y1p)*s, yy - (ygp2 + lg2y2p)*s);
                setcolor(MAGENTA);
                line(xx + xgp1*s, yy - ygp1*s, xx +
    (xgp1 + lg1z1p)*s, yy - (ygp1 + lg1z2p)*s);
                line(xx + xgp2*s, yy - ygp2*s, xx +
    (xgp2 + lg2z1p)*s, yy - (ygp2 + lg2z2p)*s);
                }
            if(qdr = = 0)
                {gotoxy(1, 1);
                printf("  ww: %.0f",
    ww*180.0/M_PI);
                gotoxy(1, 2);
```

```
                                        printf("  \xE1:  %.0f",
            ii*180.0/M_PI);
                                        gotoxy(1, 3);
                                        printf("  \xE0:  %.0f",
            oo*180.0/M_PI);
                                        gotoxy(70, 1);
                                        printf("t: %.2f", (t-tt)*0.01118);
                                        gotoxy(70, 2);
                                        printf(" r: %.2f ", r/20.0);
                                        setcolor(YELLOW);
                                        outtextxy(5, 470, "Pause:continue.
            Esc:exit.");
                                        setcolor(WHITE);
                                        outtextxy(255, 470, "  \x1B,
            \x1A:\xE0");
                                        setcolor(RED);
                                        outtextxy(360, 470, "  \x18,
            \x19:\xE1");
                                        setcolor(MAGENTA);
                                        outtextxy(475, 470, "  PageUp,
            PageDown:ww");
                                        }
                                press = getch();  // รอรับคำสั่งทางแป้นพิมพ์
                                } //while((press != 32) || (press != 27))
                        } //if(d != 'n')
            else
                        quit = 0;
            } //if(ch = = 32)
switch(ch)
            {case 75: oo = oo - M_PI/180.0;
            break;
            case 77: oo = oo + M_PI/180.0;
```

```
                break;
                case 72: ii = ii + M_PI/180.0;
                break;
                case 80: ii = ii - M_PI/180.0;
                break;
                case 73: ww = ww + M_PI/180.0;
                break;
                case 81: ww = ww - M_PI/180.0;
                break;
                case 99: bk_num = bk_num + 1;
                break;
                case 104: co_num = co_num + 1;
                break;
                case 43: if(size_num < 11)
                                size_num  + = 1;
                break;
                case 45: if(size_num > 0)
                                size_num  - = 1;
                break;
                case 113: qdr = 0;
                break;
                case 119: qdr = 1;
                break;
                case 101: qdr = 2;
                break;
                case 114: qdr = 3;
                break;
                case 116: qdr = 4;
                break;
                }
        if(ch = = 27)
                quit = 0;
```

```
                    if(press = = 27)
                            quit = 0;
                    }  //while(quit != 0)
            }  //if(out != 2)
        closegraph();
        }//while(out != 2)
return(0);
}  // จบการทำงานของฟังก์ชัน main()


int Input()
{int gdrive = DETECT, gmode = VGAHI;
 int i, ch, n, x, y, j, quit = 1, out = 1;
 char press, *str;
 initgraph(&gdrive, &gmode, "\\TC\\BGI");
 while(quit != 2)
        {cleardevice();
        setcolor(YELLOW);
        setbkcolor(BLACK);
        settextstyle(1, 0, 4);
        outtextxy(195, 200, "GALACTIC INTERACTIONS SIMULATION");
        setfillstyle(SOLID_FILL, GREEN);
        if(count = = 1)
                setfillstyle(SOLID_FILL, LIGHTGREEN);
        bar(209, 400, 289, 420);
        setfillstyle(SOLID_FILL, GREEN);
        if(count = = 2)
                setfillstyle(SOLID_FILL, LIGHTGREEN);
        bar(353, 400, 433, 420);
        settextstyle(0, 0, 1);
        setcolor(BLACK);
        outtextxy(218, 410, "CONTINUE");
```

```
        outtextxy(380, 410, "EXIT");
        ch = getch();
        switch(ch)
                {case 75: if(count = = 2)
                                count = count - 1;
                        else
                                count = 2;
                break;
                case 77: if(count = = 1)
                                count = count + 1;
                        else
                                count = 1;
                break;
                case 13: delay(100);
                        quit = 2;
                        if(count = = 2)
                                out = 2;
                break;
                }
        }
closegraph();
if(out != 2)
        {textbackground(BLACK);
        clrscr();
        window(1, 1, 80, 1);
        textbackground(7);
        clrscr();
        textcolor(WHITE);
        gotoxy(2, 1);
        cputs("Enter Parameters");
        window(1, 2, 80, 25);
        textbackground(1);
```

```
clrscr();
textcolor(14);
gotoxy(2, 2);  cputs("r :");
gotoxy(40, 2); cputs("q :");
gotoxy(2, 4); cputs("M1:");
gotoxy(40, 4); cputs("M2:");
gotoxy(2, 6); cputs("i1:");
gotoxy(40, 6); cputs("i2:");
gotoxy(2, 8); cputs("\xEA"); gotoxy(3, 8); cputs("1:");
gotoxy(40, 8); cputs("\xEA"); gotoxy(41, 8); cputs("2:");
gotoxy(2, 10); cputs("e :");
gotoxy(2, 12); cputs("Size of G1:");
gotoxy(24, 12); cputs("(0 < = size < = 10)");
gotoxy(2, 14); cputs("Size of G2:");
gotoxy(24, 14); cputs("(0 < = size < = 10)");
gotoxy(2, 16); cputs("Reverse G1's Rotation(y/n):");
gotoxy(2, 18); cputs("Reverse G2's Rotation(y/n):");
window(5, 3, 14, 3);
textbackground(BLACK);
clrscr();
window(43, 3, 52, 3);
textbackground(BLACK);
clrscr();
window(5, 5, 14, 5);
textbackground(BLACK);
clrscr();
window(43, 5, 52, 5);
textbackground(BLACK);
clrscr();
window(5, 7, 14, 7);
textbackground(BLACK);
clrscr();
```

```
window(43, 7, 52, 7);
textbackground(BLACK);
clrscr();
window(5, 9, 14, 9);
textbackground(BLACK);
clrscr();
window(43, 9, 52, 9);
textbackground(BLACK);
clrscr();
window(5, 11, 14, 11);
textbackground(BLACK);
clrscr();
window(13, 13, 22, 13);
textbackground(BLACK);
clrscr();
window(13, 15, 22, 15);
textbackground(BLACK);
clrscr();
window(29, 17, 38, 17);
textbackground(BLACK);
clrscr();
window(29, 19, 38, 19);
textbackground(BLACK);
clrscr();
window(1, 2, 80, 25);
gotoxy(5, 2);
n = 1;
x = 5;
y = 2;
j = 1;
ch = getch();
if(ch = = 27)
```

```
            {ch = 13;
             n = 13;
             out = 2;
            }
    while(!((ch = = 13) && (n = = 13)))
            {switch(ch)
                    {case 83: Move(n, &x, &y);
                              cputs("          ");
                              Move(n, &x, &y);
                              for(i = 0; i < 10; i++)
                                      {press = ' ';
                                       Getpress(n, i, press);
                                      }
                     break;
                     case 8 : if(j > 1)
                                      {x = x - 1; gotoxy(x, y);
                                       cputs(" ");
                                       gotoxy(x, y);
                                       j = j - 1;
                                       press = ' ';
                                       Getpress(n, i, press);
                                      }
                     break;
                     case 13: if(n < 13)
                                      n = n + 1;
                              else
                                      n = n;
                              Move(n, &x, &y);
                              j = 1;
                     break;
                     case 77: if((n > = 1) && (n < 13))
                                      n = n + 1;
```

```
            else
                n = 1;
            Move(n, &x, &y);
            j = 1;
break;
case 75: if(( n < = 13) && (n > 1))
                n = n - 1;
            else
                n = 13;
            Move(n, &x, &y);
            j = 1;
break;
case 46: press = '.';
            Getpress(n, j - 1, press);
            cputs("."); x = x + 1; j = j + 1;
break;
case 48: press = '0';
            Getpress(n, j - 1, press);
            cputs("0"); x = x + 1; j = j + 1;
break;
case 49: press = '1';
            Getpress(n, j - 1, press);
            cputs("1"); x = x + 1; j = j + 1;
break;
case 50: press = '2';
            Getpress(n, j – 1, press);
            cputs("2"); x = x + 1; j = j + 1;
break;
case 51: press = '3';
            Getpress(n, j - 1, press);
            cputs("3"); x = x + 1; j = j + 1;
break;
```

```
case 52: press = '4';
        Getpress(n, j - 1, press);
        cputs("4"); x = x + 1; j = j + 1;
break;
case 53: press = '5';
        Getpress(n, j - 1, press);
        cputs("5"); x = x + 1; j = j + 1;
break;
case 54: press = '6';
        Getpress(n, j – 1, press);
        cputs("6"); x = x + 1; j = j + 1;
break;
case 55: press = '7';
        Getpress(n, j - 1, press);
        cputs("7"); x = x + 1; j = j + 1;
break;
case 56: press = '8';
        Getpress(n, j - 1, press);
        cputs("8"); x = x + 1; j = j + 1;
break;
case 57: press = '9';
        Getpress(n, j - 1, press);
        cputs("9"); x = x + 1; j = j + 1;
break;
case 45: press = '-';
        Getpress(n, j - 1, press);
        cputs("-"); x = x + 1; j = j + 1;
break;
case 121:press = 'y';
        Getpress(n, j - 1, press);
        cputs("y"); x = x + 1; j = j + 1;
break;
```

```
                case 110:press = 'n';
                         Getpress(n, j - 1, press);
                         cputs("n"); x = x + 1; j = j + 1;
                 break;
                } //switch
        if((j = = 11) && (n != 13))
                {n = n + 1;
                j = 1;
                Move(n, &x, &y);
                }
        if((j = = 11) && (n = = 13))
                {n = 1;
                j = 1;
                Move(n, &x, &y);
                }
        ch = getch();
        if(ch = = 27)
                {ch = 13;
                 n = 13;
                 out = 2;
                }
        } //while
str = "0000000000";
str = cr;
r = atof(str)*20.0;
str = "          "; str = cq;
q = atof(str)*20.0;
str = "          "; str = ce;
e = atof(str);
str ="          "; str = cM1;
M1 = atof(str);
str = "          "; str = cM2;
```

```
        M2 = atof(str);
        str = "        "; str = ci1;
        i1 = atof(str);
        str = "        "; str = ci2;
        i2 = atof(str);
        str = "        "; str = co1;
        o1 = atof(str);
        str = "        "; str = co2;
        o2 = atof(str);
        str = "        "; str = cN12;
        N12 = atof(str);
        str = "        "; str = cN22;
        N22 = atof(str);
        r1 = cr1[0];
        r2 = cr2[0];
        window(1, 1, 80, 25);
        textbackground(BLACK);
        clrscr();
        textcolor(WHITE);
        }   //if(out != 2)
return(out);
}



void Move(int n, int *x, int *y)
{switch(n)
        {case 1 : gotoxy(5, 2); *x = 5; *y = 2;
         break;
         case 2 : gotoxy(43, 2); *x = 43; *y = 2;
         break;
         case 3 : gotoxy(5, 4); *x = 5; *y = 4;
```

```
            break;
            case 4 : gotoxy(43, 4); *x = 43; *y = 4;
            break;
            case 5 : gotoxy(5, 6); *x = 5; *y = 6;
            break;
            case 6 : gotoxy(43, 6); *x = 43; *y = 6;
            break;
            case 7 : gotoxy(5, 8); *x = 5; *y = 8;
            break;
            case 8 : gotoxy(43, 8); *x = 43; *y = 8;
            break;
            case 9 : gotoxy(5, 10); *x = 5, *y = 10;
            break;
            case 10: gotoxy(13, 12); *x = 13; *y = 12;
            break;
            case 11: gotoxy(13, 14); *x = 13; *y = 14;
            break;
            case 12: gotoxy(29, 16); *x = 29; *y = 16;
            break;
            case 13: gotoxy(29, 18); *x = 29, *y = 18;
            break;
            }  //switch
}


void Getpress(int n, int j, char c)
{switch(n)
            {case 1 : cr[j] = c;
            break;
            case 2 : cq[j] = c;
            break;
            case 3 : cM1[j] = c;
```

```
break;
case 4 : cM2[j] = c;
break;
case 5 : ci1[j] = c;
break;
case 6 : ci2[j] = c;
break;
case 7 : co1[j] = c;
break;
case 8 : co2[j] = c;
break;
case 9 : ce[j] = c;
break;
case 10: cN12[j] = c;
break;
case 11: cN22[j] = c;
break;
case 12: cr1[j] = c;
break;
case 13: cr2[j] = c;
break;
}
}


void G_Coor(float *lxg1, float *lxg2, float *lxg3, float *lyg1, float *lyg2, float *lyg3, float *lzg1,
        float *lzg2, float *lzg3, float i, float o)
{float l = 50.0;
 *lxg1 = l*cos(o);
 *lxg2 = l*sin(o);
 *lxg3 = 0.0;
 *lyg1 = - l*sin(o)*cos(i);
```

```
*lyg2 = l*cos(o)*cos(i);
*lyg3 = l*sin(i);
*lzg1 = l*sin(o)*sin(i);
*lzg2 = - l*cos(o)*sin(i);
*lzg3 =  l*cos(i);
}


void Rotate_Gco(float *lx1, float *lx2, float *ly1, float *ly2, float *lz1, float *lz2, float ww,
                 float ii, float oo, float l1, float l2, float l3, float l4, float l5, float l6, float l7,
                 float l8, float l9)
{*lx1 =  l1*( cos(ww)*cos(oo) - sin(ww)*sin(oo)*cos(ii))
        +l2*( - sin(ww)*cos(oo) - cos(ww)*sin(oo)*cos(ii)) + l3*sin(ii)*sin(oo);
 *lx2 = l1*( cos(ww)*sin(oo) + sin(ww)*cos(oo)*cos(ii))
        +l2*( - sin(ww)*sin(oo) + cos(ww)*cos(oo)*cos(ii)) - l3*sin(ii)*cos(oo);
 *ly1 =  l4*( cos(ww)*cos(oo) - sin(ww)*sin(oo)*cos(ii))
        +l5*( - sin(ww)*cos(oo) - cos(ww)*sin(oo)*cos(ii)) + l6*sin(ii)*sin(oo);
 *ly2 =  l4*( cos(ww)*sin(oo) + sin(ww)*cos(oo)*cos(ii))
        +l5*( - sin(ww)*sin(oo) + cos(ww)*cos(oo)*cos(ii)) - l6*sin(ii)*cos(oo);
 *lz1 = l7*( cos(ww)*cos(oo) - sin(ww)*sin(oo)*cos(ii))
       +l8*( - sin(ww)*cos(oo) - cos(ww)*sin(oo)*cos(ii)) + l9*sin(ii)*sin(oo);
 *lz2 = l7*( cos(ww)*sin(oo) + sin(ww)*cos(oo)*cos(ii))
        +l8*( - sin(ww)*sin(oo) + cos(ww)*cos(oo)*cos(ii)) - l9*sin(ii)*cos(oo);
}


void Rotate_Coor(float *lx1, float *lx2, float *ly1, float *ly2, float *lz1, float *lz2, float ww,
                  float ii, float oo)
{float l = 100.0;
 *lx1 =  l*( cos(ww)*cos(oo) - sin(ww)*sin(oo)*cos(ii));
 *lx2 =  l*( cos(ww)*sin(oo) + sin(ww)*cos(oo)*cos(ii));
 *ly1 =  l*( - sin(ww)*cos(oo) - cos(ww)*sin(oo)*cos(ii));
```

```
*ly2 =  l*( - sin(ww)*sin(oo) + cos(ww)*cos(oo)*cos(ii));
*lz1 =  l*sin(ii)*sin(oo);
*lz2 = - l*sin(ii)*cos(oo);
}


void Rotate(float *x, float *y, float ww, float ii, float oo, float a, float b, float c)
{*x = a*( cos(ww)*cos(oo) - sin(ww)*sin(oo)*cos(ii))
      + b*( - sin(ww)*cos(oo) - cos(ww)*sin(oo)*cos(ii)) + c*sin(ii)*sin(oo);
 *y =  a*( cos(ww)*sin(oo) + sin(ww)*cos(oo)*cos(ii))
      +b*( - sin(ww)*sin(oo) + cos(ww)*cos(oo)*cos(ii)) - c*sin(ii)*cos(oo);
}


float rkp(float w, float M1, float M2, float q)
{float f;
 f = sqrt(M1 + M2)*(1.0 + cos(w))*(1.0 + cos(w))/sqrt(8.0*q*q*q);
 return(f);
}


float rk(float hh, float M1, float M2, float aa, float e)
{float f;
 f = sqrt(M1 + M2)*(1.0 + e*cos(hh))*(1.0 + e*cos(hh)) / (sqrt((e*e - 1.0)*(e*e - 1.0)*(e*e –
     1.0))*sqrt(aa*aa*aa));
 return(f);
}


float rkell(float hh, float M1, float M2, float aa, float e)
{float f;
 f = sqrt(M1 + M2)*(1.0 + e*cos(hh))*(1.0 + e*cos(hh)) / (sqrt((1.0 - e*e)*(1.0 - e*e)*(1.0 –
```

```
     e*e))*sqrt(aa*aa*aa));
 return(f);
}


float L(float x, float y, float c, float d, float g)
{float l, b;
 l = sqrt((c - x)*(c - x) + (d - y)*(d - y) + g*g);
 b = l*l*l;
 return(b);
}


float rs(float x, float y, float z)
{float r, l;
 r = sqrt(x*x + y*y + z*z);
 l = r*r*r;
 return(l);
}


float gx(float x, float y, float r, float c, float d, float g, float M1, float M2)
{float k;
 k = - x*M2/(r*r*r) - (c - x)*M2/L(x, y, c, d, g) - c*M1/rs(c, d, g);
 return(k);
}

float gy(float x, float y, float r, float c, float d, float g, float M1, float M2)
{float k;
 k = - y*M2/(r*r*r) - (d - y)*M2/L(x, y, c, d, g) - d*M1/rs(c, d, g);
 return(k);
}
```

```
float gz(float x, float y, float c, float d, float g, float M1, float M2)
{float k;
 k = - g*M2/L(x, y, c, d, g) - g*M1/rs(c, d, g);
 return(k);
}
```

ประวัติผู้เขียนวิทยานิพนธ์

ชื่อ นามสกุล : นาย ศราวุธ วันไชยธนวงศ์

วัน เดือน ปี และสถานที่เกิด : 4 มกราคม 2522 จังหวัดเชียงราย

การศึกษา : สำเร็จการศึกษาระดับปริญญาบัณฑิตสาขาวิชาฟิสิกส์จาก

มหาวิทยาลัยเชียงใหม่ ปีการศึกษา 2545