หนังสืออ้างอิง

Athans, M., and P. L. Falb, Optimal Control : An Introduction to the Theory and Its Application, McGraw-Hill, New York, 1964.

Athans, M., and F. C. Schweppe, "Gradient Matrices and Matrix Calculations," Technical Report TN 1965-63, Lincoln Lab., MIT., Lexington, 1965.

Athans, M., "The Matrix Minimum Principle," Inform. Contr., 11, 592-606, 1967.

Athans, M., "The Role and Use of the Stochastic Linear-Quadratic-Gaussian Problem in Control System Design," IEEE Trans. on AC., 16(6), 529-552, December 1971.

Anderson, B.D.O., and J. B. Moore, Linear Optimal Control, Prentice-Hall, New Jersey, 1971.

Bartels, R.H., and G.W. Stewart, "Solution of the Equation AX + XB = Q (Algorithm 432)," Commun. Ass. Comput. Mach., 15, 820-826, 1972.

Bengiamin, N. M., and W. C. Chan, "Multilevel Load-Frequency Control of Interconnected Power Systems," Proc. IEE, 125(6), 521-526, June 1978.

Bengtsson, G., and S. Lindahl, "A Design Scheme for Incomplete State or Output Feedback with Application to Boiler and Power System Control," Automatica, 10, 15-30, 1974.

Basar, T., "On the Relative Leadership Property of Stackelberg Strategies," JOTA., 11(6), 655-661, 1973.

Calovic, M.S., "Dynamic State-Space Models of Electric Power Systems," Departments of Electrical and Mechanical Engineering Report, University of Illinois, Urbana, 1971.

Calovic, M. S., "Linear Regulator Design for a Load and Frequency Control," IEEE Trans. on PAS., 91, 2271-2285, 1972.

Calovic, M. S., S. P. Bingulac, and N. M. Cuk, "An Output Feedback Proportional-Plus-Integral Regulator for Automatic Generation Control," IEEE PES Summer Meeting, Paper No. C73-489-2, Vancouver, Canada, July 1973.

Calovic, M. S., N. M. Cuk, and M. Djorovic, "Autonomous Area Generation Control of Interconnected Power Systems," Proc. IEE, 124(4), 393-402, April 1977.

Chen, C. I., and J. B. Cruz, Jr., "Stackelberg Strategies of Two-Person Games with Biased Information Patterns," IEEE Trans. on AC., 17, 791-798, 1972.

Chen, C. T., Linear System Theory and Design, Holt Rinehart & Winston, New York, 1984.

Choi, S. S., and H. R. Sirisena, "Computation of Optimal Output Feedback Gains for Linear Multivariable Systems," IEEE Trans. on AC., 19, 257-258, June 1974.

Cohn, N., Control of Generation and Power Flow on Interconnected Systems, John Wiley, New York, 1966.

Cruz, J. B. Jr., "Leader-Follower Strategies for Multilevel Systems," IEEE Trans. on AC., 23, 244-258, April 1978.

Davison, E. J., "The Robust Decentralized Control of General Servomechamism Problem," IEEE Trans. on AC., 21, 14-24, 1976.

Davison, E. J., and N. K. Tripathi, "The Optimal Decentralized Control of a Large Power System: Load and Frequency Control," IEEE Trans. on AC., 23, 312-325, 1978.

Elgerd, O. I., and C. E. Fosha, Jr., "Optimum Megawatt-Frequency

Control of Multiarea Electric Energy System," IEEE Trans. on PAS, 89(4), 556-563, April 1970.

Elgerd, O. I., Electric Energy Systems Theory : An Introduction, McGraw-Hill, New York, 2nd edition, 1982.

Ermer, E.M., "Output Feedback Gains for a Class of Stochastic Control Problems," Ph.D. Dissertation, The Johns Hopkins University, 1972.

Ermer, E. M., and V. D. Vandelinde, "Output Feedback Gain for a Linear Discrete Stochastic Control Problem," IEEE Trans. on AC., 18, 154-157, 1973.

Fosha, C. E. Jr., and O. I. Elgerd, "The Megawatt-Frequency Control Problem : A New Approach via Optimal Control Theory," IEEE Trans. on PAS., 89(4), 563-571, April 1970.

Gardner, B.F. Jr., and J. B. Cruz, Jr., "Feedback Stackelberg Strategies for a Two-Player Game," IEEE Trans. on AC., 22, 270-271, April 1977.

Gardner, B. F. Jr., and J. B. Cruz, Jr., "Feedback Stackelberg Strategies for M-Level Hierarchical Games," IEEE Trans. on AC., 23, 489-491, June 1978.

Glankwamdee, S., and J. B. Cruz, Jr., "Decentralized Stackelberg Strategies for Interconnected Stochastic Dynamic Systems," Proc. of 7th Triennial World Congress of IFAC. (Niemi, A., B. Wahlstrom, and J. Virkkunen, eds.), pp.1017-1023, Helsinki, Finland, 12-16 June 1978.

Glankwamdee, S., "Two-Level Coordination for Automatic Generation Control of Interconnected Power Systems," Proc. of the International Conference of Industrial Systems Engineering and Management in Developing Countries, pp. 737-744, Bangkok, Thailand, 3-6 November 1980.

Grad, J., and M. A. Brebner, "Algorithm 343 : Eigenvalues and Eigenvectors of a Real General Matrix (F2)," Commum. Ass. Comput. Mach., 11, 820-826, 1968.

Horisberger, H. P., and P. R. Belanger, "Solution of the Optimal Constant Output Feedback Problems by Conjugate Gradients," IEEE Trans. on AC., 19, 434-435, August 1974.

Jacoby, S.L.S., J. S. Kawalik, and J. T. Pizzo, Iterative Methods for Nonlinear Optimization Problems, Prentice-Hall, New Jersey, 1972.

Jamshidi, M., Large-Scale Systems : Modeling and Control, North-Holland, U.S.A., 1983.

Kavin, R. K., M.C. Budge, and P. Rasmussen, "An Optimal Linear Systems Approach to Load-Frequency Control," IEEE Trans. on PAS., 90, 2472-2482, 1971.

Kleinman, D. L., "On an Iterative Technique for Riccati Equation Computation," IEEE Trans. on AC., 13(1), 114-115, February 1968a.

Kleinman, D. L., and M. Athans, "The Design of Suboptimal Linear Time-Varying Systems," IEEE Trans. on AC., 13(2), 150-158, April 1968b.

Kleinman, D. L., T. Fortmann, and M. Athans, "On the Design of Linear Systems with Piecewise-Constant Feedback Gains," IEEE Trans. on AC., 13(4), 354-361, August 1968c.

Kleinman, D. L., "An Easy Way to Stabilize a Linear Control System," IEEE Trans. on AC., 15(5), 692, December 1970.

Knapp, C. H., and S. Basuthakur, "On Optimal Output Feedback," IEEE Trans. on AC., 17(6), 823-825, December 1972.

Kosut, R. L., "Suboptimal Control of Linear Time-Invariant Systems Subject to Control Structure Constraints," IEEE Trans. on AC., 15(6), 557-563, December 1970.

Kurtaran, B., "Optimal Instanteneous Output-Feedback Controllers for Linear Stochastic Systems," Int. J. Control, 19(4), 797-816, 1974.

Kwakernaak, H., and R. Sivan, Linear Optimal Control Systems, John Wiley, New York, 1972.

Levine, W. S., and M. Athans, "On the Determination of the Optimal Constant Output Feedback Gains for Linear Multivariable Systems," IEEE Trans. on AC., 15, 44-48, February 1970.

Levine, W. S., T. L. Johnson, and M. Athans, "Optimal Limited State Variable Feedback Controllers for Linear Systems," IEEE Trans. on AC., 16, 785-793, December 1971.

Luenburger, D. G., "Observers for Multivariable Systems," IEEE Trans. on AC., 11(2), 190-197, April 1966.

McLane, P., "Linear Optimal Stochastic Control Using Instanteneous Output Feedback," Int. J. Control, 13(2), 383-396, 1971.

Meditch, J. S., Stochastic Optimal Linear Estimation and Control, McGraw-Hill, New York, 1969.

Mendel, J. M., "A Concise Derivation of Optimal Constant Limited State Feedback Gains," IEEE Trans. on AC., 19, 447-448, 1974.

Miniesy, S. M., and E. V. Bohn, "Two-Level Control of Interconnected Power Plants," IEEE Trans. on PAS., 90, 2742-2748, 1971.

Miniesy, S. M., and E. V. Bohn, "Optimum Load-Frequency Continuous Control with Unknown Deterministic Power Demand," IEEE Trans. on PAS., 91, 1910-1915, 1972.

Quazza, G. "Noninteracting Controls of Interconnected Electric Power

Systems," IEEE Trans. on PAS., 85, 727-741, 1966.

Saksena, V. R., J. B. Cruz, Jr., W. R. Perkins, and T. Basar, "Information Induced Multimodel Solutions in Multiple Decisionmaker Problems," IEEE TRans. on AC., 28(6), 716-728, June 1983.

Sandell, N. R. Jr., P. Varaiya, M. Athans, and M. G. Safonov, "Survey of Decentralized Control Methods for Large Scale Systems," IEEE Trans. on AC., 23, 108-128, 1978.

Simaan, M. and J. B. Cruz, Jr., "Stackelberg Solution for Games with Many Players," IEEE Trans. on AC., 18, 322-324, 1973.

Singh, M., Decentralized Control, North-Holland, The Netherlands, 1981.

Smith, B. T., J. M. Boyle, J. J. Dongara, B.S. Garbow, Y. Ikebe, V. C. Klema, and C. B. Moler, Matrix Eigensystem Routines - EISPACK Guide, Springer-Verlag, Germany, 2nd edition, 1976.

Starr, A. W., "Nonzero-sum Differential Games: Concepts and Models," Technical Report No. 590, Division of Engineering and Applied Physics, Harvard University, June 1969.

Walsh, P.M., "On Symmetric Matrices and the Matrix Minimum Principle," IEEE Trans. on AC., 22 (6), 995-996, December 1977.

Walsh. P. M. and J. B. Cruz Jr., "An Example of the Impact of Information Structure on Decentralized Multicriterion Control Structure," IEEE Trans. on AC., 24, 978-800, December 1979.

Venkateswarlu, K., and A. K. Mahalanabis, "Design of Decentralized Load-Frequency Regulators," Proc. IEE, 124 (9), 817-820, September 1977.

ภาคผนวก

ภาคผนวก ก.

โมเดลระบบ LFC ของระบบไฟฟ้ากำลังสองเขตที่เชื่อมโยงกัน

โมเดลระบบ LFC ของระบบไฟฟ้ากำลังสองเขตที่ใช้ในวิทยานิพนธ์ฉบับนี้ตลอดจนในงานวิจัยของ Calovic (1972,1973,1977) แต่ละเขตประกอบด้วยหน่วยผลิตพลังน้ำหนึ่งหน่วยและหน่วยผลิตพลังไอน้ำหนึ่งหน่วย หน่วยผลิตพลังไอน้ำใช้เครื่องจักรพลังไอน้ำแบบ reheat ซึ่งแสดงไว้ในรูปที่ ก.1 กัฟเวอร์เนอร์ที่ใช้กับเครื่องจักรพลังไอน้ำดังกล่าวเป็นแบบ accelerotachometric ซึ่งแสดงไว้ในรูปที่ 2.4 ส่วนหน่วยผลิตพลังน้ำเป็นโรงจักรพลังน้ำที่มีเขื่อนกักเก็บน้ำดังแสดงไว้ในรูปที่ ก.2 กัฟเวอร์เนอร์ที่ใช้คู่กันเป็นแบบ transient speed-droop ซึ่งแสดงไว้ในรูปที่ 2.5 ดังนั้นระบบไฟฟ้ากำลังแต่ละเขตจึงเป็นโรงจักรผสมระหว่างพลังน้ำและพลังไอน้ำ เหตุผลที่เลือกโมเดลของหน่วยผลิตแต่ละหน่วยเป็นชนิดดังกล่าวคือ โรงจักรทั้งสองแบบเป็นที่นิยมใช้กันทั่วไปในระบบไฟฟ้ากำลัง รายละเอียดของโมเดลตลอดจนสมการสถานะของส่วนประกอบต่าง ๆ ในระบบไฟฟ้ากำลังอาทิเช่น กัฟเวอร์เนอร์, เทอร์ไบน์, เครื่องกำเนิดไฟฟ้า เป็นต้น ค้นหาได้จาก Calovic (1971)

ภายใต้สมมติฐานของพลวัตเนื่องจากสัญญาณขนาดเล็ก โมเดลของระบบไฟฟ้ากำลังสองเขตที่เชื่อมโยงกันด้วยเส้นเชื่อมต่อสำหรับใช้อธิบายพฤติกรรมของระบบระหว่างกำลังจริงกับความถี่สามารถถูกทำให้เป็นเชิงเส้นรอบจุดทำงานปกติ ตัวแปรของระบบต่าง ๆ ที่ใช้เป็นตัวแปรอินครีเมนทัล (incremental variable) ทั้งสิ้นและในที่นี้ตัวแปรทุกตัวใช้หน่วยเป็นเปอร์ยูนิตทั้งสิ้น โมเดลดังกล่าวเขียนได้เป็น

$$\mathring{x}(t) = Ax(t) + Bu(t) + Ez(t) \qquad (ก.1)$$

โดยที่ $x(t) = [x_1^T(t), \Delta P_{12}, x_2^T(t)]^T$ เป็นสถานะของระบบทั้งหมดมิติ 17

$x_i(t) = [\Delta a_T, \Delta p_{t1}, \Delta p_{t2}, \Delta p_{t3}, \Delta a_H, \Delta q, \Delta f]_i^T, \quad i=1,2$

เป็นสถานะของระบบไฟฟ้าแต่ละเขตมิติ 8

$\Delta P_{12}(t)$ เป็นการเบี่ยงเบนของกำลังจริงในเส้นเชื่อมต่อ

$u(t) = [\Delta f_{mT1}, \Delta f_{mH1}, \Delta f_{mT2}, \Delta f_{mH2}]^T$

เป็นเวกเตอร์ควบคุมของหน่วยผลิตทั้งหมดมิติ 4

$z(t) = [\Delta P_{L1}, \Delta P_{L2}]^T$

เป็นเวกเตอร์การเปลี่ยนแปลงของโหลดดีมานด์ทั้งหมดมิติ 2

ตัวแปรในสมการ (ก.1) เป็นตัวแปรอินครีเมนทัลซึ่งมีหน่วยเป็นเปอร์ยูนิตทั้งหมดโดยมีคำอธิบายดังในตารางที่ ก.1 ในแต่ละเขตหน่วยผลิตทั้งหมดถูกควบคุมด้วยตัวควบคุมเขตของตัวเดียวกันตามหลักการของการควบคุมร่วมกันในเขต (area-control concept) จำนวนตัวแปรควบคุมของระบบทั้งหมดในสมการ (ก.1) ลดลงมาเหลือเท่ากับจำนวนของเขตทั้งหมด ตัวควบคุมของเขตอธิบายได้ด้วยสมการ

$$u(t) = \nu w(t) \qquad (ก.2)$$

โดยที่ $w(t) = [\ w_1(t), w_2(t)\ ]^T$ เป็นเวกเตอร์ควบคุมของเขตทั้งหมด มิติ 2

ν เป็นเมตริกซ์ของการปันส่วนในการควบคุม มิติ 4 x 2

เมื่อในระบบมีการใส่อินทิกรัลของ ACE สถานะของระบบที่เพิ่มขึ้นได้แก่

$$\dot{v}(t) = ACE = Dx(t) \qquad (ก.3)$$

โดยที่ $v(t) = [\ v_1, v_2\ ]^T$ เป็นเวกเตอร์อินทิกรัลของ ACE มิติ 2

ระบบไฟฟ้ากำลังแต่ละเขตมีการวัดดังนี้

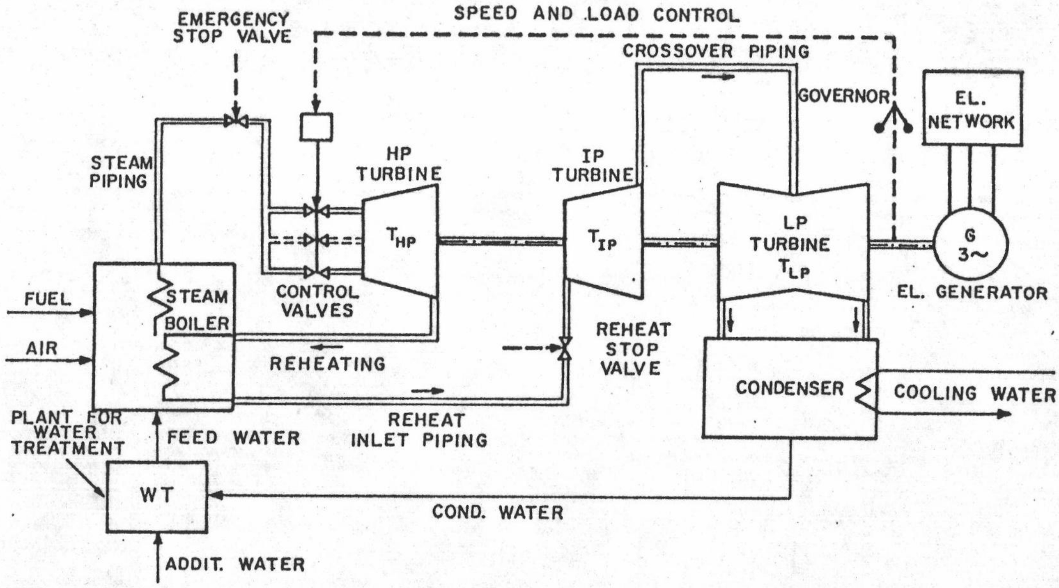
$$y^i = C^i x \qquad (ก.4)$$

โดยที่ $y^i = [\Delta P_{Ti}, \Delta P_{Hi}, \Delta f_i, \Delta P_{12}, v_i]^T$

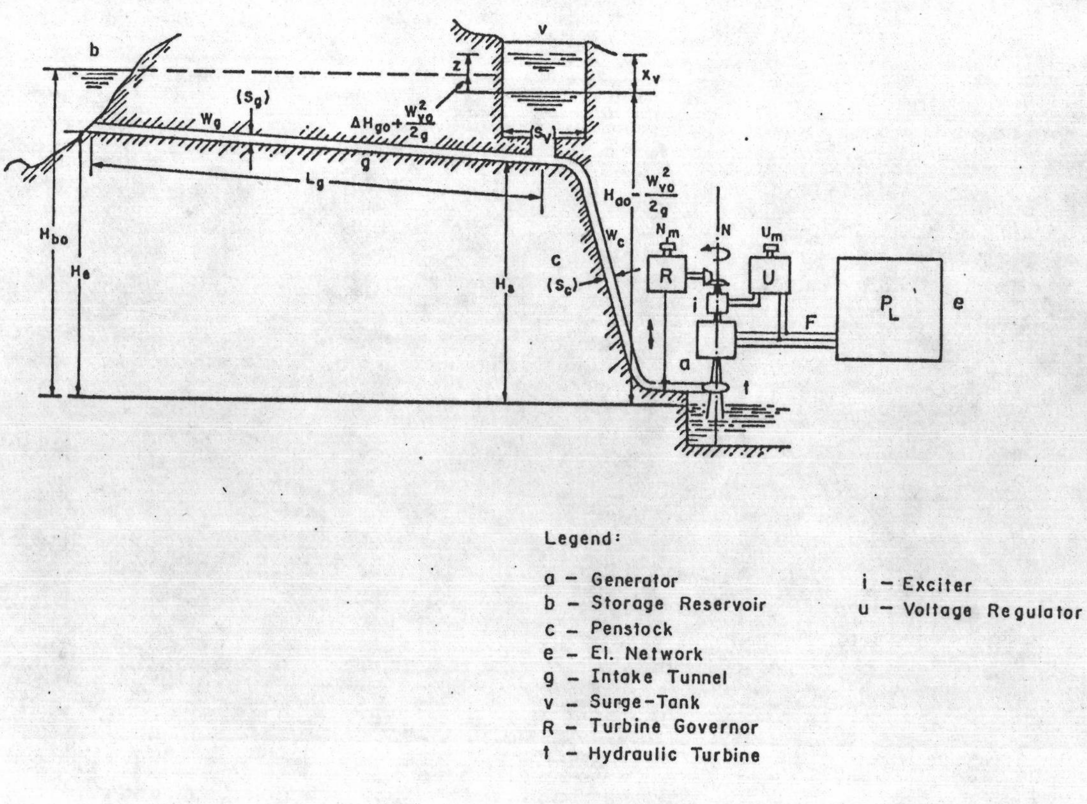
P_{Ti} เป็นอินครีเมนทัลหรือการเบี่ยงเบนของกำลังขาออกของเทอร์ไบน์พลังไอน้ำในเขตที่ i

P_{Hi} เป็นอินครีเมนทัลหรือการเบี่ยงเบนของกำลังขาออกของเทอร์ไบน์พลังน้ำในเขตที่ i

เมตริกซ์ต่าง ๆ ในสมการ (ก.1)-(ก.4) ตลอดจนค่าตัวเลขต่าง ๆ แสดงไว้ในรูปที่ ก.3



รูปที่ ก.1 โรงจักรพลังไอน้ำแบบ reheat



รูปที่ ก.2 โรงจักรพลังน้ำแบบเขื่อนกักเก็บน้ำ

$$A = \begin{bmatrix} A_1 & a_{12} & 0 \\ m & 0 & -m \\ 0 & -\alpha a_{21} & A_2 \end{bmatrix}, \quad B = \begin{bmatrix} B_1 & 0 \\ 0 & 0 \\ 0 & B_2 \end{bmatrix},$$

$$E = \begin{bmatrix} E_1 & 0 \\ 0 & 0 \\ 0 & E_2 \end{bmatrix}, \quad D = \begin{bmatrix} D_1 & 1 & 0 \\ 0 & -\alpha & D_2 \end{bmatrix},$$

$$C^1 = \begin{bmatrix} C & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}, \quad C^2 = \begin{bmatrix} C & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

$$A_1 = \begin{bmatrix} -0.2 & & & & & & & \\ 4.75 & -5.0 & & & & & & -4.0 \\ & 0.1667 & -0.1667 & & & & & \\ & & 2.0 & -2.0 & & & & \\ & -0.08 & -0.0747 & -0.112 & -3.9944 & 10.0 & -0.928 & -9.1011 \\ & & & & 0.2 & -0.50 & & \\ & & & & 1.3194 & & -1.3889 & -0.2778 \\ & 0.01 & 0.0093 & 0.014 & -0.0632 & & 0.1160 & -0.1124 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} -0.2 & & & & & & & \\ 4.75 & -5.0 & & & & & & -4.0 \\ & 0.1667 & -0.1667 & & & & & \\ & & 2.0 & -2.0 & & & & \\ & -0.10 & -0.0933 & -0.14 & -4.096 & 10.0 & -0.7422 & -9.1079 \\ & & & & 0.2 & -0.50 & & \\ & & & & 1.3194 & & -1.3889 & -0.2778 \\ & 0.0125 & 0.0117 & 0.0175 & -0.0506 & & 0.0928 & -0.1115 \end{bmatrix}$$

$$B_1 = B_2 = \begin{bmatrix} 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 10 & 0 & 0 & 0 \end{bmatrix}^T$$

$$C = \begin{bmatrix} 0 & 0.30 & 0.28 & 0.42 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1.52 & 0 & 2.78 & 0.217 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$E_1 = E_2 = [\,0 \quad 0 \quad 0 \quad 0 \quad 0.6667 \quad 0 \quad 0 \quad -.0833\,]^T$$

$$D_1 = D_2 = [\,0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 10\,]$$

$$m = [\,0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 22.2144\,]$$

$$\sim\ = \begin{bmatrix} 0.4 & 0.6 & 0 & 0 \\ 0 & 0 & 0.5 & 0.5 \end{bmatrix}^T$$

$$a_{21} = a_{12} = E_1 = E_2, \quad \alpha = 1.0$$

รูปที่ ก.3 แสดงค่าของเมตริกซ์ต่าง ๆ ของโมเดลระบบ LFC

symbol	description
Δa_T	the incremental steam turbine valve opening in p.u.
Δp_{t1}	the incremental high-pressure output of steam turbine in p.u.
Δp_{t2}	the incremental intermediate-pressure output of steam turbine in p.u.
Δp_{t3}	the incremental low-pressure output of steam turbine in p.u.
Δa_H	the incremental hydroturbine gate opening in p.u.
Δv	the incremental dashpot piston position realative to the lever of permanent speed droop of hydroturbine governor in p.u.
Δq	the incremental hydroturbine flow in p.u.
Δf	the frequency deviation in p.u.
ΔP_{12}	the incremental tie-line power flow in p.u.
Δf_{mT}	the incremental steam unit control input in p.u.
Δf_{mH}	the incremental hydrounit control input in p.u.
ΔP_L	the incremental area-load demand or area-load disturbance in p.u

ตารางที่ ก.1 อธิบายสัญญลักษณ์ของตัวแปรต่าง ๆ ในโมเดลของระบบ LFC

ภาคผนวก ข

เอกลักษณ์สำคัญที่เกี่ยวกับผลบวกเฉียงของเมตริกซ์

จากในบทที่ 2 และบทที่ 5 ของวิทยานิพนธ์ฉบับนี้ เห็นได้ว่า การหาเงื่อนไขจำเป็นสำหรับคำตอบของการป้อนกลับสัญญาณออกแบบเล็งเลิศ และแบบ Stackelberg เกี่ยวกับการหาอนุพันธ์ (derivative) ของฟังก์ชันจำนวนจริงเทียบกับเมตริกซ์ตลอดเวลา และฟังก์ชันจำนวนจริงดังกล่าวมีรูปแบบเฉพาะ คือ อยู่ในรูปผลบวกเฉียง (trace) ของฟังก์ชันเมตริกซ์ ดังนั้นในที่นี้จะแสดงเอกลักษณ์ที่จำเป็นเกี่ยวกับผลบวกเฉียงของเมตริกซ์ และอนุพันธ์ของฟังก์ชันดังกล่าวเทียบกับเมตริกซ์ สำหรับรายละเอียดและขั้นตอนการพิสูจน์หาได้ใน Athans and Schweppe (1965), Athans (1967), Ermer (1972) และ Walsh (1977)

กำหนดให้ $f = f(X)$ และ $g = g(X)$ เป็นฟังก์ชันค่าเมตริกซ์จัตุรัส ดังนั้น $tr\,[f(X)]$ จึงเป็นจำนวนจริง X และ Y เป็นเมตริกซ์จำนวนจริงมิติ $n \times m$ และ $m \times n$ ตามลำดับ N และ L เป็นเมตริกซ์จำนวนจริงมิติ $n \times n$ และ $m \times m$ ตามลำดับ เอกลักษณ์ดังกล่าวได้แก่

1) $tr\,(N) = tr\,(N^T)$

2) $tr\,(NL) = tr\,(LN)$

3) $\frac{d}{dX} tr\, f(X) = \left[\frac{d}{dx_{ij}} tr\, f(X) \right]_{n \times m}$

4) $\frac{d}{dX} tr\,(NX) = N^T$

5) $\frac{d}{dX} tr\,(NX^T) = N$

6) $\frac{d}{dX} tr\,(NXL) = N^T L^T$

7) $\frac{d}{dX} tr\,(NXLX^T) = N^T X L^T + NXL$

8) $\frac{d}{dX} tr\,[f(X)g(X)] = \frac{d}{dX} tr\,[f(X)g^c(X)] + \frac{d}{dX} tr\,[f^c(X)g(X)]$

where the superscript "c" means that the term is held constant and does not undergo the differentiation operation.

ภาคผนวก ค

โปรแกรมที่ใช้

User: EEEB815818 -at PRO

<STUD>THESIS>EEEB815818>LINPAKDIR>STKOUT.F77

```
********************************************************************************
********************************************************************************

WWWWWW WWWWWW WWWWWW WWWWW   WWWW     W   WWWWWW  WWWW     W    WWWW
W      W      W      W    W W    W   WW   W      W    W   WW   W    W
W      W      W      W    W W    W    W   WWWWW  W    W    W   W    W
WWWWW  WWWWW  WWWWW  WWWWW   WWWW     W         W  WWWW     W    WWWW
W      W      W      W    W W    W    W         W W    W    W   W    W
W      W      W      W    W W    W    W   W    W W    W    W   W    W
WWWWWW WWWWWW WWWWWW WWWWW   WWWW    WWW   WWWW   WWWW    WWW   WWWW


 WWWW  WWWWWW W    W  WWWW  W    W WWWWWW        WWWWWW WWWWWW WWWWWW
W    W   W    W   W  W    W W    W   W           W           W      W
W        W    W W    W    W W    W   W           W         W      W
 WWWW    W    WW     W    W W    W   W           WWWWW    W      W
     W   W    W W    W    W W    W   W           W       W      W
W    W   W    W   W  W    W W    W   W      WW   W      W      W
 WWWW    W    W    W  WWWW   WWWW    W      WW   W      W      W

********************************************************************************
********************************************************************************
```

Label: PRT002 -form

Pathname: <STUD>THESIS>EEEB815818>LINPAKDIR>STKOUT.F77
File last modified: 88-04-12.18:58:12.Tue

Spooled: 88-04-12.18:58:24.Tue [Spooler rev 19.4]
Started: 88-04-12.18:59:40.Tue on: PRO by: PRO

```
C      PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG, 12 MARCH 1988.


C      PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG, 12 MARCH 1988.
C
C-----------------------------------------------------------------------
C                          M.Eng THESIS TITLED
C             "DECENTRALIZED LOAD-FREQUENCY CONTROL OF 2-AREA
C                  POWER SYSTEMS VIA OUTPUT FEEDBACK"
C-----------------------------------------------------------------------
       PROGRAM MAIN
       IMPLICIT INTEGER*2 (I-N)
       REAL*4  A(17,17),B(17,4),F(17,2),D(2,17),KP(2,17),KI(2,2),
     &         R(2,2),NU(4,2),A1(19,19),B1(19,2),
     &         BNU(17,2),F1(19,2),XO(19),
     &         DT
       REAL*4  AC(19,19),GRAD1(5),GRAD2(5),FF1(5),FF2(5),C1(5,19)
     &         ,C2(5,19)
       REAL*4  U1(361),U2(361),U3(361),RR(19),RI(19)
C       REAL*4  U1(361),U2(361),U3(361),RR(19),RI(19),Q(19,19)
       INTEGER*2  N,L,M,K,IND,IPOINT
       INTEGER*4  INN
       REAL*8   EPS,DFX
       REAL*8  DA(19,19),DB(19,2),DC1(5,19),DC2(5,19),R1,R2
C       REAL*8  DA(19,19),DB(19,2),DC1(5,19),DC2(5,19),DQ(19,19),R1,R2
       REAL*8  DAC(19,19),KO(19,19),K1(19,19),L1(19,19),DF1(5),DF2(5)
       REAL*8  DGRAD1(5),DGRAD2(5),J1,J2,Q1(19,19),Q2(19,19)
       REAL*8  C13(3,19),C23(3,19)
       INTEGER*2  LIMIT,IN,IM1,IM2,IR1,IR2,IR13,IR23
C
       IND = 1
       N=17
       M=4
       K=2
       L=2
C
       IN  = 19
       IM1 = 1
       IM2 = 1
       IR13 = 3
       IR23 = 3
       IR1 = 5
       IR2 = 5
C*************************************************************
       CALL DEFINE(A,B,D,F,NU,Q,R,N,L,M,K)
C       CALL PRINTMAT(A,17,17)
C       CALL PRINTMAT(B,17,4)
C       CALL PRINTMAT(D,2,17)
C       CALL PRINTMAT(F,17,2)
C       CALL PRINTMAT(Q,19,19)
C       CALL PRINTMAT(R,2,2)
C       CALL PRINTMAT(NU,4,2)
C
C      CHECK EIGENVALUES OF A
C
       CALL EGVP(A,17,RR,RI,IND)
       WRITE(*,*)' EIGEN OF A'
       CALL PRINTMAT(RR,17,1)
       CALL PRINTMAT(RI,17,1)

       CALL GMPRD(B,NU,BNU,17,4,2)            /*    FIND BNU    */
       CALL SCLA(U1,0.0,17,2,0)               /*    U1 = [0]17X2   */
       CALL SCLA(U2,0.0, 2,2,0)               /*    U2 = [0] 2X2   */
```

```
C      PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG,  12 MARCH 1988.


       CALL JOIN(A,U1,D,U2,N,N,L,L,A1)        /*   FORM A1     */
       CALL RTIE(BNU,U2,B1,N,L,O,O,L)         /*   FORM B1     */
C       CALL PRINTMAT(A1,19,19)
C       CALL PRINTMAT(B1,19,2 )
C       CALL PRINTMAT(BNU,17,2)

       CALL DEFINE_H(C1,C2,IR1,IR2)
       CALL DFINEC3( C13,C23,IR13,IR23 )
       WRITE(*,*)' A-closed loop = A - B1F1C1 - B2F2C2'
       WRITE(*,*)' INPUT F1 = ....'
       READ(*,*)( FF1(JJ),JJ=1,IR1)
       WRITE(*,*)' INPUT F2 = ...'
       READ(*,*)( FF2(JJ),JJ=1,IR2)
       WRITE(*,*)' AREA#1 OR AREA#2 IS THE LEADER ? CHOOSE (1) OR (2)'
       READ(*,*)  ICHOOSE

C      CHECK INITIAL GUESS
       IF ( ICHOOSE.EQ.1) THEN
C
          CALL GMPRD(FF1,C1,U1,IM1,IR1,IN)
          CALL GMPRD(B1,U1,U2,IN,IM1,IN)
          CALL GMSUB(A1,U2,AC,IN,IN)
          CALL SINGTRAN(C23,U3,IR23,IN)
          CALL GMPRD(FF2,U3,U1,IM2,IR23,IN)
          CALL GMPRD(B1(1,2),U1,U2,IN,IM2,IN)
          CALL GMSUB(AC,U2,AC,IN,IN)
C
       ELSE
C
          CALL SINGTRAN(C13,U3,IR13,IN)
          CALL GMPRD(FF1,U3,U1,IM1,IR13,IN)
          CALL GMPRD(B1,U1,U2,IN,IM1,IN)
          CALL GMSUB(A1,U2,AC,IN,IN)
          CALL GMPRD(FF2,C2,U1,IM2,IR2,IN)
          CALL GMPRD(B1(1,2),U1,U2,IN,IM2,IN)
          CALL GMSUB(AC,U2,AC,IN,IN)
C
       END IF
C
       CALL EGVP(AC,19,RR,RI,IND)
       DO 100 II=1,IN
          WRITE(*,101)RR(II),RI(II)
101       FORMAT(2F15.6)
100    CONTINUE

       WRITE(*,*)' WAITING !! '
       EPS = 1D-4
       LIMIT = 200
       CALL DBLETRAN( A1,DA,19,19 )
       CALL DBLETRAN( B1,DB,19,2 )
       CALL DBLETRAN( C1,DC1,IR1,IN )
       CALL DBLETRAN( C2,DC2,IR2,IN )
C       CALL DBLETRAN( Q,DQ,19,19 )
       CALL DBLETRAN( FF1,DF1,IM1,IR1 )
       CALL DBLETRAN( FF2,DF2,IM2,IR2 )
       R1 = 1.0D0
       R2 = 1.0D0
       IF ( ICHOOSE.EQ.1 ) THEN
             CALL DEFINEQ(Q1,Q2)
         ELSE
```

```
C     PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG,  12 MARCH 1988.


              CALL DEFINQQ(Q1,Q2)
          END IF

      WRITE(*,*)'Q1 = ',Q1(8,8),Q1(9,9),Q1(17,17),Q1(18,18),Q1(19,19)
      WRITE(*,*)'Q2 = ',Q2(8,8),Q2(9,9),Q2(17,17),Q2(18,18),Q2(19,19)
      WRITE(*,*)' START RUN..'
C
              IF  ( ICHOOSE.EQ.1 ) THEN
C
C     AREA # 1 IS THE LEADER AND RECEIVES MORE INFORMATION
      CALL STKOUT(DA,DB,DB(1,2),DC1,C23,Q1,Q2,R1,R2,DAC,KO,K1,L1,DF1,
     &     DF2,DGRAD1,DGRAD2,J1,J2,EPS,LIMIT,IN,IM1,IM2,IR1,IR23 )
C
              ELSE
C
C     AREA # 2 IS THE LEADER AND RECEIVES MORE INFORMATION
      CALL STKOUT(DA,DB(1,2),DB,DC2,C13,Q2,Q1,R2,R1,DAC,KO,K1,L1,DF2,
     &     DF1,DGRAD2,DGRAD1,J2,J1,EPS,LIMIT,IN,IM2,IM1,IR2,IR13 )
C
              END IF
C
          IF  ( ICHOOSE.EQ.1 ) THEN
C
      WRITE(*,*)' AREA # 1 IS THE LEADER AND RECEIVES MORE INFORMATION'
C
          ELSE
C
      WRITE(*,*)' AREA # 2 IS THE LEADER AND RECEIVES MORE INFORMATION'
C
          END IF
C
      WRITE(*,*)' KO = '
      CALL DPRINTMAT(KO,19,19)
      WRITE(*,*)' K1 = '
      CALL DPRINTMAT(K1,19,19)
      WRITE(*,*)' L1 = '
      CALL DPRINTMAT( L1,19,19 )
      WRITE(*,*)' J2 = ',J2
      WRITE(*,*)' J1 = ',J1
      WRITE(*,*)' GRAD1 ='
      CALL DPRINTMAT( DGRAD1,1,IR1 )
      WRITE(*,*)' GRAD2 ='
      CALL DPRINTMAT( DGRAD2,1,IR2 )
      WRITE(*,*)' F1 = '
      CALL DPRINTMAT( DF1,1,IR1 )
      WRITE(*,*)' F2 = '
      CALL DPRINTMAT( DF2,1,IR2 )
C
      CALL SINGTRAN( DAC,AC,19,19 )
      CALL SINGTRAN( DF1,FF1,IM1,IR1 )
      CALL SINGTRAN( DF2,FF2,IM2,IR2 )
      CALL SINGTRAN( DGRAD1,GRAD1,IM1,IR1 )
      CALL SINGTRAN( DGRAD2,GRAD2,IM2,IR2 )
      CALL EGVP(AC,19,RR,RI,IND)
      DO 200 II=1,IN
          WRITE(*,201)RR(II),RI(II)
201       FORMAT(2E17.7)
200   CONTINUE
C
      IF ( ICHOOSE.EQ.1) THEN
```

```
C      PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG,  12 MARCH 1988.


C
          CALL GMPRD(FF1,C1,U1,IM1,IR1,IN)
          CALL GMPRD(B1,U1,U2,IN,IM1,IN)
          CALL GMSUB(A1,U2,AC,IN,IN)
C          CALL SINGTRAN(C23,U3,IR23,IN)
          CALL GMPRD(FF2,U3,U1,IM2,IR23,IN)
          CALL GMPRD(B1(1,2),U1,U2,IN,IM2,IN)
          CALL GMSUB(AC,U2,AC,IN,IN)
C
       ELSE
C
C          CALL SINGTRAN(C13,U3,IR13,IN)
          CALL GMPRD(FF1,U3,U1,IM1,IR13,IN)
          CALL GMPRD(B1,U1,U2,IN,IM1,IN)
          CALL GMSUB(A1,U2,AC,IN,IN)
          CALL GMPRD(FF2,C2,U1,IM2,IR2,IN)
          CALL GMPRD(B1(1,2),U1,U2,IN,IM2,IN)
          CALL GMSUB(AC,U2,AC,IN,IN)
C
       END IF
       CALL EGVP( AC,19,RR,RI,IND )
       WRITE(*,*)' EIGENVALUES OF A -B1F1C1 - B2F2C2'
       DO 300 II=1,IN
          WRITE(*,301)RR(II),RI(II)
301       FORMAT(2E17.7)
300    CONTINUE


       STOP
       END

       SUBROUTINE PRINTMAT(A,IROW,ICOL)
       IMPLICIT INTEGER*2 (I-N)
       REAL*4 A(1)
       INTEGER IROW,ICOL,I,J
C
C      CAN BE USED UP TO ICOL=20
C
       DO 10 I=1,IROW
          WRITE(*,100) ( A(I+(J-1)*IROW) , J=1,ICOL )
100       FORMAT(20E14.5)
10     CONTINUE
       WRITE(*,*)
       RETURN
       END

       SUBROUTINE DPRINTMAT(A,IROW,ICOL)
       IMPLICIT INTEGER*2 (I-N)
       REAL*8 A(1)
       INTEGER IROW,ICOL,I,J
C
C      CAN BE USED UP TO ICOL=20
C
       DO 10 I=1,IROW
          WRITE(*,100) ( A(I+(J-1)*IROW) , J=1,ICOL )
100       FORMAT(20E14.5)
10     CONTINUE
       WRITE(*,*)
       RETURN
       END
```

```
C       PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG, 12 MARCH 1988.


        SUBROUTINE DEFINEQ ( Q1,Q2 )
        REAL*8      Q1(1),Q2(1)
        CALL DSCLA(Q1,ODO,19,19,0)
        CALL DSCLA(Q2,ODO,19,19,0)
        Q1(141)    = 1DO      /* F1    */
        Q1(161)    = 0.5DO      /* Ptie */
        Q1(341) = 1DO         /* V1    */
        Q2(161)    = 0.5DO      /* Ptie */
        Q2(321) = 1DO         /* F2    */
        Q2(361) = 1DO         /* V2    */
        RETURN
        END

        SUBROUTINE DEFINQQ ( Q1,Q2 )
        REAL*8      Q1(1),Q2(1)
        CALL DSCLA(Q1,ODO,19,19,0)
        CALL DSCLA(Q2,ODO,19,19,0)
        Q1(141)    = 1DO      /* F1    */
        Q1(161)    = 0.5DO      /* Ptie */
        Q1(341) = 1DO         /* V1    */
        Q2(161)    = 0.5DO      /* Ptie */
        Q2(321) = 1DO         /* F2    */
        Q2(361) = 1DO         /* V2    */
        RETURN
        END

        SUBROUTINE DEFINE(A,B,D,F,NU,Q,R,N,L,M,K)
        IMPLICIT INTEGER*2 (I-N)
        REAL*4  A(1),B(1),D(1),F(1),NU(1),Q(1),R(1),QQ
        INTEGER*2  N,L,M,K
        CALL SCLA(A ,0.0,N     ,N     ,0)
        CALL SCLA(B ,0.0,N     ,M     ,0)
        CALL SCLA(D ,0.0,L     ,N     ,0)
        CALL SCLA(F ,0.0,N     ,K     ,0)
        CALL SCLA(NU,0.0,M     ,L     ,0)
        CALL SCLA(Q ,0.0,(N+L),(N+L),0)
        CALL SCLA(R ,0.0,L     ,L     ,0)
C
C       SET R
C
        R(1) = 1
        R(4) = 1
C
C       SET Q
C
        QQ = 1.0
        Q(141) = QQ
        Q(161) = QQ
        Q(321) = QQ
        Q(341) = QQ
        Q(361) = QQ
C
C       SET NU
C
        NU(1) = 0.4
        NU(2) = 0.6
        NU(7) = 0.5
        NU(8) = 0.5
C
```

```
C       PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG, 12 MARCH 1988.


C       SET F
C
        F(5) =   0.666667
        F(8) =  -0.083333
        F(31)=   0.666667
        F(34)=  -0.083333
C
C       SET D
C
        D(15) = 10.0
        D(17) = 1.0
        D(18) = -1.0
        D(34) = 10.0
C
C       SET B
C
        B(1)  = 4.0
        B(22) = 10.0
        B(44) = 4.0
        B(65) = 10.0
C
C       SET A
C
        A(1)  = -0.2
        A(2)  =  4.75

        A(19) = -5.0
        A(20) =  0.166667
        A(22) = -0.08
        A(25) =  0.01

        A(37) = -0.166667
        A(38) =  2.0
        A(39) = -0.07466
        A(42) =  0.0093333

        A(55) = -2.0
        A(56) = -0.112
        A(59) =  0.014

        A(73) = -3.9944
        A(74) =  0.2
        A(75) =  1.31944
        A(76) = -0.063194

        A(90) =  10.0
        A(91) = -0.5

        A(107)= -0.92778
        A(109)= -1.38889
        A(110)=  0.115972

        A(120)= -4.0
        A(124)= -9.10111
        A(126)= -0.27778
        A(127)= -0.112361
        A(128)= 22.21439

        A(141)=  0.666667
        A(144)= -0.083333
```

```
C       PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG, 12 MARCH 1988.

        A(150)= -0.666667
        A(153)=  0.083333

        A(163)= -0.2
        A(164)=  4.75

        A(181)= -5.0
        A(182)=  0.166667
        A(184)= -0.1
        A(187)=  0.0125

        A(199)= -0.166667
        A(200)=  2.0
        A(201)= -0.093334
        A(204)=  0.0116625

        A(217)= -2.0
        A(218)= -0.14
        A(221)=  0.0175

        A(235)= -4.09552
        A(236)=  0.2
        A(237)=  1.31944
        A(238)= -0.0505555

        A(252)= 10.0
        A(253)= -0.5

        A(269)= -0.7422224
        A(271)= -1.38889
        A(272)=  0.092776

        A(281)= -22.21439
        A(282)= -4.0
        A(286)= -9.107862
        A(288)= -0.27778
        A(289)= -0.1115167

        RETURN
        END

        SUBROUTINE DEFINE_H(H1,H2,IR1,IR2)
        IMPLICIT INTEGER*2 (I-N)
        REAL*4  H1(5,19),H2(5,19)
        INTEGER*2 IR1,IR2
        CALL SCLA( H1,0.0,IR1,19,0 )
        H1(1,2) = 0.3
        H1(1,3) = 0.28
        H1(1,4) = 0.42
        H1(2,5) = -1.52
        H1(2,7) = 2.78
        H1(2,8) = 0.217
        H1(3,8) = 1.0
         H1(4,9) = 1.0
        H1(5,18) = 1.0
C
        CALL SCLA( H2,0.0,IR2,19,0 )
        H2(1,11) = 0.3
        H2(1,12) = 0.28
        H2(1,13) = 0.42
```

```
C      PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG, 12 MARCH 1988.


      H2(2,14) = -1.52
      H2(2,16) = 2.78
      H2(2,17) = 0.217
      H2(3,17) = 1.0
      H2(4,9)  = 1.0
      H2(5,19) = 1.0
      RETURN
      End

      SUBROUTINE DFINEC3(H1,H2,IR13,IR23)
      REAL*8  H1(3,19),H2(3,19)
      INTEGER*2 IR1,IR2
      CALL DSCLA( H1,0.0D0,IR13,19,0 )
      H1(1,8) = 1.0D0
      H1(2,9) = 1.0D0
      H1(3,18) = 1.0D0
C
      CALL DSCLA( H2,0.0D0,IR23,19,0 )
      H2(1,17) = 1.0D0
      H2(2,9)  = 1.0D0
      H2(3,19) = 1.0D0
      RETURN
      End

      SUBROUTINE EGVP(A,N,RR,RI,IND)
      IMPLICIT INTEGER*2 (I-N)
C        ****** OK. TEST ON 18 SEP. 1986 ******
      DIMENSION A(1),RR(1),RI(1),C(400)
      DIMENSION L1(20)
      CALL MCPY(A,C,N,N,0)
      CALL HSBG(N,C,N)
      CALL ATEIG(N,C,RR,RI,L1,N)
      RETURN
      END

      SUBROUTINE JOIN(A11,A12,A21,A22,N1,M1,N2,M2,A)
      IMPLICIT INTEGER*2 (I-N)
      REAL*4  A11(1),A12(1),A21(1),A22(1),A(1)
      INTEGER*2  N1,M1,N2,M2,N,M,IA
      N = N1+N2
      M = M1+M2
      IA = N*M1+1
      CALL RTIE(A11,A21,A     ,N1,M1,0,0,N2)
      CALL RTIE(A12,A22,A(IA),N1,M2,0,0,N2)
      RETURN
      END

      SUBROUTINE RCUT1(A,L,R,S,N,M,MS)
      IMPLICIT INTEGER*2 (I-N)
      REAL*4 A(1),R(1),S(1)
      INTEGER*2 L,N,M,MS,NL
      NL = N-L
      DO 10 I=1,M
         DO 20 J=1,L
            R( J+(I-1)*L ) = A( J+(I-1)*N )
20       CONTINUE
         DO 30 J=1,NL
            S( J+(I-1)*NL ) = A( J+(I-1)*N+L )
30       CONTINUE
10    CONTINUE
```

```
C     PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG,  12 MARCH 1988.


      RETURN
      END

      SUBROUTINE DJOIN(A11,A12,A21,A22,N1,M1,N2,M2,A)
      IMPLICIT INTEGER*2 (I-N)
      REAL*4   A11(1),A12(1),A21(1),A22(1),A(1)
      INTEGER  N1,M1,N2,M2,N,M,IA
      N = N1+N2
      M = M1+M2
      IA = N*M1+1
      CALL RCUT1(A    ,N1,A11,A21,N,M1,0)
      CALL RCUT1(A(IA),N1,A12,A22,N,M2,0)
      RETURN
      END

      REAL*8 FUNCTION TR(K,N)
      IMPLICIT INTEGER*2 (I-N)
      REAL*8      K(1)
      INTEGER*2  N,I,J
      TR = K(1)
      J = N+1
      DO 10 I=1,(N-1)
         J = J+1
         TR = TR + K(J)
         J = J+N
10    CONTINUE
      RETURN
      END

      SUBROUTINE DBLETRAN( A,DA,IR,IC )
      IMPLICIT INTEGER*2 (I-N)
      REAL*4    A(1)
      REAL*8    DA(1)
      INTEGER  IR,IC,I,J
      I = IR*IC
      DO 10 J=1,I
10       DA(J) = DBLE(A(J))
      RETURN
      END

      SUBROUTINE SINGTRAN( DA,A,IR,IC )
      IMPLICIT INTEGER*2 (I-N)
      REAL*4    A(1)
      REAL*8    DA(1)
      INTEGER  IR,IC,I,J
      I = IR*IC
      DO 10 J=1,I
10       A(J) = SNGL(DA(J))
      RETURN
      End

      SUBROUTINE BTLSTW (A,X,Q,H,Y,U,N,IOPTION,IERR)
      IMPLICIT INTEGER*2 (I-N)
      IMPLICIT REAL*8 (A-H,O-Z)
      INTEGER*2  N,IOPTION,IERR
      REAL*8      A(N,N),X(N,N),Q(N,N),H(N,N),Y(N,N),U(N,N)
C      REAL*8      TIME1,TIME2,TTT,CTIM$A
C
C     =======================
C     SOLVE LYAPUNOV EQUATION     AT*X + X*A + Q = 0  ( IOPTION = 0 )
```

```
C     PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG, 12 MARCH 1988.


C     ======================
C                                     A*X + X*AT + I = O  ( IOPTION = 1 )
C                                     A*X + X*AT + Q = O  ( IOPTION = 2 )
C
C     IF IERR = O   , IT DENOTES A IS STABLE   AND X WILL     BE COMPUTED
C     IF IERR = 999 , IT DENOTES A IS UNSTABLE AND X WILL NOT BE COMPUTED
C                     ALTHOUGH THIS ROUTINE CAN SOLVE IT.
C
C     A IS NOT DESTROYED ON RETURN.
C     A, P, Q, H, HT, Y, U ARE ALL ( NXN ) REAL*8 MATRIX.
C
      REAL*8     HNORM, EPS, AUX1(20), RR(20), RI(20), A1(16), B1(4), R1(64),
     &           T11, T12, T21, T22, H11, H12, H21, H22, R11, R12, R21, R22
      INTEGER*2  ID(20), K, COLUMN, ICOUNT, IS1, IS2, IROW, ICOL, KS, N4, M,
     &           IR1, IC1
C
C      TIME1 = CTIM$A(TTT)

      EPS = 1D-10
      N4 = 4
      IF (IOPTION. EQ. 0) THEN
                     CALL DMCPY(A, H, N, N, 0)
            ELSE
                     CALL DGMTRA(A, U, N, N)
                     CALL DMCPY(U, H, N, N, 0)
            END IF
C
      IS1 = 1
      IS2 = N
      CALL ORTHES(N, N, IS1, IS2, H, AUX1)
      CALL ORTRAN(N, N, IS1, IS2, H, AUX1, U)
      CALL HQRORT(N, N, IS1, IS2, H, RR, RI, U, IERR)
C
      DO 10 K=1, N
         IF (RR(K). GT. -1D-6) GO TO 1000
 10   CONTINUE
C
C     NOW U CAN TRANSFORM (A)T INTO LOWER TRIANGULAR MATRIX
C

 555  CONTINUE
      IF ( IOPTION. EQ. 0 ) THEN
                     CALL DGTPRD(A, U, Y, N, N, N)
            ELSE
                     CALL DGMPRD(A, U, Y, N, N, N)
            END IF
      CALL DGTPRD(U, Y, H, N, N, N)
C
C     NOW H IS A LOWER GUASI-TRIANGULAR MATRIX.
C     THEN COMPUTE  T = UT*Q*U
C
      IF ( IOPTION. EQ. 1 ) GO TO 19
      CALL DGMPRD(Q, U, X, N, N, N)
      CALL DGTPRD(U, X, Q, N, N, N)
C
C     FIND  K  AND ID(I), I=1, K
C
 19   CONTINUE
      HNORM = ODO
      DO 20 K=1, N
```

```
C     PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG, 12 MARCH 1988.

 20        HNORM = HNORM + DABS(H(K,1))
C
      K = 0
      COLUMN = 1
      DO 30 IS1=1,N
          IF (COLUMN.GE.N) GO TO 31
          K = K + 1
          IF ( DABS(H(COLUMN,COLUMN+1)).LT.DABS(EPS*HNORM) ) THEN
                      ID(K) = 1
                      COLUMN = COLUMN + 1
              ELSE
                      ID(K) = 2
                      COLUMN = COLUMN + 2
              END IF
 30   CONTINUE
 31   CONTINUE
      IF (COLUMN.EQ.N) THEN
                      K = K + 1
                      ID(K) = 1
                      END IF
C
C     SOLVE FOR Y IN   H*Y + Y*HT + T = 0
C
      IROW = 1
      DO 100 IS1=1,K
          ICOUNT = IS1
          ICOL = 1
          DO 99 IS2=1,ICOUNT
              IF ( ID(IS1).EQ.1 ) THEN
                      IF ( ID(IS2).EQ.1 ) THEN
C
C     1111111111111111111111111111111111111111111111111111111111111111111
      T11 = Q(IROW,ICOL)
      DO 110 M=1,(IROW-1)
 110      T11 = T11 + H(IROW,M)*Y(M,ICOL)
      DO 120 M=1,(ICOL-1)
 120      T11 = T11 + Y(IROW,M)*H(ICOL,M)
C
      Y(IROW,ICOL) = -T11/( H(IROW,IROW)+H(ICOL,ICOL) )
C
      IF (IROW.NE.ICOL) Y(ICOL,IROW) = Y(IROW,ICOL)
C     1111111111111111111111111111111111111111111111111111111111111111111
C
                          ELSE
C
C     2222222222222222222222222222222222222222222222222222222222222222222
      IC1 = ICOL + 1
      T11 = Q(IROW,ICOL)
      T12 = Q(IROW,IC1 )
      DO 130 M=1,(IROW-1)
          T22 = H(IROW,M)
          T11 = T11 + T22*Y(M,ICOL)
 130      T12 = T12 + T22*Y(M,IC1 )
      DO 140 M=1,(ICOL-1)
          T22 = Y(IROW,M)
          T11 = T11 + T22*H(ICOL,M)
 140      T12 = T12 + T22*H(IC1 ,M)
C
      H11 = H(IROW,IROW) + H(ICOL,ICOL)
      H12 = H(ICOL,IC1 )
```

```
C       PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG, 12 MARCH 1988.


        H21 = H(IC1 ,ICOL)
        H22 = H(IROW,IROW) + H(IC1 ,IC1 )
        T21 = H11*H22 - H12*H21
C
        Y(IROW,ICOL) = ( -T11*H22 + T12*H12 ) / T21
        Y(IROW,IC1 ) = ( -T12*H11 + T11*H21 ) / T21
C
        Y(ICOL,IROW) = Y(IROW,ICOL)
        Y(IC1 ,IROW) = Y(IROW,IC1 )
C       2222222222222222222222222222222222222222222222222222222222222222
C
                            END IF
              ELSE  IF ( ID(IS2).EQ.1 ) THEN
C
C       3333333333333333333333333333333333333333333333333333333333333333
        IR1 = IROW + 1
        T11 = Q(IROW,ICOL)
        T21 = Q(IR1 ,ICOL)
        DO 150 M=1,(IROW-1)
           T12 = Y(M,ICOL)
           T11 = T11 + T12*H(IROW,M)
 150       T21 = T21 + T12*H(IR1 ,M)
        DO 160 M=1,(ICOL-1)
           T12 = H(ICOL,M)
           T11 = T11 + T12*Y(IROW,M)
 160       T21 = T21 + T12*Y(IR1 ,M)
C
        H11 = H(IROW,IROW) + H(ICOL,ICOL)
        H12 = H(IROW,IR1 )
        H21 = H(IR1 ,IROW)
        H22 = H(IR1 ,IR1 ) + H(ICOL,ICOL)
        T12 = H11*H22 - H12*H21
C
        Y(IROW,ICOL) = ( -T11*H22 + T21*H12 ) / T12
        Y(IR1 ,ICOL) = ( -T21*H11 + T11*H21 ) / T12
C
        Y(ICOL,IROW) = Y(IROW,ICOL)
        Y(ICOL,IR1 ) = Y(IR1 ,ICOL)
C       3333333333333333333333333333333333333333333333333333333333333333
C
                           ELSE
C
C       4444444444444444444444444444444444444444444444444444444444444444
        IR1 = IROW + 1
        IC1 = ICOL + 1
        T11 = Q(IROW,ICOL)
        T12 = Q(IROW,IC1 )
        T21 = Q(IR1 ,ICOL)
        T22 = Q(IR1 ,IC1 )
        DO 170 M=1,(IROW-1)
           H11 = H(IROW,M)
           H12 = H(IR1 ,M)
           H21 = Y(M,ICOL)
           H22 = Y(M,IC1 )
           T11 = T11 + H11*H21
           T12 = T12 + H11*H22
           T21 = T21 + H12*H21
 170       T22 = T22 + H12*H22
        DO 180 M=1,(ICOL-1)
           H11 = Y(IROW,M)
```

```
C     PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG, 12 MARCH 1988.


         H12 = Y(IR1 ,M)
         H21 = H(ICOL,M)
         H22 = H(IC1 ,M)
         T11 = T11 + H11*H21
         T12 = T12 + H11*H22
         T21 = T21 + H12*H21
 180     T22 = T22 + H12*H22
C
      R11 = H(ICOL,ICOL)
      R12 = H(IC1 ,ICOL)
      R21 = H(ICOL,IC1 )
      R22 = H(IC1 ,IC1 )
      H11 = H(IROW,IROW)
      H12 = H(IROW,IR1 )
      H21 = H(IR1 ,IROW)
      H22 = H(IR1 ,IR1 )
C
      A1(1)  = H11 + R11
      A1(5)  = H12
      A1(9)  = R21
      A1(13) = ODO
      A1(2)  = H21
      A1(6)  = H22 + R11
      A1(10) = ODO
      A1(14) = R21
      A1(3)  = R12
      A1(7)  = ODO
      A1(11) = H11 + R22
      A1(15) = H12
      A1(4)  = ODO
      A1(8)  = R12
      A1(12) = H21
      A1(16) = H22 + R22
C
      B1(1) = -T11
      B1(2) = -T21
      B1(3) = -T12
      B1(4) = -T22
      CALL DSIMQ(A1,B1,N4,KS)
C
      Y(IROW,ICOL) = B1(1)
      Y(IR1 ,ICOL) = B1(2)
      Y(IROW,IC1 ) = B1(3)
      Y(IR1 ,IC1 ) = B1(4)
C
      IF (IROW.EQ.ICOL) GO TO 190
          Y(ICOL,IROW) = B1(1)
          Y(ICOL,IR1 ) = B1(2)
          Y(IC1 ,IROW) = B1(3)
          Y(IC1 ,IR1 ) = B1(4)
 190  CONTINUE
C     444444444444444444444444444444444444444444444444444444444444444444
C
          END IF
          ICOL = ICOL + ID(IS2)
 99     CONTINUE
        IROW = IROW + ID(IS1)
 100  CONTINUE
C
C     NOW WE OBTAIN Y FOR THE SOLUTION OF H*Y + Y*HT + T = 0
```

```
C       PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG, 12 MARCH 1988.


C       THEN FIND       X = U*Y*UT
C
        CALL DGMTRA(U, X, N, N)
        CALL DGMPRD(Y, X, H, N, N, N)
        CALL DGMPRD(U, H, X, N, N, N)
C
        IERR = 0

C        TIME2 = CTIM$A(TTT)
C        WRITE(*,*)' TIME USED IN BTLSTW = ', (TIME2-TIME1)
 999    RETURN
C
 1000   IERR = 999
        GO TO 999
        END

        SUBROUTINE STKOUT( A, B0, B1, C0, C1, Q0, Q1, R0, R1, AC, K0, K1, L1, F0,
     &             F1, GRAD0, GRAD1, J0, J1, EPS, LIMIT, IN, IM0, IM1, IR0, IR1    )
C
        IMPLICIT INTEGER*2 (I-N)
        INTEGER*2 LIMIT, IN, IM0, IM1, IR0, IR1
        REAL*8      A(1), B0(1), B1(1), C0(1), C1(1), Q0(1), Q1(1), R0(1), R1(1),
     &              AC(1), K0(1), K1(1), L1(1), F0(1), F1(1),
     &              GRAD0(1), GRAD1(1), J0, J1, EPS
C
        INTEGER*2 N, I, J, K, IK, IROW, ICOL, IGRAD, IGRADF, IFCN, ICOUNT
        REAL*8      OLDGRAD(20), SEARCH(20), OLDF0(20), LAMBDA, SRHNORM, OLDJ0,
     &              MCHEPS, DMAX, DX, DX1, DPHI, GNORM
        REAL*8      FA, FB, FC, XA, XB, XC, AA, BB, CC, DD, OLDLMD, LAMBDA1
        REAL*8      H(20,20), YK(20), PK(20), HYK(20)
        REAL*8       TIME1, TIME2, TTT, CTIM$A
        DATA        MCHEPS /7.105428D-15/
C
C       INITIALIZE
        I=0
        N = IM0 * IR0
        DMAX = 0.01D0
C
        MCHEPS1 = 1D-8
C
C       ==============================================
C       COMPUTE  f(x) AND grad(x) AT INITIAL GUESS
C       ==============================================
        CALL FCNLEAD( A, B0, B1, C0, C1, Q0, Q1, R0, R1, AC, K0, K1, L1, F0, F1, GRAD0,
     &                GRAD1, J0, J1, EPS, IN, IM0, IM1, IR0, IR1, IERR )
        CALL GRDLEAD( A, B0, B1, C0, C1, Q0, Q1, R0, R1, AC, K0, K1, L1, F0, F1, GRAD0,
     &                GRAD1, J0, J1, EPS, IN, IM0, IM1, IR0, IR1  )
        WRITE(*,*)' LEADER : FOLLOWER'
        WRITE(*,*)' J0 = ', J0
        CALL DPRINTMAT(F0, IM0, IR0)
        WRITE(*,*)' J1 = ', J1
        CALL DPRINTMAT(F1, IM1, IR1)
C
C       ================================
C       SET h = -grad(x)   AND  SET H = I
C       ================================
3       CONTINUE

        DO 7 IROW=1, N
           DO 5 ICOL=1, N
```

```
C      PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG, 12 MARCH 1988.


                IF (IROW.NE.ICOL) THEN
                      H(IROW,ICOL) = 0.0D0
                   ELSE
                      H(IROW,ICOL) = 1.0D0
                   END IF
5          CONTINUE
7       CONTINUE
C
10      DO 15 K=1,N
           SEARCH(K) = -GRADO(K)
15      CONTINUE
C
C           STORE OLD VALUES OF FO AND GRADO AND JO
C           ====================================
20      CONTINUE
C
        TIME1 = CTIM$A(TTT)
C
        DO 22 K=1,N
           OLDGRAD(K) = GRADO(K)
           OLDFO(K)   = FO(K)
22      CONTINUE
        OLDJO = JO
C
        WRITE(*,*)' GRADIENT # 0'
        CALL DPRINTMAT(GRADO,IMO,IRO)
C
C=================================
C           FIND STEP-SIZE OR LAMBDA
C=================================
C       FIND NORM OF SEARCH OR SRHNORM
        SRHNORM = 0.0D0
        GNORM   = 0.0D0
        DO 225 K=1,N
           GNORM   = GNORM   + GRADO(K) *GRADO(K)
225        SRHNORM = SRHNORM + SEARCH(K)*SEARCH(K)
        SRHNORM = DSQRT(SRHNORM)
        GNORM   = DSQRT(GNORM)
        DO 975 K=1,N
           IF( DABS(GRADO(K)) .LT.  1D-2 ) RETURN
 975    CONTINUE
        IF (GNORM.LT.5D-2) RETURN
C
        WRITE(*,*)' SRHNORM = ',SRHNORM
C
C        DO 2251 K=1,N
C 2251      SEARCH(K) = SEARCH(K)/SRHNORM
C
C
C           COMPUTE DIRECTIONAL DERIVATIVE, DPHI
C           =================================
        DPHI = 0.0D0
        DO 226 K=1,N
226        DPHI = DPHI + GRADO(K)*SEARCH(K)

C       ----------------------------------------------------------
C       IF DIRECTIONAL DERIVATIVE > 0 ,THEN SEARCH ALONG STEEPEST
C       DESCENT DIRECTION
C       ----------------
        IF (DPHI) 300,125,3
```

```
C      PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG,  12 MARCH 1988.


300    CONTINUE
C
       LAMBDA = DABS( 0.2D0*J0/DPHI )
       IF ( SRHNORM.LT.1D3   ) LAMBDA = 0.20D0/SRHNORM
       IF ( SRHNORM.LT.1D-2  ) LAMBDA = 0.02D0/SRHNORM
C
C           INITIALIZE FOR FINDING BRACKET OF MIN POINT
C           ==========================================
       FA = J0
       XA = 0.0D0
       LAMBDA = DMAX/SRHNORM
C
C           CALCULATE J0 AT THE NEXT POINT,  XB
C           ==================================
 30    CONTINUE
       DO 301 K=1,N
 301      F0(K) = OLDF0(K) + LAMBDA*SEARCH(K)
       CALL FCNLEAD( A,B0,B1,C0,C1,Q0,Q1,R0,R1,AC,K0,K1,L1,F0,F1,GRAD0,
     &                  GRAD1,J0,J1,EPS,IN,IM0,IM1,IR0,IR1,IERR )
C
       IF ( IERR.NE.0 ) THEN
              LAMBDA = LAMBDA * 0.666666666667D0
              GO TO 30
              END IF
C
       FB = J0
       XB = LAMBDA
C
       IF (J0.LT.FA) GO TO 303
C           IF ( FB<FA ) THEN  CONTINUE FINDING J0 AT XC
C                        ELSE  IT DENOTES THAT THE MINIMUM HAS BEEN PASSED
            XC = XB
            FC = FB
3019        CONTINUE
            LAMBDA = LAMBDA / 3.0D0
            DO 302 K=1,N
302            F0(K) = OLDF0(K) + LAMBDA*SEARCH(K)
            CALL FCNLEAD( A,B0,B1,C0,C1,Q0,Q1,R0,R1,AC,K0,K1,L1,F0,F1,
     &                       GRAD0,GRAD1,J0,J1,EPS,IN,IM0,IM1,IR0,IR1,IERR )
            IF (J0.GT.FA) THEN
                   FC = J0
                   XC = LAMBDA
                   GO TO 3019
            ELSE
                   XB = LAMBDA
                   FB = J0
            END IF
C
C           NOW WE OBTAIN THE BOUND OF THE MIN. POINT
C
            GO TO 329
C
303    CONTINUE
       LAMBDA = LAMBDA + LAMBDA
C             NOW LAMBDA IS USED AS INTERVAL LENGTH
3031   CONTINUE
       XC = XB + LAMBDA
       DO 304 K=1,N
 304      F0(K) = OLDF0(K) + XC*SEARCH(K)
       CALL FCNLEAD( A,B0,B1,C0,C1,Q0,Q1,R0,R1,AC,K0,K1,L1,F0,F1,GRAD0,
```

```
C       PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG, 12 MARCH 1988.


     &                GRAD1, JO, J1, EPS, IN, IMO, IM1, IRO, IR1, IERR )
        IF (IERR.NE.O) THEN
                LAMBDA = LAMBDA*0.666666666667DO
                GO TO 3031
                END IF
        FC = JO
C
        IF ( FC.LT.FB ) THEN      /*  SEARCH FURTHER   */
                FA = FB
                XA = XB
                FB = FC
                XB = XC
                GO TO 303
        END IF      /* ELSE IT DENOTES THAT THE MINIMUM HAS BEEN BOUNDED */
                    /* THEN PERFORM QUADRATIC INTERPOLATION              */
C       ==============================================
C       QUADRATIC INTERPOLATION IS PERFORMED HERE.
C       ==============================================
  329   CONTINUE
        ICOUNT = O
        LAMBDA = XB
  330   CONTINUE
C
        OLDLMD = LAMBDA
        AA = FA*(XB*XB-XC*XC) + FB*(XC*XC-XA*XA) + FC*(XA*XA-XB*XB)
        BB = FA*(XB-XC)       + FB*(XC-XA)       + FC*(XA-XB)
        LAMBDA = AA/BB/2DO
C
        ICOUNT = ICOUNT + 1
        DO 331 K=1,N
  331      FO(K) = OLDFO(K) + LAMBDA*SEARCH(K)
        CALL FCNLEAD( A, BO, B1, CO, C1, QO, Q1, RO, R1, AC, KO, K1, L1, FO, F1, GRADO,
     &                GRAD1, JO, J1, EPS, IN, IMO, IM1, IRO, IR1, IERR )
C
        IF ( LAMBDA.GT.XB ) THEN
                     IF (JO.LT.FB) THEN
                              XA = XB
                              FA = FB
                              XB = LAMBDA
                              FB = JO
                          ELSE
                              XC = LAMBDA
                              FC = JO
                          END IF
             ELSE  IF (JO.LT.FB) THEN
                              XC = XB
                              FC = FB
                              XB = LAMBDA
                              FB = JO
                          ELSE
                              XA = LAMBDA
                              FA = JO
             END IF
C
        IF ( DABS(OLDLMD-LAMBDA).LT.DABS(5D-4*OLDLMD) ) GO TO 400
        IF ( DABS(FA-FB).LT.DABS(1D-10*FB) ) GO TO 400
        IF ( DABS(FB-FC).LT.DABS(1D-10*FB) ) GO TO 400
        IF ( ICOUNT.GE.5 ) GO TO 400
C
C       REDUCE THE INTERVAL IN CASE OF UNBALANCED INTERVAL
```

```
C     PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG, 12 MARCH 1988.


C     SO AS TO REDUCE THE COMPUTATION IN QUADRATIC INTERPOLATION
C     PROCESS
      IF ( DABS(XB-XA) .GT. DABS(0.1DO*(XC-XB)) ) GO TO 330
      IF ( DABS(XC-XB) .GT. DABS(0.1DO*(XB-XA)) ) GO TO 330
      IF ( DABS(XB-XA) .LT. DABS(XC-XB) ) THEN
C               ======================================
C               XC-XB IS VERY MUCH LARGER THAN XB-XA
C               ======================================
                LAMBDA1 = XB + 0.3DO*( XC-XB )
3099            CONTINUE
C
                DO 3001 K=1,N
 3001              FO(K) = OLDFO(K) + LAMBDA1*SEARCH(K)
                CALL FCNLEAD( A,BO,B1,CO,C1,GO,Q1,RO,R1,AC,KO,K1,L1,FO,
     &             F1,GRADO,GRAD1,JO,J1,EPS,IN,IMO,IM1,IRO,IR1,IERR )
C
                IF (JO.GT.FB) THEN
                      XC = LAMBDA1
                      FC = JO
                ELSE
                      XA = XB
                      FA = FB
                      XB = LAMBDA1
                      FB = JO
                END IF
          ELSE
                LAMBDA1 = XB - 0.3DO*( XB-XA )
3098            CONTINUE
C
                DO 3002 K=1,N
 3002              FO(K) = OLDFO(K) + LAMBDA1*SEARCH(K)
                CALL FCNLEAD( A,BO,B1,CO,C1,GO,Q1,RO,R1,AC,KO,K1,L1,FO,
     &               F1,GRADO,GRAD1,JO,J1,EPS,IN,IMO,IM1,IRO,IR1,IERR )
C
                IF ( JO.GT.FB) THEN
                     XA = LAMBDA1
                     FA = JO
                 ELSE
                     XC = XB
                     FC = FB
                     XB = LAMBDA1
                     FB = JO
                 END IF
      END IF
      GO TO 330
C
C         HAVING TERMINATED FROM LINE SEARCH PROCESS
C         ========================================
400       CONTINUE
C         ------------------------------------------------------------
C         SELECT WHICH IS MINIMUM AMOUNG THREE POINTS, THEN EXIT LINE-SEARCH
C         ------------------------------------------------------------
          IF ( FA.GT.FB ) THEN
                 LAMBDA = XB
                 JO = FB
             ELSE
                 LAMBDA = XA
                 JO = FA
          END IF
          IF ( FC.LT.JO ) THEN
```

```
C      PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG, 12 MARCH 1988.


                    LAMBDA = XC
                    JO = FC
         END IF
C        -------------
C        UPDATE VECTOR
C        -------------
         DO 409 K=1,N
 409        FO(K) = OLDFO(K) + LAMBDA*SEARCH(K)
C        ----------------------------------------------
C        CHECK TERMINATION , USING THE GIVEN MACHINE EPS
C        ----------------------------------------------
         IF ( DABS(JO).LT.1D-5 ) GO TO 119
            IF ( DABS(JO-OLDJO).LT.DABS(OLDJO*MCHEPS) ) GO TO 125
 119     CONTINUE
         DO 120 K=1,N
            DX  = DABS( OLDFO(K)-FO(K) )
            DX1 = DABS( OLDFO(K) )*EPS + MCHEPS
            IF ( DX.GT.DX1 ) GO TO 130
120      CONTINUE
C******************************************************************
125      RETURN
C******************************************************************
130      I = I+1
C
         TIME2 = CTIM$A(TTT)
         WRITE(*,*)' TIME USED IN THIS ITERATION = ',(TIME2-TIME1)
         ITER = I
C
         WRITE(*,*)'*******************************************'
         WRITE(*,116)I
116      FORMAT(' I = ',I5)
         WRITE(*,*)' LAMBDA =',LAMBDA
         WRITE(*,*)' JO = ',JO
         CALL DPRINTMAT(FO,IMO,IRO)
         WRITE(*,*)' J1 = ',J1
         CALL DPRINTMAT(F1,IM1,IR1)
         WRITE(*,*)'*******************************************'
C
         IF (I.GT.LIMIT) GO TO 999
C
C      CALCULATE THE GRADIENT OF THE NEXT POINT
C
       CALL GRDLEAD( A,BO,B1,CO,C1,QO,Q1,RO,R1,AC,KO,K1,L1,FO,F1,GRADO,
     &               GRAD1,JO,J1,EPS,IN,IMO,IM1,IRO,IR1 )
C      --------------------
C      UPDATE METRIC MATRIX
C      --------------------
       DO 773 K=1,N
          YK(K) = GRADO(K) - OLDGRAD(K)
          PK(K) = FO(K) - OLDFO(K)
 773   CONTINUE
C
       DO 775 IK=1,N
          HYK(IK) = 0.0DO
          DO 774 K=1,N
             HYK(IK) = HYK(IK) + H(IK,K)*YK(K)
 774      CONTINUE
 775   CONTINUE
C
       DX  = 0.0DO
```

```
C     PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG,  12 MARCH 1988.


      DX1 = 0.0D0
      DO 776 K=1,N
          DX  = DX  + HYK(K) * YK(K)
          DX1 = DX1 + PK(K)  * YK(K)
 776  CONTINUE
C
      DO 800 IK=1,N
         DO 801 K=1,N
            H(IK,K) = H(IK,K) + PK(IK)*PK(K)/DX1 - HYK(IK)*HYK(K)/DX
 801     CONTINUE
 800  CONTINUE
C
C     SET SEARCH DIRECTION FOR THE NEXT ITERATION
C     ============================================
      DO 803 K=1,N
         SEARCH(K) = 0.0D0
         DO 802 IK=1,N
            SEARCH(K) = SEARCH(K) - H(K,IK) * GRADO(IK)
 802     CONTINUE
 803  CONTINUE
      GO TO 20
C
999   WRITE(*,*)' THE ITERATIONS CANNOT CONVERGE IN LIMIT.'
      RETURN
      End

      SUBROUTINE FCNLEAD ( A,B0,B1,C0,C1,Q0,Q1,R0,R1,AC,K0,K1,L1,F0,F1,
     &             GRAD0,GRAD1,J0,J1,EPS,IN,IM0,IM1,IR0,IR1,IERR )
      IMPLICIT INTEGER*2 (I-N)
      INTEGER*2  IN,IM0,IM1,IR0,IR1,IERR
      REAL*8     A(1),B0(1),B1(1),C0(1),C1(1),Q0(1),Q1(1),R0(1),R1(1),
     &           AC(1),K0(1),K1(1),L1(1),F0(1),F1(1),
     &           GRAD0(1),GRAD1(1),J0,J1,EPS
      INTEGER*2  ITMAX,ITER,N0,N1,IW
      REAL*8     AA(361),U(361),V(361),HH(361),EPS1,HDIFF,BUFFER,
     &           F1BUFFER(10),J1BUFFER,J0FRNT,J0BACK,MEPS
      REAL*8     TR,DMAX1,DMIN1
      DATA       MEPS /7.0D-14/
C
C++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
C      START TO FIND OPTIMAL REACTION DUE TO F0
C++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
C
C     COMPUTE AA = A - B0_F0_C0
C
      CALL DGMPRD(F0,C0,U ,IM0,IR0,IN)       /*     U = F0_C0        */
      CALL DGMPRD(B0,U,V,IN,IM0,IN)          /*     V = B0_F0_C0     */
      CALL DGMSUB(A,V,AA,IN,IN)
C
      EPS1 = EPS * 0.01D0
      ITMAX = 100
      CALL DFP( F1,J1,GRAD1,EPS1,ITMAX,AA,B1,C1,K1,L1,AC,Q1,R1,
     &          IN,IM1,IR1,ITER,IERR )
      IF (IERR.NE.0) GO TO 990

C      DO 444 IW=1,IR1
C 444     WRITE(*,*)' GRAD = ',GRAD1(IW)
C      CALL DPRINTMAT(AC,IN,IN)

C
```

```
C      PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG, 12 MARCH 1988.


C      COMPUTE  V = ( QO + CO'FO'RO_FO_CO )
C
       CALL DGMPRD(RO,U,V,IMO,IMO,IN)        /*  V = RO_FO_CO              **/
       CALL DGTPRD(FO,V,U,IMO,IRO,IN)        /*  U = FO'RO_FO_CO           **/
       CALL DGTPRD(CO,U,V,IRO,IN,IN)         /*  V = CO'FO'RO_FO_CO        **/
       CALL DGMADD(QO,V,V,IN,IN)             /*  V = QO +CO'FO'RO_FO_CO **/

C       CALL DPRINTMAT(V,IN,IN)
C       ITMAX = 30
C       CALL DMCPY( AC,U,IN,IN,O )
C       CALL DLYAP( U,V,KO,1.ODO,1.OD-14,IN,ITMAX,O )

       CALL BTLSTW(AC,KO,V,HH,AA,U,IN,O,IERR)
       JO = TR(KO,IN)*0.5DO
C
 990   RETURN
       END

       SUBROUTINE GRDLEAD ( A,BO,B1,CO,C1,QO,Q1,RO,R1,AC,KO,K1,L1,FO,F1,
     &             GRADO,GRAD1,JO,J1,EPS,IN,IMO,IM1,IRO,IR1 )
       IMPLICIT INTEGER*2 (I-N)
       INTEGER*2   IN,IMO,IM1,IRO,IR1
       REAL*8      A(1),BO(1),B1(1),CO(1),C1(1),QO(1),Q1(1),RO(1),R1(1),
     &             AC(1),KO(1),K1(1),L1(1),FO(1),F1(1),
     &             GRADO(1),GRAD1(1),JO,J1,EPS
       INTEGER*2   I,J,NO,N1,IERR1
       REAL*8      HDIFF,BUFFER,F1BUFFER(10),J1BUFFER,JOFRNT,JOBACK,MEPS
       REAL*8      TR,DMAX1,DMIN1
       DATA        MEPS /7.OD-14/
C
       HDIFF = DMAX1( DMIN1(((3*JO*MEPS)**(1DO/3DO)),1D-3) , 2D-5 )
       NO = IMO * IRO
       N1 = IM1 * IR1
       DO 250 I=1,NO
          BUFFER = FO(I)
C
          FO(I) = BUFFER + HDIFF
          DO 150 J=1,N1
 150         F1BUFFER(J) = F1(J)
          CALL FCNLEAD( A,BO,B1,CO,C1,QO,Q1,RO,R1,AC,KO,K1,L1,FO,
     &                  F1BUFFER,GRADO,GRAD1,JOFRNT,J1BUFFER,EPS,IN,IMO,
     &                  IM1,IRO,IR1,IERR1 )
          FO(I) = BUFFER - HDIFF
          DO 160 J=1,N1
 160         F1BUFFER(J) = F1(J)
          CALL FCNLEAD( A,BO,B1,CO,C1,QO,Q1,RO,R1,AC,KO,K1,L1,FO,
     &                  F1BUFFER,GRADO,GRAD1,JOBACK,J1BUFFER,EPS,IN,IMO,
     &                  IM1,IRO,IR1,IERR1 )
          GRADO(I) = ( JOFRNT-JOBACK )/( 2.ODO*HDIFF )
C
          FO(I) = BUFFER
 250   CONTINUE
C
 990   RETURN
       END

       SUBROUTINE FUNCT( F,VAL,GRAD,
     &                   A1,B1,C,DK,DL,AC,Q,R,IN,IM,IR,IERR )
C
C     &                   A1,B1,C,DK,DL,AC,Q,R,IN,IM,IR )
```

```
C     PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG, 12 MARCH 1988.


C
      IMPLICIT INTEGER*2 (I-N)
      REAL*8  F(1),VAL,GRAD(1),A1(1),B1(1),C(1),DK(1),DL(1),AC(1),Q(1),
     &         R(1),    TR
      INTEGER*2    IN,IM,IR,IERR
      REAL*8       U(361),V(361),HH(361),YY(361)
C
C      INTEGER*2    IN,IM,IR,     IND,ITER
C      REAL*8       U(400),V(400),RR(20),RI(20),UV(400)
C======================================================================
CCC      COMMON /OUT/ A1(19,19),B1(19,2),C(5,19),DK(19,19),DL(19,19),
C    &             AC(19,19),Q(19,19),R(2,2),        IN,IM,IR
C======================================================================
C
C     COMPUTE AC = A-BFC
C
      CALL DGMPRD(F,C,U ,IM,IR,IN)
      CALL DGMPRD(B1,U,V,IN,IM,IN)          /*    V = BFC     */
      CALL DGMSUB(A1,V,AC,IN,IN)
C
C     COMPUTE  V = ( Q+C'F'RFC )
C
      CALL DGMPRD(R,U,V,IM,IM,IN)         /*  V = RFC   **/
      CALL DGTPRD(F,V,U,IM,IR,IN)         /*  U = F'RFC **/
      CALL DGTPRD(C,U,V,IR,IN,IN)         /*  V = C'F'RFC **/
      CALL DGMADD(Q,V,V,IN,IN)           /*  V = Q+C'F'RFC **/

C ********************************************
C     SOLVE TWO LYAPUNOV'S EQUATIONS.
C ********************************************
C      CALL DMCPY(AC,UV,IN,IN,0)
C      ITER = 30
C      CALL DLYAP(UV,V,DK,1.0D0,1.0D-14,IN,ITER,0)
C      CALL DMCPY(AC,UV,IN,IN,0)
C      CALL DLYAP(UV,U,DL,1.0D0,1.0D-14,IN,ITER,1)
C
      CALL BTLSTW(AC,DK,V,HH,YY,U,IN,0,IERR)
      IF (IERR.NE.0) GO TO 1000
      VAL = TR(DK,IN)*0.5D0
C     ====================
C     CALCULATE GRADIENT
C     GRAD = RFCLC'-B'KLC'
C     ====================
C
C     COMPUTE U = I
C
      CALL DSCLA(V,0.0D0,IN,IN,0)
      CALL DDCLA(V,1.0D0,IN,0)
      CALL BTLSTW(AC,DL,V,HH,YY,U,IN,1,IERR)
C
      CALL DGMTRA(C,U,IR,IN)              /*     U = C'        */
      CALL DGMPRD(DL,U,V,IN,IN,IR)         /*     V = LC'       */
      CALL DGMPRD(C,V,U,IR,IN,IR)         /*     U = CLC'      */
      CALL DGMPRD(F,U,V,IM,IR,IR)         /*     V = FCLC'     */
      CALL DGMPRD(R,V,GRAD,IM,IM,IR)      /*   GRAD = RFCLC'   */
      CALL DGMTRA(C,U,IR,IN)              /*     U = C'        */
      CALL DGMPRD(DL,U,V,IN,IN,IR)         /*     V = LC'       */
      CALL DGMPRD(DK,V,U,IN,IN,IR)         /*     U = KLC'      */
      CALL DGTPRD(B1,U,V,IN,IM,IR)        /*     V = B'KLC'    */
      CALL DGMSUB(GRAD,V,GRAD,IM,IR)      /*  GRAD = RFCLC'-B'KLC' */
```

```
C      PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG, 12 MARCH 1988.


998    RETURN
C-----------------------------------
999    WRITE(*,*)' THE SYSTEM IS NOT STABLE!! '
1000   RETURN
C-----------------------------------
       END

       SUBROUTINE DFP( X,FX,GRAD,EPS,LIMIT,
      &                A1,B1,C,DK,DL,AC,Q,R,IN,IM,IR,ITER,IERR )
C     &                A1,B1,C,DK,DL,AC,Q,R,IN,IM,IR,ITER )
       IMPLICIT INTEGER*2 (I-N)
       REAL*8     X(1),GRAD(1),FX,EPS,A1(1),B1(1),C(1),DK(1),DL(1),AC(1),
      &           Q(1),R(1)
       INTEGER*2 LIMIT,IN,IM,IR,ITER,IERR
       INTEGER*2 N,I,J,K,IK
       REAL*8     GRADO(20),SEARCH(20),XO(20),LAMBDA,SRHNORM,GAMMA,DX,
      &           GNORM,FA,FB,XA,XB,DPHIA,DPHIB,DPHI,Z,W,MU,DMAX,DX1,
      &           H(20,20),YK(20),PK(20),HYK(20),FOLD,MEPS
       DATA       MEPS /7.105428D-15/
C==========================================================================
C       REAL*8     A1,B1,C,DK,DL,AC,Q,R,IN,IM,IR
C       COMMON /OUT/ A1(19,19),B1(19,2),C(5,19),DK(19,19),DL(19,19),
C      &           AC(19,19),Q(19,19),R(2,2),        IN,IM,IR
C==========================================================================
C
C      INITIALIZE
       I = 0
       ITER = 0
       N = IM * IR
C       DMAX = 0.01D0
C      ==========================================
C      COMPUTE  f(x) AND grad(x) AT INITIAL GUESS
C      ==========================================
       CALL FUNCT(X,FX,GRAD,A1,B1,C,DK,DL,AC,Q,R,IN,IM,IR,IERR)
C
C      IF ITER=0 AND IERR<>0 , IT DENOTES THE INITIAL GUESS IS NOT A.S.
C
       IF (IERR.NE.0) GO TO 1000

C       CALL DPRINTMAT(X,IM,IR)
C       WRITE(*,*)' J = ',FX

C      ================================
C      SET h = -grad(x)  AND  SET H = I
C      ================================
3      CONTINUE

       DO 7 IROW=1,N
          DO 5 ICOL=1,N
             IF (IROW.NE.ICOL) THEN
                      H(IROW,ICOL) = 0.0D0
                ELSE
                      H(IROW,ICOL) = 1.0D0
                END IF
5         CONTINUE
7      CONTINUE
C
10     DO 15 K=1,N
          SEARCH(K) = -GRAD(K)
15     CONTINUE
```

```
C      PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG, 12 MARCH 1988.


C
C          STORE OLD VALUES OF X AND GRAD AND FX
C          =====================================
20     CONTINUE
       DO 22 K=1,N
          GRADO(K) = GRAD(K)
          XO(K) = X(K)
22     CONTINUE
       FOLD = FX
C=================================
C          FIND STEP-SIZE OR LAMBDA
C=================================
C      FIND NORM OF SEARCH OR SRHNORM
       SRHNORM = 0.0D0
       DO 225 K=1,N
225       SRHNORM = SRHNORM + SEARCH(K)*SEARCH(K)
       SRHNORM = DSQRT(SRHNORM)

C      COMPUTE DIRECTIONAL DERIVATIVE, DPHIA
       DPHIA = 0.0D0
       DO 226 K=1,N
226       DPHIA = DPHIA + GRAD(K)*SEARCH(K)

C
C      IF DIRECTIONAL DERIVATIVE > 0 ,THEN SEARCH ALONG STEEPEST
C      DESCENT DIRECTION
C
       IF (DPHIA) 300,125,3
300    CONTINUE
C
C       IF (I.GT.3) DMAX = 0.1D0
C       IF (SRHNORM.LT.1D-3) DMAX = 0.01D0
C
       LAMBDA = DABS(0.2D0*FX/DPHIA)
       IF (SRHNORM.LT.1D3 ) LAMBDA = 0.4D0 /SRHNORM
       IF (SRHNORM.LT.1D-2) LAMBDA = 0.02D0/SRHNORM
C
       FA = FX
       XA = 0.0D0
C
C       LAMBDA = DMAX/SRHNORM
C
30     CONTINUE
       DO 301 K=1,N
301       X(K) = XO(K) + LAMBDA*SEARCH(K)
       CALL FUNCT(X,FX,GRAD,A1,B1,C,DK,DL,AC,Q,R,IN,IM,IR,IERR)
       IF (IERR.NE.0) THEN
               LAMBDA = LAMBDA*0.6666666666666667D0
               GO TO 30
               END IF
C
C          COMPUTE DPHIB, THE DIRECTIONAL DERIVATIVE AT THE NEW POINT
       DPHIB = 0.0D0
       DO 302 K=1,N
302       DPHIB = DPHIB + GRAD(K)*SEARCH(K)
C          CHECK WHETHER THE MINIMUM HAS BEEN BOUNDED OR NOT
       IF (DPHIB) 310,400,320
310        DPHIA = DPHIB
           XA = LAMBDA
           FA = FX
```

```
C      PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG,  12 MARCH 1988.


             LAMBDA = LAMBDA + LAMBDA
             GO TO 30
320     XB = LAMBDA
        FB = FX
C
C            CUBIC INTERPOLATION IS PERFORMED HERE.
C
             ICOUNT = 0
330          Z = 3*(FA-FB)/(XB-XA) + DPHIA + DPHIB
             W = DSQRT( Z*Z - DPHIA*DPHIB )
             MU = ( DPHIB + W - Z )/( DPHIB - DPHIA + 2*W )
             ICOUNT = ICOUNT + 1
             LAMBDA = XB - MU*(XB-XA)

C             WRITE(*,555)XA,FA,XB,FB
C 555         FORMAT('XA FA , XB,FB' , 4E15.6)

C
             DO 331 K=1,N
331             X(K) = XO(K) + LAMBDA*SEARCH(K)
             CALL FUNCT(X,FX,GRAD,A1,B1,C,DK,DL,AC,Q,R,IN,IM,IR,IERR)
             DPHI = 0.0D0
             DO 332 K=1,N
332             DPHI = DPHI + GRAD(K)*SEARCH(K)

C             WRITE(*,777) DPHI,LAMBDA
C 777         FORMAT(' DPHI = ',2E15.6)

C
C            CHECK WHETHER THE MINIMUM IS CLOSED ENOUGH OR NOT
             IF ( DABS(DPHI/SRHNORM).LT.1D-3 ) GO TO 400
             IF (ICOUNT.GE.4) GO TO 400
             IF ( DPHI.LT.0.0D0 ) GO TO 334
333              XB = LAMBDA
                 FB = FX
                 DPHIB = DPHI
                 GO TO 330
334          XA = LAMBDA
             FA = FX
             DPHIA = DPHI
             GO TO 330
C
C            HAVING TERMINATED FROM LINE SEARCH PROCESS
C            ===========================================
400          CONTINUE
C            CHECK TERMINATION ; USING THE GIVEN MACHINE EPS
             IF ( DABS(FX).LT.1D-5 ) GO TO 119
                 IF ( DABS(FX-FOLD).LT.DABS(FOLD*MEPS) ) GO TO 125
 119         CONTINUE
             DO 120 K=1,N
                DX = DABS( XO (K)-X(K) )
                DX1 = DABS(XO(K))*EPS + MEPS
                IF ( DX.GT.DX1 ) GO TO 130
120          CONTINUE
C***********************************************************************
125          RETURN
C***********************************************************************
130          I = I+1
             ITER = I
C             WRITE(*,116)I
```

```
C      PROGRAMME LISTING WRITTEN BY SUCHIN ARUNSAWATWONG, 12 MARCH 1988.


C116        FORMAT(' I = ',I8)
C           CALL DPRINTMAT(X,IM,IR)
C           WRITE(*,*)' J = ',FX

          IF (I.GT.LIMIT) GO TO 999
C      ---------------------
C      UPDATE METRIC MATRIX
C      ---------------------
       DO 773 K=1,N
          YK(K) = GRAD(K) - GRADO(K)
          PK(K) = X(K) - XO(K)
 773   CONTINUE
C
       DO 775 IK=1,N
          HYK(IK) = 0.0D0
          DO 774 K=1,N
             HYK(IK) = HYK(IK) + H(IK,K)*YK(K)
 774      CONTINUE
 775   CONTINUE
C
       DX  = 0.0D0
       DX1 = 0.0D0
       DO 776 K=1,N
          DX  = DX  + HYK(K) * YK(K)
          DX1 = DX1 + PK(K)  * YK(K)
 776   CONTINUE
C
       DO 800 IK=1,N
          DO 801 K=1,N
             H(IK,K) = H(IK,K) + PK(IK)*PK(K)/DX1 - HYK(IK)*HYK(K)/DX
 801      CONTINUE
 800   CONTINUE
C
C      SET SEARCH DIRECTION
C      ====================
       DO 803 K=1,N
          SEARCH(K) = 0.0D0
          DO 802 IK=1,N
             SEARCH(K) = SEARCH(K) - H(K,IK) * GRAD(IK)
 802      CONTINUE
 803   CONTINUE
       GO TO 20
C
999    WRITE(*,*)' THE ITERATIONS CANNOT CONVERGE IN LIMIT.'
1000   RETURN
       END
```

ประวัติผู้เขียน

นาย สุชิน อรุณสวัสดิ์วงศ์ เกิดวันที่ 6 มกราคม พ.ศ. 2506 ที่กรุงเทพมหานคร สำเร็จปริญญาวิศวกรรมศาสตร์บัณฑิต สาขาวิศวกรรมไฟฟ้า จากจุฬาลงกรณ์มหาวิทยาลัย เมื่อปี พ.ศ. 2527 หลังจากนั้นได้เข้าศึกษาต่อปริญญาโทในภาควิศวกรรมไฟฟ้า สาขาระบบควบคุม ที่จุฬาลงกรณ์มหาวิทยาลัย ระหว่างปีการศึกษา 2528 ถึง 2529 ได้ทำหน้าที่เป็นทั้งผู้ช่วยสอนและผู้ช่วยวิจัยของห้อง วิจัยระบบควบคุม ภาควิศวกรรมไฟฟ้า