

รายการอ้างอิง

- [1] D.J. Murray-Smith. Continuous System Simulation. London : Chapman & Hall, 1995.
- [2] James A. Payne. Introduction To Simulation. New York : McGraw-Hill Book Co, 1988.
- [3] Rickard Nilsson. Robot Arm[Online]. 2002. Available from :
<http://www.flamingo3d.com/gallery/mech/testrending003.htm> [2006,October 20]
- [4] Russel C. Hibbeler. Structural Analysis. New Jersey : Prentice-Hall, Inc, 1995.
- [5] S.S Tezcan. Matrix Simplified Formulation of Stiffness Matrices. Journal of Structural Division 89, 6 (1963) : 445-449.
- [6] Aslam Kassimali. Matrix Analysis of Structures. California : Brooks / Cole Publishing Company, 1999.
- [7] Daniel L. Schodek. Structures. New Jersey : Prentice Hall, 2004.
- [8] Hayrettin Kardestuncer. Elementary Matrix Analysis of Structures. New York : McGraw-Hill Book Co, 1974.
- [9] C. T. F. Ross. Computer Analysis of Skeletal Structures. London : Spon Press, 1981.
- [10] J. N. Reddy. Finite Element Method. New York : McGraw-Hill Book Co, 1993.
- [11] Saeed Moaveni. Finite Element Analysis. New Jersey : Prentice-Hall, 1999.
- [12] R.L. Sack. Matrix Structural Analysis, Illinois : Waveland Press Inc, 1994.
- [13] William Weaver and James M. Gere. Matrix Analysis of Framed Structures. New York : Chapman & Hall, 1990.
- [14] Wong Kp. Solving power system optimization problems using simulated annealing. Engr Appl Artif Intell, 8,6 (1995): 665-670.
- [15] Irving M. R. and Sterling M. J. H. Optimum network tearing using simulated annealing. IEE Proceedings C, 137,1 (1990) : 69-72.
- [16] Satoh T. and Nara K. Maintenance Scheduling by using simulated annealing method. Power Systems, IEEE Transaction on. , 6,2 (1991):850-856.
- [17] Zhuang F. and Galiana F. D. Unit commitment by simulated annealing, IEEE Transaction, Power systems. 5 (1990):311-317.
- [18] Laarhoven PJMV, Aarts EHL, Lenstra JK. Job shop scheduling by simulated annesling. Operations Research, 40 (1992): 113-25.
- [19] Kalyanmoy Deb, Surendra Gulati. Design of truss-structures for minimum weight using genetic algorithms. Finite element in analysis and design, 37,5 (2001): 447-465.

- [20] Chai Shan, Sun Huanchun, 1997. A combinatorial algorithm for the discrete optimization of structures. Applied mathematics and mechanics, 18, 9 (1997): 847-856.
- [21] M. Kripka, Discrete optimization of trusses by simulated annealing, Journal of the Brazilian Society of Mechanical Sciences and Engineering. 26,2 (2004): 170-173
- [22] Ting-Yu Chen, Jyh-Jye Su, 2002. Efficiency improvement of simulated annealing in optimal structural designs. Advances in engineering software, 33,7-10 (2002): 675-680.
- [23] Elaine Rich. Artificial Intelligence. New York : McGraw-Hill Book, 1991.
- [24] Kentucky University. Simulated Annealing[Online]. (n.d.). Available from : <http://www.ccs.uky.edu/csep/mo/node28.html> [2004, March 21]
- [25] Boonserm Kijirikul. Artificial Intelligence[Online]. (n.d.). Available from : <http://zeus.cp.eng.chula.ac.th/~boonserm/teaching/ai1.0.2.pdf> [2006, October 8]
- [26] Dennis Lam, Thien-Cheong Ang, Sing-Ping Chiew. Structural Steelwork Design to Limit State Theory: Elsevier Butterworth-Heinemann, 2004.
- [27] N.S. Trahair, M.A. Bradford, D.A. Nethercot . The Behavior and Design of Steel Structures to BS5950. London: Spon Press, Inc, 2001.
- [28] AISC, Specification for Structural Steel Buildings. Chicago, Illinois : American Institute of steel Construction, March 9, 2005.
- [29] Richard Rosen. Matrix bandwidth minimization. Proceedings of the 1968 23rd ACM national conference 23 (1968) :585-595.
- [30] E. Cuthill and J.McKee. Reducing the bandwidth of sparse symmetric matrices. Proceedings of the 1969 24th ACM national conference 24 (1968): 157-172.
- [31] Ferdinand P. Beer and E. Russell Johnston. Jr.. Mechanics of materials Berkshire England : McGraw-Hill Inc, 1992.
- [32] Alongkorn Lamom and Tanongsak Bisarnsin. Structural Analysis Software for Education. Proceeding of Eighth National Convention on Civil Engineering 8 (2002) : 119-123.
- [33] Tartaglione, Louis, C. Structural Analysis. Singapore : McGraw-Hill Book Co., 1991.
- [34] Aslam Kassimali. Structural Analysis. United States : Thomson, 2005.

ภาคผนวก

ภาคผนวก ก.

ตารางที่ ก.1 ตารางคุณสมบัติท่อเหล็กกลมกลวงที่ใช้ในการทดลอง

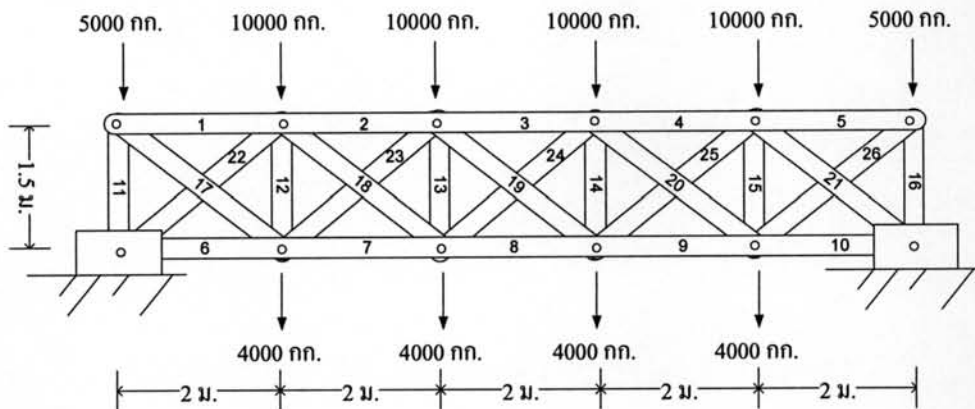
ID	Section	Area (m ²)	Modulus of Elastic (Kg/m ²)	Moment of Inertia (m ⁴)	r (cm)	S _x (cm ³)	D (cm)	t (cm)
1	P15X2	0.000124	2.00E+10	6.07E-09	0.7	0.56	21.7	2
2	P20X2	0.000158	2.00E+10	1.26E-08	0.89	0.93	27.2	2
3	P20X2.3	0.00018	2.00E+10	1.41E-08	0.88	1.03	27.2	2.3
4	P25X2.3	0.000229	2.00E+10	2.89E-08	1.12	1.7	34	2.3
5	P32X2.3	0.000292	2.00E+10	5.97E-08	1.43	2.8	42.7	2.3
6	P40X2.3	0.000335	2.00E+10	8.99E-08	1.64	3.7	48.6	2.3
7	P40X2.8	0.000403	2.00E+10	1.06E-07	1.62	4.36	48.6	2.8
8	P50X2.3	0.000421	2.00E+10	1.78E-07	2.06	5.9	60.5	2.3
9	P40X3.2	0.000456	2.00E+10	1.18E-07	1.61	4.86	48.6	3.2
10	P50X3.2	0.000576	2.00E+10	2.37E-07	2.03	7.84	60.5	3.2
11	P65X2.8	0.000647	2.00E+10	4.37E-07	2.6	11.5	76.3	2.8
12	P50X4	0.00071	2.00E+10	2.85E-07	2	9.41	60.5	4
13	P65X3.2	0.000735	2.00E+10	4.92E-07	2.59	12.9	76.3	3.2
14	P80X2.8	0.000759	2.00E+10	7.07E-07	3.05	15.9	89.1	2.8
15	P80X3.2	0.000864	2.00E+10	7.98E-07	3.04	17.9	89.1	3.2
16	P90X3.2	0.000989	2.00E+10	1.20E-06	3.48	23.6	101.6	3.2
17	P100X3.2	0.001117	2.00E+10	1.72E-06	3.93	30.2	114.3	3.2
18	P90X4	0.001226	2.00E+10	1.46E-06	3.45	28.8	101.6	4
19	P125X3.6	0.00154	2.00E+10	3.57E-06	4.82	51.1	139.8	3.6
20	P100X4.5	0.001552	2.00E+10	2.34E-06	3.89	41	114.3	4.5
21	P125X4	0.001707	2.00E+10	3.94E-06	4.8	56.3	139.8	4
22	P125X4.5	0.001913	2.00E+10	4.38E-06	4.79	62.7	139.8	4.5
23	P150X4.5	0.002272	2.00E+10	7.34E-06	5.68	88.9	165.2	4.5
24	P150X5	0.002516	2.00E+10	8.08E-06	5.67	97.8	165.2	5
25	P125X6	0.002522	2.00E+10	5.66E-06	4.74	80.9	139.8	6
26	P200X4.5	0.002994	2.00E+10	1.68E-05	7.49	155	216.3	4.5
27	P150X6	0.003001	2.00E+10	9.52E-06	5.63	115	165.2	6
28	P200x5.8	0.003836	2.00E+10	2.12E-05	7.45	197	216.3	5.8
29	P200X7	0.004603	2.00E+10	2.52E-05	7.4	233	216.3	7

โดยที่ r = รัศมีไจเรชัน S_x = โมเมนต์หน้าตัด D = เส้นผ่านศูนย์กลางท่อเหล็กกลม t = ความหนาท่อเหล็กกลมกลวง

ภาคผนวก ข.

ข.1 ตัวอย่างการตรวจสอบความถูกต้องของคำตอบการเลือกขนาดวัสดุที่เหมาะสม

เพื่อแสดงให้เห็นแนวทางในการตรวจสอบความถูกต้องของคำตอบที่ได้จากการทำงานของอัลกอริทึมการจำลองการอบเหนียวและฮิวริสติกอัลกอริทึมสำหรับการเลือกขนาดวัสดุ ดังนั้นในภาคผนวก ข. จะสาธิตวิธีการตรวจสอบคำตอบโดยใช้ตัวอย่างที่ 2 ดังแสดงอยู่ในบทที่ 4 และบทที่ 5 เป็นตัวอย่างเพื่อการสาธิต ตัวอย่างที่ 2 เป็น โครงถักกระนาบมีองค์อาคารจำนวน 26 ชิ้น มีลักษณะและแรงกระทำดังรูปที่ ข.1 สูง 1.5 ม. แต่ละช่วงห่าง 2 ม. การเชื่อมต่อระหว่างองค์อาคารในทุกจุดเป็นแบบข้อหมุน ยกเว้นองค์อาคารหมายเลข 1, 2, 3, 4 และ 5 ที่เชื่อมต่อเป็นชั้นเดียวกัน และองค์อาคารหมายเลข 6, 7, 8, 9 และ 10 ที่เชื่อมต่อเป็นชั้นเดียวกันแบบข้อแข็ง มีฐานรองรับเป็นแบบที่รองรับชนิดหมุน หน่วยแรงดึงที่จุดคราก 2500 กก./ตร.ซม. โมดูลัสยืดหยุ่นของเหล็ก 2000000 กก./ตร.ซม.



รูปที่ ข.1 รูปแบบการเลือกขนาดวัสดุใน โครงสร้าง

จำนวนชุดวัสดุที่ต้องการเลือกมีทั้งหมด 5 ชุด ข้อมูลการแบ่งชุดวัสดุแสดงดังตารางที่ ข.1

ตารางที่ ข.1 ข้อมูลการแบ่งชุดหน้าตัดสำหรับตัวอย่างที่ 2

ชุดหน้าตัด	ชิ้นส่วน
1	1-5
2	6-10
3	11-16
4	17-20, 23-26
5	21, 22

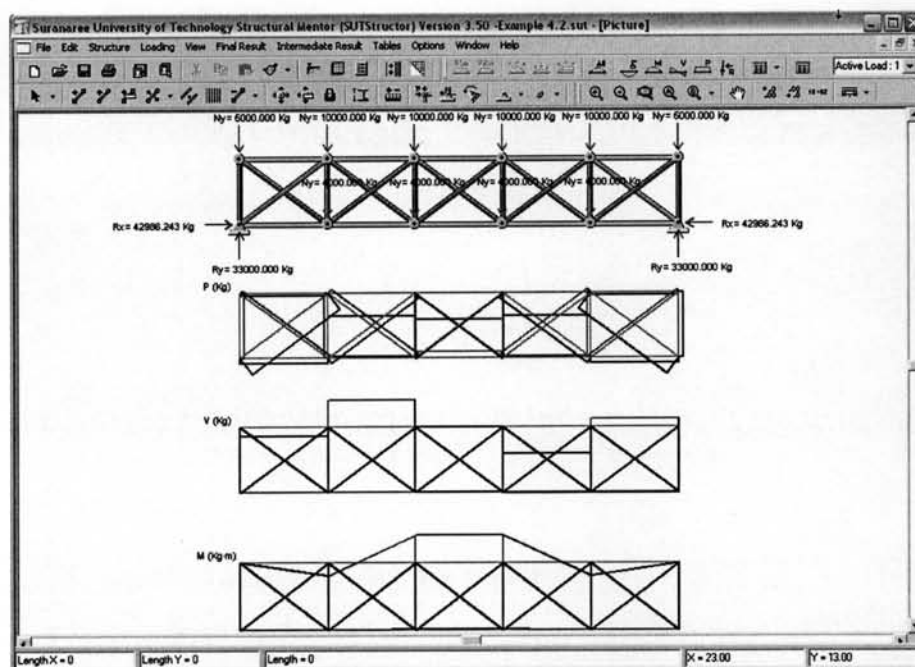
คำตอบที่ได้จากการเลือกหน้าตัด โดยอัลกอริทึมการจำลองการอบเหนียวและฮิวริสติก อัลกอริทึมสำหรับการเลือกขนาดวัสดุแสดงดังตารางที่ ข.2

ตารางที่ ข.2 ผลการเลือกหน้าตัด

ชุดหน้าตัด	ผลการเลือกหน้าตัดโดยอัลกอริทึมการจำลองการอบเหนียว
1	P200x7
2	P90x4
3	P65x3.2
4	P100x3.2
5	P200x4.5
ปริมาตรเหล็ก	102215 ลบ.ซม.

เพื่อตรวจสอบว่าหน้าตัดที่ถูกเลือกมาดังแสดงในตารางที่ ข.1 เป็นคำตอบที่ถูกต้องสามารถรับน้ำหนักบรรทุกได้ถูกต้องตามมาตรฐานหรือไม่ มีกระบวนการหลัก 2 ส่วน คือ

- 1) จำลองพฤติกรรมโครงสร้างเพื่อหาแรงที่เกิดขึ้นกับโครงสร้าง
- 2) ตรวจสอบว่าขนาดหน้าตัดที่เลือกเพียงพอกับแรงที่เกิดขึ้นหรือไม่โดยสมการที่แสดงในบทที่ 2 หัวข้อที่ 2.8



รูปที่ ข.2 แผนผังแรงที่เกิดขึ้นกับโครงสร้างที่ได้จากการจำลองพฤติกรรม

ผลการจำลองพฤติกรรมโครงสร้างแสดงดังรูปที่ ข.2 ซึ่งในรูปจะแสดงแผนผังแรงตามแนวแกน แรงเฉือน และแรงดัด ได้ลงมาตามลำดับ ส่วนรายละเอียดของแรงต่างๆ ที่เกิดขึ้นในแต่ละองค์อาคารแสดงอย่างละเอียดในตารางที่ ข.3

ตารางที่ ข.3 รายละเอียดแรงภายในที่เกิดในแต่ละองค์อาคาร

องค์อาคาร	แรงอัดสูงสุด (กก.)	แรงดึงสูงสุด (กก.)	แรงเฉือนสูงสุด (กก.)	แรงดัดบวกสูงสุด (กก.-ม.)	แรงดัดลบสูงสุด (กก.-ม.)
1	5454.306	0	189.996	0	379.993
2	45851.946	0	600.554	821.115	379.993
3	54499.578	0	0	821.115	0
4	45851.946	0	600.554	821.115	379.993
5	5454.306	0	189.996	0	379.993
6	10859.867	0	4.485	8.969	0
7	0	4172.635	12.351	33.672	0
8	0	13374.463	0	33.672	0
9	0	4172.635	12.351	33.672	0
10	10859.867	0	4.485	8.969	0
11	8900.733	0	0	0	0
12	0	7100.784	0	0	0
13	1517.876	0	0	0	0
14	1517.876	0	0	0	0
15	0	7100.784	0	0	0
16	8900.733	0	0	0	0
17	0	6817.882	0	0	0
18	0	10339.08	0	0	0
19	1163.206	0	0	0	0
20	11972.745	0	0	0	0
21	40157.971	0	0	0	0
22	40157.971	0	0	0	0
23	11972.745	0	0	0	0
24	1163.206	0	0	0	0
25	0	10339.08	0	0	0
26	0	6817.882	0	0	0

ข.1.1 การตรวจสอบความสามารถในการรับแรงของชุดหน้าตัดที่ 1

ชุดหน้าตัดที่ 1 เกิดแรงอัดสูงสุดที่องค์อาคารหมายเลข 3 ซึ่งในชุดหน้าตัดที่ 1 ประกอบด้วยองค์อาคารหมายเลข 1 ถึง 5 และหน้าตัดที่เลือกสำหรับชุดหน้าตัดที่ 1 คือ P200x7 มีพื้นที่หน้าตัด 46.03 ตร.ซม. รัศมีจอยเรชัน 7.4 ซม. ตามหลักการที่แสดงในหัวข้อที่ 2.8.2 เรื่ององค์อาคารรับแรงอัด การตรวจสอบความสามารถในการรับแรงเป็นดังนี้

กำหนดหา λ และ λ_c เพื่อตรวจสอบว่าองค์อาคารมีพฤติกรรมแบบใดจากสมการที่ 2.9 และสมการที่ 2.10 ในบทที่ 2

$$\lambda = \frac{KL}{r} = \frac{1 \times 200}{7.4} = 27.03 \quad (\text{ข.1})$$

$$\lambda_c = 4.71 \sqrt{\frac{E}{F_y}} = 4.71 \sqrt{\frac{2000000}{2500}} = 133 \quad (\text{ข.2})$$

เมื่อค่า $\lambda < \lambda_c$ เสาจะมีพฤติกรรมการพังในช่วงอินอีลาสติก ซึ่งแรงอัดที่ยอมให้เกิดในองค์อาคารจะคำนวณได้จากสมการที่ 2.11

$$P_a = \left(0.658 \frac{F_y}{F_e} \right) \frac{F_y A_g}{\Omega_c} = \left(0.658 \frac{2500}{27007} \right) \frac{2500 \times 46.03}{1.67} = 66213 \text{ กก.} \quad (\text{ข.3})$$

โดยที่ F_e คำนวณได้จาก

$$F_e = \frac{\pi^2 E}{\left(\frac{KL}{r} \right)^2} = \frac{3.141^2 \times 2000000}{(27.03)^2} = 27007 \text{ กก./ตร.ซม.} \quad (\text{ข.4})$$

เมื่อเปรียบเทียบแรงอัดที่ยอมให้ 66213 กก. กับแรงอัดที่เกิดขึ้นในองค์อาคารที่ 3 ขนาด 54500 กก. พบว่าแรงอัดที่ยอมให้มีความมากกว่าแรงอัดที่เกิดขึ้นในองค์อาคาร ดังนั้นหน้าตัดที่เลือกสำหรับองค์อาคารนี้สามารถรับแรงอัดที่เกิดขึ้นในองค์อาคารได้ แต่เนื่องจากองค์อาคารนี้มีแรงอัดเกิดขึ้นในองค์อาคารด้วย จึงจำเป็นต้องตรวจสอบความสามารถในการรับแรงอัดและแรงคดพร้อมกันตามสมการที่ 2.15 หรือ 2.16 จากหลักการที่แสดงในหัวข้อที่ 2.8.5 เรื่ององค์อาคารรับแรงอัดและแรงคดพร้อมกัน สามารถตรวจสอบความสามารถในการรับแรงได้ดังนี้

ตรวจสอบอัตราส่วนของแรงที่เกิดขึ้นจริงกับแรงที่ยอมให้สำหรับแรงในแนวแกนดังนี้

$$\frac{P_r}{P_c} = \frac{54500}{66213} = 0.82 \quad (\text{ข.5})$$

ถ้าอัตราส่วนของแรงที่เกิดขึ้นจริงกับแรงที่ยอมให้สำหรับแรงในแนวแกนมีค่าน้อยกว่า 0.2 จะทำการตรวจสอบความสามารถในการรับน้ำหนักบรรทุกโดยสมการที่ 2.16

$$\frac{P_r}{P_c} + \frac{8}{9} \left(\frac{M_{rx}}{M_{cx}} \right) = \frac{54500}{66213} + \frac{8}{9} \left(\frac{883}{7400} \right) = 0.93 \leq 1 \quad (\text{ข.6})$$

โดยที่ M_{cx} คำนวณได้จาก

$$M_{cx} = \frac{F_y \times 1.27 \times S_x}{\Omega} = \frac{2500 \times 1.27 \times 233}{1.67} = 7400 \text{ กก.-ม.} \quad (\text{ข.7})$$

และ M_{rx} คำนวณได้จาก

$$M_{rx} = B_1 M_m = 1.075 \times 821 = 883 \text{ กก.-ม.} \quad (\text{ข.8})$$

โดยที่

$$B_1 = \frac{C_m}{(1 - 1.6 P_r / P_{el})} = \frac{1}{(1 - 1.6 \times 54500 / 1243101)} = 1.075 \geq 1 \quad (\text{ข.9})$$

เมื่อ

$$C_m = 0.6 - 0.4(M_1/M_2) = 0.6 - 0.4(-1 \times (821/821)) = 1 \quad (\text{ข.10})$$

และ

$$P_{el} = \frac{\pi^2 EI}{(KL)^2} = \frac{3.141^2 \times 2000000 \times 2520}{(1 \times 200)^2} = 1243101 \text{ กก.} \quad (\text{ข.11})$$

จากสมการที่ ข.6 ค่าที่ได้มีค่าน้อยกว่า 1 แสดงว่าองค์อาคารนี้สามารถรับแรงที่เกิดขึ้นในโครงสร้างได้ ดังนั้นการตรวจสอบความสามารถในการรับแรงอัดและแรงดัดร่วมกันของชุดหน้าตัดที่ 1 ผ่านการตรวจสอบ

ข.1.2 การตรวจสอบความสามารถในการรับแรงของชุดหน้าตัดที่ 2

ชุดหน้าตัดที่ 2 แรงที่เกิดขึ้นสูงสุดเป็นแรงดัดเกิดขึ้นที่องค์อาคารหมายเลข 8 ซึ่งในชุดหน้าตัดที่ 2 ประกอบด้วยองค์อาคารหมายเลข 6 ถึง 10 และหน้าตัดที่เลือกสำหรับชุดหน้าตัดที่ 1 คือ P90x4 มีพื้นที่หน้าตัด 46.03 ตร.ซม. ตามหลักการที่แสดงในหัวข้อที่ 2.8.1 เรื่ององค์อาคารรับแรงดัด สามารถตรวจสอบความสามารถในการรับแรงได้ดังนี้

หน้าหนักบรรทุกสูงสุดที่ยอมให้เกิดขึ้นได้ดังสมการ

$$T_a \leq F_y A_g / \Omega_t = 2500 \times 12.26 / 1.67 = 18353 \text{ กก.} \quad (\text{ข.12})$$

เมื่อเปรียบเทียบแรงดัดที่ยอมให้ 18353 กก. กับแรงดัดที่เกิดขึ้นในองค์อาคารที่ 8 ขนาด 13374 กก. พบว่าแรงดัดที่ยอมให้มีความมากกว่าแรงดัดที่เกิดขึ้นในองค์อาคาร ดังนั้นหน้าตัดที่เลือกสำหรับองค์อาคารนี้สามารถรับแรงที่เกิดขึ้นในองค์อาคารได้

ข.1.3 การตรวจสอบความสามารถในการรับแรงของชุดหน้าตัดที่ 3

ชุดหน้าตัดที่ 3 เกิดแรงอัดสูงสุดที่องค์อาคารหมายเลข 11 ซึ่งในชุดหน้าตัดที่ 3 ประกอบด้วยองค์อาคารหมายเลข 11 ถึง 16 และหน้าตัดที่เลือกสำหรับชุดหน้าตัดที่ 11 คือ P65x3.2 มีพื้นที่หน้าตัด 7.35 ตร.ซม. รัศมีไจเรชัน 2.59 ซม. ตามหลักการที่แสดงในหัวข้อที่ 2.8.2 เรื่ององค์อาคารรับแรงอัด การตรวจสอบความสามารถในการรับแรงเป็นดังนี้

คำนวณหา λ และ λ_c เพื่อตรวจสอบว่าองค์อาคารมีพฤติกรรมแบบใดจากสมการที่ 2.9 และสมการที่ 2.10

$$\lambda = \frac{KL}{r} = \frac{1 \times 150}{2.59} = 57.92 \quad (\text{ข.13})$$

$$\lambda_c = 4.71 \sqrt{\frac{E}{F_y}} = 4.71 \sqrt{\frac{2000000}{2500}} = 133 \quad (\text{ข.14})$$

เมื่อค่า $\lambda < \lambda_c$ เสาจะมีพฤติกรรมการพังในช่วงอินอีลาสติก ซึ่งแรงดัดที่ยอมให้เกิดขึ้นในองค์อาคารจะคำนวณได้จากสมการที่ 2.11

$$P_a = \left(0.658^{\frac{F_y}{F_{ce}}} \right) \frac{F_y A_g}{\Omega_c} = \left(0.658^{\frac{2500}{5882}} \right) \frac{2500 \times 7.35}{1.67} = 9210 \text{ กก.} \quad (\text{ข.15})$$

โดยที่ F_{ce} คำนวณได้จาก

$$F_{ce} = \frac{\pi^2 E}{\left(\frac{KL}{r} \right)^2} = \frac{3.141^2 \times 2000000}{(57.92)^2} = 5882 \text{ กก./ตร.ซม.} \quad (\text{ข.16})$$

เมื่อเปรียบเทียบแรงดัดที่ยอมให้ 9210 กก. กับแรงดัดที่เกิดขึ้นในองค์อาคารที่ 11 ขนาด 8901 กก. พบว่าแรงดัดที่ยอมให้มีความมากกว่าแรงดัดที่เกิดขึ้นในองค์อาคาร ดังนั้นหน้าตัดที่เลือกสำหรับองค์อาคารนี้สามารถรับแรงที่เกิดขึ้นในองค์อาคารได้

ข.1.4 การตรวจสอบความสามารถในการรับแรงของชุดหน้าตัดที่ 4

ชุดหน้าตัดที่ 4 เกิดแรงอัดสูงสุดที่องค์อาคารหมายเลข 20 ซึ่งในชุดหน้าตัดที่ 4 ประกอบด้วยองค์อาคารหมายเลข 17 ถึง 20 และ 23 ถึง 26 หน้าตัดที่ถูกเลือกสำหรับชุดหน้าตัดที่ 4 คือ P100x3.2 มีพื้นที่หน้าตัด 11.17 ตร.ซม. รัศมีไจเรชัน 3.93 ซม. ตามหลักการที่แสดงในหัวข้อที่ 2.8.2 เรื่ององค์อาคารรับแรงอัด การตรวจสอบความสามารถในการรับแรงเป็นดังนี้

คำนวณหา λ และ λ_c เพื่อตรวจสอบว่าองค์อาคารมีพฤติกรรมแบบใดจากสมการที่ 2.9 และสมการที่ 2.10

$$\lambda = \frac{KL}{r} = \frac{1 \times 250}{3.93} = 63.61 \quad (\text{ข.17})$$

$$\lambda_c = 4.71 \sqrt{\frac{E}{F_y}} = 4.71 \sqrt{\frac{2000000}{2500}} = 133 \quad (\text{ข.18})$$

เมื่อค่า $\lambda < \lambda_c$ เสาจะมีพฤติกรรมการพังในช่วงอินอีลาสติก ซึ่งแรงอัดที่ยอมให้เกิดในองค์อาคารจะคำนวณได้จากสมการที่ 2.11

$$P_a = \left(0.658^{\frac{F_y}{F_e}} \right) \frac{F_y A_g}{\Omega_c} = \left(0.658^{\frac{2500}{4877}} \right) \frac{2500 \times 11.17}{1.67} = 13493 \text{ กก.} \quad (\text{ข.19})$$

โดยที่ F_e คำนวณได้จาก

$$F_e = \frac{\pi^2 E}{\left(\frac{KL}{r} \right)^2} = \frac{3.141^2 \times 2000000}{(63.61)^2} = 4877 \text{ กก./ตร.ซม.} \quad (\text{ข.20})$$

เมื่อเปรียบเทียบแรงอัดที่ยอมให้ 13493 กก. กับแรงอัดที่เกิดขึ้นในองค์อาคารที่ 20 ขนาด 11973 กก. พบว่าแรงอัดที่ยอมให้มีความมากกว่าแรงอัดที่เกิดขึ้นในองค์อาคาร ดังนั้นหน้าตัดที่เลือกสำหรับองค์อาคารนี้สามารถรับแรงที่เกิดขึ้นในองค์อาคารได้

ข.1.5 การตรวจสอบความสามารถในการรับแรงของชุดหน้าตัดที่ 5

ชุดหน้าตัดที่ 5 เกิดแรงอัดสูงสุดที่องค์อาคารหมายเลข 21 หน้าตัดที่ถูกเลือกสำหรับชุดหน้าตัดที่ 5 คือ P200x4.5 มีพื้นที่หน้าตัด 29.94 ตร.ซม. รัศมีไจเรชัน 7.49 ซม. ซึ่งในชุดหน้าตัดที่ 5 ประกอบด้วยองค์อาคารหมายเลข 21 และ 16 ตามหลักการที่แสดงในหัวข้อที่ 2.8.2 เรื่ององค์อาคารรับแรงอัด การตรวจสอบความสามารถในการรับแรงเป็นดังนี้

คำนวณหา λ และ λ_c เพื่อตรวจสอบว่าองค์อาคารมีพฤติกรรมแบบใดจากสมการที่ 2.9 และสมการที่ 2.10

$$\lambda = \frac{KL}{r} = \frac{1 \times 250}{7.49} = 33.38 \quad (\text{ข.21})$$

$$\lambda_c = 4.71 \sqrt{\frac{E}{F_y}} = 4.71 \sqrt{\frac{2000000}{2500}} = 133 \quad (\text{ข.22})$$

เมื่อค่า $\lambda < \lambda_c$ เสาจะมีพฤติกรรมการพังในช่วงอินอีลาสติก ซึ่งแรงอัดที่ยอมให้เกิดในองค์อาคารจะคำนวณได้จากสมการที่ 2.11

$$P_a = \left(0.658^{\frac{F_y}{F_e}} \right) \frac{F_y A_g}{\Omega_c} = \left(0.658^{\frac{2500}{17709}} \right) \frac{2500 \times 29.94}{1.67} = 42249 \text{ กก.} \quad (\text{ข.23})$$

โดยที่ F_e คำนวณได้จาก

$$F_e = \frac{\pi^2 E}{\left(\frac{KL}{r} \right)^2} = \frac{3.141^2 \times 2000000}{(33.38)^2} = 17709 \text{ กก./ตร.ซม.} \quad (\text{ข.24})$$

เมื่อเปรียบเทียบแรงอัดที่ยอมให้ 42249 กก. กับแรงอัดที่เกิดขึ้นในองค์อาคารที่ 21 ขนาด 40158 กก. พบว่าแรงอัดที่ยอมให้มีความมากกว่าแรงอัดที่เกิดขึ้นในองค์อาคาร ดังนั้นหน้าตัดที่เลือกสำหรับองค์อาคารนี้สามารถรับแรงที่เกิดขึ้นในองค์อาคารได้ จากการตรวจสอบ ทุกชุดหน้าตัดสามารถรับแรงที่เกิดขึ้นในองค์อาคารได้ทั้งหมด ดังนั้นค้ำคอบหรือชุดหน้าตัดที่ถูกเลือกโดยอัลกอริทึมการจำลองการอบเหนียวและฮิวริสติกอัลกอริทึมจึงเป็นค้ำคอบที่ถูกต้อง

ภาคผนวก ก.

การทดสอบทางสถิติในการวิจัย

ก.1 การทดสอบทางสถิติในการวิจัย

1. กำหนดค่าสถิติเบื้องต้น เช่น ค่าเฉลี่ย ค่าเบี่ยงเบนมาตรฐาน ค่าความแปรปรวน ค่าฐานนิยม
2. ใช้การแจกแจงวิธี T-test ในการทดสอบเพื่อตรวจสอบความแตกต่างของค่าตอบอย่างมีนัยสำคัญ โดยมีขั้นตอนดังนี้
 - การเปรียบเทียบเชิงปริมาณของผลลัพธ์ระหว่างวิธีการจำลองอบเหนือกับวิธีที่ใช้อยู่ในปัจจุบัน
 - การเปรียบเทียบเชิงจำนวนรอบของการคำนวณค่าตอบที่เหมาะสมระหว่างวิธีการจำลองอบเหนือกับวิธีฮิวริสติก

ก.2 ขั้นตอนการเปรียบเทียบเชิงปริมาณของผลลัพธ์ระหว่างวิธีการจำลองอบเหนือกับวิธีที่ใช้อยู่ในปัจจุบัน

1. การตั้งสมมติฐานเพื่อทดสอบ
 - สมมติฐานหลัก (H_0): ค่าเฉลี่ยของปริมาณผลลัพธ์ที่เหมาะสมด้วยวิธีการจำลองการอบเหนือ มีค่าเท่ากับค่าเฉลี่ยของปริมาณผลลัพธ์ที่เหมาะสมด้วยวิธีที่ใช้ในปัจจุบัน
 - สมมติฐานรอง (H_1): ค่าเฉลี่ยของปริมาณผลลัพธ์ที่เหมาะสมด้วยวิธีการจำลองการอบเหนือมีความแตกต่างกับค่าเฉลี่ยของปริมาณผลลัพธ์ที่เหมาะสมด้วยวิธีที่ใช้ในปัจจุบัน

2. กำหนดสถิติทดสอบ
$$t = \frac{\bar{x} - a}{\frac{S.D.}{\sqrt{n}}}$$

โดยที่ค่า x = ค่าเฉลี่ยของชุดข้อมูลที่นำมาเปรียบเทียบ

a = ค่าเฉลี่ยของชุดข้อมูลมาตรฐาน

S.D. = ค่าเบี่ยงเบนมาตรฐานของชุดข้อมูลที่นำมาเปรียบเทียบ

n = จำนวนชุดข้อมูล

3. กำหนดระดับนัยสำคัญหรือ α ของการทดสอบ

4. การหาค่าวิกฤติเพื่อสรุปผลการทดสอบ : $t_{1-\frac{\alpha}{2}, n-1}$



5. การสรุปผลทดสอบ

5.1 ปฏิเสธ H_0 (ยอมรับ H_1) ถ้าค่าสถิติทดสอบ $|t|$ ที่คำนวณได้มากกว่า $t_{1-\frac{\alpha}{2}, n-1}$

ค.3 การเปรียบเทียบเชิงจำนวนรอบของของการคำนวณหาค่าตอบที่เหมาะสมระหว่างวิธีการจำลอง
รอบเหนียวกับวิธีฮิวริสติก

พิจารณาค่ายกัน เพียงแต่เปลี่ยนสมมุติฐานดังนี้

สมมุติฐานหลัก (H_0) : ค่าเฉลี่ยของจำนวนรอบในการหาค่าตอบที่เหมาะสมด้วยวิธีการ
จำลองการอบเหนียว มีค่าเท่ากับค่าเฉลี่ยของจำนวนรอบในการหาค่าตอบที่เหมาะสมด้วยวิธีฮิวริ
สติก

สมมุติฐานรอง (H_1) : ค่าเฉลี่ยของจำนวนรอบในการหาค่าตอบที่เหมาะสมด้วยวิธีการ
จำลองการอบเหนียว มีความแตกต่างกับค่าเฉลี่ยของจำนวนรอบในการหาค่าตอบที่เหมาะสมด้วย
วิธีฮิวริสติก

$$2. \text{ คำนวณสถิติทดสอบ } t = \frac{\bar{x} - a}{\frac{S.D.}{\sqrt{n}}}$$

โดยที่ค่า x = ค่าเฉลี่ยของชุดข้อมูลที่นำมาเปรียบเทียบ

a = ค่าเฉลี่ยของชุดข้อมูลมาตรฐาน

S.D. = ค่าเบี่ยงเบนมาตรฐานของชุดข้อมูลที่นำมาเปรียบเทียบ

n = จำนวนชุดข้อมูล

3. กำหนดระดับนัยสำคัญหรือ α ของการทดสอบ

4. การหาค่าวิกฤติเพื่อสรุปผลการทดสอบ : $t_{1-\frac{\alpha}{2}, n-1}$

5. การสรุปผลทดสอบ

5.1 ปฏิเสธ H_0 (ยอมรับ H_1) ถ้าค่าสถิติทดสอบ $|t|$ ที่คำนวณได้มากกว่า $t_{1-\frac{\alpha}{2}, n-1}$

ค.4 ผลการศึกษา

การเปรียบเทียบประสิทธิภาพของวิธีการหาปริมาตรเหล็ก โดยการวิเคราะห์สถิติเบื้องต้น
สรุปไว้ในตารางที่ ค.1 - ค.5 และรูปภาพที่ ค.1 - ค.4

ตารางที่ ค.1 ปริมาตรเหล็กและจำนวนรอบในการหาค่าตอบ โดยวิธีการจำลองอบเหนียว

ค่าสถิติ	ตัวอย่างที่							
	1		2		3		4	
	ปริมาตรเหล็ก ลบ.ซม.	จำนวนรอบ	ปริมาตรเหล็ก ลบ.ซม.	จำนวนรอบ	ปริมาตรเหล็ก ลบ.ซม.	จำนวนรอบ	ปริมาตรเหล็ก ลบ.ซม.	จำนวนรอบ
ค่าสูงสุด(Max.)	92407	971	116051	1147	100406	1169	193684	861
ค่าต่ำสุด(Min.)	83073	751	102215	883	86395	938	162485	566
ค่าพิสัย (Range)	9334	220	13836	264	14011	231	31199	295
ค่าเฉลี่ย(Average)	86327.560	873.320	107376.720	999.440	91780.600	1042.760	172373.080	679.280
ค่าเบี่ยงเบนมาตรฐาน(Stdev.)	2507.019	61.988	3866.608	70.573	3331.620	68.308	7080.429	78.943
ค่าความแปรปรวน(Variance)	6285143.757	3842.477	14950657.043	4980.507	11099689.083	4665.940	50132474.327	6231.960
ค่าสัมประสิทธิ์ความผันแปร(Cv)	0.029	0.071	0.036	0.071	0.036	0.066	0.041	0.116
ค่าฐานนิยม (Mode)	83073	938	106634	938	88765	960	-	598
จำนวนฐานนิยม (Count of Mode)	6	3	2	3	4	3	0	2

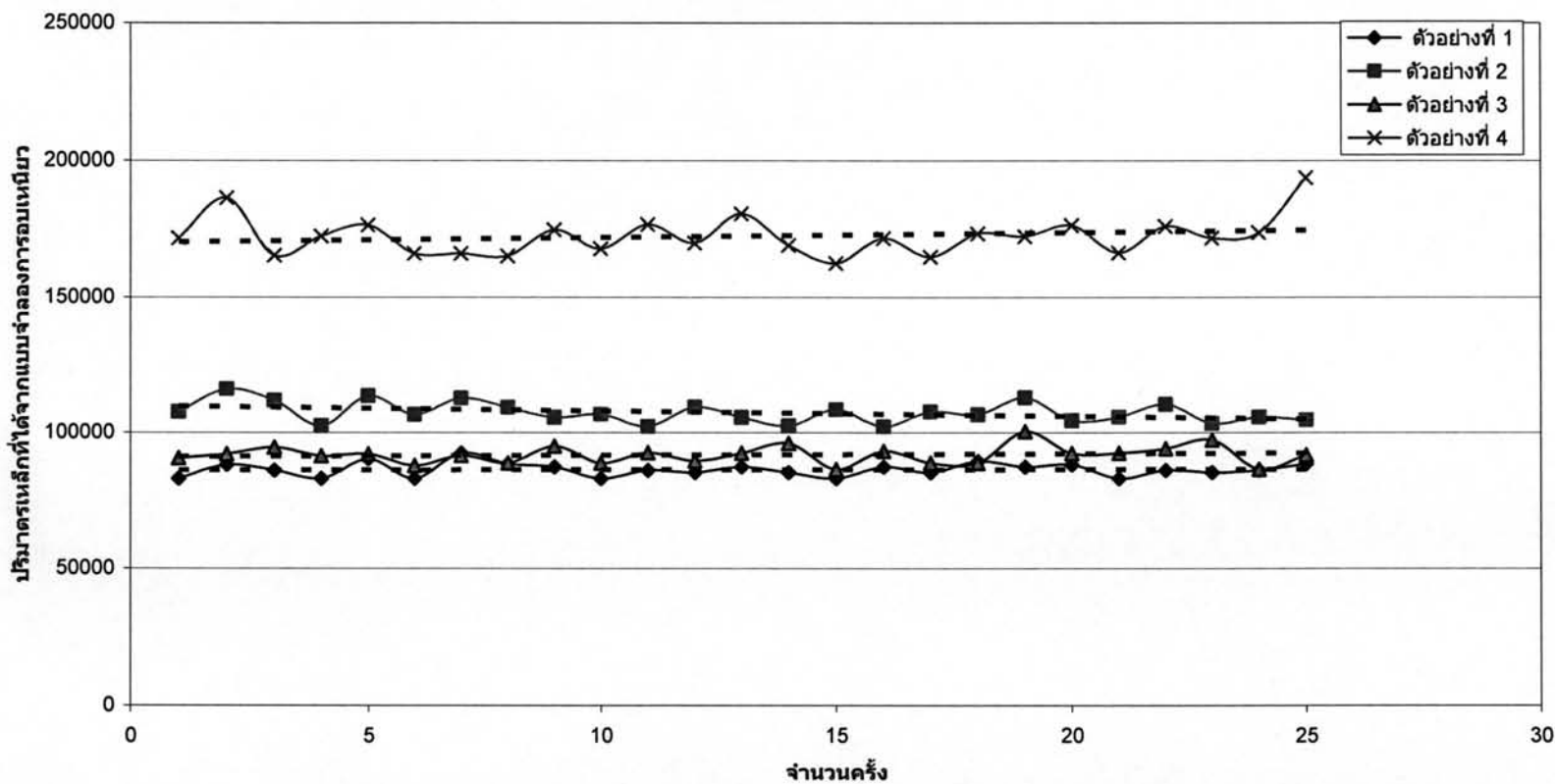
ตารางที่ ค.2 ปริมาตรเหล็กและจำนวนรอบในการหาค่าตอบโดยวิธีฮิวริสติก

ค่าสถิติ	ตัวอย่างที่							
	1		2		3		4	
	ปริมาตรเหล็ก ลบ.ชม.	จำนวนรอบ	ปริมาตรเหล็ก ลบ.ชม.	จำนวนรอบ	ปริมาตรเหล็ก ลบ.ชม.	จำนวนรอบ	ปริมาตรเหล็ก ลบ.ชม.	จำนวนรอบ
ค่าสูงสุด(Max.)	84800	80	124388	44	93769	80	165460	61
ค่าต่ำสุด(Min.)	83073	20	102215	22	86395	26	162485	24
ค่าพิสัย (Range)	1727	60	22173	22	7374	54	2975	37
ค่าเฉลี่ย(Average)	83211.160	41.040	110316.360	29.520	90087.480	36.960	162836.520	39.840
ค่าเบี่ยงเบนมาตรฐาน(Stdev.)	478.185	13.427	5885.570	5.501	2034.806	10.830	750.544	10.359
ค่าความแปรปรวน(Variance)	228660.557	180.290	34639939.907	30.260	4140436.427	117.290	563316.677	107.307
ค่าสัมประสิทธิ์ความผันแปร(Cv)	0.006	0.327	0.053	0.186	0.023	0.293	0.005	0.260
ค่าฐานนิยม (Mode)	83073	38	-	25	89375	38	162485	45
จำนวนฐานนิยม (Count of Mode)	23	2	0	4	5	3	19	2

ตารางที่ ๓.3 ปริมาตรเหล็กและจำนวนรอบ โดยวิธีที่ใช้อยู่ในปัจจุบัน

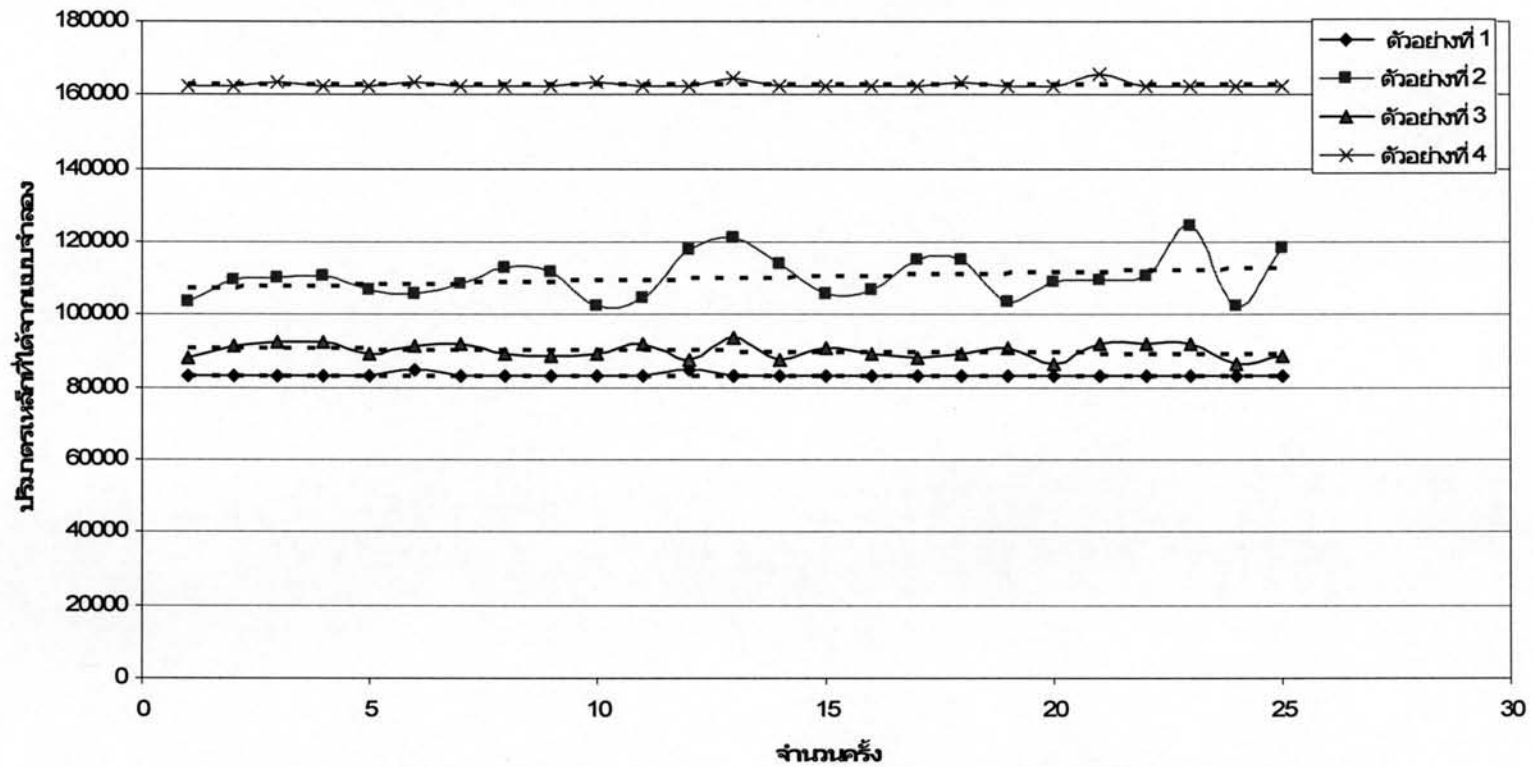
ค่าสถิติ	ตัวอย่างที่							
	1		2		3		4	
	ปริมาตรเหล็ก ลบ.ชม.	จำนวนรอบ	ปริมาตรเหล็ก ลบ.ชม.	จำนวนรอบ	ปริมาตรเหล็ก ลบ.ชม.	จำนวนรอบ	ปริมาตรเหล็ก ลบ.ชม.	จำนวนรอบ
ค่าสูงสุด(Max.)	-	-	-	-	-	-	-	-
ค่าต่ำสุด(Min.)	-	-	-	-	-	-	-	-
ค่าพิสัย (Range)	-	-	-	-	-	-	-	-
ค่าเฉลี่ย(Average)	127658	2	143640	3	107858	3	185234	3
ค่าเบี่ยงเบนมาตรฐาน(Stdev.)	-	-	-	-	-	-	-	-
ค่าความแปรปรวน(Variance)	-	-	-	-	-	-	-	-
ค่าสัมประสิทธิ์ความผันแปร(Cv)	-	-	-	-	-	-	-	-
ค่าฐานนิยม (Mode)	-	-	-	-	-	-	-	-
จำนวนฐานนิยม (Count of Mode)	-	-	-	-	-	-	-	-

ความสัมพันธ์ระหว่างปริมาณเหล็กจากแบบจำลองการอบเหนียวในชุดตัวอย่างที่ 1,2,3 และ 4



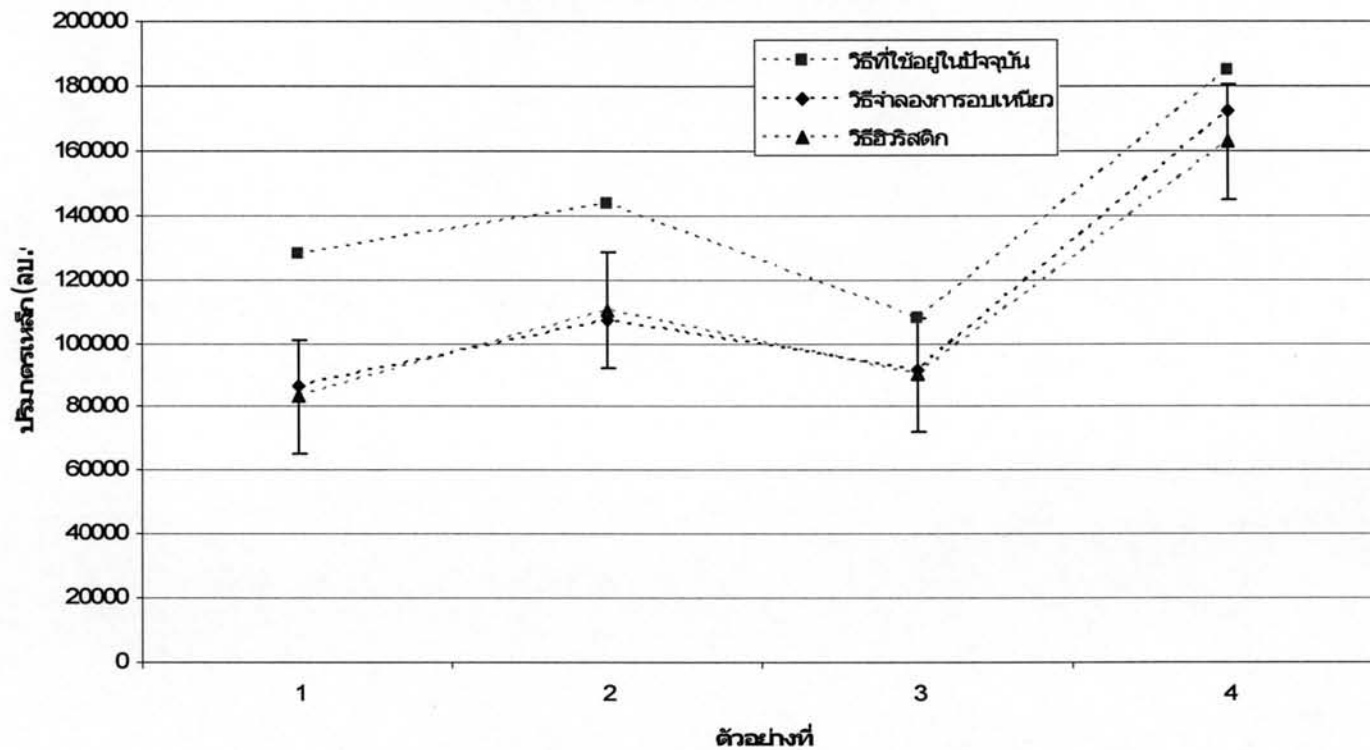
รูปที่ ค.1 ความสัมพันธ์ระหว่างปริมาณเหล็กจากแบบจำลองการอบเหนียวในชุดตัวอย่างที่ 1, 2, 3 และ 4

ความสัมพันธ์ระหว่างปริมาณเหล็กจากแบบจำลองฮิวริสติกในชุดตัวอย่างที่ 1,2,3 และ 4



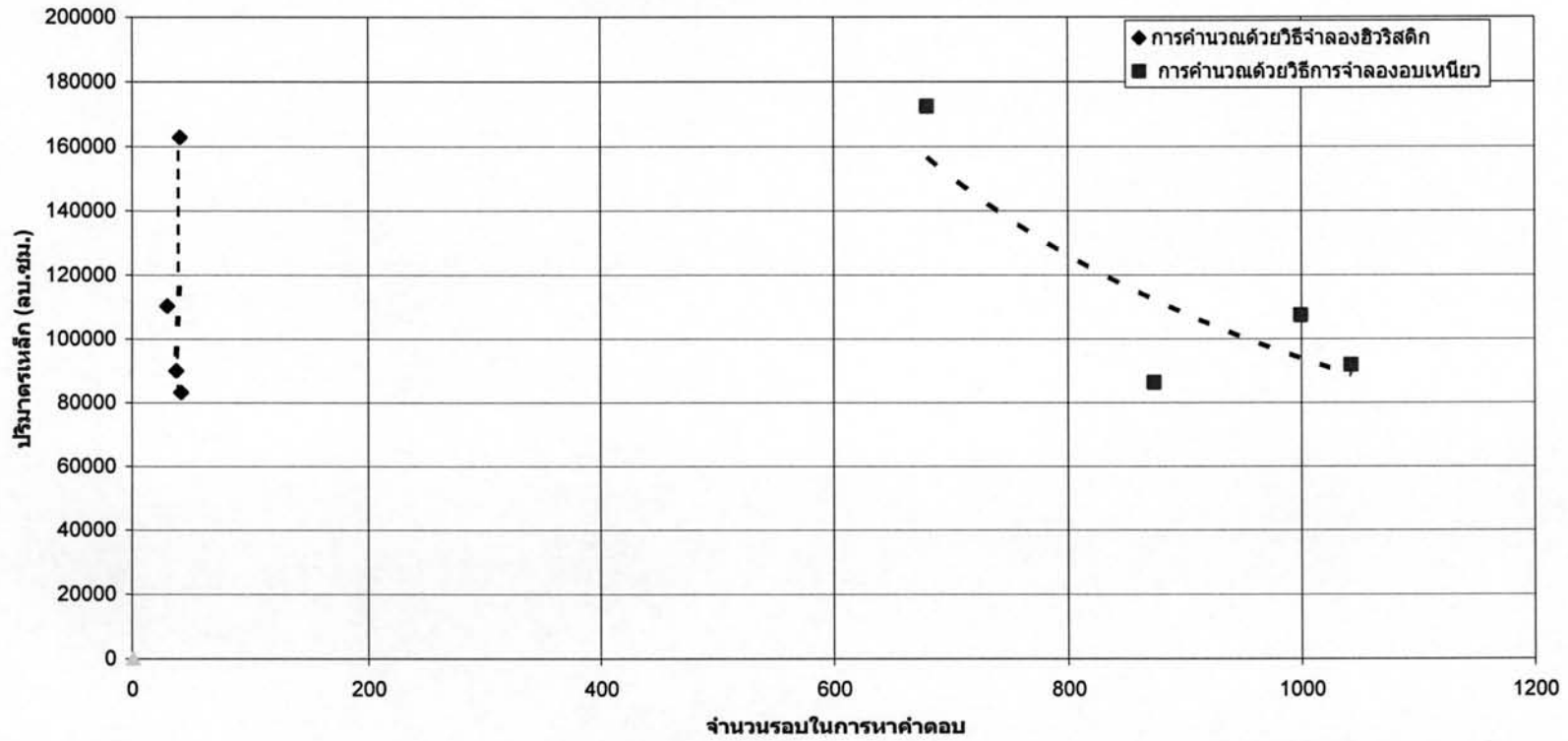
รูปที่ ค.2 ความสัมพันธ์ระหว่างปริมาณเหล็กจากแบบจำลองฮิวริสติกในชุดตัวอย่างที่ 1, 2, 3 และ 4

การเปรียบเทียบปริมาณเหล็กที่ได้จาก 3 วิธี



รูปที่ ค.3 การเปรียบเทียบปริมาณเหล็กที่ได้จาก 3 วิธี

การเปรียบเทียบจำนวนรอบในการหาค่าตอบที่เหมาะสม



รูปที่ ค.4 การเปรียบเทียบจำนวนรอบในการหาค่าตอบที่เหมาะสม

ตารางที่ ค.4 ตารางการคำนวณค่าทางสถิติด้วยวิธี T-test

	n = 25	n-1 = 24						
Reliability (%) =	99.9	99.5	99	95	90	85	80	75
$\mu =$	0.001	0.005	0.010	0.050	0.100	0.150	0.200	0.250
$\mu/2 =$	0.0005	0.0025	0.0050	0.0250	0.0500	0.0750	0.1000	0.1250
$1-\mu/2 =$	0.9995	0.9975	0.9950	0.9750	0.9500	0.9250	0.9000	0.8750
$t_{1-\mu/2, n-1}$	$t_{0.9995, 24}$	$t_{0.9975, 24}$	$t_{0.995, 24}$	$t_{0.975, 24}$	$t_{0.95, 24}$	$t_{0.925, 24}$	$t_{0.9, 24}$	$t_{0.875, 24}$
$t_{1-\mu/2, n-1}$: ค่าวิกฤต	3.745398618	3.090513547	2.7969395	2.063898547	1.710882067	1.487136	1.317836	1.1788845

การสรุปผลทดสอบ ถ้า $|t| > t_{1-\frac{\mu}{2}, n-1}$ ยอมรับ สมมติฐานรอง (H_1)

ปริมาตรเหล็กโดยวิธีทั่วไป	127658	143640	107858	185234
	ตัวอย่างที่ 1	ตัวอย่างที่ 2	ตัวอย่างที่ 3	ตัวอย่างที่ 4
วิธีทั่วไปกับ SA; ปริมาตรเหล็ก; t =	-82.42945442	-46.89288449	-24.128505	-9.082020358
วิธีทั่วไปกับ HA; ปริมาตรเหล็ก; t =	-464.7455776	-28.30960915	-43.666369	-149.2082572
วิธี HA กับวิธี SA; ปริมาตรเหล็ก; t =	-32.58573879	2.497328006	-4.16039613	-63.53096406
จำนวนรอบ โดยวิธี HA กับวิธี SA; t =	-309.9228831	-881.5997476	-464.356087	-308.6430146

หมายเหตุ : การทดสอบนัยสำคัญ เป็นการทดสอบความน่าเชื่อถือของข้อมูล ในการศึกษาี้ เลือกใช้วิธี t-test

ตารางที่ ค.5 ผลการวิเคราะห์ความน่าเชื่อถือของข้อมูลที่ระดับความเชื่อมั่น 99 %

	ตัวอย่างที่ 1	ตัวอย่างที่ 2	ตัวอย่างที่ 3	ตัวอย่างที่ 4
วิธีทั่วไปกับ SA; ปริมาตรเหล็ก ;	Sig. (2-tailed)	Sig. (2-tailed)	Sig. (2-tailed)	Sig. (2-tailed)
วิธีทั่วไปกับ HA; ปริมาตรเหล็ก ;	Sig. (2-tailed)	Sig. (2-tailed)	Sig. (2-tailed)	Sig. (2-tailed)
วิธี HA กับวิธี SA ; ปริมาตรเหล็ก ;	Sig. (2-tailed)	Sig. (2-tailed)*	Sig. (2-tailed)	Sig. (2-tailed)
จำนวนรอบโดยวิธี HA กับวิธี SA ;	Sig. (2-tailed)	Sig. (2-tailed)	Sig. (2-tailed)	Sig. (2-tailed)

หมายเหตุ : *เฉพาะการเปรียบเทียบปริมาตรเหล็กที่ได้จากการประเมิน โดยวิธีฮิวริสติกกับวิธีการจำลองการอบเหนียวในตัวอย่างที่ 2 พบว่ามีความแตกต่างกันที่ระดับความเชื่อมั่น 95 %

ผลการเปรียบเทียบเชิงปริมาตรของเหล็กระหว่างวิธีการจำลองอบเหนียวกับวิธีทั่วไปที่ใช้อยู่ในปัจจุบัน

ผลของการเปรียบเทียบปริมาตรเหล็กที่ได้จากการเลือกโดยอัลกอริทึมการจำลองการอบเหนียวเทียบกับวิธีทั่วไปที่แสดงในหัวข้อที่ 2.7 ตารางที่ 2.3 พบว่า วิธีที่ใช้อัลกอริทึมการจำลองการอบเหนียว ได้ปริมาตรเหล็กในการคำนวณแตกต่างกับวิธีทั่วไปโดยเฉลี่ยทั้ง 4 ตัวอย่าง มีความแตกต่างกันที่ระดับนัยสำคัญ (Significant level) 0.001 หรือมีความแตกต่างกันที่ระดับความเชื่อมั่น 99.9 % โดยที่วิธีการจำลองการอบเหนียวเลือกเหล็กได้ปริมาตรน้อยกว่าและปริมาตรเหล็กจากการทดสอบโดยอัลกอริทึมการจำลองการอบเหนียวใน 4 ตัวอย่างตัวอย่างละ 25 ครั้งแสดงดังรูปที่ ค.1

ผลการเปรียบเทียบเชิงปริมาตรของเหล็กระหว่างวิธีฮิวริสติกกับวิธีที่ใช้อยู่ในปัจจุบัน

ผลของการเปรียบเทียบปริมาตรเหล็กที่ได้จากการเลือกโดยอัลกอริทึมฮิวริสติกเทียบกับวิธีทั่วไปพบว่า วิธีที่ใช้อัลกอริทึมฮิวริสติกได้ปริมาตรเหล็กในการคำนวณแตกต่างกับวิธีทั่วไป โดยเฉลี่ยทั้ง 4 ตัวอย่าง มีความแตกต่างกันที่ระดับนัยสำคัญ (Significant level) 0.001 หรือมีความแตกต่างกันที่ระดับความเชื่อมั่น 99.9 % โดยที่วิธีที่ใช้อัลกอริทึมฮิวริสติกเลือกเหล็กได้ปริมาตรน้อยกว่าและปริมาตรเหล็กจากการทดสอบโดยอัลกอริทึมฮิวริสติกใน 4 ตัวอย่างตัวอย่างละ 25 ครั้งแสดงดังรูปที่ ค.2

ผลการเปรียบเทียบเชิงปริมาตรของเหล็กระหว่างวิธีฮิวริสติกกับวิธีการจำลองอบเหนียว

ผลของการเปรียบเทียบปริมาตรเหล็กที่ได้จากการเลือกโดยอัลกอริทึมฮิวริสติกเทียบกับวิธีการจำลองอบเหนียวพบว่า วิธีที่ใช้อัลกอริทึมฮิวริสติกได้ปริมาตรเหล็กในการคำนวณแตกต่างกับวิธีการจำลองอบเหนียวโดยเฉลี่ยสำหรับตัวอย่างที่ 1 และ 4 ที่เป็นโครงข้อหมุนมีความแตกต่าง

กันที่ระดับนัยสำคัญ (Significant level) 0.001 และตัวอย่าง 2 และ 3 ที่เป็นโครงสร้างผสมมีความแตกต่างที่ระดับนัยสำคัญ 0.15 หรือมีความแตกต่างกันที่ระดับความเชื่อมั่น 85 % โดยที่วิธีที่ใช้ อัลกอริทึมฮิวริสติกเลือกเหล็กได้ปริมาณน้อยกว่าและผลการเปรียบเทียบปริมาณเหล็กเฉลี่ยที่เลือกได้ด้วยวิธีต่างๆ แสดงดังรูปที่ ค.3

ผลการเปรียบเทียบจำนวนรอบที่ใช้ในการคำนวณปริมาณเหล็กที่เหมาะสมระหว่างวิธีการจำลองการอบเหนียวกับอัลกอริทึมฮิวริสติก

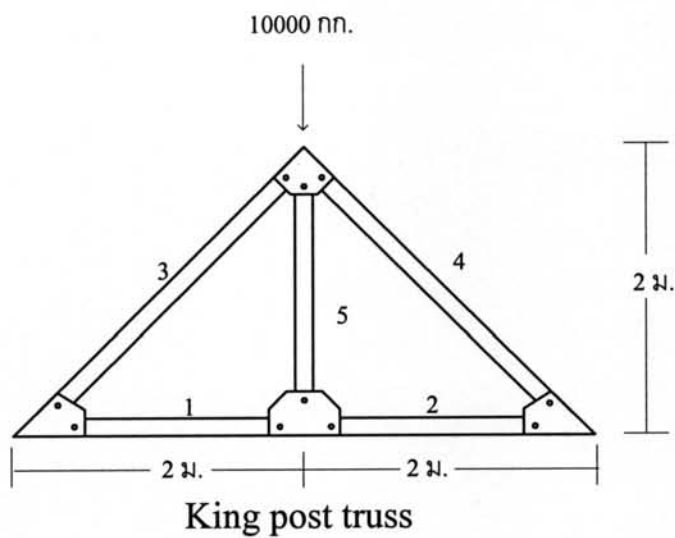
ผลของการเปรียบเทียบจำนวนรอบที่ใช้ในการคำนวณปริมาณเหล็กที่เหมาะสมระหว่างอัลกอริทึมการจำลองการอบเหนียวเทียบกับวิธีที่ใช้อัลกอริทึมฮิวริสติกพบว่าจากรูปที่ ค.4 จำนวนรอบที่ได้จากการทดสอบกับตัวอย่างทั้ง 4 พบว่าจำนวนรอบในการสร้างคำตอบด้วยอัลกอริทึมการจำลองการอบเหนียวต้องสร้างและตรวจคำตอบมากกว่า 600 ครั้งในขณะที่ฮิวริสติกอัลกอริทึมสร้างคำตอบและตรวจคำตอบไม่เกิน 200 ครั้งก็เข้าสู่คำตอบ และจากการคำนวณค่าทางสถิติด้วยวิธี T-test ตามตารางที่ ค.4 พบว่าวิธีที่ใช้อัลกอริทึมแบบฮิวริสติกใช้จำนวนรอบในการคำนวณแตกต่างกับวิธีการจำลองอบเหนียว โดยเฉลี่ยทั้ง 4 ตัวอย่าง มีความแตกต่างกันที่ระดับนัยสำคัญ (Significant level) 0.001 หรือมีความแตกต่างกันที่ระดับความเชื่อมั่น 99 % โดยที่วิธีที่ใช้อัลกอริทึมฮิวริสติกใช้จำนวนรอบน้อยกว่า

ภาคผนวก ง.

ง.1 การทดสอบอัลกอริทึมการจำลองการอบเหนียวและฮิวริสติกอัลกอริทึมกับตัวอย่างรูปแบบอื่นเพิ่มเติม

ในหัวข้อนี้ตัวอย่าง โครงสร้างข้อมนรูปแบบต่างๆ อีก 10 ตัวอย่างได้ถูกนำมาทดสอบกับอัลกอริทึมการจำลองการอบเหนียวและฮิวริสติกอัลกอริทึมเพื่อศึกษาเพิ่มเติม ซึ่งตัวอย่างที่ทดสอบเป็นตัวอย่าง โครงสร้างข้อมนที่มีจุดรองรับแบบธรรมดา (Simple Support) ที่พบได้ทั่วไป [4, 33, 34] และผลการทดสอบในทุกตัวอย่างจะแสดงในตอนท้ายของภาคผนวก

ตัวอย่างที่ 1

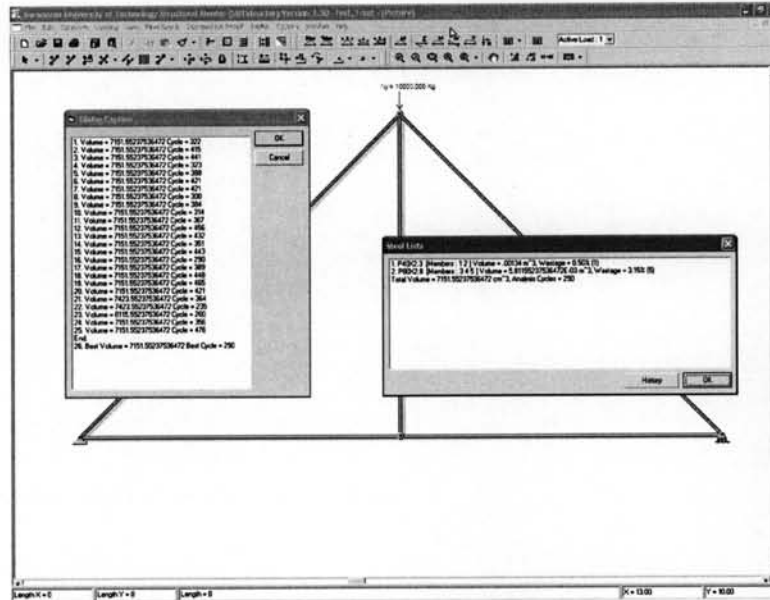


รูปที่ ง.1 โครงสร้างข้อมนตัวอย่างที่ 1

ตารางที่ ง.1 ข้อมูลการแบ่งชุดหน้าตัดสำหรับตัวอย่างที่ 1

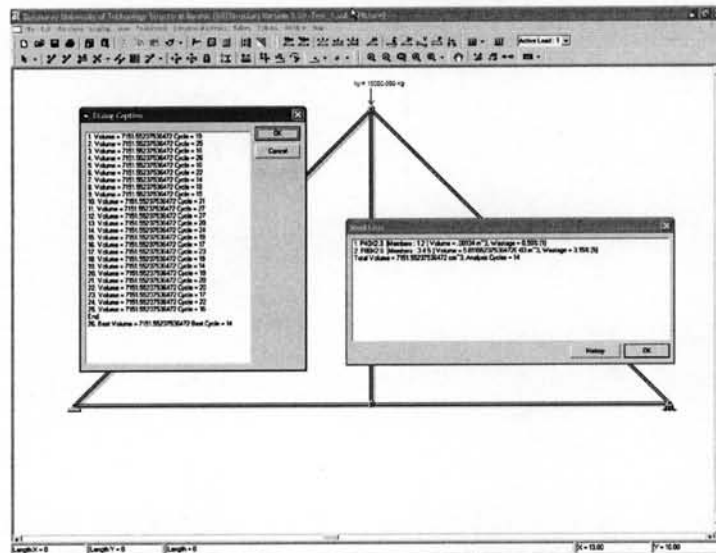
ชุดหน้าตัด	ชิ้นส่วน
1	1-2
2	3-5

ผลการทำงานของอัลกอริทึมการจำลองการอบเหนียว



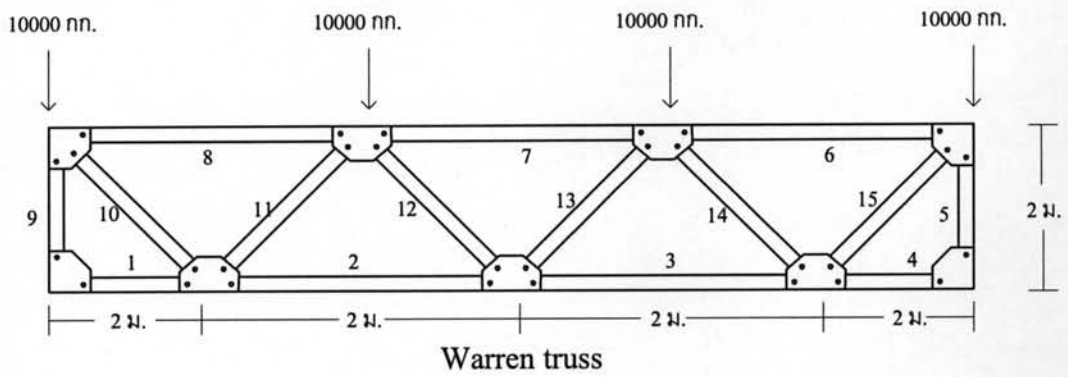
รูปที่ ง.2 ผลการทำงานของอัลกอริทึมการจำลองการอบเหนียวกับตัวอย่างที่ 1

ผลการทำงานของอัลกอริทึมฮิวริสติก



รูปที่ ง.3 ผลการทำงานของฮิวริสติกอัลกอริทึมกับตัวอย่างที่ 1

ตัวอย่างที่ 2

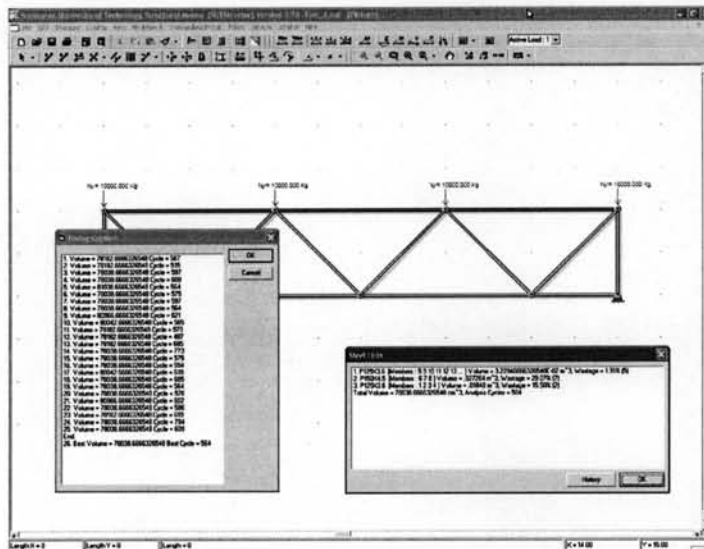


รูปที่ ๓.4 โครงสร้างข้อหมุนตัวอย่างที่ 2

ตารางที่ ๓.2 ข้อมูลการแบ่งชุดหน้าตัดสำหรับตัวอย่างที่ 2

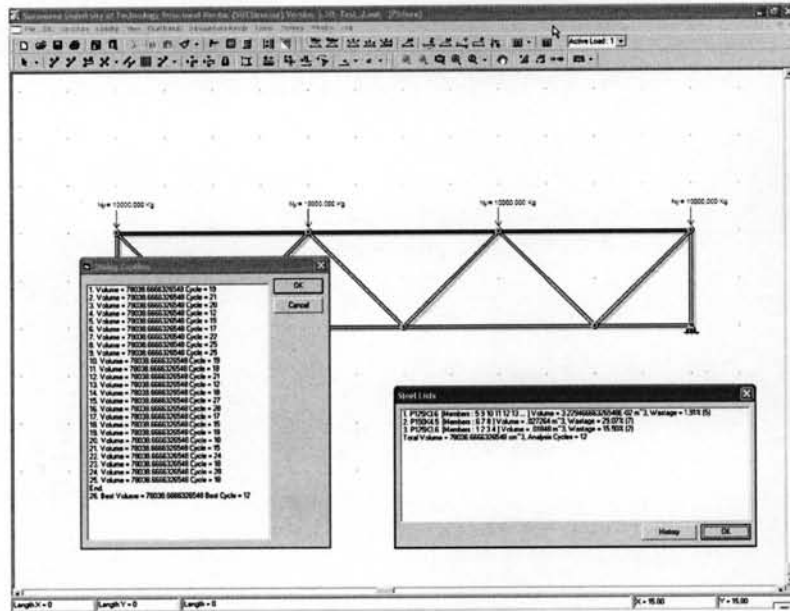
ชุดหน้าตัด	ชิ้นส่วน
1	1-2
2	5,9-15
3	6-8

ผลการทำงานของอัลกอริทึมการจำลองการอบเหนียว



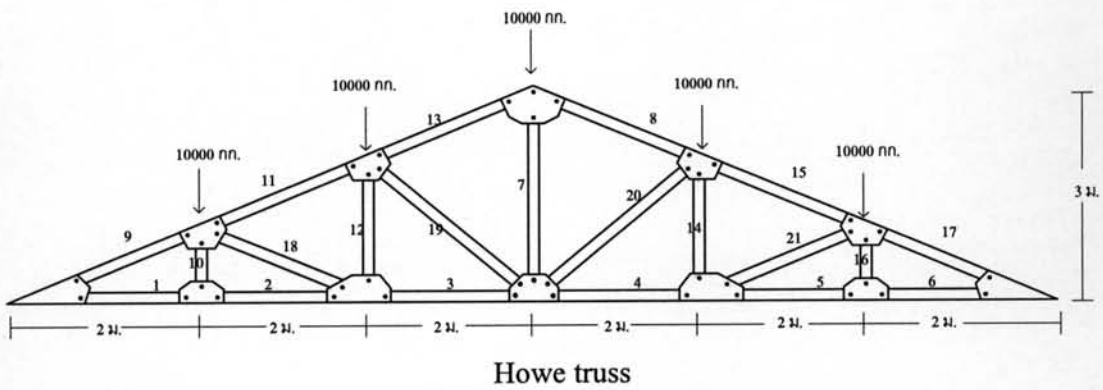
รูปที่ ๓.5 ผลการทำงานของอัลกอริทึมการจำลองการอบเหนียวกับตัวอย่างที่ 2

ผลการทำงานของอัลกอริทึมฮิวริสติก



รูปที่ ๖.6 ผลการทำงานของฮิวริสติกอัลกอริทึมกับตัวอย่างที่ 2

ตัวอย่างที่ 3

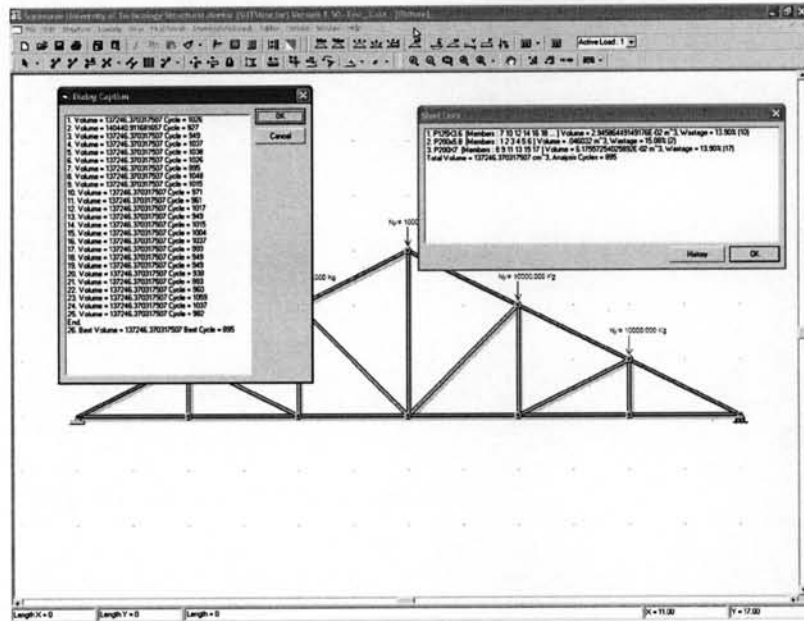


รูปที่ ๖.7 โครงสร้างข้อหมุนตัวอย่างที่ 3

ตารางที่ ๖.3 ข้อมูลการแบ่งชุดหน้าตัดสำหรับตัวอย่างที่ 3

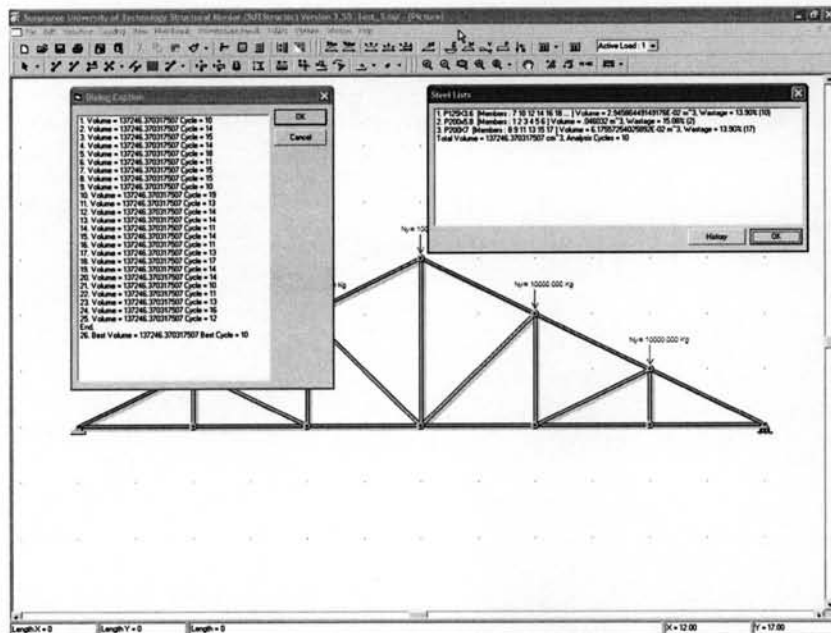
ชุดหน้าตัด	ชิ้นส่วน
1	7,10,12,14,16,18
2	1-6
3	8,9,11,13,15,17

ผลการทำงานของอัลกอริทึมการจำลองการอบเหนียว



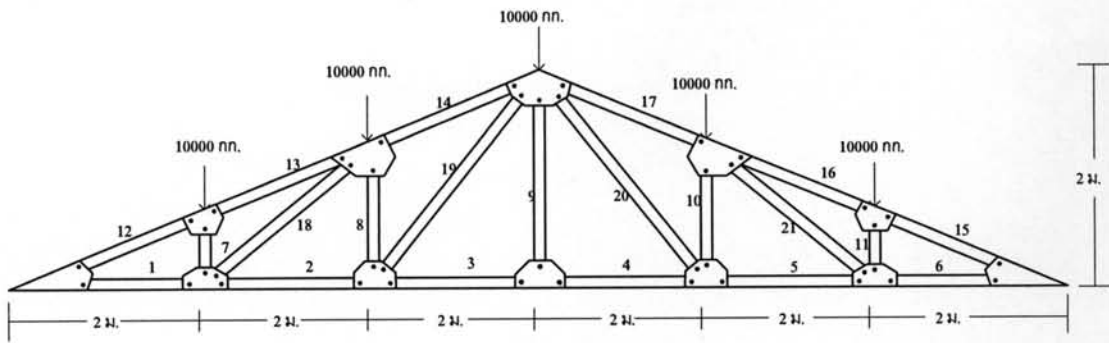
รูปที่ ๖.8 ผลการทำงานของอัลกอริทึมการจำลองการอบเหนียวกับตัวอย่างที่ 3

ผลการทำงานของอัลกอริทึมฮิวริสติก



รูปที่ ๖.9 ผลการทำงานของฮิวริสติกอัลกอริทึมกับตัวอย่างที่ 3

ตัวอย่างที่ 4



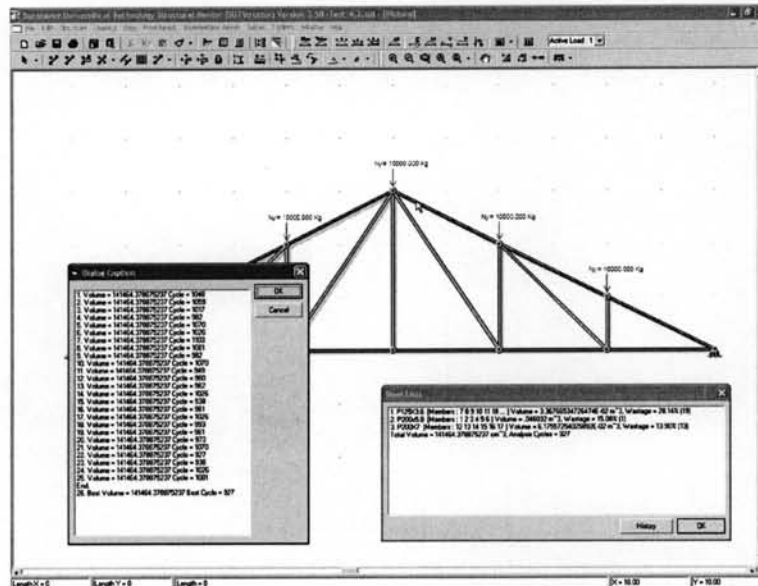
Pratt truss

รูปที่ ง.10 โครงสร้างข้อหมุนตัวอย่างที่ 4

ตารางที่ ง.4 ข้อมูลการแบ่งชุดหน้าตัดสำหรับตัวอย่างที่ 4

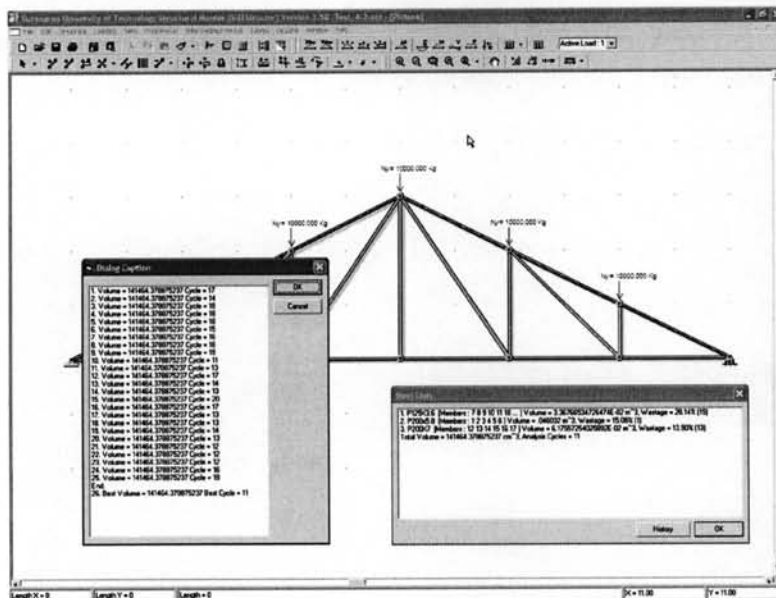
ชุดหน้าตัด	ชิ้นส่วน
1	7,10,12,14,16,18
2	1-6
3	8,9,11,13,15,17

ผลการทำงานของอัลกอริทึมการจำลองการอบเหนียว



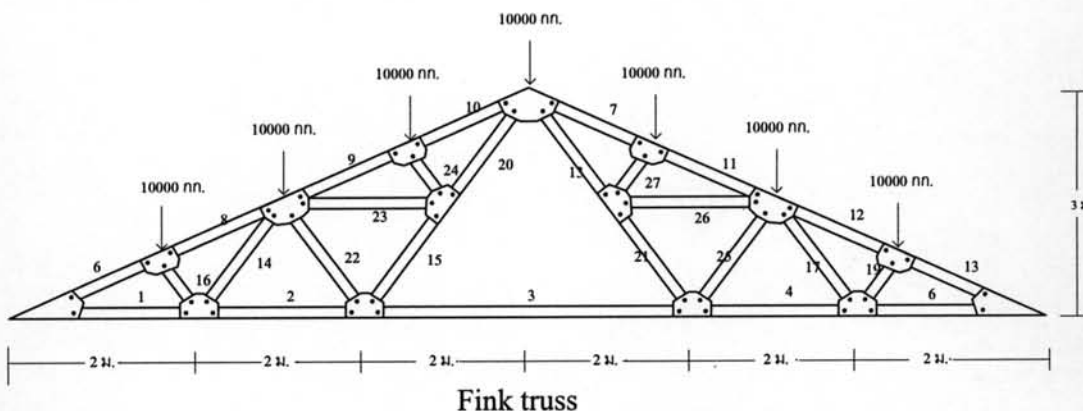
รูปที่ ง.11 ผลการทำงานของอัลกอริทึมการจำลองการอบเหนียวกับตัวอย่างที่ 4

ผลการทำงานของอัลกอริทึมฮิวริสติก



รูปที่ ง.12 ผลการทำงานของอัลกอริทึมการจำลองการอบเหนียวกับตัวอย่างที่ 4

ตัวอย่างที่ 5



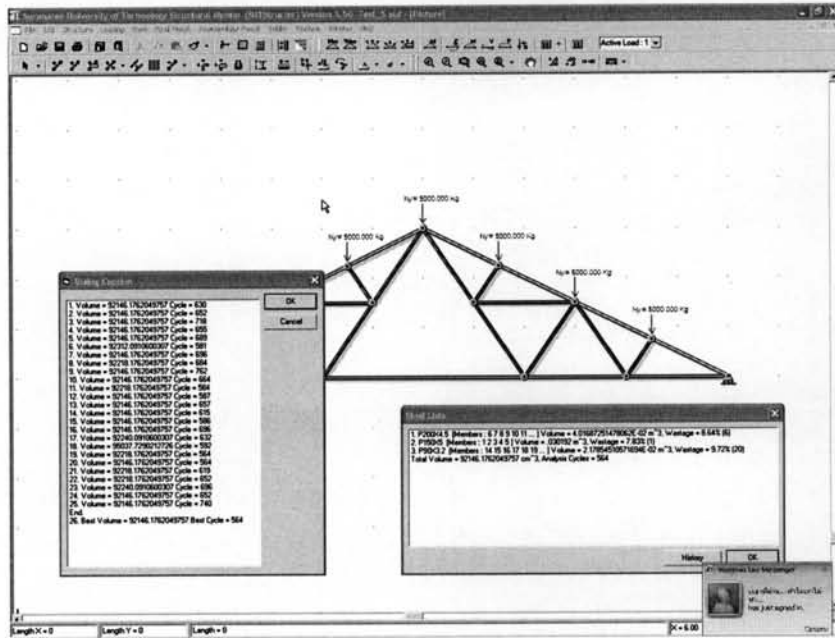
Fink truss

รูปที่ ง.13 โครงสร้างข้อหมุนตัวอย่างที่ 5

ตารางที่ ง.5 ข้อมูลการแบ่งชุดหน้าตัดสำหรับตัวอย่างที่ 5

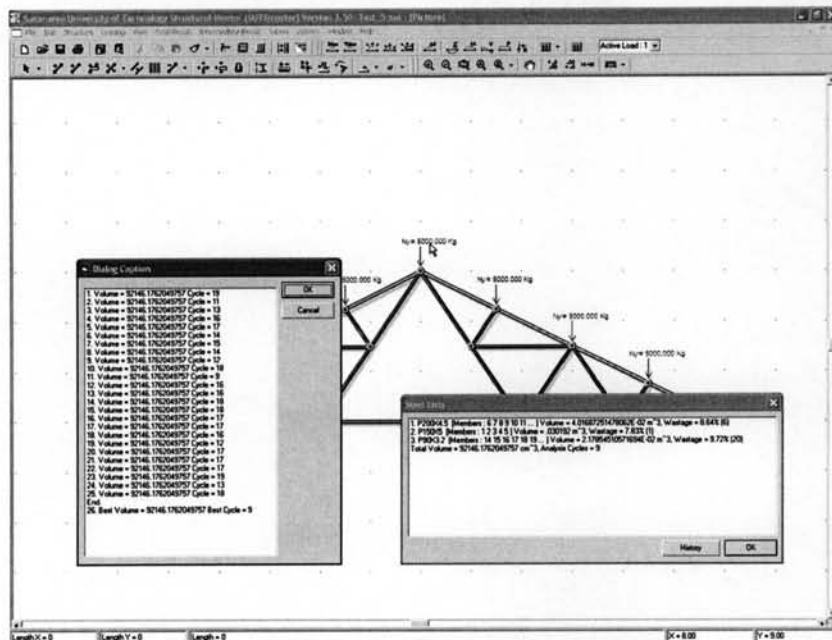
ชุดหน้าตัด	ชิ้นส่วน
1	6-11
2	1-5
3	14-19

ผลการทำงานของอัลกอริทึมการจำลองการอบเหนียว



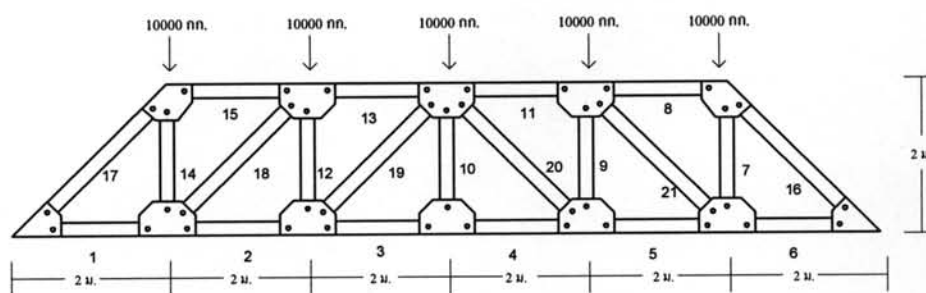
รูปที่ ง.14 ผลการทำงานของอัลกอริทึมการจำลองการอบเหนียวกับตัวอย่างที่ 5

ผลการทำงานของอัลกอริทึมฮิวริสติก



รูปที่ ง.15 ผลการทำงานของฮิวริสติกอัลกอริทึมกับตัวอย่างที่ 5

ตัวอย่างที่ 6



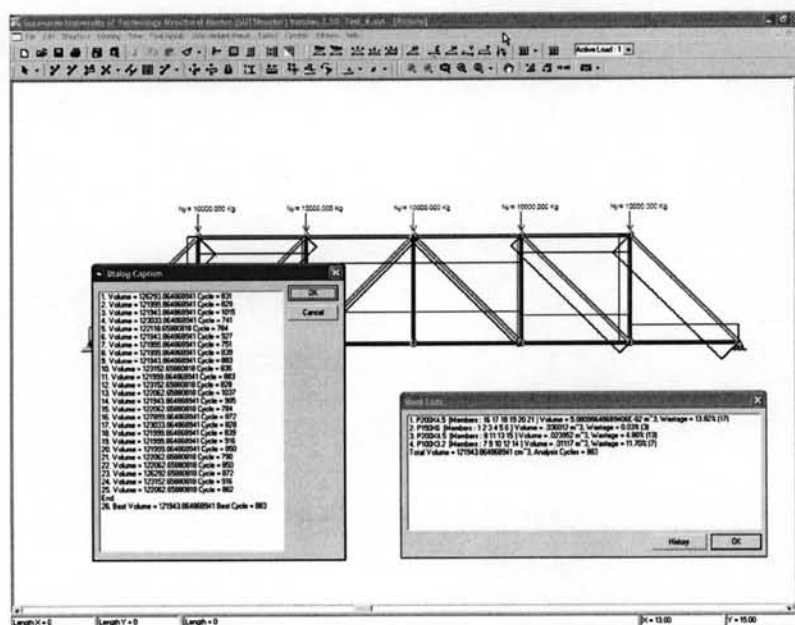
Howe truss

รูปที่ ง.16 โครงสร้างข้อหมุนตัวอย่างที่ 6

ตารางที่ ง.6 ข้อมูลการแบ่งชุดหน้าตัดสำหรับตัวอย่างที่ 6

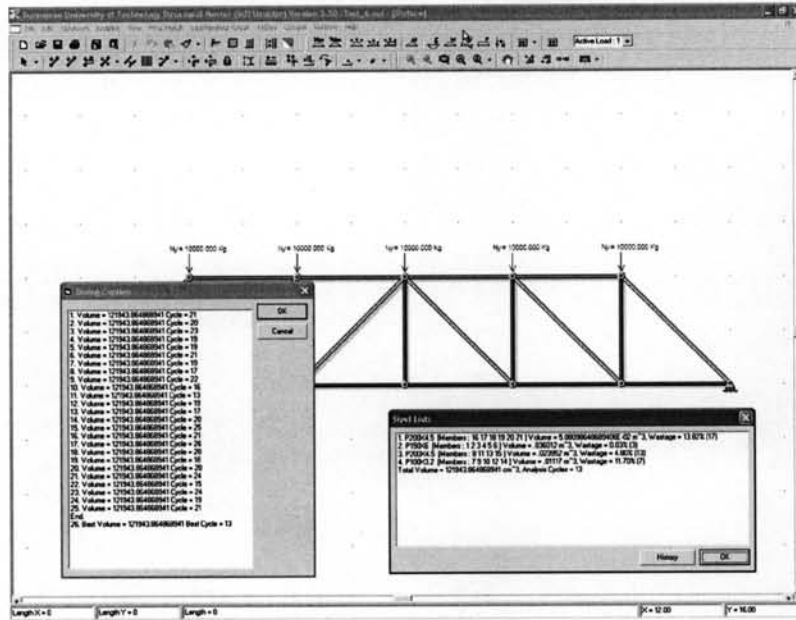
ชุดหน้าตัด	ชิ้นส่วน
1	16-21
2	1-6
3	8, 11, 13, 15
4	7, 9, 10, 12, 14

ผลการทำงานของอัลกอริทึมการจำลองการอบเหนียว



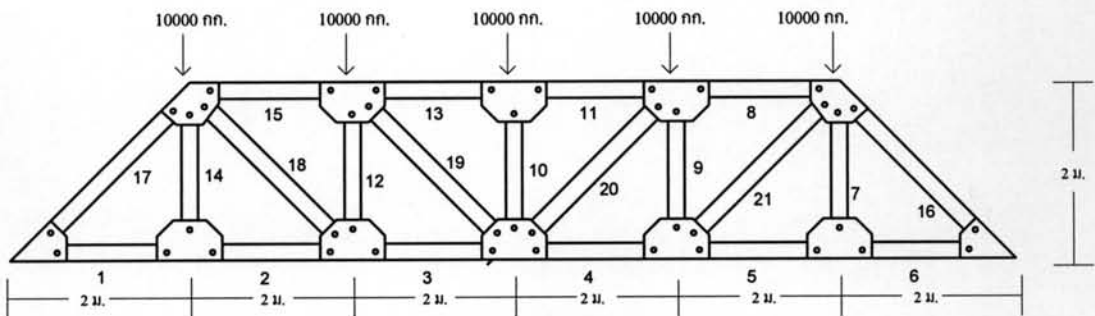
รูปที่ ง.17 ผลการทำงานของอัลกอริทึมการจำลองการอบเหนียวกับตัวอย่างที่ 6

ผลการทำงานของอัลกอริทึมฮิวริสติก



รูปที่ ง.18 ผลการทำงานของฮิวริสติกอัลกอริทึมกับตัวอย่างที่ 6

ตัวอย่างที่ 7



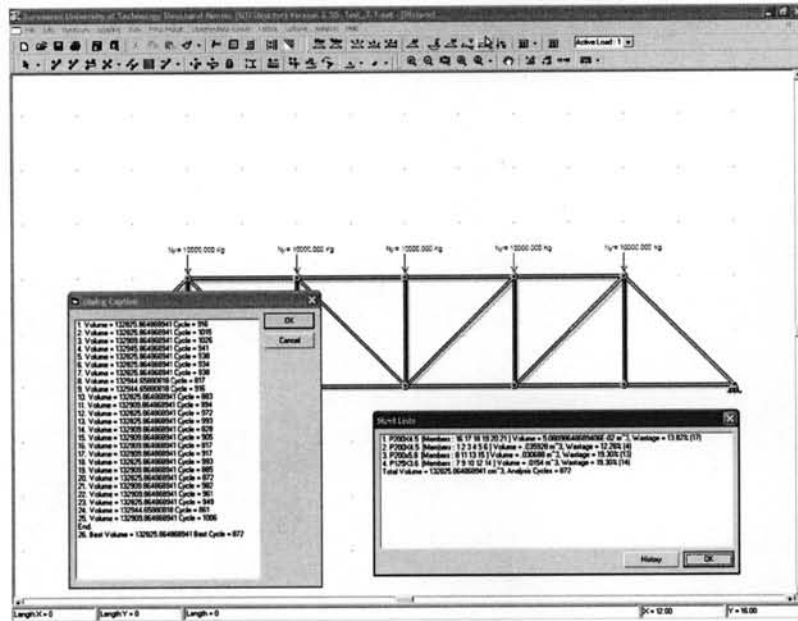
Pratt truss

รูปที่ ง.19 โครงสร้างข้อหมุนตัวอย่างที่ 7

ตารางที่ ง.7 ข้อมูลการแบ่งชุดหน้าตัดสำหรับตัวอย่างที่ 7

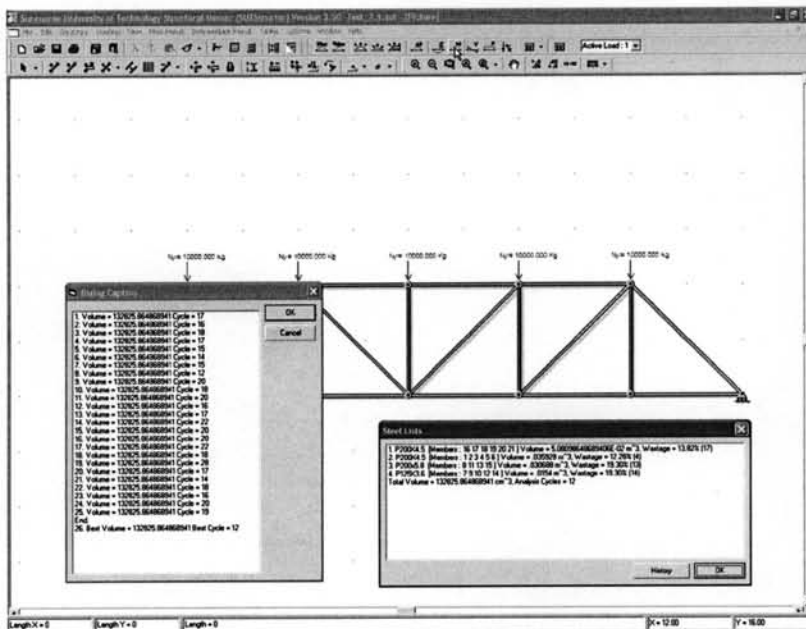
ชุดหน้าตัด	ชิ้นส่วน
1	16-21
2	1-6
3	8, 11, 13, 15
4	7, 9, 10, 12, 14

ผลการทำงานของอัลกอริทึมการจำลองการอบเหนียว



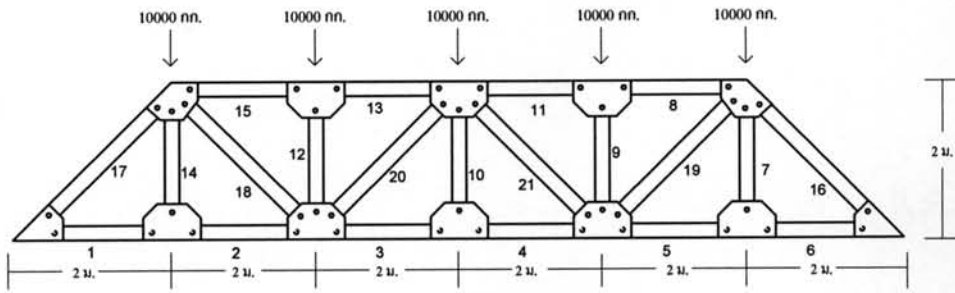
รูปที่ ง.20 ผลการทำงานของอัลกอริทึมการจำลองการอบเหนียวกับตัวอย่างที่ 7

ผลการทำงานของอัลกอริทึมฮิวริสติก



รูปที่ ง.21 ผลการทำงานของฮิวริสติกอัลกอริทึมกับตัวอย่างที่ 7

ตัวอย่างที่ 8



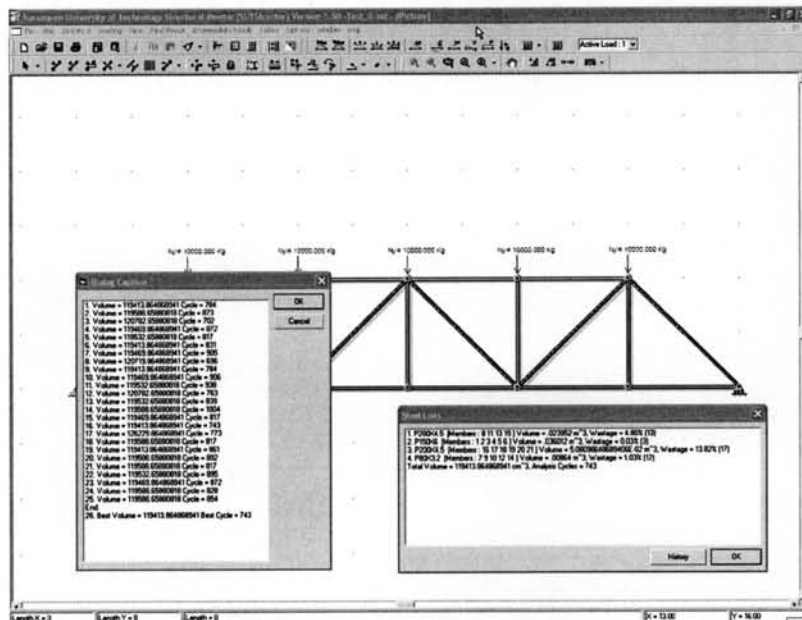
Warren truss

รูปที่ ง.22 โครงสร้างข้อหมุนตัวอย่างที่ 8

ตารางที่ ง.8 ข้อมูลการแบ่งชุดหน้าตัดสำหรับตัวอย่างที่ 8

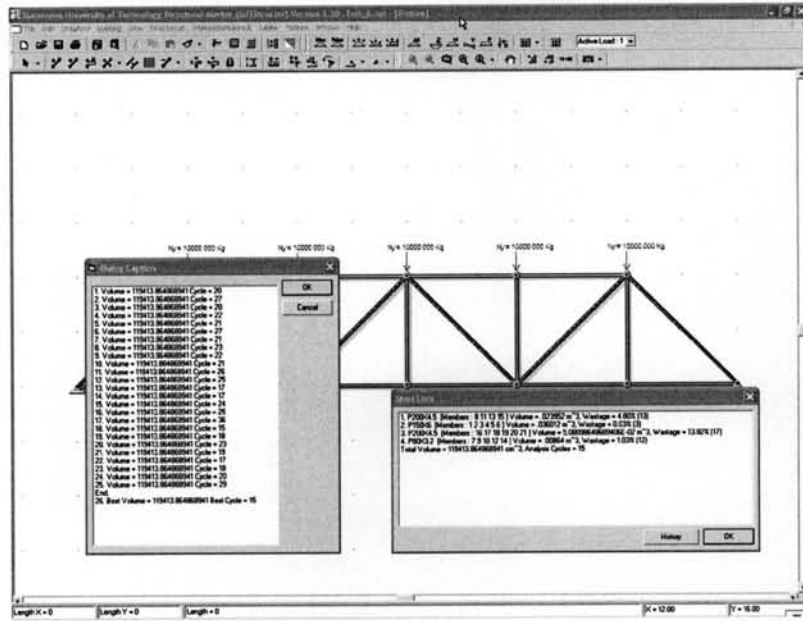
ชุดหน้าตัด	ชิ้นส่วน
1	8, 11, 13, 15
2	1-6
3	16-21
4	7, 9, 10, 12, 14

ผลการทำงานของอัลกอริทึมการจำลองการอบเหนียว



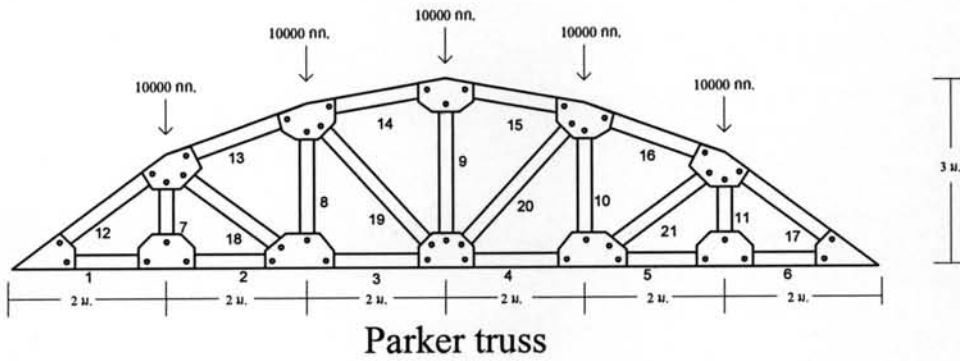
รูปที่ ง.23 ผลการทำงานของอัลกอริทึมการจำลองการอบเหนียวกับตัวอย่างที่ 8

ผลการทำงานของอัลกอริทึมฮิวริสติก



รูปที่ ง.24 ผลการทำงานของฮิวริสติกอัลกอริทึมกับตัวอย่างที่ 8

ตัวอย่างที่ 9

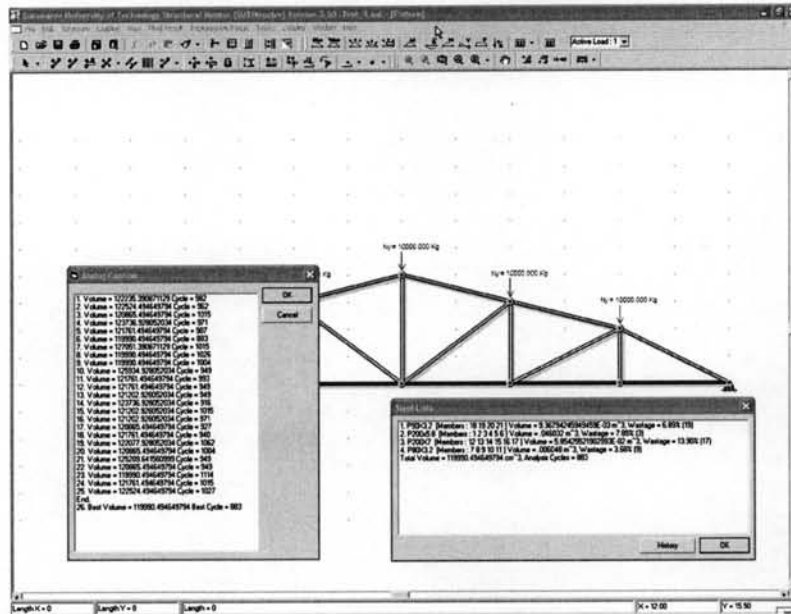


รูปที่ ง.25 โครงสร้างข้อหมุนตัวอย่างที่ 9

ตารางที่ ง.9 ข้อมูลการแบ่งชุดหน้าตัดสำหรับตัวอย่างที่ 9

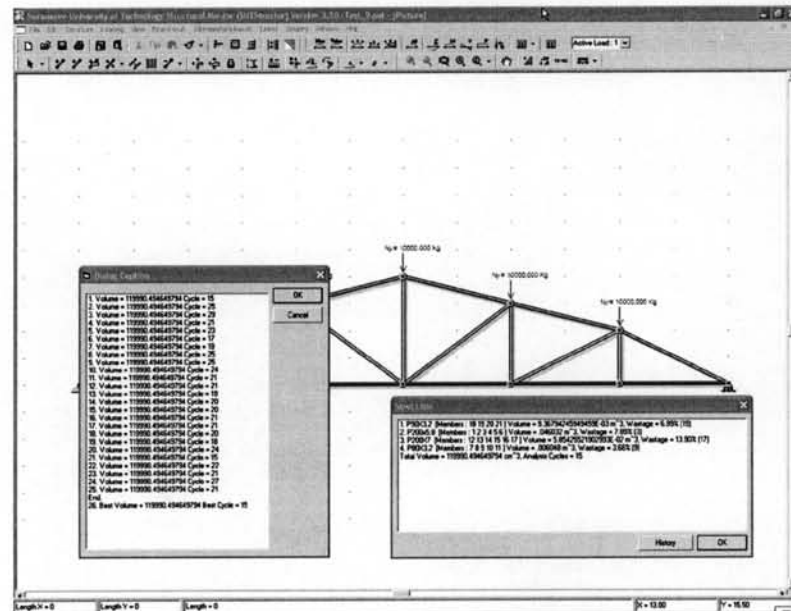
ชุดหน้าตัด	ชิ้นส่วน
1	18-21
2	1-6
3	12-17
4	7-11

ผลการทำงานของอัลกอริทึมการจำลองการอบเหนียว



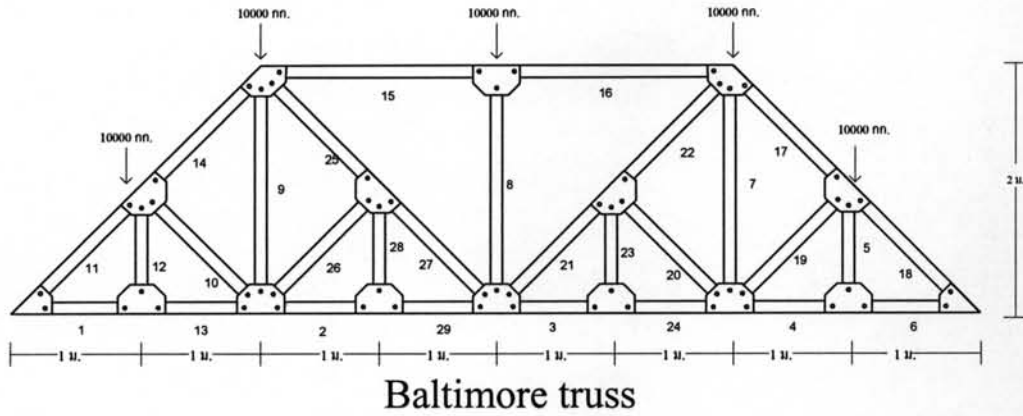
รูปที่ ง.26 ผลการทำงานของอัลกอริทึมการจำลองการอบเหนียวกับตัวอย่างที่ 9

ผลการทำงานของอัลกอริทึมฮิวริสติก



รูปที่ ง.27 ผลการทำงานของฮิวริสติกอัลกอริทึมกับตัวอย่างที่ 9

ตัวอย่างที่ 10

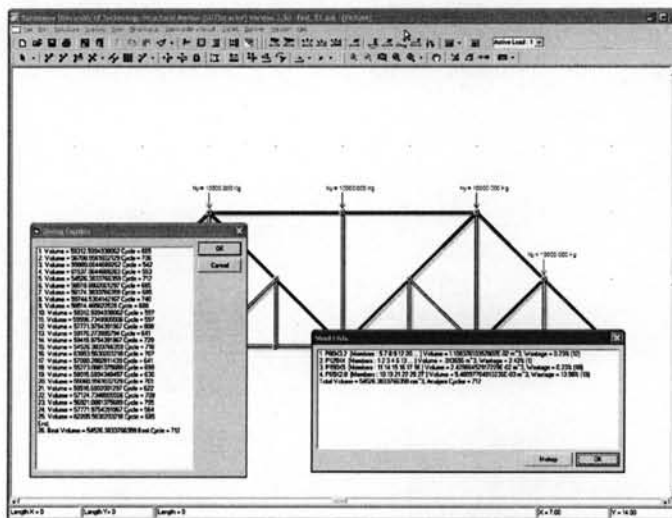


รูปที่ ง.28 โครงสร้างข้อหมุนตัวอย่างที่ 10

ตารางที่ ง.10 ข้อมูลการแบ่งชุดหน้าตัดสำหรับตัวอย่างที่ 10

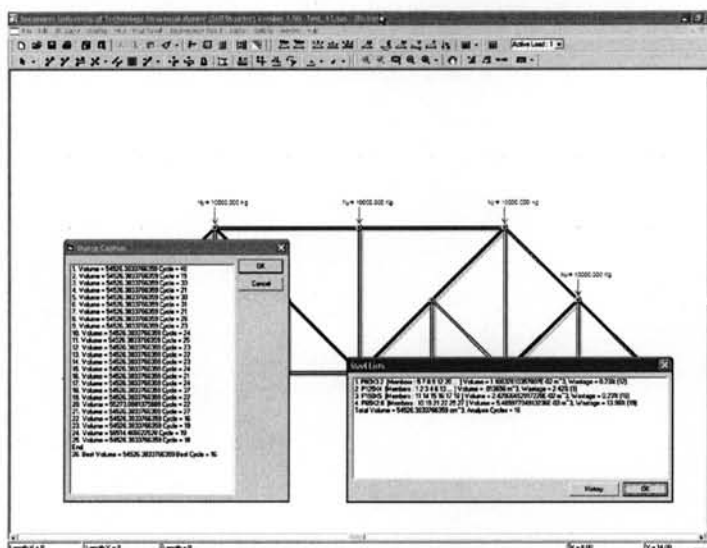
ชุดหน้าตัด	ชิ้นส่วน
1	5, 7, 9, 12, 20
2	1-4, 6, 13
3	11, 14-18
4	10, 19, 21, 22, 25, 27

ผลการทำงานของอัลกอริทึมการจำลองการอบเหนียว



รูปที่ ง.29 ผลการทำงานของอัลกอริทึมการจำลองการอบเหนียวกับตัวอย่างที่ 10

ผลการทำงานของอัลกอริทึมฮิวริสติก



รูปที่ 3.30 ผลการทำงานของฮิวริสติกอัลกอริทึมกับตัวอย่างที่ 10

รวมผลการทดสอบกับ 10 ตัวอย่าง

ตารางที่ 3.11 ผลการเลือกหน้าตัดโดยพิจารณาว่าที่ดีที่สุด

ตัวอย่าง โครงสร้างแบบที่	อัลกอริทึมการจำลองการอบ เหนียว		อัลกอริทึมฮิวริสติก	
	ปริมาตร ¹	รอบ ²	ปริมาตร ¹	รอบ ²
1	7151.55	290	7151.55	14
2	78038.67	564	78038.67	12
3	78038.67	564	78038.67	12
4	137246.37	895	137246.37	10
5	92146.18	564	92146.18	9
6	121943.86	883	121943.86	13
7	132825.86	872	132825.86	12
8	119413.86	743	119413.86	15
9	119990.49	883	119990.49	15
10	54526.38	712	54526.38	16

หมายเหตุ 1. ปริมาตรเหล็กที่หาได้จากอัลกอริทึมหน่วยเป็น ลบ.ซ.ม.

2. จำนวนรอบในการเรียกฟังก์ชันตรวจคำตอบ

ตารางที่ ง.12 ผลการเลือกหน้าตัดโดยพิจารณาค่าเฉลี่ยจากการทดสอบ 25 ครั้ง

ตัวอย่าง โครงสร้างแบบที่	อัลกอริทึมการจำลองการอบ เหนียว		อัลกอริทึมฮิวริสติก	
	ปริมาตรเฉลี่ย ¹	รอบเฉลี่ย ²	ปริมาตรเฉลี่ย ¹	รอบเฉลี่ย ²
1	7151.552	378.880	7151.552	19.840
2	78587.787	605.200	78038.667	19.800
3	137374.152	991.400	137246.370	13.520
4	141464.379	1009.960	141464.379	15.200
5	92290.388	646.680	92146.176	15.560
6	122808.743	858.680	121943.864	19.960
7	132878.520	926.360	132825.864	17.960
8	119927.398	835.320	119413.864	22.120
9	122034.894	979.760	119990.494	21.320
10	58237.980	673.960	54715.774	24.400

หมายเหตุ 1. ปริมาตรเหล็กที่หาได้จากอัลกอริทึมหน่วยเป็น ลบ.ซ.ม.

2. จำนวนรอบเฉลี่ยจากการทดสอบ 25 ครั้งในการเรียกฟังก์ชันตรวจคำตอบ

ผลการทดสอบ

จากผลการทดสอบอัลกอริทึมการจำลองการอบเหนียวและฮิวริสติกอัลกอริทึมกับตัวอย่าง ทั้ง 10 ตัวอย่างแสดงให้เห็นว่าผลการทดสอบมีแนวโน้มของคำตอบเช่นเดียวกับอีก 4 ตัวอย่างที่ถูกศึกษาในบทที่ 4 และบทที่ 5 และเมื่อทดสอบทางสถิติด้วยวิธี T-test กับตัวอย่างทั้ง 10 ตัวอย่าง พบว่าฮิวริสติกอัลกอริทึมใช้จำนวนรอบในการสร้างคำตอบและจำลองพฤติกรรม โครงสร้างเพื่อตรวจคำตอบน้อยกว่าอัลกอริทึมการจำลองการอบเหนียวที่ระดับนัยสำคัญ 0.001

ภาคผนวก จ.

บทความและผลงานวิจัยที่ได้รับการตีพิมพ์

1. The 2nd ECTI Annual Conference (ECTI-CON) 2005 Pattaya, Thailand (May 12-13, 2005): 742-745.
2. The ECTI Transaction on Computer and Information Technology 1, 2 (November 2005): 108-117.
3. American Journal of Applied Science. 8, 5 (2008): 943-951.

A New Technique for 2D Structure Simulation

Alongkorn Lamom^{1,2} and Wanchai Rivepiboon¹

Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University, Thailand¹
 Faculty of Engineering, Mahasarakham University, Thailand²
 E-mail: alongkorn_lamom@hotmail.com, wanchai.r@chula.ac.th

ABSTRACT

To simulate a building structure by finite element method, the connection in the simulation depends on element types. Most 2-D structural analysis software provides two element types; frame and truss elements, to simulate plane frame and plane truss respectively. As both elements are separately implemented into two modules, there are inconvenient for improving and debugging the software. The paper proposes a new technique to simulate both plane frame and plane truss structures by using only frame element. Since two types of structure simulation are integrated into single element, time and effort to implement the software is reduced.

Keywords: Structure Simulation, Code Numbers

1. INTRODUCTION

In traditional technique for simulating plane frame and plane truss structures by finite element method, software is composed of two modules; truss and frame simulation modules [1-5]. Each module is implemented for simulating each type of the structure; plane frame and plane truss structures. The difference of simulation between plane frame and plane truss structures is the connection in the structure. The connection employed in the simulation is in accordance as the module used for simulation. The truss simulation module simulates the connection in the structure as hinged connection, while the frame simulation module simulates connections in the structure as rigid connection. Figure 1 describes how the traditional software works.

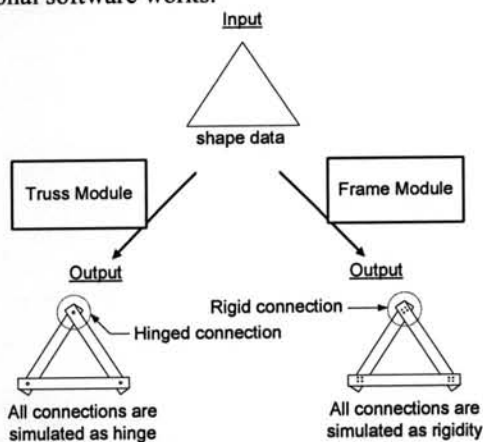


Fig.1: The traditional technique

To simulate by using traditional technique, user has to input the shape of the structure, and then choose a simulation module. The module has to be selected based on type of connection the user wants to simulate. For example, if the user wants to simulate the structure as plane truss, of which all connections in the structure are hinged connection, the plane truss simulation module is selected. On the other hand, if the user wants to simulate the structure as plane frame, the plane frame module is selected instead.

Since the conventional simulation technique for the plane frame and plane truss structures composes of two modules, improving and debugging these modules are performed separately. This paper proposes a new simulation technique that requires only one module for simulating both plane frame and plane truss structures. By this technique, time and effort of improving and debugging structural analysis software is reduced. In next section the convention technique is described, and then the new technique is illustrated later.

2. SIMULATION PROCESS

This section reviews the process of simulation by finite element method. Simulation process of this method can be divided into two parts. The first part forms while the second part solves the equation. The equation used for structure simulation is defined as.

$$[Q] = [K] [D] \quad (1)$$

Where

[Q] is structural external force matrix,
 [K] is structural stiffness matrix and
 [D] is structural displacement matrix

The above equation shows the relationship between force and displacement of a structure. Structural stiffness matrix [K] is a constant matrix used to link the relationship between forces and displacements. To form the equation [1,3-5], structure is divided into finite element called member as shown in Figure 2. Each member has a private stiffness matrix [k], called member stiffness matrix, which is calculated from properties of the member. These member stiffness matrices are combined to be a structural stiffness matrix [K] according

to code numbers [1]. These code numbers are generated by the algorithm, called nodal numbering scheme.

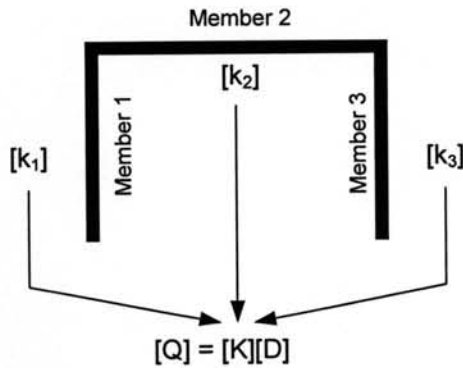


Fig.2: Equation performing

To implement the software, software is divided into two modules; truss and frame modules. Both modules have similar process, but they are different in type and size of element matrix. To integrate simulating ability from two modules, a new technique uses only element matrix of frame and redesigns nodal numbering scheme. However, in the other steps, the conventional process is still employed. Redesigning this scheme follows up the relationship between connection and code number. This relationship shall be described after reviewing the traditional nodal numbering scheme.

3. THE TRADITIONAL NODAL NUMBERING SCHEME

In traditional nodal numbering scheme, code number of a member is referred from nodes, and the member exists in two nodes. The scheme will generate code number for every node in x, y and z directions. Then the member refers to both nodes at its edge for code numbers.

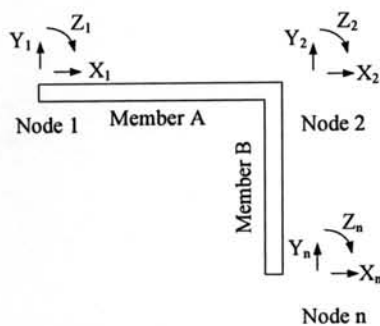


Fig.3: Traditional nodal numbering scheme

The algorithm for generating code numbers for every node is presented as follows:

Define
 n is a node,
 X_n is a code number in x-axis of node n,
 Y_n is a code number in y-axis of node n and
 Z_n is a code number in z-axis of node n

1. Assign ordering number to all nodes.
2. Code number = 0
3. For all node
 - Code number = Code number + 1
 - X_n = Code number
 - Code number = Code number + 1
 - Y_n = Code number
 - Code number = Code number + 1
 - Z_n = Code number
- End loop

This algorithm is simple but it is used for only one type of the structure. It is noteworthy that the amount of code numbers in each node is always equal to three degrees, and values of these code numbers involve connecting members together. This relationship is described in the next section, and it will lead to a new idea for generating code numbers.

4. RELATIONSHIP BETWEEN CONNECTION AND CODE NUMBER

Code number represents a direction of force or displacement in a structure. To get the force-displacement equation matrix of structure, Code number is referred for forming many minor force-displacement equations in each direction of each member. Each minor force-displacement equation is the relationship between external force and internal force of all members that are connected and transferred in each code numbers direction [1]. To form these equations using code numbers, relationships between characteristic of connection and code number are given below:

1. Direction, of which members are connected and transferred together, will use the same code number.
2. Direction, of which member are connected and not transferred together, will use a different code number.

As members connected by either hinged connection or rigid connection always transfer together in x and y directions, these members will share a code number in these directions. On the other hand, in z direction, rotating of members connected together depends on the connected member data. Hence generating code numbers will be determined by characteristic of connection. These specifications lead to a new idea of generating the code numbers. The idea is described below.

1. The conventional frame programming model will be redesigned in process of generating code numbers.
2. A new algorithm is created to support the model and to generate code numbers for each node of member according to these constrains.

In x and y directions
 - If members are connected together, they will use the same code number.

In z direction

- If members are connected by rigid connection, they will use the same code number in z direction.
- If members are connected together, but they independently rotate, they will use the different code numbers in z direction.

It is more convenient to implement the software if these constrains are rewritten as an algorithm, which is described in the next section.

5. THE NEW NODAL NUMBERING SCHEME

The new algorithm generates code numbers based on characteristic of connections in a structure. An amount of code numbers in each node is not a constant. It varies in accordance characteristic of connected members in each node. This new algorithm is called "Dynamic Nodal Numbering Scheme". Generating code numbers by new scheme is created in order of a member instead of a node (Figure 4). Each member consists of two nodes; a beginning node and an ending node. An extreme node is identified by the head of the black arrow in the middle of the member. Each node has three degrees of freedom or three code numbers for x, y and z axes. Code numbers are assigned for each member at the beginning node and then the ending node as the following algorithm.

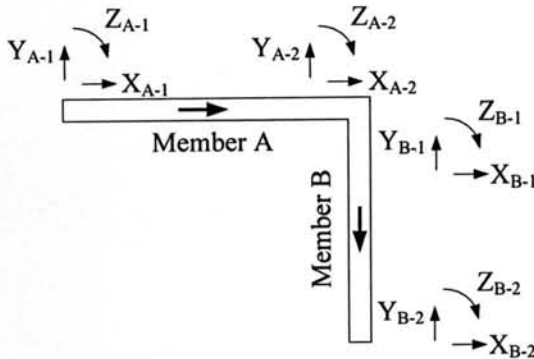


Fig.4: Dynamic nodal numbering scheme

Define,

m is a member,

X_{m-1} is a code number in x-axis of the beginning node of the member m

Y_{m-1} is a code number in y-axis of the beginning node of the member m

Z_{m-1} is a code number in z-axis of the beginning node of the member m

X_{m-2} is a code number in x-axis of the ending node of the member m

Y_{m-2} is a code number in y-axis of the ending node of the member m

Z_{m-2} is a code number in z-axis of the ending node of the member m

1. Assign order number to all members.
2. Code number = 0
3. For all member

If the beginning node already exists then

X_{m-1} = Code number in x-axis from the an existing node

Y_{m-1} = Code number in y-axis from the an existing node

If the structure is simulated as plane frame

Z_{m-1} = Code number in z-axis from the existing node

Else

Code number = Code number + 1

Z_{m-1} = Code number

End if

Else

Code number = Code number + 1

X_{m-1} = Code number

Code number = Code number + 1

Y_{m-1} = Code number

Code number = Code number + 1

Z_{m-1} = Code number

End if

If an ending node already exists then

X_{m-2} = Code number in x-axis from the an existing node

Y_{m-2} = Code number in y-axis from the an existing node

If the structure is simulated as plane frame

Z_{m-2} = Code number in z-axis from the an existing node

Else

Code number = Code number + 1

Z_{m-2} = Code number

End if

Else

Code number = Code number + 1

X_{m-2} = Code number

Code number = Code number + 1

Y_{m-2} = Code number

Code number = Code number + 1

Z_{m-2} = Code number

End if

End loop

6. GENERATING CODE NUMBER FOR PLANE FRAME SIMULATION

This section illustrates how to generate code numbers following the new scheme for plane frame. A sample plane frame structure is shown in Figure 5, of which all connections are rigid connection. Generating this structure starts at member A in the beginning node and then ending node, i.e. Nodes 1 and 2, in x, y and z directions in order. Since the beginning node and ending node of member A do not exist, all code numbers are created newly from numbers 1 to 6. Next, code numbers are generated for member B at the beginning node and the ending node; Nodes 2 and 3. Because the beginning node of member B is connected as rigid connection to the ending node of member A, of which code numbers are

already generated, so code numbers in x and y axes of the beginning node of member B are copied from the ending node of member A. In z axis, member A and member B are connected by rigid connection. They rotate concurrently. So, the code number in z direction of member B is copied from member A. Afterward, code numbers are created for the ending node of member B. As the result, the code numbers for ending node of member B never exist; code numbers are refreshed from 7 to 9. The next section shows the code number generation for plane truss.

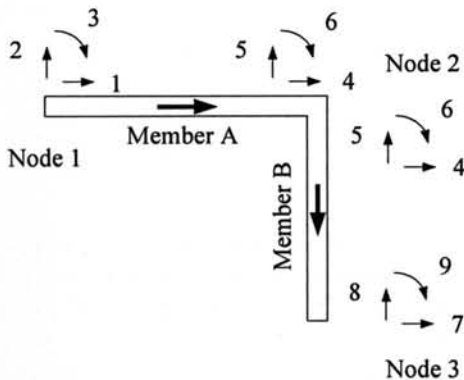


Fig.5: Code numbers for the plane frame

7. GENERATING CODE NUMBER FOR PLANE TRUSS SIMULATION

This section shows an example that illustrates how to generate code numbers for a plane truss structure by the new nodal numbering scheme. The example is shown in Figure 6. The freely rotating structure has two elements, connected by a hinged connection at Node 2.

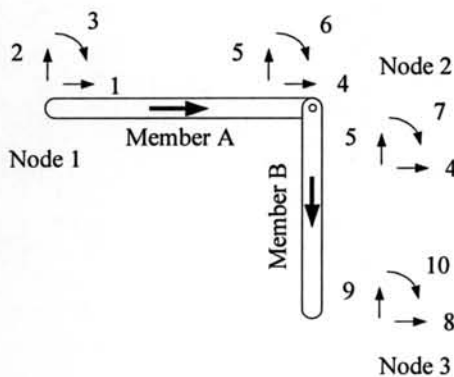


Fig.6. Code number for plane truss

Code number generating are started at the beginning node and ending node of member A. These nodes are Nodes 1 and 2. Since Node 1 does not connect to any other member, and the code numbers for Node 2 are not yet generated, code numbers are generated newly for the beginning node and ending node of member A (from 1 to 6). Then code numbers are generated for member B. As the beginning node of member B connects to the ending node of member A, code numbers in x and y axes are

copied from the existing node. These code numbers are 4 and 5 respectively. As the structure is simulated as plane truss, a code number for the beginning node of member B in z direction is generated newly as 7. Then new code numbers are created for the ending node of member B. As this node does not connect to any other member, the code numbers in x, y and z axes are 8, 9 and 10.

8. CONCLUSIONS

This paper proposes a new technique which allows plan truss and plane frame structures to be simulated by using only frame elements. A new model is created by improving the conventional plane frame simulation method by redesign of the algorithm in code number generation process. A new scheme of automatic nodal numbering, called Dynamic Nodal Numbering Scheme, is proposed for supporting the new programming model in generating the code number process. The new technique uses a strategy, which has only frame module and can manipulate both types of structure. So, the new technique reduces software developing, debugging and expanding time while increases the simulation flexibility.

As the new technique can simulate only plane truss and plane frame structures, which consist of only two types of connection; rigid and hinged connection, we are planning to improve the nodal numbering scheme for simulating a structure consisting of various formats of connection in the future.

- [1] X2. Aslam Kassimali, *Matrix Analysis of Structures*, Brooks/Cole Publishing Company, 1999, ch. 3.
- [2] J. N. Reddy, *Finite Element Method*, McGraw-Hill Book Co, 1993.
- [3] Russel C. Hibbeler, *Structural Analysis*, Prentice-Hall, United States of America, 1995.
- [4] Saeed Moaveni, *Finite Element Analysis*, Prentice-Hall, United States of America, 1999.
- [5] William Weaver, Jr. and James M. Gere, *Matrix Analysis of Framed Structures*, Chapman & Hall, 1990.

A Nodal Numbering Algorithm for 2D Structural Simulation

Alongkorn Lamom and Wanchai Rivepiboon, Non-members

ABSTRACT

This paper proposes the new algorithm, called Dynamic Nodal Numbering Scheme, for generating the code numbers. Code numbers are used to formulate equations for structural simulation by the finite element method. In the traditional technique, developing the 2D structural analysis software provides two element types: frame and truss elements. These two elements are used to simulate plane frame and plane truss structures respectively. Connection types of structural simulation depend on element types. Using the new algorithm, only the frame element is required while the capabilities of simulation are not different. Since two types of structural simulation are integrated into a single element type, time and effort to implement and debug the software are reduced. At the end of this paper, the algorithm comparisons are shown. It is found that both algorithms have the same complexity of $O(n)$. The proposed algorithm has an advantage over the traditional one when it comes to the problem that consists of many non-closed loop structures. For the plane truss structure, it is reported that the proposed algorithm generates more degrees of freedom than the traditional algorithm. This disadvantage is compensated by report resulting in z axis, using rotation in z axis to draw deflection curve by double integration method which is more computationally effective than the traditional technique to which many nodes are added.

Keywords: Structural Simulation, Code Numbers

1. INTRODUCTION

In the traditional technique for developing a structural analysis software to simulate plane frame and plane truss structures by the finite element method, the software is comprised of two components: truss and frame simulation modules [1-5] as shown in figure 1. Each module provides one element type, truss element or frame element, for simulating plane truss and plane frame structures, respectively. The types of connections are simulated according to the modules employed. The truss simulation module simulates all connections in the structure as hinged connection, while the frame simulation module simulates all connections in the

structure as rigid connection. As a result, the traditional technique has to provide two modules for simulating each type of a structure. There are many processes that are implemented in each module. It is not convenient for coding, debugging and extending the software.

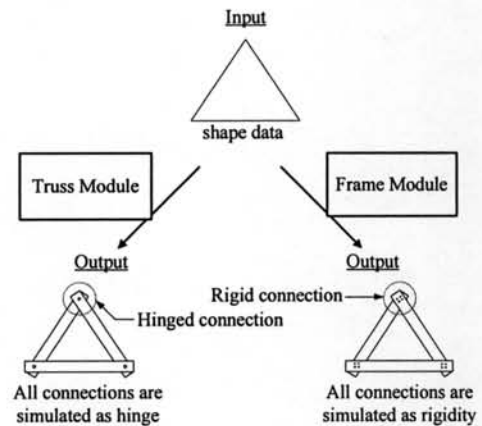


Fig.1: The traditional technique

Although the software implemented by the traditional technique contains two modules, both modules have similar processes, forming and solving equations, as shown in figure 2. There are a few differences in minor processes of forming equations, for example, sizes of member matrices of each module are not equal, and numbers of degrees of freedom and code number [6] per node are different. In this new technique, we found that there are direct relationships between code numbers and behaviors of connections. If degrees of freedom of each node can be adjusted according to the characteristics of connections, only one element type which is a frame element is required.

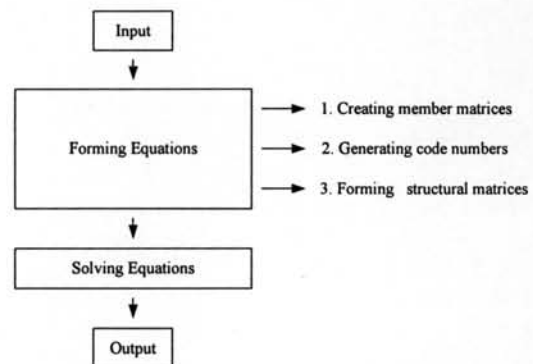


Fig.2: Simulation processes

The authors are with Software Engineering Laboratory, Department of Computer Engineering, Chulalongkorn University, Bangkok 10330 Thailand. E-mail: alongkorn_lamom@hotmail.com and wanchai.R@chula.ac.th

After the traditional technique had been proposed, there were some techniques proposed to handle code numbers to reduce the matrix bandwidth [7-8]. However, there has not been a technique for generating code numbers which can reduce cost, effort, and time of software development. It is one of the most important factors of software development.

This paper proposes the new algorithm for generating code numbers which are used to formulate structural equations. Using the new algorithm, the frame element supports the simulation of both plane frame and plane truss structures. Time, cost and effort of improving and debugging the structural analysis software are reduced. The new algorithm can also be applied with slight changes in the traditional technique. In the next section, the conventional technique will be illustrated and described followed by the detailed explanation of the new technique. In the last section of this paper, comparisons between the traditional and the new nodal numbering algorithms will be analyzed and discussed.

2. SIMULATION PROCESS

This section reviews the process of simulation by the finite element method. Simulation process of this method can be divided into two parts. The first part forms while the second part solves the equation. The equation used for a structural simulation is defined as:

$$[P] = [S] [d]$$

where

[P] is a structural external force matrix,
[S] is a structural stiffness matrix and
[d] is a structural displacement matrix

The above equation shows the relationship between force and displacement of a structure. The structural stiffness matrix [S] is a coefficient matrix used to link the relationship between forces and displacements. To form the equation [1,3-5], the structure is divided into finite elements called members as shown in figure 3.

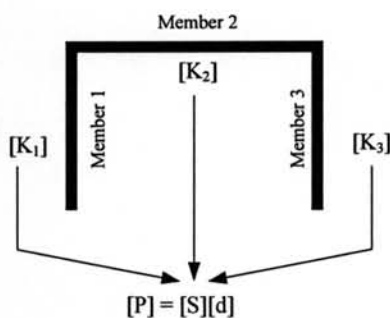


Fig.3: Equation forming

Each member has a private stiffness matrix [K] called member stiffness matrix in global axis which is

calculated from properties of the members. These member stiffness matrices are combined to be the structural stiffness matrix [S] according to code numbers [1]. These code numbers are generated by the algorithm called nodal numbering scheme.

To implement the structural analysis software, it is divided into two modules: truss and frame modules. Both modules have similar process, but they are different in type and size of element matrix. To integrate simulating abilities from two modules, the new technique uses only the element matrix of frame and redesigns the nodal numbering scheme. However, in the following steps, the conventional process is still employed. Redesigning this scheme follows up the relationship between connection types and code numbers. This relationship will be described after reviewing the traditional nodal numbering scheme.

3. THE TRADITIONAL NODAL NUMBERING SCHEME

In the traditional nodal numbering scheme, code numbers of a member are referred from nodes, and the member exists between two nodes. The scheme will generate code numbers for every node in x, y and z directions. Then the member refers to both nodes at its edge for code numbers.

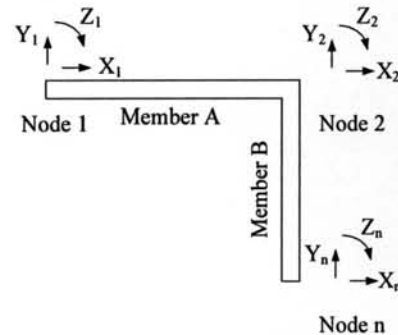


Fig.4: The traditional nodal numbering scheme

The algorithm for generating code numbers for every node is presented as follows:

Define

n is a node,

X_n is a code number in x-axis of node n,

Y_n is a code number in y-axis of node n and

Z_n is a code number in z-axis of node n

1. Assign ordering number to all nodes.

2. Set code number = 0

3. For all nodes

Code number = Code number + 1

X_n = Code number

Code number = Code number + 1

Y_n = Code number

Code number = Code number + 1

Z_n = Code number

End loop

This algorithm is simple but is used only for one type of structure. It is important to note that the number of code numbers at each node is always equal to three degrees, and values of these code numbers involve behaviors of a member connection. This relationship is described in the next section, and, as a result, it will lead to a new idea for generating code numbers.

4. RELATIONSHIPS BETWEEN CONNECTION TYPES AND CODE NUMBERS

A code number represents a direction of force or displacement at node in a structure. To formulate the force-displacement equation matrix of a structure, code numbers are needed to form many minor force-displacement equations. Each minor force-displacement equation is the relationship between external force and displacement in each direction at node from all members that are connected [1]. To form these equations using code numbers, relationships between characteristics of connections and code numbers are given below:

1. Direction, of which members are connected and transferred together, will use the same code number.
2. Direction, of which members are connected and not transferred together, will use a different code number.

As members connected by either hinged connection or rigid connection always transfer together in x and y directions, these members will share a code number in these directions. On the other hand, in z direction, the rotation of members connected together depends on the connected member data. Hence generating code numbers will be determined by characteristics of connections. These specifications lead to the new idea of generating the code numbers. The idea is described below.

1. The conventional frame programming model will be redesigned in process of generating code numbers.
2. A new algorithm is created to support the model and to generate code numbers for each node of members according to these constraints.

In x and y directions

- If members are connected, they will use the same code number.

In z direction

- If members are connected by rigid connection, they will use the same code number in z direction.

- If members are connected, but independently rotate, they will use the different code numbers in z direction.

These relationships are used to create the new algorithm for generating code numbers. Before the new algorithm is proposed, these relationships will be proved first in the next section.

5. FORMING OF STRUCTURAL EQUATIONS

In the previous section, the relationships between connections and code numbers are shown. In this section, the derivation of these relationships is presented. These relationships involve using member equations to form structural equations. The example of forming structural equations is shown in figure 5. At the end of forming, the relationships are indicated.

The structural equations, $[P] = [S][d]$ are external forces-structural displacement relationships, where $[P]$ is an external forces matrix, $[d]$ is a structural displacement matrix and $[S]$ is a force-displacement relationship coefficient matrix called structural stiffness matrix. In this case, only forming of the structural equation in x-axis of Node 2 is derived for an easier and better understanding. Forming equations for other directions uses the same processes. The equation of this direction is the relationship between the external force, P_4 , and structural displacements in every direction of every node, d_1 to d_9 , where S_{ij} are coefficient of these relationships, i is a direction of a force and j is a direction of a displacement.

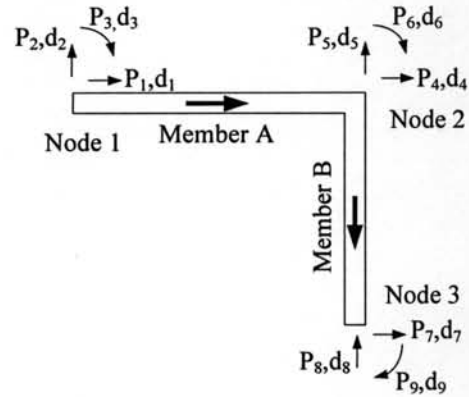


Fig.5: An example structure

In the first step, member equations, $[F] = [K][v]$, are formulated, where $[F]$ is a member end forces matrix, $[K]$ is a force-displacement coefficient matrix, called global member stiffness matrix, and $[v]$ is a displacement matrix at the end of the member. We can write the member stiffness relation, $[F] = [K][v]$, in expanded form as

$$\begin{bmatrix} F_1 \\ \vdots \\ F_6 \end{bmatrix} = \begin{bmatrix} K_{11} & \dots & K_{16} \\ \vdots & \ddots & \vdots \\ K_{61} & \dots & K_{66} \end{bmatrix} \begin{bmatrix} v_1 \\ \vdots \\ v_6 \end{bmatrix} \tag{1}$$

Directions of forces and displacements of the member are always referred as shown in figure 6. Force numbers are started from x-axis, y-axis and z-axis of the beginning node, and then the ending node. The beginning node and the ending node are assigned by the arrow at the middle of the members.

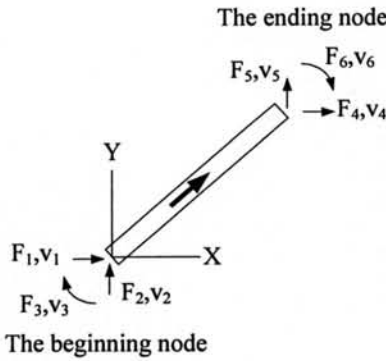


Fig.6: Member forces and displacements

Directions of forces and displacements of member A and B are shown in figure7.

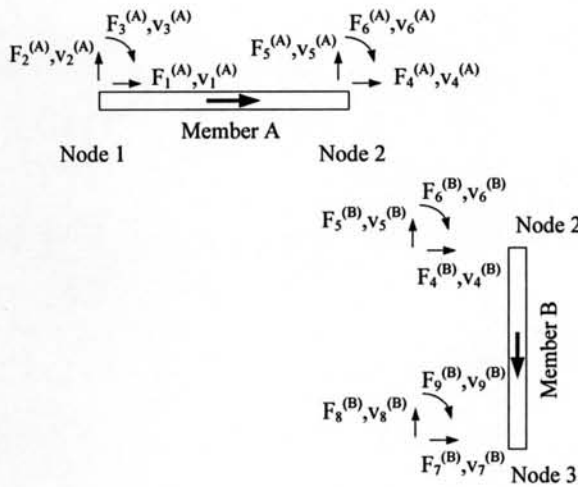


Fig.7: Member forces and displacements

The force-displacement equations for member A are the same as equation (1) while equations for member B are

$$\begin{bmatrix} F_4 \\ \vdots \\ F_9 \end{bmatrix} = \begin{bmatrix} K_{44} & \dots & K_{49} \\ \vdots & \ddots & \vdots \\ K_{94} & \dots & K_{99} \end{bmatrix} \begin{bmatrix} v_4 \\ \vdots \\ v_9 \end{bmatrix} \quad (2)$$

In the next step, the free body diagram of the structure is drawn in figure 8.

Then member equations are used to formulate the structural equations in x-axis on Node 2 by applying them into equilibrium equations.

$$\sum F_x = 0, \quad (3)$$

There are three forces in x direction of Node 2, P_4 , $F_4^{(A)}$, $F_4^{(B)}$. Consequently, a summation of forces is

$$\sum F_x = P_4 - F_4^{(A)} - F_4^{(B)}, \quad (4)$$

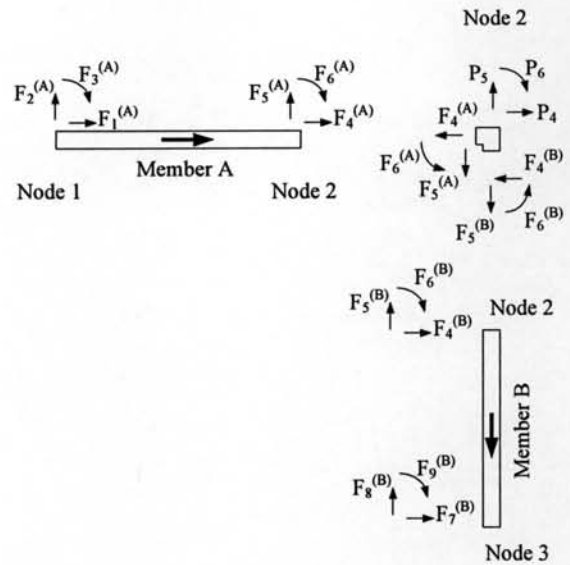


Fig.8: Free body diagram of the structure

Substitution of equation (4) into (3) forms a result in equation (5).

$$P_4 = F_4^{(A)} + F_4^{(B)}, \quad (5)$$

Substitutions of the equation (1) and (2) into (5) yield equation (6).

$$P_4 = K_{41}^{(A)} v_1^{(A)} + K_{42}^{(A)} v_2^{(A)} + K_{43}^{(A)} v_3^{(A)} + K_{44}^{(A)} v_4^{(A)} + K_{45}^{(A)} v_5^{(A)} + K_{46}^{(A)} v_6^{(A)} + K_{44}^{(B)} v_4^{(B)} + K_{45}^{(B)} v_5^{(B)} + K_{46}^{(B)} v_6^{(B)} + K_{47}^{(B)} v_7^{(B)} + K_{48}^{(B)} v_8^{(B)} + K_{49}^{(B)} v_9^{(B)} \quad (6)$$

Compare figure 5 with figure 7, and convert member displacements to structural displacements, $d_1 = v_1^{(A)}$, $d_2 = v_2^{(A)}$, $d_3 = v_3^{(A)}$, $d_4 = v_4^{(A)} = v_4^{(B)}$, $d_5 = v_5^{(A)} = v_5^{(B)}$, $d_6 = v_6^{(A)} = v_6^{(B)}$, $d_7 = v_7^{(B)}$, $d_8 = v_8^{(B)}$, $d_9 = v_9^{(B)}$. Then structural displacements, d_1 to d_9 , are substituted into equation (6), equation (7) is obtained as follows.

$$P_4 = K_{41}^{(A)} d_1 + K_{42}^{(A)} d_2 + K_{43}^{(A)} d_3 + K_{44}^{(A)} d_4 + K_{45}^{(A)} d_5 + K_{46}^{(A)} d_6 + K_{44}^{(B)} d_4 + K_{45}^{(B)} d_5 + K_{46}^{(B)} d_6 + K_{47}^{(B)} d_7 + K_{48}^{(B)} d_8 + K_{49}^{(B)} d_9 \quad (7)$$

Equation (7) can be derived into a simple form as equation (8).

$$P_4 = K_{41}^{(A)} d_1 + K_{42}^{(A)} d_2 + K_{43}^{(A)} d_3 + (K_{44}^{(A)} + K_{44}^{(B)}) d_4 + (K_{45}^{(A)} + K_{45}^{(B)}) d_5 +$$

$$(K_{46}^{(A)} + K_{46}^{(B)})d_6 + K_{47}^{(B)}d_7 + K_{48}^{(B)}d_8 + K_{49}^{(B)}d_9 \tag{8}$$

Represent K_{ij} in term of S_{ij} , equation (9) is obtained.

$$P_4 = S_{41}d_1 + S_{42}d_2 + S_{43}d_3 + S_{44}d_4 + S_{45}d_5 + S_{46}d_6 + S_{47}d_7 + S_{48}d_8 + S_{49}d_9 \tag{9}$$

Consider equation (8) and (9), S_{ij} is a summation of K_{ij} of member A and member B in the same direction, where i is a direction of an external force, and j is a direction of a displacement. Accordingly, S_{ij} can be computed by summation of K_{ij} according to the position. For example, in $S_{44} = K_{44}^{(A)} + K_{44}^{(B)}$, the first subscript of $K_{44}^{(A)}$ and $K_{44}^{(B)}$, 4 means a direction of a force in x axis of Node 2, while in the second subscript, 4 means a direction of a displacement in x axis of Node 2. Therefore, forming of S can be established easily by summation of all K_{ij} which share the same subscript. These subscriptions are called code numbers. In the programming view, the force-displacement coefficients of a member are kept in the matrix format as shown below

$$[K]^{(A)} = \begin{bmatrix} K_{11}^{(A)} & K_{12}^{(A)} & K_{13}^{(A)} & K_{14}^{(A)} & K_{15}^{(A)} & K_{16}^{(A)} \\ K_{21}^{(A)} & K_{22}^{(A)} & K_{23}^{(A)} & K_{24}^{(A)} & K_{25}^{(A)} & K_{26}^{(A)} \\ K_{31}^{(A)} & K_{32}^{(A)} & K_{33}^{(A)} & K_{34}^{(A)} & K_{35}^{(A)} & K_{36}^{(A)} \\ K_{41}^{(A)} & K_{42}^{(A)} & K_{43}^{(A)} & K_{44}^{(A)} & K_{45}^{(A)} & K_{46}^{(A)} \\ K_{51}^{(A)} & K_{52}^{(A)} & K_{53}^{(A)} & K_{54}^{(A)} & K_{55}^{(A)} & K_{56}^{(A)} \\ K_{61}^{(A)} & K_{62}^{(A)} & K_{63}^{(A)} & K_{64}^{(A)} & K_{65}^{(A)} & K_{66}^{(A)} \end{bmatrix}$$

and

$$[K]^{(B)} = \begin{bmatrix} K_{44}^{(B)} & K_{45}^{(B)} & K_{46}^{(B)} & K_{47}^{(B)} & K_{48}^{(B)} & K_{49}^{(B)} \\ K_{54}^{(B)} & K_{55}^{(B)} & K_{56}^{(B)} & K_{57}^{(B)} & K_{58}^{(B)} & K_{59}^{(B)} \\ K_{64}^{(B)} & K_{65}^{(B)} & K_{66}^{(B)} & K_{67}^{(B)} & K_{68}^{(B)} & K_{69}^{(B)} \\ K_{74}^{(B)} & K_{75}^{(B)} & K_{76}^{(B)} & K_{77}^{(B)} & K_{78}^{(B)} & K_{79}^{(B)} \\ K_{84}^{(B)} & K_{85}^{(B)} & K_{86}^{(B)} & K_{87}^{(B)} & K_{88}^{(B)} & K_{89}^{(B)} \\ K_{94}^{(B)} & K_{95}^{(B)} & K_{96}^{(B)} & K_{97}^{(B)} & K_{98}^{(B)} & K_{99}^{(B)} \end{bmatrix}$$

[S] is formulated by summation of $[K]^{(A)}$ and $[K]^{(B)}$ following code numbers. From the equation formulation technique using code numbers, it was found that forming the structural stiffness matrix, [S], is a summation of K_{ij} in the same directions and same code numbers. Therefore, whichever members that are connected and transferred together will use the same code numbers. On the other hand, whichever members that are connected but not

transferred together will use the different code numbers. These relationships are applied in the new algorithm.

6. THE NEW NODAL NUMBERING SCHEME

The new algorithm generates code numbers based on characteristics of connections in the structure. The number of code numbers in each node is not a constant. It varies in accordance with the characteristics of connected members in each node. This new algorithm is called "Dynamic Nodal Numbering Scheme". In this new nodal numbering scheme, code numbers are generated according to a member instead of a node (figure 9). Each member consists of two nodes: the beginning node and the ending node. An extreme node is identified by the head of the black arrow in the middle of the member. Each node has three degrees of freedom or three code numbers for x, y and z axes. Code numbers are assigned for each member at the beginning node and then the ending node as the following algorithm.

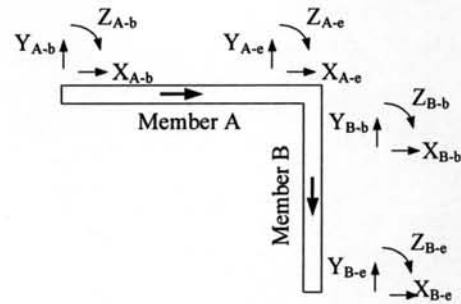


Fig.9: Dynamic nodal numbering scheme

Define,

m is a member,

X_{m-b} is a code number in x-axis of the beginning node of the member m

Y_{m-b} is a code number in y-axis of the beginning node of the member m

Z_{m-b} is a code number in z-axis of the beginning node of the member m

X_{m-e} is a code number in x-axis of the ending node of the member m

Y_{m-e} is a code number in y-axis of the ending node of the member m

Z_{m-e} is a code number in z-axis of the ending node of the member m

1. Assign order number to all members.

2. Set code number = 0

3. For all members

If the beginning node already exists then

X_{m-b} = Code number in x-axis from the existing node

Y_{m-b} = Code number in y-axis from the existing node

If the structure is simulated as plane frame

Z_{m-b} = Code number in z-axis from the existing node

```

Else
    Code number = Code number + 1
    Zm-b = Code number
End if
Else
    Code number = Code number + 1
    Xm-b = Code number
    Code number = Code number + 1
    Ym-b = Code number
    Code number = Code number + 1
    Zm-b = Code number
End if
If the ending node already exists then
    Xm-e = Code number in x-axis from the
    existing node
    Ym-e = Code number in y-axis from the
    existing node

If the structure is simulated as plane frame
    Zm-e = Code number in z-axis from the
    existing node
Else
    Code number = Code number + 1
    Zm-e = Code number
End if
Else
    Code number = Code number + 1
    Xm-e = Code number
    Code number = Code number + 1
    Ym-e = Code number
    Code number = Code number + 1
    Zm-e = Code number
End if
End loop
    
```

For the new technique, it uses only the frame element. All of the simulation processes are similar to the traditional frame simulation except the process of generating code numbers as shown in figure 2.

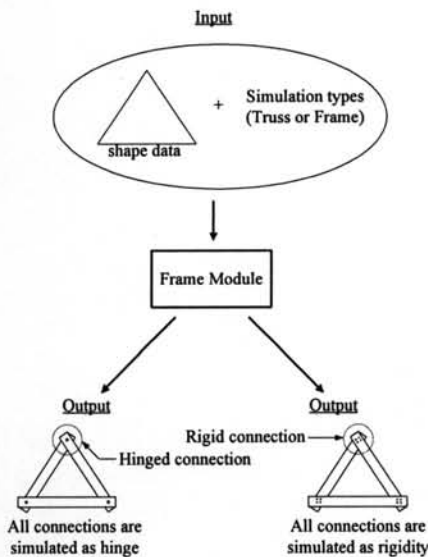


Fig.10: The new technique

The traditional nodal numbering algorithm for the frame simulation will be replaced by the new nodal numbering algorithm. The input data consists of the shape information and the simulation type information. Code numbers are generated following the simulation type information. Other simulation will use the same processes as the frame simulation, so the new technique has only the frame module as shown in figure 10. Nevertheless, it supports the simulation of both types of structures. Therefore, the new technique is an integration of the plane frame and plane truss simulations. In the next section, an example of generating code number for the plane frame simulation will be presented.

7. GENERATING CODE NUMBERS FOR THE PLANE FRAME SIMULATION

This section illustrates how to generate code numbers according to the new scheme for plane frame. The sample plane frame structure is shown in figure 11, of which all connections are rigid connections.

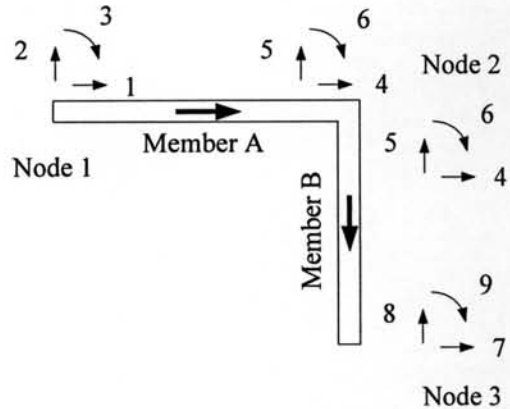


Fig.11: Code numbers for the plane frame

Code number generation of this structure starts at member A in the beginning node and then the ending node, i.e. Node 1 and Node 2, in x, y and z directions, respectively. Since the beginning node and the ending node of member A do not exist, all code numbers are newly created from numbers 1 to 6. Next, code numbers are generated for member B at the beginning node and the ending node; Node 2 and Node 3. Because the beginning node of member B is connected as rigid connection to the ending node of member A, of which code numbers are already generated, so code numbers in x and y axes of the beginning node of member B are copied from the ending node of member A. In z-axis, member A and member B are connected by rigid connection. They rotate concurrently. So, the code numbers in z direction of member B is copied from member A. Afterward, code numbers are created for the ending node of member B. As a result, the code numbers for the ending node of member B never exist; code numbers are refreshed from 7 to 9. The next section will show code numbers generation for the plane truss structure.

8. GENERATING CODE NUMBERS FOR THE PLANE TRUSS SIMULATION

This section shows the example that illustrates how to generate code numbers for the plane truss structure using the new nodal numbering scheme. The example is shown in figure 12. The freely rotating structure has two elements, connected by a hinged connection at Node 2.

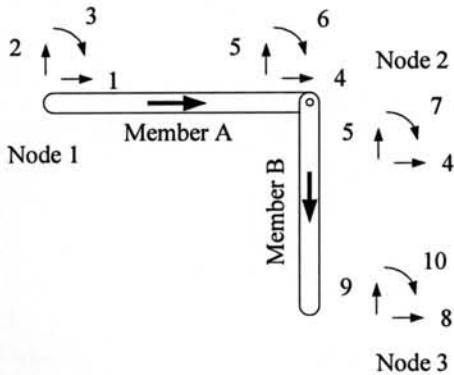


Fig.12. Code numbers for plane truss

Generation of code numbers starts at the beginning node and ending node of member A. These are Node 1 and 2. Since Node 1 does not connect to any members, and code numbers for Node 2 are not yet generated, code numbers are newly generated for the beginning node and the ending node of member A (from 1 to 6). Then, code numbers are generated for member B. Since the beginning node of member B connects to the ending node of member A, code numbers in x and y axes are copied from the existing node. These code numbers are 4 and 5 respectively. Because the structure is simulated as the plane truss, a code number for the beginning node of member B in z direction is generated newly as 7. Then new code numbers are created for the ending node of member B. Since this node does not connect to any members, code numbers in x, y and z axes are 8, 9 and 10.

9. ALGORITHM COMPARISON

Although the new algorithm uses only one element type for simulating both types of structures, which reduces software developing time, there are some issues that should be considered before using the new algorithm. The analysis of this algorithm is presented in this section, which consists of algorithm complexity and resource usage issues.

9.1 Algorithm Complexity

With regard to the traditional and the new algorithm that are proposed in the previous section, the complexity of both algorithms is $O(n)$. It seems to be similar but in fact it is actually not quite so. n in the traditional algorithm is the number of nodes while n in the new algorithm is the number of members. So both algorithms are suitable for a structure in different cases. We select two example structures to show how they are different.

The first example is the structure which consists of five members connected together as two closed loops shown in figure 13. The structure has four nodes and five members. In this case, the traditional algorithm is better than the new algorithm because cycles of code numbers generation which follow node are less than those which follow member.

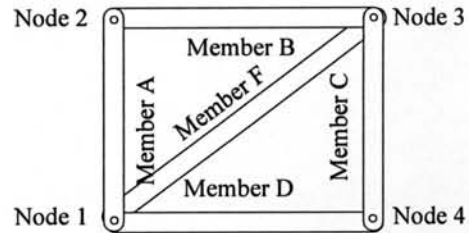


Fig.13. The first example

The second example, shown in figure 14, has two members and three nodes.

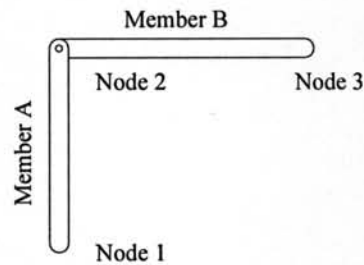


Fig.14. The second example

In this case, the new algorithm is better than the traditional algorithm. From these examples, it was found that the traditional algorithm is suitable for the structure that has many closed loops while the new algorithm is better when the problem consists of many non-closed loops structures.

However, the number of loops of code number generating has very little significant for simulating the real problem. Consider the large structure in figure 15 which has one hundred stories and ten bays. It contains approximately one thousand nodes and members. It can be easily dealt with the computing power of the current computer. In general, most problems in the 2D structural simulation are usually smaller than this problem.

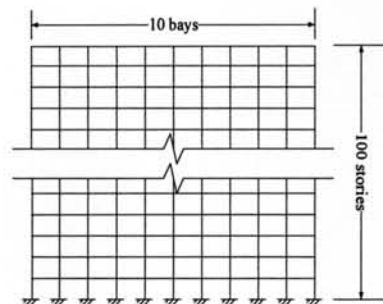


Fig.15. The large structure example

9.2 Resource Usages and Performance

To simulate a structure, the numbers of degrees of freedom and code numbers in a structure affect the number of equations and resource usages. The traditional technique provides truss and frame elements. The number of degrees of freedom depends on types of elements and the number of nodes. Truss element, as shown in figure 15, requires two degrees of freedom per node. These degrees of freedom consist of x-axis and y-axis. One degree of freedom in z-axis is neglected. Therefore, total degrees of freedom of truss structure are $2n$, when n is the number of nodes. There are three nodes in figure 16, so the total number of degrees of freedom is six.

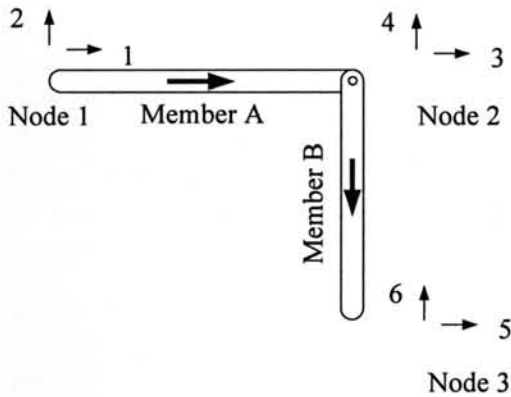


Fig.16. Degrees of freedom for truss structure created by the traditional algorithm

For frame element as shown in figure 17, it requires three degrees of freedom per node. These degrees of freedom are x-axis, y-axis and z-axis. Hence, the number of degrees of freedom for frame structure is $3n$, where n is the number of nodes. In figure 17, there are three nodes, so the number of degrees of freedom of this structure is nine.

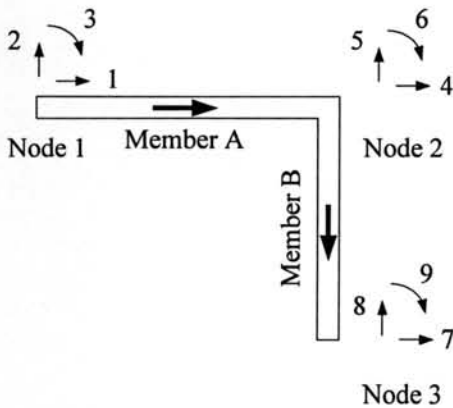


Fig.17. Degrees of freedom for frame structure created by the traditional algorithm

For the new algorithm, the number of degrees of freedom created for the frame simulation is the same as

traditional algorithm, but the truss simulation is different. In this case, the number of degrees of freedom does not only depend on the number of nodes but is also based on the number of members.

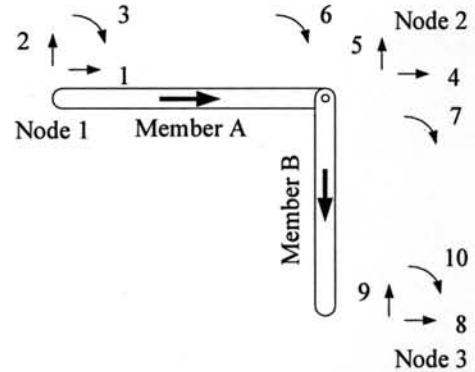


Fig.18. Degrees of freedom for truss structure created by the proposed algorithm

See figure 18, each node composes of two degrees of freedom, x-axis and y-axis. Each member contains two nodes. It has an individual degree of freedom in z-axis at the beginning node and the end node. As a result, the number of degrees of freedom is two degrees added-up per member in z-axis at the both end of the member.

In the example, degrees of freedom of number, 1,2,4,5,8 and 9 are created based on node while degrees of freedom of number 3 and 6 are set up for the member A, and degrees of freedom of number 7 and 10 for the member B. Consequently, the number of degrees of freedom is $2n + 2m$, where n is the number of nodes and m is the number of members.

Table 1: The number of degrees of freedom generated for a simulation

	Traditional Algorithm	New Algorithm
Plane Frame	$3n$	$3n$
Plane Truss	$2n$	$2n+2m$

Although the new algorithm creates more degrees of freedom than the traditional algorithm when it is used to simulate a truss structure, it reports more results in z-axis. The rotation in z-axis is required to simulate the deflection curve as shown in figure 19. The problem is the truss structure in which there is a load act on the member between Node C and Node B. To find the deflection curve in members as shown in figure 20 [9] by the traditional technique of the truss simulation, many nodes are created between Node C and Node B. It affects computation time. Using the new technique, it is not necessary to create nodes between members to find a deflection curve. Rotations in z axis are used to compute the deflection curve by double integration method [10-11]. The deflection curve is smoother, and the computational efficiency is better. For example, if 100 nodes are added between Node B and C, to draw the deflection curve using the traditional technique, degrees of freedom are 200 increased. In case of the new technique, it is not necessary to add nodes, only the

distance from the node to the position of the deflection that ones wants to know is required to substitute into the formula to compute the deflection curve. Moreover, the new technique reduces cost, time and effort of software development significantly because it requires only the frame element for structural analysis software development.

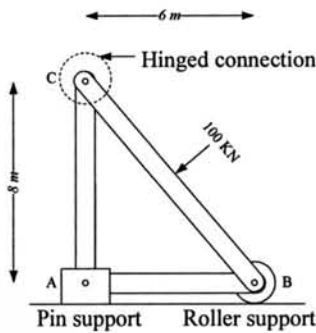


Fig.19. Truss structure with a load at the middle of a member

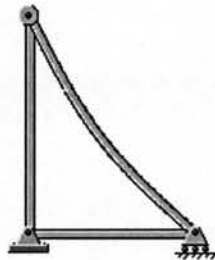


Fig.20. Deflection curve of truss structure in figure 19

10. CONCLUSIONS

This paper proposes the new algorithm for the 2D structural simulation. The new algorithm allows plane truss and plane frame structures to be simulated by using only one element type: frame element. This algorithm generates code numbers, which are used in the structural formulation processes. The new algorithm is created by improving the conventional plane frame simulation method by redesigning the algorithm in code number generation process. Since the number of code numbers generated by the new algorithm is changed according to types of connections in the structure, the new algorithm is called "Dynamic Nodal Numbering Scheme". Using the new algorithm, only the frame element type can manipulate both types of structures. Therefore, it reduces effort, time and cost of software developing, debugging and expanding.

As the new technique can simulate only plane truss and plane frame structures, which consist of only two types of connection: rigid and hinged connection, we are planning to improve the nodal numbering scheme for simulating the structure consisting of various formats of connection such as slider connection and hybrid connection, which is a combination of rigid and hinged connection, in the future.

References

- [1] Aslam Kassimali, *Matrix Analysis of Structures*, Brooks / Cole Publishing Company, California, 1999, ch. 3.
- [2] J. N. Reddy, *Finite Element Method*, McGraw-Hill Book Co, 1993, ch. 4.
- [3] Russel C. Hibbeler, *Structural Analysis*, Prentice-Hall, Inc., New Jersey, 1995, ch. 13-15.
- [4] Saeed Moaveni, *Finite Element Analysis*, Prentice-Hall, New Jersey, 1999. ch. 2, 8.
- [5] William Weaver, Jr. and James M. Gere, *Matrix Analysis of Framed Structures*, Chapman & Hall, New York, 1990. ch. 4.
- [6] Tezcan, S.S., "Matrix Simplified Formulation of Stiffness Matrices," *Journal of Structural Division*, ASCE 89(6), pp. 445-449, 1963.
- [7] Richard Rosen, "Matrix bandwidth minimization", *Proceedings of the 1968 23rd ACM national conference*, pp. 585-595, 1968.
- [8] E. Cuthill and J.McKee, "Reducing the bandwidth of sparse symmetric matrices", *Proceedings of the 1969 24th ACM national conference*, pp. 157-172, 1968.
- [9] Alongkorn Lamom and Tanongsak Bisarnsin, "Structural Analysis Software for Education", *Proceeding of Eighth National Convention on Civil Engineering (NCCE 8)*, pp. STR 119-123, 2002.
- [10] Russel C. Hibbeler, *Structural Analysis*, Prentice-Hall, Inc., New Jersey, 1995, ch. 8.
- [11] Ferdinand P. Beer and E. Russell Johnston, Jr., *Mechanics of materials*, McGraw-Hill Inc., Berkshire, England, 1992, ch. 8.



Alongkorn Lamom was born in 1976. He received the B.S. degree in Civil Engineering from Suranaree University of Technology in 1997 and the M.S. degree in Civil Engineering (structure) from Suranaree University of Technology in 2001. He is currently pursuing the Ph.D. degree in Computer Engineering at Chulalongkorn University where he has joined Software Engineering Laboratory since 2003. His research interests include structural simulation, distributed system and parallel computing.



Wanchai Rivepiboon was born in 1949. He is an associate professor of the School of Computer Engineering, Chulalongkorn University. He received the Ph.D. degree in Computer Science from University Grenoble 1. His research interests include structural simulation, software engineering and artificial intelligence.

Heuristic algorithm in optimal discrete structural designs

Alongkorn Lamom, Thaksin Thepchatri and Wanchai Rivepiboon
 Faculty of Engineering, Chulalongkorn University, Thailand, 10330
 Tel. +66-83-719995

Abstract: This paper proposes a Heuristic Algorithm for Material Size Selection (HAMSS). It is developed to handle discrete structural optimization problems. The proposed algorithm (HAMSS), Simulated Annealing Algorithm (SA) and the conventional design algorithm obtained from a structural steel design software are studied with three selected examples. The HAMSS, in fact, is the adaptation from the traditional SA. Although the SA is one of the easiest optimization algorithms available, a huge number of function evaluations deter its use in structural optimizations. To obtain the optimum answers by the SA, possible answers are first generated randomly. Many of these possible answers are rejected because they do not pass the constraints. To effectively handle this problem, the behavior of optimal structural design problems is incorporated into the algorithm. The new proposed algorithm is called the HAMSS. The efficiency comparison between the SA and the HAMSS is illustrated in term of number of finite element analysis cycles. Results from the study show that HAMSS can significantly reduce the number of structural analysis cycles while the optimized efficiency is not different.

Key words: Heuristic algorithm, Steel design, Optimization algorithm

INTRODUCTION

There are many techniques used to handle structural optimization problem. Deb and Gulati^[1] proposed techniques to design truss structures for minimum weight using genetic algorithm. Shan and Huanchun^[2] combined two algorithms to handle the discrete optimization of structures. Chen^[3] used the SA to place active/passive member in truss structures. Szewczyk and Hajela^[4] incorporated the SA and counter propagation neural network to perform structural optimization. Benage and Dhingra^[5] proposed three strategies in using the SA to solve single and multi-objective structural optimization problems. Chen and Su^[6] suggested two methods to improve SA efficiency in optimal structural designs. Although these referenced techniques can be used to handle structural optimization problem, large numbers of finite element analysis are needed to improve the result. The main problems of using huge number of finite element analysis cycles are that many rejected answers are generated. Consequently, a waste of computer time occurs.

Although the SA is a simple and quite easy technique for implementation, there are many function evaluations needed to find optimal answers in structural optimization problems. Since possible answers given by the SA are randomly generated, the percentage of accepted answers is low. Only answers which pass constraint checks are kept. Many generated answers, of course, are rejected because they do not pass constraints. These constraints, known as filters, involve member abilities to support both tension load and compression load. After possible answers are filtered by constraints, the best answer is searched. A typical

optimization problem is modeled as shown in Fig 1. Theoretically, the probability to generate a new answer at the left hand side and right hand side of the current answer is equal. In structural optimization problems, however, the typical optimized algorithm can be modified to reduce the computation time. In this study, the problem understanding is used to help developing the new algorithm, the HAMSS. Because the structural analysis process spends most of the computation time in optimization process, the HAMSS will reduce the structural analysis cycles.

Comparison results of the HAMSS, the SA and Multiframe 4D software with three truss examples are presented to demonstrate algorithm's effectiveness at the end of the paper. The first example is a typical 3 bar planar truss, the second one is a 26 bar planar truss and the last one is a 30 bar curvature planar truss.

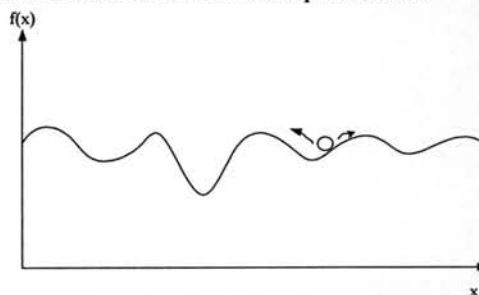


Fig. 1. Structural optimization problem
HEURISTIC ALGORITHM

Heuristic searching uses knowledge called heuristic to improve the searching efficiency. Heuristic will indicate trend of answers and guide which search route should be better. To show an overview of this method,

the classical problem in heuristic algorithm is exemplified. The traveling salesman problem is shown in Fig 2. The objective of this example is to discover the shortest path of salesman's traveling.



Fig. 2. Traveling salesman problem

As shown in Fig 2^[7], there are 7 cities. Salesman has to go to all cities and comeback to the started city. The basic method to find the shortest path is generating all routes that is possible, there are $(7-1)!/2$ or 360 routes, then measuring distance of each route. A route which has shortest distance is the best solution. However, this method is not practically effective when there are many cities. For example, if 100 cities are goaled, there are 4.67×10^{155} traveling paths. In this case, it will take long time to find the optimal answer.

A better method to find the shortest path is to use the knowledge of problem understanding to guess the answer. It is obvious that the next selected city for going should be the nearest city. If the every decisions of city selection are nearest city from the current city, the good answer may be investigated. This is an example of using the knowledge to help solving the problem. The selected path may not be the best answer but it is an acceptable one. Using an uncompleted knowledge to handle a problem or a reasonable guessing is called heuristic.

In the SA, although there are many proposed techniques for predicting feasible zone, answers generated for optimal structural problem is randomized. New answers are generated around the present answer as shown in Fig 1. Each new answer is then checked if it passes all constraints. In this study, behavior of optimal structural problem is studied and is found to be similar to that shown in Fig 3 with y-axis is the steel volume and x-axis is the cross sectional area of steel in one group. Trend of the best answer occurs at the left hand side of the graph and locates above the load constraint which is converted to be allowable minimum cross sectional area constraint. This constraint varies with steel cross sectional area.

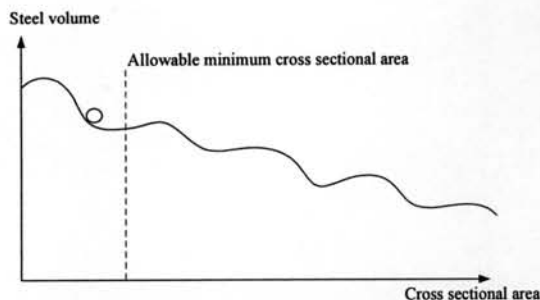


Fig. 3. Structural optimization problem behavior

If the gap between answer and allowable minimum cross sectional area is small, the best answer should be resulted. The proposed algorithm is developed from this knowledge. The algorithm will be demonstrated in the next section.

HEURISTIC ALGORITHM FOR MATERIAL SIZE SELECTION

Firstly, the maximum size from the given material data is assigned to be the started answer. Then a new answer is randomly created by reducing size of material as shown in Fig 4. If this material size is below the allowable minimum cross sectional area line, i.e. the size does not pass the load constraints. A new answer is created by increasing material size one step. This procedure is repeated until corrected size is obtained as shown at Fig 5. The algorithm will repeat the process until the steel volume of a new answer does not change for 2 times. The last answer is the best result as shown in Fig 6.

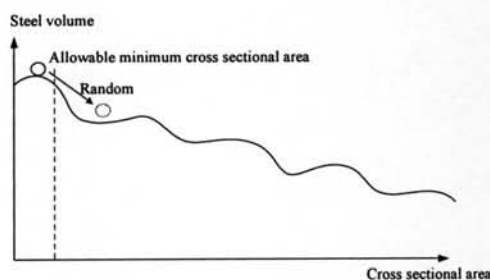


Fig. 4. Reducing material size

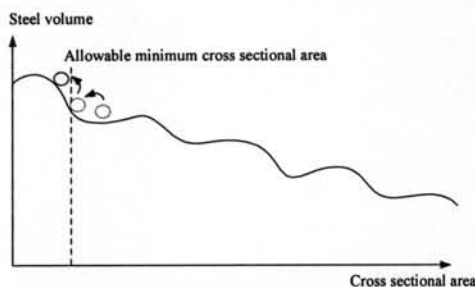


Fig. 5. Increasing material size

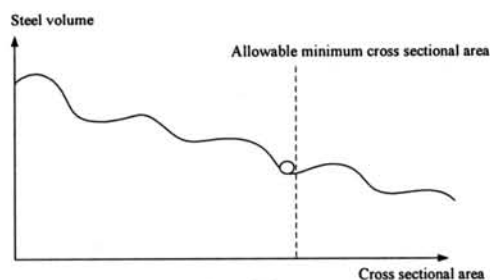


Fig. 6. Algorithm stop

The Heuristic Algorithm for Material Size Selection is shown in table 1. Step 1, all members in a structure are assigned to be the biggest size. Step 2 is the stopping evaluation. If the steel volume of the new answer is repeated twice, the algorithm will stop and the last best answer is a final result. In case that the steel volume can be reduced, Step 2.1 will operate. All member sizes are reduced randomly in this step. Members having the same group number, however, will have the same size. Next step is the constraint examination. In step 2.2, all members are verified if their sizes are large enough for load constraints. If material size of any member group is unacceptable, material size of that member group is increased one size at step 2.2.1 until all member sizes pass the load constraints. In step 2.3, the new answer is evaluated. If the new answer is better than the previous one, the current answer is replaced by the new answer. If the new answer is equal to the current answer, the new answer is assigned to be an initial answer for the next loop. These processes at step 2 are recursively operated until the constraint at step 2 is qualified. Then, the last best answer is set to be the final result.

Table 1: Heuristics algorithm for material size selection

1.	Evaluate the initial state which is the biggest size.
2.	UNTIL a volume of current state and a volume of previous state are equal 2 time continuously DO
2.1	Reduce sizes of materials randomly.
2.2	UNTIL size of materials are enough for loads DO
2.2.1	Increase sizes of materials one step
2.3	Evaluate the new state.
2.3.1	Compute the structural volume
	IF volume of new state=volume of previous state then
	Return new state and quit
	ELSE IF the new state is better than the current state
	THEN { current state := new state }
	ELSE { current state := old state }
3.	Return the last state as the answer.

NUMERICAL EXAMPLE

Three truss structures are used to verify the proposed algorithm. Each example is studied by using a commercial steel design software and two algorithms. The selected steel design software is Multiframe 4D.^[8] This program is a specialized steel design software widely used in civil engineering. The two algorithms are the Simulated Annealing Algorithm (SA) and the Heuristic Algorithm for Material Size Selection (HAMSS). These algorithms are implemented in SUTStructure^[9, 10]. SUTStructure is an education structural analysis software used to analyze two dimensional truss and frame structures. Structural design criteria used in the Multiframe and the SA is the Allowable Stress Design (ASD) given in the AISC 1989 specification. The HAMSS algorithm however uses the ASD from both the AISC 1989 and AISC 2005 specifications^[11].

Steel shape used in the study is the hollow round steel as listed in Table 2. There are 29 sizes. Modulus of Elasticity, E , is 196×10^6 kN/m² (2×10^{10} kg/m²) and yield stress, F_y , is 245000 kN/m² (25000000 kg/m²).

Table 2: Hollow round steel

Number	Section	Outside diameter (mm)	Thickness (mm)	Area (m ²)	Radius of gyration (mm)
1	P15×2	21.7	2	0.000124	7
2	P20×2	27.2	2	0.000158	8.9
3	P20×2.3	27.2	2.3	0.000180	8.8
4	P25×2.3	34.0	2.3	0.000229	11.2
5	P32×2.3	42.7	2.3	0.000292	14.3
6	P40×2.3	48.6	2.3	0.000335	16.4
7	P40×2.8	48.6	2.8	0.000403	16.2
8	P50×2.3	60.5	2.3	0.000421	20.6
9	P40×3.2	48.6	3.2	0.000456	16.1
10	P50×3.2	60.5	3.2	0.000576	20.3
11	P65×2.8	76.3	2.8	0.000647	26
12	P50×4	60.5	4.0	0.000710	20
13	P65×3.2	76.3	3.2	0.000735	25.6
14	P80×2.8	89.1	2.8	0.000759	30.5
15	P80×3.2	89.1	3.2	0.000864	30.4
16	P90×3.2	101.6	3.2	0.000989	34.8
17	P100×3.2	114.3	3.2	0.001117	39.3
18	P90×4	101.6	4.0	0.001226	34.5
19	P125×3.6	139.8	3.6	0.001540	48.2
20	P100×4.5	114.3	4.5	0.001552	38.9
21	P125×4	139.8	4.0	0.001707	48
22	P125×4.5	139.8	4.5	0.001913	47.9
23	P150×4.5	165.2	4.5	0.002272	56.8
24	P150×5	165.2	5	0.002516	56.7
25	P125×6	139.8	6	0.002522	47.4
26	P200×4.5	216.3	4.5	0.002994	74.9

27	P150×6	165.2	6	0.003001	56.3
28	P200×5.8	216.3	5.8	0.003836	74.5
29	P200×7	216.3	7	0.004603	74

Example 1

The first example is a typical planar truss structure. The structure is a 3 bar planar truss. All members in the structure are connected by hinged connections. Load 49 kN (5000 kg) acts along both X and Y-axis as shown in Fig 7. Material group information is given in Table 3. Materials sizes are divided into 2 groups. Material sizes of member 1 and member 3 are equal and assigned to be group 1 while member 2 is different and assigned to be group 2.

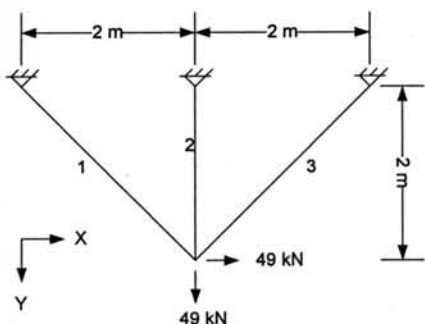


Fig. 7. A 3 bar planar truss. (Example 1)

Table 3: Group information (Example 1)

Group	Members
1	1, 3
2	2

The problem optimization is studied by using the Multiframe software program, the SA and the HAMSS algorithms. In the Multiframe, the optimal steel design required 2 important commands, the analysis and the design commands. To find the best answer, the analysis command will be called first then followed by the design command. These processes will be repeated until sizes of the designed steel do not change. Then the biggest steel size in each group is selected to be the selected size of that group. From the study, the optimum steel volume of the structure is 2528 cm³. Selected steel sizes are shown in Table 4.

In the SA, the design criteria used with this algorithm is based on the AISC 1989. The example is run with this algorithm for 25 times. The best design from these results is selected to be the answer. To obtain the optimal answer, the SA uses 196 finite element analyses. The optimum steel volume of the structure designed by the SA is 2528 cm³. Selected steel sizes are listed in Table 3. The intermediated results from log file are plotted as convergent graph in Fig 8. It should be noted that both the Multiframe and the SA algorithm yield the same optimum steel volume.

In the HAMSS, both AISC 1989 and AISC 2005 (ASD) are studied. The example is tested with this algorithm for 25 times. The optimum steel volume of the structure designed by the HAMSS with both AISC/1989 and AISC /2005 (ASD) is 2528 cm³. The intermediated results from the HAMSS for AISC 2005 (ASD) standard are plotted and shown in Fig 9. The gray points show all answers generated in the process of trial to find the optimal solution. The dark points on the graph are accepted answers. The HAMSS used only 16 finite element analyses. The final results reported from the implemented program are demonstrated in Fig 10.

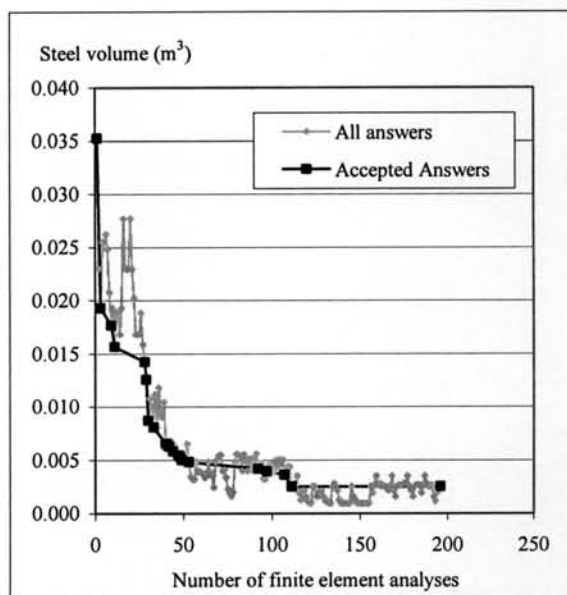


Fig. 8. Convergent graph of the SA (Example 1)

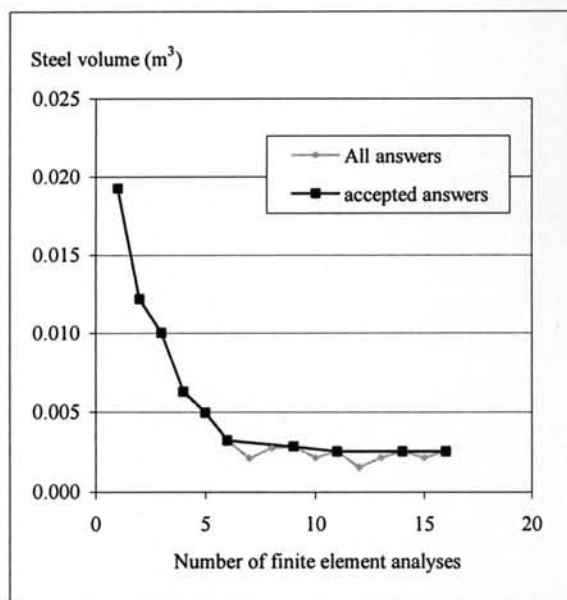


Fig. 9. Convergent graph of the HAMSS (Example 1)

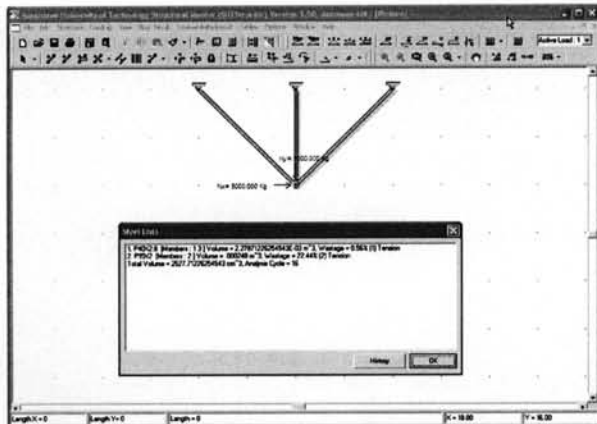


Fig. 10. Results from the implemented program (Example 1)

Table 4 reports steel volume and selected steel sizes designed by the Multiframe, the SA and the HSMSS algorithms. All design techniques give the same results. However, it should be pointed out that the HAMSS required only 16 finite element analyses when compares to 269 finite element analyses used in the SA.

Table 4: Results from algorithms (Example 1)

Group	Multiframe (ASD1989)	SA (ASD1989)	HAMSS (ASD1989)	HAMSS (ASD2005)
1	P40x2.8	P40x2.8	P40x2.8	P40x2.8
2	P15x2	P15x2	P15x2	P15x2
Volume	2528 cm ³	2528 cm ³	2528 cm ³	2528 cm ³

Example 2

The second example is a 26 bar planar truss with joint loads as shown in Fig 11. All connections in the structure are hinged connections. The truss is also pin supported at both ends. The structure is 1.5 m high and 10 m long.

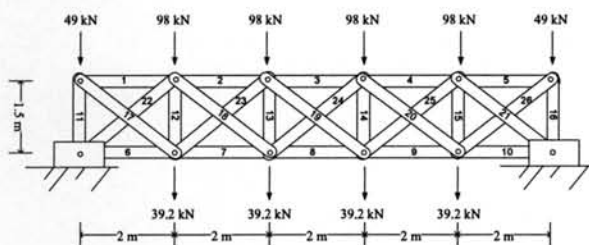


Fig. 11. A 26 bar planar truss. (Example 2)

Material sizes in the structure are assigned into 4 groups. Information of members using the same steel size is shown in Table 5.

Table 5: Group information (Example 2)

Group	Members
1	1, 5, 6-10, 11-16
2	2, 3, 4
3	17-20, 23-26
4	21, 22

The steel volume of the structure obtained from the Multiframe is 103289 cm³. The selected steel sizes are shown in Table 6.

In the SA, the example is run with this algorithm for 25 times. The best design from these results is selected to be the answer. Fig 12 shows convergent performance of the SA. To find the optimal answer in this example, the SA requires 719 finite element analyses. The optimum steel volume of the structure designed by the SA is 112785 cm³. Selected steel sizes are shown in Table 6. According to the information in Fig 12, it can be seen that many answers are generated but they are mostly rejected. Most of them do not pass steel design criteria. Many rejections cause much computation time for answer validations. These times are totally waste.

In the HAMSS, The optimum steel volume of the structure designed using the ASD 1989 standard is 89010 cm³ and it requires only 20 finite element analyses. Also in the HAMSS with ASD 2005, the optimum steel volume of the structure is 83073 cm³ and it spends only 20 finite element analyses to get the answer. These results show that the HAMSS is more powerful than the SA in term of number of finite element analyses. The convergent performance graph of HAMSS for the ASD 2005 version is shown in Fig 13. Only 14 rejected answers are created. The final results reported from the implemented program are presented in Fig 14.

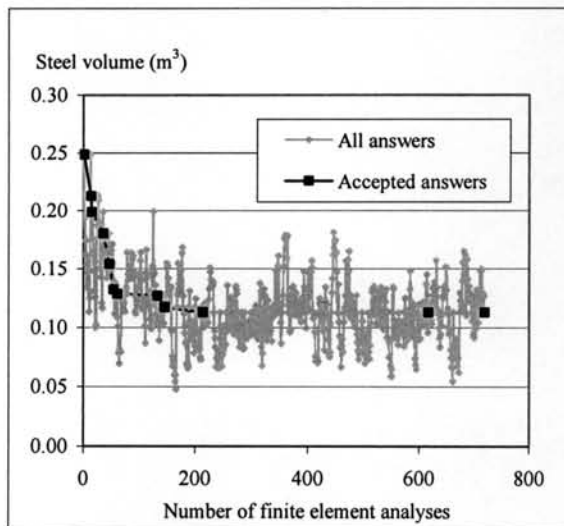


Fig. 12. Convergent graph of the SA (Example 2)

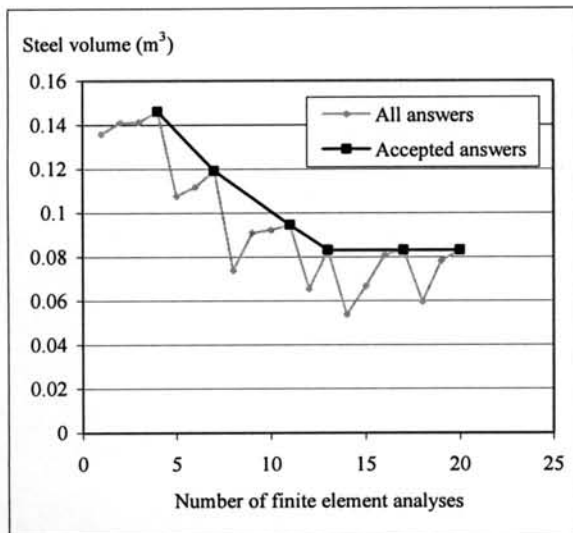


Fig. 13. Convergent graph of the HAMSS (Example 2)

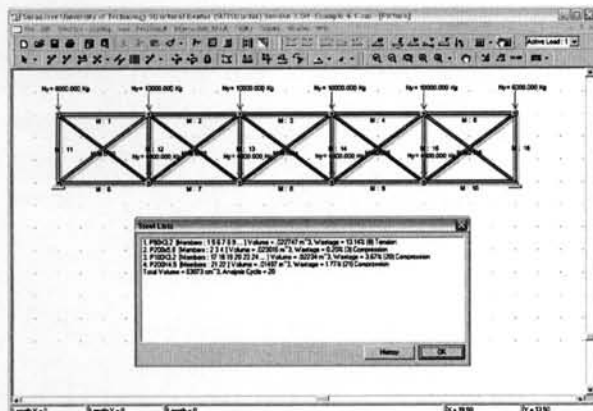


Fig. 14. Results from the implemented program (Example 2)

Table 6 shows the optimum structural steel volume and selected material sizes designed by each technique.

Table 6: Results from algorithms (Example 2)

Group	Multiframe (ASD1989)	SA (ASD1989)	HAMSS (ASD1989)	HAMSS (ASD2005)
1	P100x3.2	P125x4.5	P80x3.2	P90x3.2
2	P200x7	P200x5.8	P200x7	P200x5.8
3	P125x3.6	P125x3.6	P100x3.2	P100x3.2
4	P200x5.8	P200x4.5	P200x5.8	P200x4.5
Volume	103289 cm ³	112785 cm ³	89010 cm ³	83073cm ³

According to information in Table 6, the optimum steel volumes of the structure designed using both versions of the HAMSS, AISC/1989 and AISC/2005, are also significantly better than the optimum steel volume of the structure designed using the Multiframe and the SA.

Example 3

The third example is a 30 bar planar truss with joint loads as shown in Fig 15. The structure is 5.5 m high and 28 m long. The structure is pin supported at both ends. All connections in the structure are hinged connections. Material sizes in the structure are assigned into 5 groups. Members that use the same steel size are grouped and shown in Table 7.

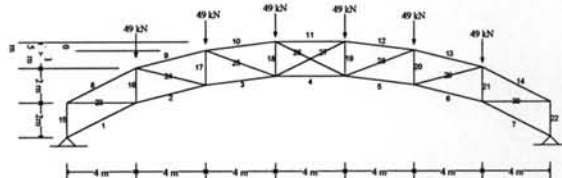


Fig. 15. A 26 bar planar truss (Example 3)

Table 7: Group information (Example 3)

Group	Members
1	1-7
2	8-14
3	15-22
4	23-25, 28-30
5	26, 27

The example is computed by the Multiframe software, the SA and both versions of the HAMSS. The optimum steel volume of the structure designed by the Multiframe is 169277 cm³. Selected steel sizes are shown in Table 8.

In the SA, the design criteria used with this algorithm is based on the AISC 1989. The example is run with this algorithm for 25 times. The best design from these results is selected to be the answer. To obtain the optimal answer, the SA uses 829 finite element analyses. The optimum steel volume of the structure designed by the SA is 162485 cm³. Selected steel sizes are listed in Table 8. The intermediated results from log file are plotted as convergent graph in Fig 13. The dark points are the accepted answers and the gray points are the rejected answers. This information show that the SA can find the optimal solution, but many waste answers are created before the best answer is found. Most answers are rejected because either the answers do not pass the constraints or the answers are not better than the previous answer.

In the HAMSS, both AISC 1989 and AISC 2005 (ASD) are studied. The optimum steel volume of the structure which is designed by the HAMSS with both AISC/1989 and AISC /2005 (ASD) is 162485 cm³. The intermediated results from the HAMSS for AISC 2005 (ASD) standard are graphically shown in Fig 16. It should be noted that the number of finite element analyses in Fig 16 is less than the number of finite element analyses in Fig 15. Final results reported from

the implemented program based on HAMSS (AISC/ 2005) is shown in Fig 17.

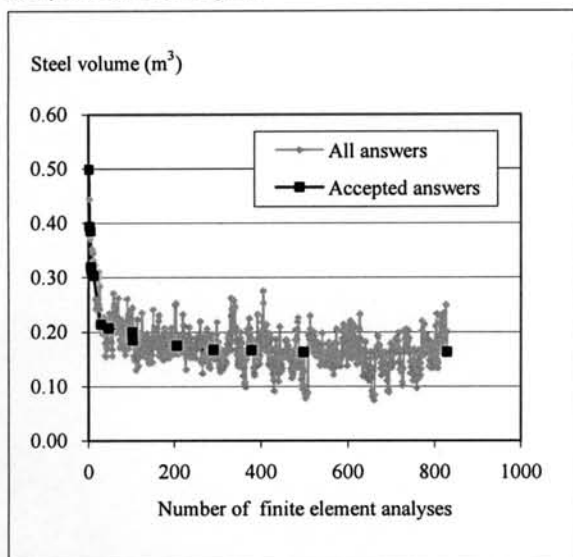


Fig. 15. Convergent graph of the SA (Example 3)

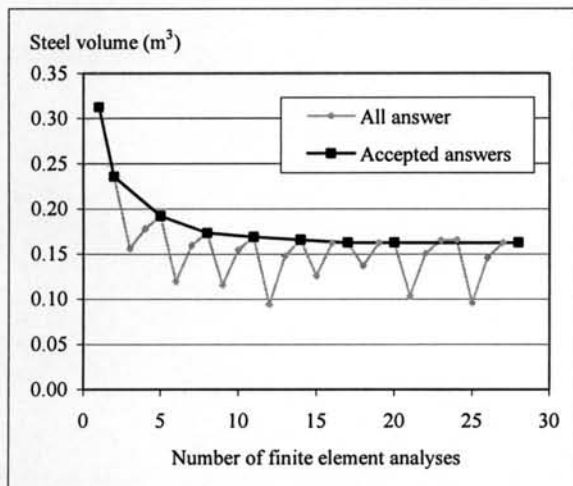


Fig. 16. Convergent graph of the HAMSS (Example 3)

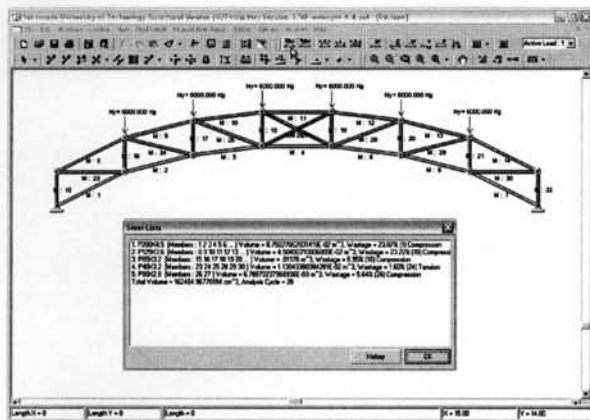


Fig. 17. Results from the implemented program (Example 3)

Table 8 shows the optimum structural steel volume and selected material size designed by each technique.

Table 8: Result from algorithms (Example 3)

Group	Multiframe (ASD1989)	SA (ASD1989)	HAMSS (ASD1989)	HAMSS (ASD2005)
1	P200x4.5	P150x6	P200x4.5	P200x4.5
2	P125x3.6	P125x3.6	P125x3.6	P125x3.6
3	P65x3.2	P80x2.8	P65x3.2	P65x3.2
4	P65x2.8	P40x3.2	P40x3.2	P40x3.2
5	P90x3.2	P80x2.8	P80x2.8	P80x2.8
Volume	169277 cm ³	163074 cm ³	162485 cm ³	162485 cm ³

From the study, the optimum steel volumes of the structure designed by both versions of the HAMSS are better than the SA. In addition, the HAMSS with AISC 2005 specifications requires only 28 finite element analyses when compares to 829 analyses used in the SA

DISCUSSION AND CONCLUSIONS

The paper proposes a new algorithm to handle discrete variable structural optimization problems. Since the proposed algorithm is developed by using knowledge of problem behavior, it is called the Heuristic Algorithm for Material Size Selection (HAMSS). The proposed algorithm is decrypted and evaluated. Three examples of planar steel truss are tested with the Simulated Annealing Algorithm (SA), the proposed algorithm (HAMSS) and a conventional steel design software (Multiframe). Results from the study show that both the SA and the HAMSS can efficiently optimize steel volume of planar truss structures. However, the HAMSS is more effective because it significantly spends less number of finite element analysis cycles when compares to the SA. In addition, any intricate mathematical function is not required for HAMSS. Hence, the HAMSS is an efficient optimization technique which is easy for understanding and simple for implementation.

In the future, the HAMSS would be used to solve other problems which are similar to the optimal discrete structural design problem. The other types of structural material and other types of structure will be tested with HAMSS.

REFERENCES

1. Kalyanmoy Deb, Surendra Gulati, 2001. Design of truss-structures for minimum weight using genetic algorithms. Finite element in analysis and design, 37(5): 447-465.
2. Chai Shan, Sun Huanchun, 1997. A combinatorial algorithm for the discrete optimization of structures. Applied mathematics and mechanics, 18(9): 847-856.
3. Chen GS, Bruno RJ, Salama M, 1991. Optimal placement of active/passive members in truss

- structures using simulated annealing. *AIAA Journal*, 29(8): 1327-1334.
4. Szewczyk Z, Hajela P, 1993. Neural network approximation in a simulated annealing based optimal structural design. *Structural and Multidisciplinary Optimization*, 5(3):159-165.
 5. Bennage WA, Dhingra AK, 1995. Single and multiobjective structural optimization in discrete-continuous variables using simulated annealing. *International Journal for Numerical Method in Engineering*, 38(16): 2753-2773.
 6. Ting-Yu Chen, Jyh-Jye Su, 2002. Efficiency improvement of simulated annealing in optimal structural designs. *Advances in engineering software*, 33(7-10): 675-680.
 7. Boonserm.k, 2003. *Artificial Intelligence*, pp: 16-17
 8. Formation Design Systems. *Multiframe4D*, <http://www.formsys.com/multiframe>
 9. Alongkorn Lamom and Tanongsak Bisarnsin, 2002. *Structural Analysis Software for Education*, in *Proceeding of the 8th National convention on civil engineering, Thailand*, pp. STR 119-123.
 10. Alongkorn Lamom and Wanchai Rivepiboon, 2005. *A Nodal Numbering Algorithm for 2D Structural Simulation*. *ECTI Transactions on Computer and Information Technology*, 1(2): 108-116.
 11. AISC, 2005. *Specification for structural steel buildings*, American institute of steel construction.

ประวัติผู้เขียนวิทยานิพนธ์

อลงกรณ์ ละม่อม เกิดวันที่ 29 กรกฎาคม พ.ศ. 2519 ในจังหวัดอุบลราชธานี เข้ารับการศึกษาในระดับปริญญาบัณฑิตที่มหาวิทยาลัยสุรนารี สาขาวิชาวิศวกรรมโยธา ในปี 2537 และสำเร็จการศึกษาในปี 2540 และเข้าทำงานในบริษัทสีมาครีเอทในตำแหน่งวิศวกรผู้ออกแบบ

ผู้เขียนเข้าศึกษาในระดับปริญญาโทบัณฑิตในปี 2542 ที่มหาวิทยาลัยเทคโนโลยีสุรนารี ในสาขาวิชาวิศวกรรมโยธาแขนงโครงสร้าง โดยทำงานวิจัยและพัฒนาโปรแกรมวิเคราะห์โครงสร้างเพื่อการเรียนการสอน (SUTSstructor) หลังจบการศึกษาในปี 2544 ได้เข้าทำงานที่มหาวิทยาลัยมหาสารคามในตำแหน่งอาจารย์ประจำคณะวิศวกรรมศาสตร์

ในปี 2546 ผู้เขียนเข้าศึกษาต่อในระดับปริญญาดุษฎีบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์ ที่จุฬาลงกรณ์มหาวิทยาลัย โดยได้รับทุนพัฒนาอาจารย์จากมหาวิทยาลัยมหาสารคาม