

การ Interface ระหว่างโปรแกรมภาษา C และโปรแกรมไฟไนต์เอลิเมนต์ ANSYS ในการโทปอลอจีมัลติออบเจกทีฟออปติไมเซชัน C Program and Finite Element Program ANSYS Interface Methodology for Topology Multi-Objective Optimization

จักษณี วิรุฬศรี^{1*} ไพรัช ตั้งพรประเสริฐ² และ Shigeru TADANO³

^{1,2}ภาควิชาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย เขตปทุมวัน กรุงเทพฯ 10330

³Research Group of Biomechanics and Robotics, Graduate School of Engineering, Hokkaido University, Japan 060-8628

*E-mail: juksanee.v@chula.ac.th

Juksanee Virulsri^{1*} Pairat Tangpomprasert² and Shigeru TADANO³

^{1,2}Department of Mechanical Engineering, Faculty of Engineering, Chulalongkorn University, Pathumwan, Bangkok 10330

³Research Group of Biomechanics and Robotics, Graduate School of Engineering, Hokkaido University, Japan 060-8628

*E-mail: juksanee.v@chula.ac.th

บทคัดย่อ

บทความนี้มีวัตถุประสงค์เพื่อนำเสนอระเบียบวิธีการ Interface ระหว่างโปรแกรมภาษา C และโปรแกรมไฟไนต์เอลิเมนต์ ANSYS โดยในปัจจุบันการคำนวณด้วยระเบียบวิธีไฟไนต์เอลิเมนต์นั้น ได้มีโปรแกรมไฟไนต์เอลิเมนต์ในเชิงการค้าต่างๆ ออกมามากมาย ดังเช่น โปรแกรม ANSYS ซึ่งมีฟังก์ชันในการคำนวณทางวิศวกรรมที่ค่อนข้างซับซ้อนได้อย่างหลากหลาย อย่างไรก็ตามโปรแกรมไฟไนต์เอลิเมนต์ในเชิงการค้าดังกล่าวก็ยังไม่ฟังก์ชันในการคำนวณเฉพาะทางบางประเภท เช่น การทำ topology multi-objective optimization ส่วนโปรแกรมภาษา C นั้นมีข้อดีคือ สามารถเขียนโปรแกรมที่ซับซ้อนได้ และยังมีความสามารถในการทำการคำนวณได้รวดเร็ว ดังนั้นการ interface ระหว่างสองโปรแกรมหาดังกล่าวซึ่งต่างก็มีข้อดีในตัวเอง ให้สามารถติดต่อกันได้จะช่วยทำให้การคำนวณ และการออกแบบงานทางด้านวิศวกรรมสามารถทำได้โดยสะดวก และมีประสิทธิภาพยิ่งขึ้นมาก นอกจากนี้โปรแกรม interface ที่ได้พัฒนาขึ้นในงานวิจัยนี้ จะนำไปทดลองใช้กับตัวอย่างของปัญหาการออกแบบรูปร่างคาน โดยวัตถุประสงค์ในการออกแบบนี้มี 2 วัตถุประสงค์ คือต้องการออกแบบให้ได้น้ำหนักน้อย ในขณะที่เดียวกันก็ต้องทำให้มีการงอตัวที่น้อยด้วย ผลลัพธ์ของการคำนวณที่ได้แสดงให้เห็นว่าระเบียบวิธีการ Interface ที่นำเสนอเป็นวิธีการหนึ่งที่จะช่วยให้การ Interface ระหว่าง topology multi-objective optimization (C program) กับ โปรแกรมไฟไนต์เอลิเมนต์ในเชิงการค้า (ANSYS program) ในการออกแบบชิ้นงานทางด้านวิศวกรรมต่างๆ สามารถทำได้เป็นผลสำเร็จได้โดยง่ายและมีประสิทธิภาพ

คำหลัก: ระเบียบวิธีการ interface, โทปอลอจีมัลติออบเจกทีฟออปติไมเซชัน, โปรแกรม ANSYS, โปรแกรมภาษา C

Abstract

The purpose of this paper is to propose a methodology of interfacing between C language program and finite element program ANSYS. Recently, there are many advanced commercial finite element program, such as ANSYS program, that is efficient for calculation of complicated engineering problems. Nevertheless, such commercial finite elements programs still not have some special functions for calculation, for example, topology multi-objective optimization. On the other hand, C language program is efficient for creation of complicated program. Moreover, it is high-speed in computation. Therefore, making the communication between these efficient programs will assist in engineering design being convenient and efficient. The developed interface program in this research was applied for a topology design problem of a cantilever beam. The objective functions in this optimization problem are to minimize both weight and compliance. The results showed that the proposed methodology is capable of interfacing between topology multi-objective optimization (C program) and commercial finite element program (ANSYS program) in design of engineering problems successfully and efficiently.

Keywords: Interface methodology, topology multi-objective optimization, ANSYS Program, C Program

1. บทนำ

ในการออกแบบทางวิศวกรรมต่าง ๆ นั้นวิธีการ optimization ถือว่ามีความสำคัญเป็นอย่างมากในการออกแบบให้มีประสิทธิภาพสูงสุดตามวัตถุประสงค์ การ optimization เพื่อหารูปร่างที่เหมาะสมที่สุด (shape optimization) และการ optimization เพื่อหาการกระจายตัวของเนื้อวัสดุที่เหมาะสมที่สุด (topology optimization) ของโครงสร้างที่มีความต่อเนื่อง (continuum structures) ก็เป็นวิธีการอย่างหนึ่งในการทำ optimization ที่จะช่วยให้ผู้ออกแบบ ได้รูปร่างชิ้นงานที่เหมาะสมในขั้นตอน ซึ่งจะช่วยลดขั้นตอนในการลองผิดลองถูก (trial & errors) ในการออกแบบ ตลอดจนสามารถช่วยในการออกแบบสำหรับผู้ที่มีประสบการณ์ในการออกแบบน้อยได้

ในช่วงสิบกว่าปีที่ผ่านมา ได้มีงานวิจัยเกี่ยวกับ topology optimization ออกมาหลายวิธี อย่างเช่น Homogenization method [1,2], Evolutionary structural optimization [3,4] หรือ Continuum structural topology design with genetic algorithms [5,6] ซึ่งในปัจจุบันได้มีโปรแกรมไฟไนต์เอลิเมนต์ในเชิงการค้าบางตัว อย่างเช่น โปรแกรม ANSYS ก็มีฟังก์ชันในการทำ topology optimization แล้ว แต่โปรแกรม ANSYS ก็ยังมีความสามารถในการทำแค่ Single objective topology optimization และยังไม่มีความยืดหยุ่นในการเลือกวัตถุประสงค์ในการคำนวณสำหรับการทำ Topology optimization อย่างไรก็ตามในการทำ optimization ในการออกแบบบางครั้งก็ประกอบไปด้วยหลายวัตถุประสงค์ และในบางครั้งแต่ละวัตถุประสงค์ก็มีความขัดแย้งซึ่งกันและกัน (conflicting objectives) ซึ่งปัญหาเหล่านี้สามารถแก้ไขได้ด้วยการทำ multi-objective optimization ผลลัพธ์จากการทำ multi-objective optimization เรียกว่า Pareto optimal solutions จะประกอบไปด้วยคำตอบมากมาย ซึ่งแต่ละคำตอบจะไม่มีคำตอบใดที่ดีกว่ากันและกัน (non-dominated optimal solutions) ซึ่งข้อดีของการทำ multi-objective optimization คือ ช่วยให้ผู้ออกแบบสามารถทำการเปรียบเทียบข้อดี ข้อเสียจากรูปแบบต่างๆของการออกแบบจากคำตอบทั้งหมดของ Pareto optimal solutions อย่างไรก็ตามจากงานวิจัยต่างๆที่ผ่านมา โดยส่วนใหญ่ฟังก์ชันของวัตถุประสงค์ (objective function) มักจะเป็นฟังก์ชันทางคณิตศาสตร์ซึ่งสามารถคำนวณ หรือทำการ differentiate ได้โดยตรง แต่ถ้าหากฟังก์ชันของวัตถุประสงค์ต้องคำนวณจากปัญหาที่มีรูปร่างซับซ้อนขึ้น โดยเฉพาะอย่างยิ่งการนำไปประยุกต์ใช้กับการทำ topology optimization แล้ว มีความจำเป็นอย่างยิ่งที่จะต้องนำระเบียบวิธีไฟไนต์เอลิเมนต์ (finite element method) มาใช้มาช่วยในการคำนวณหาค่าฟังก์ชันของวัตถุประสงค์

จากบทนำดังกล่าวจะเห็นได้ว่า การทำ interface ระหว่างสองโปรแกรมดังกล่าวซึ่งต่างก็มีข้อดีในตัวเองให้สามารถติดต่อกันได้ จะช่วยทำให้การคำนวณ และการออกแบบงานทางด้านวิศวกรรมด้วยวิธี multi-objective topology optimization สามารถทำได้โดยสะดวกและมีประสิทธิภาพยิ่งขึ้นมาก และยิ่งกว่านั้นถึงแม้จะมีบทความต่างๆ ออกมามากมายได้กล่าวถึง ระเบียบวิธีการ topology optimization และ multi-objective optimization จะเห็นได้ว่าบทความทั้งหลายดังกล่าวไม่ได้กล่าวถึงวิธีการทำ interface ซึ่งถือว่าเป็นหัวใจสำคัญของการทำ multi-objective topology optimization เช่นกัน

ดังนั้นจุดประสงค์หลักของบทความนี้คือ เพื่อนำเสนอระเบียบวิธีการ interface ระหว่างโปรแกรมภาษา C และโปรแกรมไฟไนต์เอลิเมนต์ ANSYS อย่างง่ายโดยละเอียด เพื่อช่วยให้นักวิจัยที่สนใจในศาสตร์ด้านนี้ สามารถพัฒนาโปรแกรมขึ้นมาเองได้โดยง่าย และโปรแกรม interface ที่ได้พัฒนาขึ้นในงานวิจัยนี้ ได้นำไปทดลองใช้กับตัวอย่างของการ topology multi-objective optimization ในการออกแบบรูปร่างคาน ซึ่งมีฟังก์ชันของวัตถุประสงค์ในการออกแบบอยู่ 2 วัตถุประสงค์ คือ วัตถุประสงค์เพื่อออกแบบรูปร่างของคานให้ได้น้ำหนักน้อย และอีกวัตถุประสงค์การออกแบบให้มีการงอตัวที่น้อยในขณะเดียวกันด้วย และสุดท้ายผลลัพธ์ของการคำนวณที่ได้ได้แสดงให้เห็นว่าระเบียบวิธีการ interface ที่นำเสนอสามารถช่วยให้การ interface ระหว่าง multi-objective topology optimization (C program) กับโปรแกรมไฟไนต์เอลิเมนต์ในเชิงการค้า (ANSYS program) สามารถทำได้ ซึ่งส่งผลให้การทำ topology multi-objective optimization สามารถทำได้เป็นผลสำเร็จ

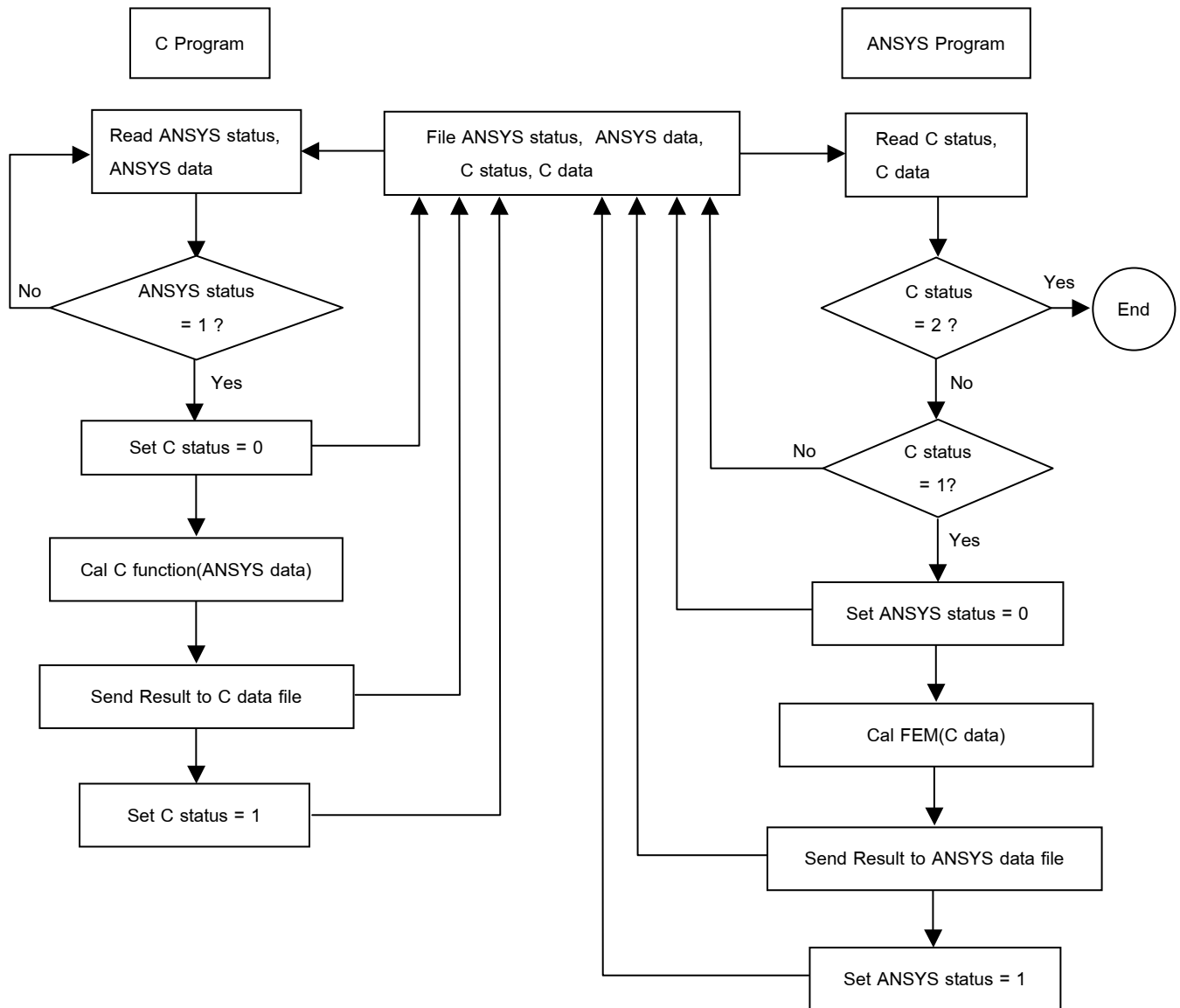
2. ระเบียบวิธีการ Interface ที่นำเสนอ

ในส่วนนี้จะแสดงถึงระเบียบวิธีการ interface ระหว่างโปรแกรมภาษา C และโปรแกรมไฟไนต์เอลิเมนต์ ANSYS อย่างละเอียด โดยหลักการของระเบียบวิธีที่นำเสนอนี้คือ อาศัยหลักการง่ายๆ ของการส่งและรับสถานะ และข้อมูลของแต่ละโปรแกรมผ่านไฟล์ โดยในการทำ topology multi-objective optimization จะมีขั้นตอนคร่าวๆ คือเมื่อโปรแกรม C คำนวณเสร็จ จะส่งข้อมูลของรูปร่างชิ้นงาน ไปให้ ANSYS คำนวณ และเมื่อโปรแกรม ANSYS คำนวณเสร็จ จะส่งกลับค่าของฟังก์ชันของวัตถุประสงค์ในการ optimize ไปให้โปรแกรมภาษา C เพื่อให้โปรแกรมภาษา C นำค่าฟังก์ชันของวัตถุประสงค์ที่ได้ไปทำการ optimize ต่อ โดยสถานะของแต่ละโปรแกรมที่กำหนดไว้ในโปรแกรม interface ในบทความนี้ จะแสดงดัง ตารางที่ 1 ส่วนผังงาน (flowchart) แสดงขั้นตอนของระเบียบวิธีที่นำเสนอนี้ได้แสดงไว้ดังรูปที่ 1

อธิบายผังงานแสดงขั้นตอนของระเบียบวิธีจากรูปที่ 1 ประกอบกับดูความหมายของสถานะของแต่ละโปรแกรมจากตารางที่ 1 ได้ว่าเมื่อเริ่มรันโปรแกรม โปรแกรม ANSYS จะหยุดพักชั่วคราวก่อน ในการหยุดพักชั่วคราวของแต่ละโปรแกรมในบทความนี้ จะหมายความว่ากรณีที่โปรแกรมหนึ่ง จะทำการวนลูปเพื่ออ่านค่าจากไฟล์สถานะที่ส่งมาจากอีกโปรแกรมหนึ่ง ในที่นี้โปรแกรม ANSYS จะเริ่มต้นด้วยการวนลูปเพื่ออ่านค่าสถานะจากโปรแกรม C (C status) โดยในขั้นแรกสถานะ

ตารางที่ 1 แสดงความหมายของ status ของแต่ละโปรแกรม

สถานะ	ความหมาย
0	โปรแกรมนั้นๆทำงานซึ่งจะส่งสถานะนี้ไปยังอีกโปรแกรมเพื่อให้อีกโปรแกรมที่ติดต่อกันอยู่หยุดพักชั่วคราว
1	โปรแกรมนั้นๆจะทำการหยุดพักชั่วคราว ซึ่งจะส่งสถานะนี้ไปยังอีกโปรแกรม เพื่อให้อีกโปรแกรมที่ติดต่อกันอยู่ทำงาน
2	สิ้นสุดการคำนวณของทุกโปรแกรม



รูปที่ 1 ระเบียบวิธีการ Interface ระหว่างโปรแกรมภาษา C และ โปรแกรมไฟไนต์เอลิเมนต์ ANSYS

ของโปรแกรม C จะเป็น 0 ก่อน นั่นคือโปรแกรม C จะเริ่มทำงานก่อน จนกระทั่งสถานะจากโปรแกรม C เป็น 1 ซึ่งหมายความว่าขณะนั้นทางฝั่ง C คำนวณเสร็จ และส่งข้อมูลที่คำนวณได้ใส่ไฟล์ C data เสร็จแล้ว โปรแกรม ANSYS จึงออกจากloop หลังจากนั้นโปรแกรม ANSYS จะตั้งค่าสถานะของโปรแกรม ANSYS (ANSYS status) ให้เป็น 0 เพื่อส่งสถานะแจ้งไปให้ฝั่งโปรแกรม C หยุดพักชั่วคราว แล้วโปรแกรม ANSYS จึงนำค่าข้อมูลที่ได้จากไฟล์ C data ไปทำการคำนวณ เมื่อทางฝั่งโปรแกรม ANSYS คำนวณเสร็จจะนำผลลัพธ์ข้อมูลที่คำนวณได้ใส่ไฟล์ ANSYS data แล้วตั้งค่าสถานะของโปรแกรม ANSYS ให้เป็น 1 เพื่อบอกให้โปรแกรม C รู้ว่าโปรแกรม ANSYS คำนวณเสร็จแล้ว และจะหยุดพักชั่วคราวจนกระทั่งโปรแกรม C ทำการคำนวณข้อมูลที่ได้จาก ANSYS data เสร็จเรียบร้อย แล้วส่งสถานะของโปรแกรม C ใส่ไฟล์มาเป็น 1 ทางด้าน ANSYS จึงคำนวณต่อโดยใช้ พารามิเตอร์จากไฟล์ C data

การ interface ระหว่างโปรแกรมทั้งสองจะดำเนินไปเรื่อยๆ จนกระทั่งโปรแกรมทั้งสองคำนวณถึงขั้นตอนสุดท้าย ในขั้นตอนสุดท้ายโปรแกรม C จะส่งค่าของโปรแกรม C เป็น 2 ไปให้โปรแกรม ANSYS เพื่อบอกให้โปรแกรม ANSYS รู้ว่าการคำนวณเสร็จสิ้นมาถึงขั้นตอนสุดท้าย การ interface จึงสิ้นสุด และจบการคำนวณทั้งหมด

นอกเหนือจากขั้นตอนดังกล่าวแล้ว ยังมีเทคนิคเพิ่มเติมที่ควรใส่ใจสำหรับวิธีการนี้คือ ต้องทำให้ไฟล์สถานะของทั้งสองโปรแกรมสามารถใช้ร่วมกันได้ในการอ่านและเขียนสถานะ เนื่องจากถ้าไฟล์สถานะไม่สามารถใช้ร่วมกันได้แล้ว จะมี error เกิดขึ้นเนื่องจากเกิดการขัดจังหวะซึ่งกันและกันระหว่างอ่านและเขียนไฟล์ ซึ่งจะแก้ปัญหาได้โดยโปรแกรมภาษา C จะมีคำสั่ง `_sopen` ซึ่งจะเปิดไฟล์ที่ระบุให้เป็นไฟล์ที่สามารถอ่าน หรือเขียนร่วมกันได้ ส่วนทางด้านโปรแกรม ANSYS ได้ใช้คำสั่ง `/NERR` ในการปิดข้อความเตือนเมื่อเปิดไฟล์ในขณะที่โปรแกรมภาษา C ใช้งานอยู่ ด้วยเทคนิคเช่นนี้

จะทำให้การรันโปรแกรม interface ด้วยวิธีการที่นำเสนอสามารถดำเนินไปได้อย่างต่อเนื่องไม่เกิดการขัดจังหวะ

3. Simulation method

3.1 Multi-objective optimization method using GA

โดยทั่วไปวิธีการ multi-objective optimization จะใช้ในกรณีที่มีวัตถุประสงค์ของการ optimize ให้พิจารณามากกว่าหนึ่งวัตถุประสงค์ และโดยส่วนใหญ่วัตถุประสงค์เหล่านั้นมักจะมีความขัดแย้งกันเอง ได้มีวิธีการดั้งเดิม (Classical method) หลายวิธีการในการแก้ปัญหาในลักษณะของ multi-objective optimization อย่างเช่น Weighted Sum Method [7] and the ϵ -constraint method [7] หากแต่วิธีการเหล่านั้นจะต้องมีการทำซ้ำ (iterated) หลายๆ ครั้งเพื่อให้ได้คำตอบแต่ละค่าที่แตกต่างกันของ Pareto-optimal solution นอกจากนี้วิธีการดั้งเดิมโดยส่วนใหญ่ จะไม่เหมาะที่จะนำมาแก้ปัญหาในกรณีที่ฟังก์ชันวัตถุประสงค์เป็น non-differentiable หรือในกรณีเป็นปัญหาที่มีความไม่ต่อเนื่องหรือเป็นปัญหาที่ให้คำตอบแบบ non-convex Pareto-optimal regions อย่างไรก็ตาม multi-objective optimization using genetic algorithm (GA) สามารถแก้ปัญหาดังกล่าวได้ดี

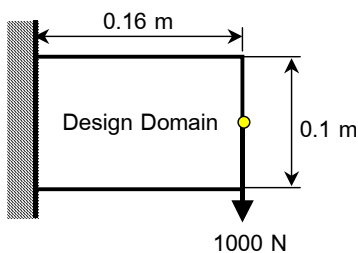
สำหรับบทความนี้ได้เลือกใช้ multi-objective optimization using genetic algorithms โดยวิธี the Elitist Non-Dominated Sorting Genetic Algorithm หรือ NSGA-II [7] ซึ่งคิดค้นขึ้นโดย K. Deb เนื่องจากเป็นวิธีการที่นำหลักการของ GA มาใช้ และสามารถให้คำตอบของ Pareto-optimal solution ที่มีลักษณะกระจายตัว ไม่เกาะกลุ่มกัน

3.2 Finite element method

โปรแกรมไฟไนต์เอลิเมนต์ที่เลือกใช้ในงานวิจัยชิ้นนี้ คือโปรแกรม ANSYS เนื่องจากเป็นโปรแกรมที่ค่อนข้างยืดหยุ่น เพราะตัวโปรแกรม ANSYS เองจะมีภาษาของตัวเองที่เรียกว่า APDL Program ซึ่งเปิดโอกาสให้สามารถเขียนคำสั่งเข้าไปช่วยในการคำนวณได้โดยการคำนวณที่ได้จากบทความนี้ได้ทดลองใช้กับโปรแกรม ANSYS เวอร์ชัน 8.1

4. Case study

โปรแกรม interface ที่ได้พัฒนาขึ้นในงานวิจัยนี้ ได้นำไปทดลองใช้กับตัวอย่างของปัญหาการ topology multi-objective optimization ในการ ออกแบบ short cantilever ที่มีอัตราส่วนความยาวต่อความกว้างเป็น 1.6:1 ซึ่งปัญหานี้เป็นปัญหาที่นิยมใช้ในการตรวจสอบความถูกต้องของโปรแกรม single objective topology optimization ที่ใช้กันอย่าง



รูปที่ 2 Short cantilever

กว้างขวาง สำหรับตัวอย่างปัญหาที่ได้ทดลองคำนวณ นี้เป็น plane stress มีความหนา 0.01 m วัสดุที่ใช้คำนวณมีค่า young modulus $E = 210$ GPa, Poisson ratio $\nu = 0.3$ มีขนาดสัดส่วน load และ boundary condition ดังแสดงในรูปที่ 2 และได้สร้างโมเดลไฟไนต์เอลิเมนต์โดยใช้ 4-node plane stress quadrilateral elements ที่มีขนาดความกว้างของแต่ละเอลิเมนต์ 5 mm เป็นจำนวน 640 เอลิเมนต์

4.1 Multi-objective optimization function

Minimize $F_1(x) = \text{compliance} \approx \text{maximum displacement}$

Minimize $F_2(x) = \text{weight}$ เนื่องจากให้ความหนาแน่นของวัสดุที่คำนวณคงที่ ดังนั้นน้ำหนักจึงพิจารณาเป็นปริมาตร หรือจำนวนของเอลิเมนต์ของวัตถุแทน

ฟังก์ชันวัตถุประสงค์ทั้งสองจะถูกคำนวณโดยโปรแกรมไฟไนต์เอลิเมนต์ โดย design variable x จะแสดงถึงการมีหรือไม่มีเนื้อวัสดุในโดเมน ดังแสดงในสมการที่ 1

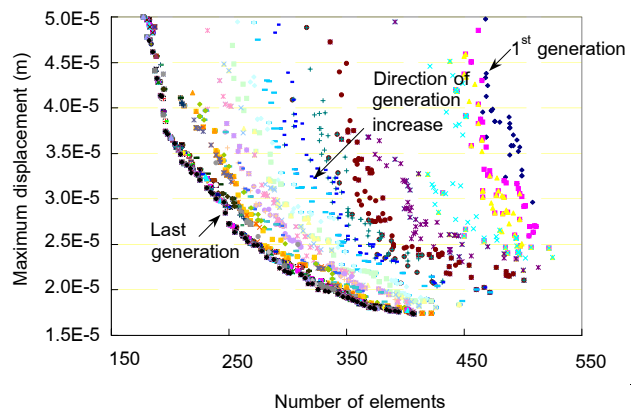
$$x = \begin{cases} 1 & \text{if material exist} \\ 0 & \text{if material doesn't exist} \end{cases} \quad (1)$$

4.2 GA parameters

บทความนี้ได้คำนวณ NSGA-II โดยใช้พารามิเตอร์ต่างๆ ดังแสดงในตารางที่ 2

ตารางที่ 2 พารามิเตอร์ต่างๆ ที่ใช้ในการ optimization

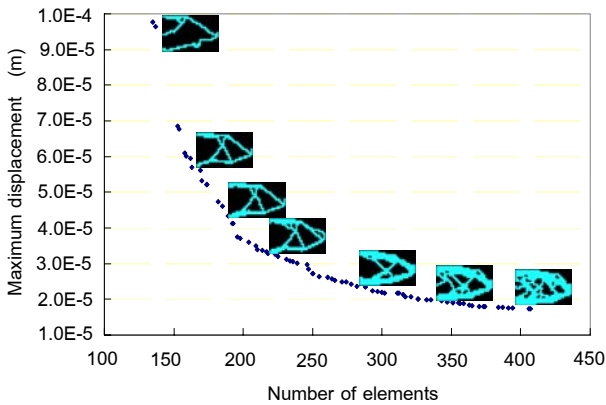
Parameters	Values
Encoding mechanism	Binary parameter (0,1)
Population size	100
Generation numbers	50
Crossover probability	0.8
Mutation probability	0.001



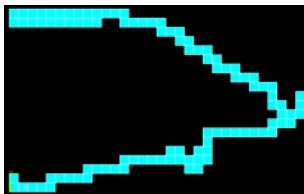
รูปที่ 3 แสดงผลการคำนวณของทุก generation

5. ผลการคำนวณ และวิเคราะห์

ผลการคำนวณของทุก generation แสดงไว้ดังรูปที่ 3 ส่วนรูปที่ 4 แสดงผลของ Pareto optimal solutions ของ generation สุดท้าย และรูปที่ 5(a-n) แสดงรูปร่างที่ได้จากการ optimize ที่ตำแหน่งต่างๆ ใน Pareto optimal solutions โดยสัญลักษณ์ Mo ที่แสดงในรูปที่ 5 คือสัดส่วนของปริมาตรของคำตอบ ต่อปริมาตรของ Design domain



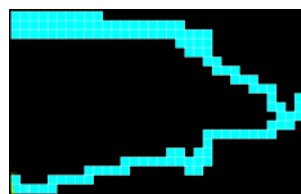
รูปที่ 4 ผลของ Pareto optimal solutions ของ generation สุดท้าย



(Mo = 0.16) 99 elements

$U_y = 0.000385$ m

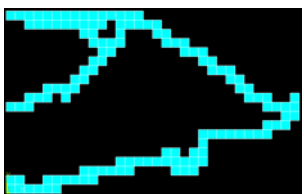
(5a)



(Mo = 0.19) 121 elements

$U_y = 0.000224$ m

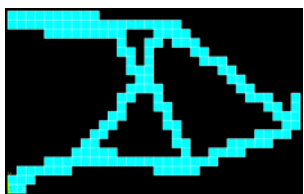
(5b)



(Mo = 0.21) 131 elements

$U_y = 0.000102$ m

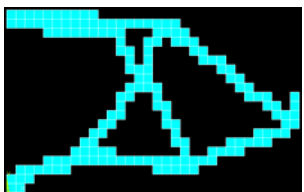
(5c)



(Mo = 0.25) 159 elements

$U_y = 0.0000602$ m

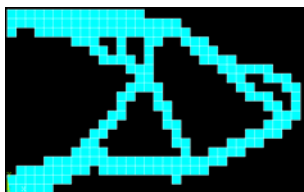
(5d)



(Mo = 0.28%) 182 elements

$U_y = 0.000047$ m

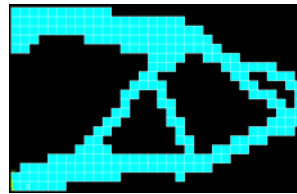
(5e)



(Mo = 0.32) 204 elements

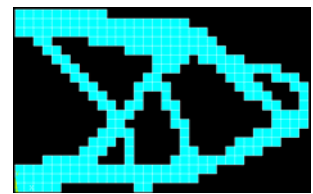
$U_y = .000036$ m

(5f)



(Mo = 0.35) 225 elements

$U_y = 0.000032$



(Mo = 0.39) 250 elements

$U_y = 0.000027$

(5g)

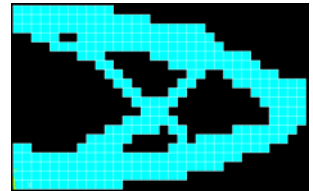
(5h)



(Mo = 0.43) 274 elements

$U_y = 0.000025$ m

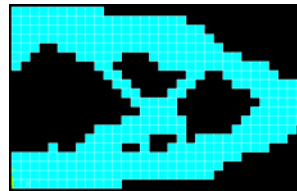
(5i)



(Mo = 0.46) 297 elements

$U_y = 0.000022$ m

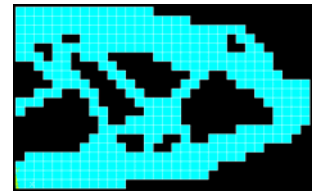
(5j)



(Mo = 0.51) 326 elements

$U_y = 0.00002$ m

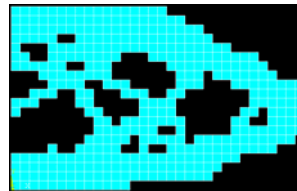
(5k)



(Mo = 0.55) 351 elements

$U_y = 0.000019$ m

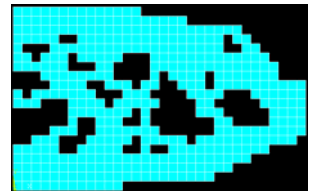
(5l)



(Mo = 0.58) 374 elements

$U_y = 0.000018$ m

(5m)

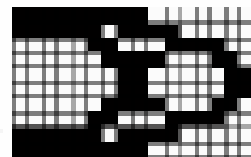


(Mo = 0.63) 406 elements

$U_y = 0.000017$ m

(5n)

รูปที่ 5(a-n) รูปร่างที่ได้จากการ optimize ที่ตำแหน่งต่างๆ ใน Pareto optimal solutions



Mo = 70/160 (0.44)

รูปที่ 6 รูปร่างที่ได้จากการ optimize ที่อัตราส่วนของน้ำหนัก 44% (Mo = 0.44) จากเอกสารอ้างอิง [8]

ซึ่งการแสดงผลในรูปที่ 4 และรูปที่ 5(a-n) นั้นส่วนที่เป็นเนื้อวัตถุจะแสดงโดยสีอ่อน และส่วนที่เป็นที่ว่างจะแสดงโดยสีเข้ม

เนื่องจากผลการคำนวณในรูปที่ 3 แสดงให้เห็นว่าเมื่อ generation เพิ่มขึ้น ค่าตอบจะมีแนวโน้มสู่ค่า minimize ของค่าวัตถุประสงค์มากขึ้น นั่นคือคำตอบเข้าใกล้จุด origin มากขึ้นเรื่อยๆ (จุด origin นี้เป็นจุดซึ่งทั้งสองวัตถุประสงค์มีค่าน้อยที่สุด แต่เนื่องจากวัตถุประสงค์ทั้งสองในการ optimize มีความขัดแย้งกัน ดังนั้นจุดนี้จึงเป็นจุด ideal ของคำตอบนั้น แสดงว่ายิ่งคำตอบเข้าใกล้ จุด ideal เท่าไหร่ คำตอบนั้นยิ่งเป็นคำตอบที่ดีกว่า) โดยคำตอบจะเริ่มเข้าสู่คำตอบสุดท้ายตั้งแต่ generation ที่ 25 และผลลัพธ์ก็ได้ถูกปรับปรุงให้ดีขึ้นเรื่อยๆ จนสู่คำตอบที่ดีที่สุดในการ generation สุดท้าย

ส่วน Pareto optimal solutions ที่ได้จากการ generation สุดท้ายนั้นได้แสดงไว้ในรูปที่ 4 โดยเมื่อพิจารณาจากรูปที่ 4 แล้ว จะเห็นว่าในการที่จะออกแบบคานให้ได้น้ำหนักที่เบาขึ้น จะต้องยอมให้มีการโก่งตัวของคานที่ออกแบบมาก ในทางกลับกันหากจะออกแบบคานให้ได้ stiffness สูงหรือมีการโก่งตัวน้อยขึ้น จะต้องยอมให้คานมีน้ำหนักมากขึ้น อย่างไรก็ตามรูปร่างที่ได้จากทุกจุดบน Pareto optimal solutions ในรูปที่ 4 เป็นรูปร่างที่ optimize แล้ว ดังนั้นผู้ออกแบบจึงสามารถเปรียบเทียบ และชั่งน้ำหนักในการตัดสินใจที่จะออกแบบได้

ส่วนรูปที่ 5(a-n) นั้นได้แสดงถึงรูปร่างที่ได้จากการ optimize ในตำแหน่งต่างๆ ของ Pareto optimal solutions ซึ่งได้แสดงตั้งแต่รูปร่าง optimize ที่น้ำหนักน้อย ไปจนถึงน้ำหนักมาก โดยแต่ละรูปร่างจะเป็นรูปแบบที่มี stiffness สูงสุด สำหรับการออกแบบนั้นๆ และเพื่อเป็นการตรวจสอบความถูกต้องของคำตอบ เมื่อเปรียบเทียบคำตอบกับรูปที่ 6 (ที่สัดส่วนของปริมาตรของคำตอบ $M_0 = 0.44$) ซึ่งเป็นคำตอบที่ได้จากวิธีการของเอกสารอ้างอิง [8] โดยการแสดงผลในรูปที่ 6 นั้นส่วนที่เป็นเนื้อวัตถุจะแสดงโดยสีเข้ม และส่วนที่เป็นที่ว่างจะแสดงโดยสีอ่อน พบว่ามีรูปร่างมีแนวโน้มที่ใกล้เคียงกับรูปที่ 5(i) (ที่สัดส่วนของปริมาตร ของคำตอบ $M_0 = 0.43$) ของคำตอบที่ได้จากการคำนวณจากบทความนี้ จึงเป็นการยืนยันความถูกต้องของโปรแกรมจากการคำนวณต่างๆ ดังแสดงในหัวข้อนี้ เป็นการแสดงให้เห็นว่า ระเบียบวิธีการ Interface ที่นำเสนอในบทความนี้สามารถช่วยให้โปรแกรม multi-objective topology optimization ที่เขียนไว้ด้วยโปรแกรมภาษา C กับโปรแกรมไฟไนต์เอลิเมนต์ในเชิงการค้า ANSYS ทำการติดต่อกันได้ ส่งผลให้สามารถนำไปใช้ในการทำ topology multi-objective optimization ได้เป็นผลสำเร็จอย่างมีประสิทธิภาพ

6. สรุป

บทความนี้ได้นำเสนอระเบียบวิธีการ Interface ระหว่างโปรแกรมภาษา C และโปรแกรมไฟไนต์เอลิเมนต์ ANSYS เพื่อการทำ topology multi-objective optimization โดยอาศัยหลักการง่ายๆ ของการส่ง และรับสถานะ และข้อมูลของแต่ละโปรแกรมผ่านไฟล์ โดยโปรแกรมที่ได้จากระเบียบวิธีที่นำเสนอนี้ ได้นำไปทดลองใช้กับตัวอย่างของปัญหาการทำ topology multi-objective optimization ในการออกแบบรูปร่างคานได้เป็นผลสำเร็จ ตลอดจนได้มีการตรวจสอบผลลัพธ์ที่ได้จากบทความนี้กับเอกสารอ้างอิงแล้วพบว่ามีความถูกต้องด้วย

กิตติกรรมประกาศ

บทความนี้ ได้รับทุนอุดหนุนจากสำนักงานคณะกรรมการการอุดมศึกษา และสำนักงานกองทุนสนับสนุนการวิจัย ความเห็นในรายงานผลการวิจัยเป็นของผู้วิจัย สำนักงานคณะกรรมการการอุดมศึกษา และสำนักงานกองทุนสนับสนุนการวิจัยไม่จำเป็นต้องเห็นด้วยเสมอไป

เอกสารอ้างอิง

- [1] M.P. Bendso, and N. Kikuchi, "Generating optimization topologies in structural design using a homogenization method," Comput. Methods Appl. Mech. Engrg., Vol. 71, 1988, pp. 197-244.
- [2] M.P. Bendsoe, and O. Sigmund, Topology optimization, Springer, Berlin, Germany, 2004.
- [3] D. N. Chu, Y.M. Xie, A. Hira, and G.P. Steven, "Evolutionary structural optimization for problems with stiffness constraints," Finite element in analysis and design, Vol. 21, 1996, pp.239-251.
- [4] X.Y. Yang, Y.M. Xie and, G.P. Steven, "Evolutionary methods for topology optimisation of continuous structures with design dependent loads", Computers and Structures, Vol. 83, 2005 pp. 956-963.
- [5] M. J. Jakiela, C. Chapman, J. Duda, A. Adewuya, and K. Saitou, "Continuum structural topology design with genetic algorithms", Comput. Methods Appl. Mech. Engrg., Vol. 186, 2000, pp. 339-356.
- [6] J.F.A. Madeira, H. Rodrigues, and H. Pina, "Multi-objective optimization of structures topology by genetic algorithms," Advances in engineering software, Vol. 36, 2005, pp. 21-28.
- [7] K. Deb, Multi-objective optimization using evolutionary algorithms, John Wiley & Sons, England, 2001.
- [8] I. Y. Kim, and O. D. Weck, "Variable chromosome length genetic algorithm for structural topology design optimization", 45th AIAA/ASME/ASCE/ASH/ASC Structures, structural dynamics & materials conference, Palm Springs, California, 19-22 April 2004. pp. 1-12.