

บทที่ 7

สรุปและข้อเสนอแนะ

วิทยานิพนธ์ฉบับนี้ได้แสดงให้เห็นถึงวิธีการการออกแบบภาษาด้านฉบับ,และรหัสกลางที่สนับสนุนการประมวลผลแบบพร้อมกัน แสดงขั้นตอนและวิธีการพัฒนาคอมไพเลอร์เพื่อแปลภาษาด้านฉบับไปเป็นรหัสกลาง และแสดงขั้นตอนและวิธีการพัฒนาอินเตอร์พรีเตอร์ เพื่อประมวลผลรหัสกลางเพื่อให้ประมวลผลพร้อมกันโดยละเอียด ซึ่งโปรแกรมที่ได้คือ

	คอมไพเลอร์	อินเตอร์พรีเตอร์
ขนาดของโปรแกรมต้นฉบับ	2,984 บรรทัด (69,792 ไบต์)	968 บรรทัด (21,047 ไบต์)
ขนาดของแฟ้ม .EXE	60,672 ไบต์	25,520 ไบต์
ความเร็วในการทำงาน	115 บรรทัดต่อวินาที	26,747 คำสั่งต่อวินาที
เวลาที่ใช้ในการจัดลำดับงาน	-	5.49.% ของเวลารวม
ความเร็วในการสลับกระบวนการ	-	3.4×10^{-6} วินาที

ภาษาที่ออกแบบซึ่งจะมีลักษณะคล้ายกับภาษาซี แต่จะมีคำสั่งที่จำเป็นสำหรับประมวลผลพร้อมกันเพิ่มเติม เมื่อแปลโปรแกรมต้นฉบับแล้วโปรแกรมส่วนใหญ่จะได้รหัสกลางที่มีขนาดเล็กกว่าโปรแกรมต้นฉบับโดยเฉลี่ย 1.7 เท่า เป็นเพราะส่วนของรหัสกลางที่ใช้แทนที่คำสั่งของภาษาด้านแบบ มีขนาดเพียงหนึ่งไบต์

File name	จำนวน โทเคน	เวลาใน การแปล (sec)	ความเร็ว (token/sec)	ขนาด โปรแกรม ต้นฉบับ	ขนาด รหัสกลาง (byte)
fac.x	99	0.16	618.75	310	196
fib.x	71	0.11	645.46	214	167
hanoi.x	402	1.10	365.45	1699	2631
sort.x	222	0.39	569.23	779	487
sum.x	168	0.22	763.64	484	195
t_gr.x	1246	1.98	629.29	5231	1766
t.x	1436	2.42	593.39	4981	2717

เมื่อนำรหัสกลางที่ได้ไปประมวลผลด้วยอินเตอร์พรีเตอร์ เราพบว่ากระบวนการต่างๆ สามารถประมวลผลพร้อมกันได้ ซึ่งได้แสดงให้เห็นในโปรแกรมปัญหาการขนส่ง จะพบว่ากระบวนการของหุ่นยนต์ทุกตัวสามารถเคลื่อนที่ไปได้พร้อมๆกันเมื่อควอนตัมมีค่าต่ำ แต่เมื่อเพิ่มค่าควอนตัมมากขึ้นหุ่นยนต์จะสลับกันเคลื่อนที่ทีละตัว ดังนั้นเมื่อค่าควอนตัมต่ำโปรแกรมจะประมวลผลพร้อมกันได้ดีกว่าโปรแกรมที่ใช้ค่าควอนตัมมาก เมื่อเทียบความเร็วในการประมวลผลจากผลการทดลองปัญหาการขนส่ง พบว่าโปรแกรมที่มีค่าควอนตัมต่ำจะประมวลผลได้ช้าทั้งนี้เป็นเพราะต้องใช้เวลาในการจัดลำดับกระบวนการมาก คือใช้เวลาในการจัดลำดับกระบวนการประมาณร้อยละ 5.5 ของเวลาในการประมวลผลทั้งหมด

อินเตอร์พรีเตอร์ที่ได้จะใช้เวลาในการสลับกระบวนการที่จะประมวลผลน้อยเมื่อเทียบกับความเร็วในการประมวลผลหนึ่งคำสั่ง คือใช้เวลาประมาณร้อยละ 5.5 ทั้งนี้เป็นเพราะโครงสร้างข้อมูลของกระบวนการที่เป็นแบบลิสต์วงกลม เมื่อต้องการเปลี่ยนกระบวนการก็ทำได้โดยเลื่อนตัวชี้ไปที่กระบวนการถัดไปทำให้สามารถเปลี่ยนกระบวนการได้เร็ว

ในส่วนของโปรแกรมกระทำการ (.EXE) ที่ได้โปรแกรมคอมพิวเตอร์จะมีขนาด 60 กิโลไบต์ และอินเตอร์พรีเตอร์มีขนาด 25 กิโลไบต์ แต่เนื่องจากในขั้นตอนการทำงานจะคอม

ไฟล์ภาษาต้นฉบับบนเครื่องคอมพิวเตอร์ส่วนบุคคลคั่งนั้นเพิ่มขนาด 60 กิโลไบต์จึงไม่เป็นอุปสรรคในการทำงาน แต่โปรแกรมอินเตอร์พรีเตอร์ที่ได้มีขนาด 25 กิโลไบต์นี้หากนำไปพัฒนาบนอุปกรณ์ที่ต่างกันเช่น ไมโครคอนโทรลเลอร์ควรจะปรับปรุงให้มีขนาดเล็กลงเพื่อให้สามารถทำงานบนอุปกรณ์ดังกล่าวได้

ปัญหาที่เกิดขึ้นในการวิจัย

1. ในขั้นตอนคอมไพล์ หากเกิดความผิดพลาดขึ้นในโปรแกรมต้นแบบ การระบุความผิดพลาดมักทำได้ไม่ถูกต้อง ทั้งนี้เป็นเพราะ เมื่อเกิดความผิดพลาดเกิดขึ้น คอมไพเลอร์จะพยายามแปลภาษาต้นแบบตามไวยากรณ์อื่นๆต่อไป จนแปลด้วยไวยากรณ์ใดๆไม่ได้แล้ว และจะแจ้งความผิดพลาดหลังจากจุดที่เกิดความผิดพลาดที่แท้จริง

2. ในระหว่างการพัฒนาโปรแกรม ซึ่งจำเป็นต้องทดลองการทำงานโดยเขียนโปรแกรมต้นแบบเพื่อทดสอบการทำงาน แต่เนื่องจากการแปลมีส่วนเกี่ยวข้องกับสองขั้นตอนคือโปรแกรมคอมไพเลอร์ และโปรแกรมอินเตอร์พรีเตอร์ หากเขียนโปรแกรมส่วนใดส่วนหนึ่งผิดพลาด การตรวจหาที่ผิดทำได้ยาก ทั้งนี้เพราะคาดเดาได้ยากว่าเกิดความผิดพลาดในขั้นตอนใด

3. ในการวัดประสิทธิภาพในการประมวลผล จำเป็นต้องจับเวลาในการประมวลผล โดยใช้ฟังก์ชันการจับเวลาของคอส ซึ่งพบว่าความละเอียดเพียง 54.936 มิลลิวินาที ซึ่งเทียบเท่ากับการประมวลผลรหัสกลาง 892 คำสั่ง ดังนั้นหากเราจับเวลาโปรแกรมที่เล็กมากๆ เช่น จับเวลาในการแปลรหัสกลางหนึ่งคำสั่ง หรือเวลาในการจัดลำดับกระบวนการ หากแทรกคำสั่งในการจับเวลาทุกๆคำสั่ง ความผิดพลาดของการจับเวลาจะมีค่าสูง แนวทางในการแก้ไขคือจะใช้การจับเวลาโปรแกรมที่มีความยาวมากขึ้นและหาค่าเฉลี่ยเวลาที่ใช้ต่อจำนวนคำสั่งที่ประมวลผลในภายหลัง

ในส่วนของการหาเวลาในการจัดลำดับกระบวนการที่มีควอนตัมแตกต่างกัน ก็ทำได้โดยหาเวลารวมที่ใช้ในการประมวลผลโปรแกรมที่มีค่าควอนตัมแตกต่างกันหลายๆ และคูณผลต่างของเวลาที่ใช้ในการประมวลผล เมื่อผลต่างของเวลามาเปรียบเทียบกับจำนวนครั้งที่สลับกระบวนการ จะได้เวลาที่ใช้ในการสลับกระบวนการหนึ่งครั้ง

ข้อเสนอแนะ

1. เนื่องจากรหัสกลางที่ได้นี้ไม่ผูกพันกับฮาร์ดแวร์ ในวิทยานิพนธ์ฉบับนี้ได้แสดงให้เห็นถึงการพัฒนาอินเตอร์พรีเตอร์บนเครื่องคอมพิวเตอร์ส่วนบุคคลเท่านั้นแต่ยังมีได้ทำการทดลองนำเอารหัสกลางไปประมวลผลบนอุปกรณ์เช่น ไมโครคอมพิวเตอร์ชิบเดี่ยวที่มีขนาดเล็ก และมีราคาถูก ซึ่งเหมาะกับการไปใช้ในงานควบคุมมากกว่า ดังนั้นหากนำไปพัฒนาบนไมโครโปรเซสเซอร์อื่นๆ เพื่อที่จะแสดงว่ารหัสกลางที่สร้างขึ้นไม่ขึ้นกับฮาร์ดแวร์ใด โดยเฉพาะ

2. เนื่องจากเป็นอินเตอร์พรีเตอร์ ที่เขียนขึ้นโดยภาษาปาสคาล จึงมีความเร็วต่ำ เมื่อเปรียบเทียบกับ การไปเป็นแปลภาษาเป้าหมายที่เป็นภาษาปาลคาลโดยตรง เราพบว่ามีความเร็วสูงขึ้นถึงสามเท่า หากต้องการประมวลผลด้วยความเร็วสูงขึ้นอาจทำได้โดยการแปลจากภาษาต้นแบบไปเป็นภาษาเป้าหมายเช่นภาษาแอสเซมบลีโดยตรงน่าจะมีความเร็วสูงขึ้น