

บทที่ 5

บทสรุป

5.1 สรุปผลการวิจัย

การออกแบบวงจรกรองอนาล็อกโดยใช้เงินเนติกอัลกอริทึมสามารถนำมาใช้ในการออกแบบวงจรกรองอนาล็อกได้ดี โดยสามารถนำตัวต้านทาน ตัวเก็บประจุ และตัวเหนี่ยวนำที่มีขายตามท้องตลาดทั่วไป มาใช้ในการออกแบบวงจรกรองอนาล็อกได้โดยตรง ซึ่งจะช่วยลดความผิดพลาดจากการเลือกใช้อุปกรณ์จริงที่มีค่าใกล้เคียงกับค่าที่ได้จากวิธีการประมาณแบบแบตเตอรี่เวอร์ต นอกจากนี้ยังมีการออกแบบวงจรให้มีตำแหน่งจุดศูนย์และจุดขั้ว ในช่วงความถี่ต่างๆ โดยกำหนดให้จุดศูนย์จะสามารถเกิดได้ในเฉพาะช่วงความถี่ที่ยอมให้สัญญาณผ่านได้ และจุดขั้วจะสามารถเกิดได้ในช่วงความถี่ที่ไม่ยอมให้สัญญาณผ่านได้ และ/หรือ ช่วงความถี่ที่มีการเปลี่ยนแปลงระดับสัญญาณ ซึ่งการออกแบบให้มีการสร้างจุดขั้วขึ้นจะมีผลช่วยทำให้ระดับแรงดันของสัญญาณด้านขาออกของวงจรตกลงต่ำเร็วกว่าเดิมมาก จึงทำให้การออกแบบวงจรโดยใช้เงินเนติกอัลกอริทึมใช้จำนวนอุปกรณ์ที่เชื่อมต่อเป็นวงจรน้อยกว่าการประมาณค่าแบบแบตเตอรี่เวอร์ต

การออกแบบวงจรกรองอนาล็อกโดยใช้เงินเนติกอัลกอริทึมนี้ จะใช้จำนวนอุปกรณ์ที่น้อยกว่าการประมาณค่าแบบแบตเตอรี่เวอร์ตมาก เมื่อนำมาใช้ออกแบบวงจรกรองอนาล็อกที่มีระยะช่วงความถี่ที่มีการเปลี่ยนแปลงระดับสัญญาณแคบมาก การออกแบบที่มีการกำหนดอัตราการลดทอนของสัญญาณในช่วงความถี่ที่ไม่ยอมให้สัญญาณผ่านได้ (K_p) สูง และการออกแบบขึ้นพื้นฐานทั่วไป แต่ในบางกรณีเช่น การออกแบบวงจรกรองอนาล็อกแบบความถี่ผ่านแถบและวงจรกรองอนาล็อกแบบความถี่ก้ำกั๊วแถบ ที่กำหนดอัตราการลดทอนของสัญญาณในช่วงความถี่ที่ยอมให้สัญญาณผ่านได้ (K_p) มีค่าต่ำมากการค้นหาโดยใช้วิธีเงินเนติกอัลกอริทึมจะทำได้ยาก โดยสังเกตได้จากวงจรที่ออกแบบได้ใช้จำนวนอุปกรณ์ในการเชื่อมต่อมากกว่าวิธีแบตเตอรี่เวอร์ต และใช้เวลาในการประมวลผลนานมาก (นานกว่า 2 ชั่วโมงสำหรับเครื่องคอมพิวเตอร์ที่มีความเร็วเท่ากับเครื่องที่ใช้ในงานวิจัย) ทั้งนี้มีสาเหตุมาจากการออกแบบวงจรกรองอนาล็อกโดยใช้เงินเนติกอัลกอริทึมนี้ เป็นการหาค่า ชนิด และโครงสร้างการเชื่อมต่อของอุปกรณ์แบบสุ่ม ซึ่งจะทำให้เกิดการเชื่อมต่ออุปกรณ์ที่ทำให้ได้ลักษณะของแรงดันด้านขาออกแบบไม่สมมาตร (Non-Symmetry) จึงทำให้ระดับแรงดันด้านขาออกของวงจรในช่วงความถี่ที่ยอมให้สัญญาณผ่านได้ไม่หักล้างกัน ทำให้มีการกระเพื่อมของระดับแรงดันมากกว่าการประมาณค่าแบบแบตเตอรี่เวอร์ต ซึ่งสามารถ

แก้ไขได้โดยการเพิ่มจำนวนประชากรให้มากขึ้น (เป็น 1,000 ตัว) ซึ่งจะช่วยให้เงินเนติกอัลกอริทึมสามารถค้นหาคำตอบได้ดีขึ้น เนื่องจากประชากรจะมีความหลากหลายมากขึ้น จึงสามารถพบวงจรที่มีจำนวนอันดับต่ำกว่าวิธีการประมาณแบบเบ็คเตอร์เวอร์ตในที่สุด ซึ่งการใช้จำนวนประชากรมากสามารถใช้กับการออกแบบที่มีข้อกำหนดง่ายได้เช่นกัน แต่จำทำให้ใช้เวลาในการประมวลผลในแต่ละรอบการทำงานนานขึ้นด้วย

การทดสอบการประมาณค่าแบบเบ็คเตอร์เวอร์ตในงานวิจัยนี้ ไม่ได้ประมาณค่าของอุปกรณ์ให้เป็นค่าของอุปกรณ์ที่มีขายอยู่ในท้องตลาด โดยจะใช้ค่าของอุปกรณ์แต่ละตัวจะเป็นเลขจำนวนจริงทศนิยม 13 ตำแหน่ง ดังนั้นหากนำอุปกรณ์เหล่านี้มาประมาณค่าให้เป็นค่าที่มีขายอยู่จริง จะทำให้ระดับแรงดันของสัญญาณด้านขาออกของวงจรไม่ตรงกับข้อกำหนดที่ออกแบบไว้อย่างแท้จริงได้

5.2 ปัญหาและวิธีการแก้ไขปัญหา

ปัญหาที่พบและมีความสำคัญมากต่อการวิจัยเกี่ยวกับการออกแบบวงจรกรองอนาล็อกโดยใช้เงินเนติกอัลกอริทึมมีดังต่อไปนี้

ก) การเข้ารหัสของโครโมโซมจำเป็นที่จะต้องครอบคลุมถึงคำตอบที่เป็นไปได้ทั้งหมด ซึ่งจะช่วยให้เงินเนติกอัลกอริทึมสามารถค้นหาคำตอบที่ดีที่สุดได้ นอกจากนี้การเข้ารหัสและถอดรหัสควรมีความสัมพันธ์กันแบบ 1 ต่อ 1 นั่นคือ โครโมโซมหนึ่งควรจะต้องถอดรหัสเป็นคำตอบได้เพียงหนึ่งคำตอบเท่านั้น ในขณะที่เดียวกันคำตอบหนึ่งคำตอบก็ควรจะเข้ารหัสได้เป็นโครโมโซม 1 แบบเท่านั้น ซึ่งจากการวิจัยพบว่า เกิดปัญหาการเข้ารหัสของอุปกรณ์ย่อย (Sub Component) ในส่วนที่เป็นการเชื่อมต่อแบบไม่มีอุปกรณ์ใดต่ออยู่ในตำแหน่งนั้น ซึ่งแทนด้วยรหัส 0 ในตำแหน่งหลักหนึ่งของแถวแรก โดยพบว่า หากมีการเกิดรหัส 0 ติดกัน 2 ตำแหน่ง จะทำให้ถอดรหัสได้คำตอบเดียวกับโครโมโซมที่ไม่มีตำแหน่งรหัส 0 อยู่เลย ซึ่งนับเป็นความซ้ำซ้อนของโครโมโซมที่อาจจะเกิดขึ้นได้ โดยแสดงโครโมโซมสองตัวที่สามารถถอดรหัสได้คำตอบเดียวกันดังรูปที่ 5.1

-1	0	0	354	-2		-1	354	-2
579	0	0	0	60		579	0	602
205	0	0	0	279	↔	205	0	279
0	23	0	0	0		0	23	0

รูปที่ 5.1 ตัวอย่างการเกิดโครโมโซมที่มีความซ้ำซ้อนกัน

เพื่อเป็นการแก้ปัญหาการเกิดรหัส 0 ขึ้นในตำแหน่งหลักติดกัน จึงกำหนดให้โปรแกรมตรวจเช็คค่ารหัส 0 เฉพาะในแถวแรกของโครโมโซมด้วย โดยในส่วนของ การสุ่มประชากรเริ่มต้น จะต้องตรวจเช็คก่อนว่า หากค่าที่สุ่มได้ในตำแหน่ง b ของแถวแรกนั้นมีค่าเป็นรหัส 0 แล้ว จะต้องตรวจเช็ครหัสหลัก $b-1$ และหลัก $b+1$ ด้วยว่ามีค่าเป็นรหัส 0 หรือไม่ ถ้าหากมีรหัส 0 อยู่แล้ว จะต้องสุ่มรหัสในหลัก b ของแถวแรกใหม่จนกว่าจะได้รหัสที่ไม่เป็น 0 และนอกจากนี้ยังจะต้องป้องกันความผิดพลาดในการตรวจเช็ครหัสของหลักด้านข้างที่ไม่มีค่าด้วย เช่นในตำแหน่งหลักที่ b เท่ากับ 1 จะไม่มีรหัสในหลักที่ $b-1$ ซึ่งคือหลักที่ 0 และในตำแหน่งหลักที่ b เท่ากับหลักสุดท้ายก็จะมีรหัสในตำแหน่งหลักที่ $b+1$ เช่นกัน

ในส่วนของ การครอสโอเวอร์ ก็มีการป้องกันการเกิดรหัส 0 ติดกันของโครโมโซมรุ่นลูกที่จะเกิดขึ้นเช่นกัน โดยกำหนดให้ถ้าตำแหน่งที่ถูกเลือกมาทำการครอสโอเวอร์มีรหัสเป็น 0 แล้วจะต้องตรวจเช็คก่อนว่าตำแหน่งหลักข้างๆที่จะนำรหัส 0 นี้ไปแทนที่มีรหัสเป็น 0 อยู่ก่อนแล้วหรือไม่ ซึ่งถ้าหากตำแหน่งหลักด้านข้างของตำแหน่งที่จะไปแทนที่มีรหัส 0 อยู่ จะต้องทำการวนสุ่มตำแหน่งที่จะไปแทนที่ใหม่จนกว่าจะได้ตำแหน่งที่มีรหัสของหลักด้านข้างไม่เป็น 0

ในส่วนของ การมิวเตชัน จะใช้วิธีการตรวจเช็คว่า ถ้ารหัสของตำแหน่งที่ถูกเปลี่ยนค่าถูกสุ่มเลือกออกมาได้เป็นรหัส 0 จะต้องตรวจเช็คด้วยว่า ตำแหน่งรหัสของหลักข้างๆนี้มีรหัส 0 อยู่หรือไม่ ซึ่งหากมีรหัส 0 อยู่ในตำแหน่งหลักข้างๆอยู่ก่อนแล้วจะต้องสุ่มหารหัสในตำแหน่งหลักนี้ใหม่จนกว่าจะได้รหัสที่เป็น 0

ข) ฟังก์ชันการหาค่าความผิดพลาดนับเป็นส่วนสำคัญที่สุดในการค้นหาคำตอบโดยใช้เงินเนติกอัลกอริทึม ซึ่งหากมีการกำหนดฟังก์ชันในการหาค่าความผิดพลาดที่ไม่ดีพอแล้ว จะทำให้การค้นหานั้นไปในแนวทางที่ผิด นั่นคือ ลักษณะของโครโมโซมที่เป็นคำตอบของประชากรจะลู่เข้าไปสู่จุดที่มีค่าความเหมาะสมที่สุดเฉพาะแห่ง (Local Solution) เท่านั้น แต่จะไม่สามารถหาค่าความเหมาะสมที่สุดอย่างแท้จริงได้ (Global Solution) ซึ่งในการออกแบบวงจรกรองอนาล็อกโดยใช้เงินเนติกอัลกอริทึมในงานวิจัยนี้ จะแบ่งฟังก์ชันการหาค่าความผิดพลาดออกเป็น 2 ส่วน คือ ฟังก์ชันค่าความผิดพลาดในช่วงความถี่ที่ยอมให้สัญญาณผ่านได้ และฟังก์ชันค่าความผิดพลาดในช่วงความถี่ที่ไม่ยอมให้สัญญาณผ่านได้ โดยจะไม่มีพิจารณาความผิดพลาดในช่วงความถี่ที่มีการเปลี่ยนแปลงระดับของสัญญาณ เนื่องจากในช่วงความถี่ดังกล่าวนี้ นับเป็นช่วงที่ไม่มีการกำหนดค่าระดับแรงดันของสัญญาณไว้ว่าควรจะเป็นอย่างไร

ปัญหาที่พบคือ การกำหนดน้ำหนักความสำคัญของค่าความผิดพลาดในช่วงความถี่ทั้งสองที่พิจารณา โดยจะต้องลดความสำคัญของการหาค่าความผิดพลาดในช่วงความถี่ที่ไม่ยอมให้สัญญาณผ่านได้ลง โดยการหารด้วยอัตราการลดทอนของสัญญาณในช่วงความถี่นี้ (K_s) ทั้งนี้

เนื่องจากหากไม่นำค่าความเหมาะสม ณ จุดความถี่ต่างๆในช่วงความถี่ที่ไม่ยอมให้สัญญาณผ่านได้นี้ มาหารด้วยค่า K_s ก่อนแล้ว จะทำให้การค้นหาของเจนเนติกอัลกอริทึม มุ่งไปที่การค้นหาให้วงจรมีระดับแรงดันของสัญญาณด้านขาออกตกลงต่ำกว่า $-K_s$ เดซิเบลในช่วงความถี่ที่ไม่ยอมให้สัญญาณผ่านได้แต่เพียงอย่างเดียว โดยโปรแกรมจะไม่สนใจค่าความเหมาะสมในส่วนในช่วงความถี่ที่ยอมให้สัญญาณผ่านได้เลย เนื่องจากค่าของความเหมาะสมที่เกิดขึ้นในช่วงความถี่ที่ยอมให้สัญญาณผ่านได้นั้นมีค่าน้อยมากเมื่อเทียบกับค่าความเหมาะสมที่เกิดขึ้นในช่วงความถี่ที่ไม่ยอมให้สัญญาณผ่านได้นั่นเอง

ค) การกำหนดค่าพารามิเตอร์ต่างๆ มีส่วนสำคัญต่อโอกาสการค้นพบโครโมโซมที่เป็นคำตอบที่ดีที่สุด และเวลาในการประมวลผลเช่นกัน โดยค่าพารามิเตอร์ที่สำคัญเหล่านี้ได้แก่ จำนวนประชากร (pop_size) ซึ่งหากกำหนดค่าไว้ต่ำเกินไปแล้วจะทำให้ไม่สามารถหาโครโมโซมที่เป็นคำตอบที่ดีที่สุดได้ เนื่องจากเจนเนติกอัลกอริทึมจะกลายเป็นเพียงการค้นหาคำตอบแบบเฉพาะแห่งเท่านั้น แต่หากกำหนดจำนวนประชากรไว้มากจนเกินไปจะทำให้ใช้เวลาในการประมวลผลมาก

อัตราการมิวเตชัน (p_m) และอัตราการครอสโอเวอร์ (p_c) เป็นตัวกำหนดการทำงานของ การมิวเตชันและครอสโอเวอร์ ซึ่งหากกำหนดค่าไว้ต่ำเกินไป จะทำให้ประชากรไม่เกิดโอกาสที่จะได้แลกเปลี่ยนยีนส์ที่ดีต่อกัน (จากการครอสโอเวอร์) และจะไม่เกิดโครโมโซมที่มีลักษณะแปลกใหม่ขึ้น (จากการมิวเตชัน) ซึ่งจะทำให้การวิวัฒนาการของประชากรเป็นไปได้ช้ามาก แต่หากกำหนดอัตราการมิวเตชันและครอสโอเวอร์ไว้สูงเกินไป จะทำให้ใช้เวลาในการประมวลผลมากเช่นกัน โดยอัตราการมิวเตชันที่ใช้ในงานวิจัยนี้จะกำหนดให้มีค่าต่ำในขณะเริ่มต้นการทำงาน และสามารถปรับให้มีค่าสูงขึ้นภายหลังได้หากโปรแกรมไม่สามารถพบโครโมโซมใหม่ที่มีค่าความผิดพลาดที่ต่ำกว่าเดิมได้ภายใน 2 รอบการทำงาน

จำนวนจุดต่อเดเคด (n_{dec}) ความถี่ที่พิจารณาต่ำสุด (f_{min}) และความถี่ที่พิจารณาสูงสุด (f_{max}) จะมีผลต่อเวลาในการประมวลผลโดยใช้โปรแกรม PSPICE และการทำงานของเจนเนติกอัลกอริทึมมาก เนื่องจากหากจำนวนจุดต่อเดเคดและช่วงความถี่ที่พิจารณามีค่าสูงเกินไปจะทำให้การประมวลผลเป็นไปได้ช้ามาก แต่หากจำนวนจุดต่อเดเคดต่ำเกินไป โปรแกรมอาจไม่สามารถมองเห็นจุดสั้นพ้องอย่างแรง (Peak) ที่อาจเกิดขึ้นช่วงความถี่ที่แคบมากๆ ได้ ในทำนองเดียวกัน หากช่วงความถี่ที่พิจารณาไม่กว้างพอที่จะครอบคลุมความถี่ในการใช้งานทั้งหมด อาจเกิดความผิดพลาดของแรงดันด้านขาออกในช่วงความถี่ที่ไม่ได้พิจารณาได้เช่นกัน ซึ่งจากการทดสอบพบว่า วงจรกรองอนาล็อกแบบความถี่ผ่านต่ำควรกำหนดค่าความถี่ต่ำสุดเป็น 1 เฮิรตซ์เสมอ และการออกแบบ วงจรกรองอนาล็อกแบบความถี่ผ่านสูง ควรกำหนดช่วงความถี่สูงสุดไว้เท่ากับความต้องการของความถี่ที่ใช้งานจริง เช่น 10 จิกะเฮิรตซ์ (10^{10} เฮิรตซ์) เป็นต้น

ง) การทำงานของเจนเนติกอัลกอริทึมสามารถค้นหาคำตอบในตำแหน่งจุดต่างๆ โดยรวมของพื้นที่ในการค้นหาทั้งหมดได้ดี แต่เจนเนติกอัลกอริทึมยังมีจุดด้อยในการค้นหาคำตอบเฉพาะแห่ง ซึ่งจะเป็นไปได้ช้ามาก ดังนั้นในงานวิจัยนี้จึงได้นำขบวนการค้นหาคำตอบเฉพาะแห่งที่เรียกว่า การปีนเขา (Hill Climbing) มาใช้ในการขยับค่าอุปกรณ์ของโครโมโซมที่ดีที่สุดที่ถูกค้นพบใหม่ในการทำงานของเจนเนติกอัลกอริทึมในแต่ละรอบไว้ด้วย โดยเรียกว่าเป็นการขยับค่า (Sliding) เช่น หากโปรแกรมวนทำงานครบรอบแล้วพบโครโมโซมที่เป็นคำตอบที่ดีกว่าโครโมโซมที่ดีที่สุดที่ค้นพบในรุ่นก่อนแล้ว จะให้โปรแกรมปรับเฉพาะค่าของอุปกรณ์ในตำแหน่งของยีนส์แต่ละตัว โดยการขยับค่านี้อาจปรับค่าครั้งละ 1 ตำแหน่ง ตำแหน่งละ 2 ค่า คือ ค่าที่เปลี่ยนแปลงไปทางด้านลบและค่าที่เปลี่ยนแปลงไปทางด้านบวก โดยการขยับค่านี้อาจต้องไม่ปรับค่ามากไปจนเกินขอบเขตรหัสของชนิดอุปกรณ์แต่ละตัว ได้แก่ ตัวต้านทานมีรหัส 1 ถึง 167 ตัวเก็บประจุมีรหัส 168 ถึง 323 และตัวเหนี่ยวนำมีรหัส 324 ถึง 700 การขยับค่าจะปรับค่าครั้งละ 1 ใน 10 ของระยะห่างจากค่ารหัสนั้นถึงขอบเขตล่าง และขอบเขตบน ซึ่งจะช่วยให้การขยับค่าเป็นไปได้รวดเร็วขึ้น ซึ่งหากวงจรหนึ่งมีจำนวนอุปกรณ์ที่ใช้ทั้งหมด 15 ตัว จะต้องขยับค่าและประมวลผลผ่านโปรแกรม PSPICE ทั้งหมด 30 วงจร และรวมกับวงจรที่ขยับค่าตัวต้านทานปิดหัวท้ายอีก 2 วงจรรวมเป็น 32 วงจรนั่นเอง โดยเมื่อขยับค่าแล้วจะหาว่าในตำแหน่งยีนส์ของรหัสใดที่ปรับค่าแล้วทำให้มีความความผิดพลาดที่ต่ำลง จะกำหนดให้โปรแกรมวนปรับค่าในตำแหน่งเหล่านั้นอีกจนกว่าจะไม่พบโครโมโซมที่มีค่าความผิดพลาดที่ต่ำกว่าเดิมอีกแล้ว จากนั้นจึงนำโครโมโซมที่ถูกพัฒนาแล้วนี้ไปวนกลับเข้าไปในประชากรแบบสุ่มจำนวน 5 เปอร์เซ็นต์ของจำนวนประชากรในทุกรอบการทำงาน เพื่อเป็นการป้องกันการสูญหายของโครโมโซมที่มีค่าความเหมาะสมที่ดีที่สุดอีกด้วย

จ) เมื่อเจนเนติกอัลกอริทึมเริ่มต้นค้นหาคำตอบ พบว่าจำนวนอันดับของวงจรในประชากรส่วนใหญ่จะมีค่าต่ำและจะเริ่มมีจำนวนอันดับที่ใกล้เคียงกันเมื่อเจนเนติกอัลกอริทึมวนทำงานในรอบต่อไป เนื่องจากการสุ่มค่าโดยที่มีจำนวนอันดับของวงจรสูงกว่า จะต้องสุ่มค่าในตำแหน่งต่างๆ มากกว่า จึงทำให้ลักษณะแรงดันของวงจร ณ ความถี่ต่างๆ ที่ได้จากการสุ่มค่ามีลักษณะที่ไม่ตรงกับข้อกำหนดในการออกแบบมากกว่าโครโมโซมที่มีจำนวนอันดับต่ำกว่า ดังนั้นผลการค้นหาคำตอบของเจนเนติกอัลกอริทึมในช่วงแรกจึงเริ่มต้นจากจำนวนอันดับของวงจรต่ำๆ ซึ่งเมื่อเจนเนติกอัลกอริทึมวนทำงานแล้ว จะไม่สามารถเพิ่มจำนวนอันดับได้เอง ทำให้เกิดการลู่เข้าหาคำตอบเฉพาะแห่ง ดังนั้นจึงกำหนดให้หากเจนเนติกอัลกอริทึมวนทำงานครบ 3 รอบแล้วยังไม่สามารถพบโครโมโซมที่มีค่าความเหมาะสมดีกว่าเดิมได้ ให้เพิ่มจำนวนอันดับขึ้น 1 อันดับ โดยใช้การสุ่มค่าในตำแหน่งหลักต่อท้ายโครโมโซมทุกตัวของประชากรทั้งหมด และเนื่องจากการเพิ่ม

จำนวนอันดับนี้ขึ้น ดังนั้นในการเขียนโปรแกรมจึงต้องกำหนดให้มีการขยายขนาดของเมตริกซ์ให้พอเพียงกับการเพิ่มจำนวนอันดับของวงจรด้วย

แต่ทั้งนี้ การเพิ่มจำนวนอันดับต่อท้ายหลักของโครโมโซมทุกตัว จะไม่ทำให้โครโมโซมที่ดีที่สุดสูญหายไปจากประชากร เนื่องจากหากโครโมโซมทุกตัวที่ถูกเพิ่มจำนวนอันดับแล้วไม่พบโครโมโซมใดที่มีค่าความผิดพลาดต่ำกว่าเดิมเลย ก็ยังมีขั้นตอนการป้อนกลับโครโมโซมที่ดีที่สุดจากรุ่นก่อนแบบสุ่มจำนวน 5 เปอร์เซ็นต์อยู่เสมอ ซึ่งจะเป็นส่วนหนึ่งที่ช่วยหน่วงไม่ให้เกิดการเพิ่มจำนวนอันดับขึ้นเองโดยไม่มีการพัฒนาค่าความผิดพลาดให้ต่ำลงได้

5.3 ข้อเสนอแนะ

การออกแบบวงจรกรองโดยใช้เงินเนติกอัลกอริทึมนี้ ยังไม่สามารถนำจำนวนอันดับของวงจรแต่ละโครโมโซมมาใช้เป็นตัวช่วยในการกำหนดฟังก์ชันการค่าความเหมาะสมที่ใช้ในการเลือกได้ เนื่องจากหากสามารถนำผลของจำนวนอันดับของวงจรแต่ละโครโมโซมมาคำนวณด้วยแล้ว จะช่วยป้องกันไม่ให้เกิดวงจรที่ออกแบบมีจำนวนอันดับที่สูงเกินไปได้ แต่ปัญหาที่สำคัญคือ หากนำจำนวนอันดับของวงจรในแต่ละโครโมโซมมาคำนวณเพื่อหาค่าความเหมาะสมด้วยแล้ว จะกำหนดฟังก์ชันค่าความเหมาะสมอย่างไร เพื่อไม่ให้ขบวนการคัดเลือกของเงินเนติกอัลกอริทึมมุ่งค้นหาแต่โครโมโซมที่ใช้จำนวนอันดับต่ำแต่เพียงอย่างเดียว โดยไม่สนใจโครโมโซมที่ดีแต่มีจำนวนอันดับสูงเลย ซึ่งจะทำให้เงินเนติกอัลกอริทึมไม่สามารถค้นหาโครโมโซมที่มีค่าความเหมาะสมที่ดีที่สุดได้

นอกจากการออกแบบควรจะคำนวณถึงจำนวนอุปกรณ์ที่ใช้แล้ว ราคาของอุปกรณ์แต่ละตัว ซึ่งมีราคาแตกต่างกันไป ก็เป็นส่วนที่สำคัญมาก ที่ควรนำมาใช้ในการพิจารณา เพื่อให้วงจรที่ออกแบบได้ เป็นวงจรที่ใช้ต้นทุนการผลิตต่ำที่สุด โดยที่ผลตอบแทนของวงจรยังตรงกับข้อกำหนดในการออกแบบทุกประการ สิ่งสำคัญในการออกแบบโดยมีการคิดผลของราคาอุปกรณ์แต่ละตัว ก็มีปัญหาเช่นเดียวกันกับการออกแบบวงจรเพื่อให้ใช้จำนวนอุปกรณ์ต่ำที่สุด เนื่องจากหากผู้เขียนโปรแกรมกำหนดน้ำหนักในการค้นหาคำตอบระหว่าง วงจรที่ตรงกับความต้องการในการออกแบบ กับวงจรที่มีราคาถูกไม่ดีพอแล้ว จะทำให้การค้นหาเป็นไปในทิศทางที่ไม่ถูกต้องได้

เงินเนติกอัลกอริทึมยังสามารถนำไปใช้ในการออกแบบวงจรนาฬิกาชนิดอื่นได้ โดยอาจกำหนดให้มีการเชื่อมต่อเป็นโครงสร้างของวงจรที่มีลักษณะแปลกใหม่ต่างๆ [9] แต่ในงานวิจัยที่ผ่านนั้น ไม่ได้คำนึงถึงการค้นหาแบบเงินเนติกอัลกอริทึมมากนัก โดยจะมุ่งใช้เงินเนติกอัลกอริทึมไปในแนวทางของการแก้ไขปัญหแบบสุ่มมากกว่า ดังจะเห็นได้จากการใช้จำนวนประชากรที่สูงมาก รวมถึงโครโมโซมของประชากรในรุ่นแรกๆ นั้นส่วนใหญ่จะไม่สามารถถอดรหัสออกเป็น

วงจรที่ถูกตัดได้เลย ซึ่งการทำงานของเงินเนติกอัลกอริทึมในลักษณะดังกล่าวทำให้สูญเสียเวลาในการประมวลผลอย่างมาก

นอกจากนี้อาจเห็นได้ว่า การออกแบบวงจรกรองอนาล็อกของเงินเนติกอัลกอริทึมในแต่ละครั้งจะต้องเริ่มต้นที่การสุ่มค่า ชนิด และโครงสร้างการเชื่อมต่อวงจรขึ้น ซึ่งหากสามารถนำผลการออกแบบวงจรต่างๆที่เงินเนติกอัลกอริทึมเคยออกแบบในครั้งก่อนเก็บไว้แยกกันเป็นส่วนว่าอุปกรณ์ค่าใดและการเชื่อมต่อแบบใดจะมีผลดีในการออกแบบวงจรกรองอนาล็อกชนิดใดในช่วงความไหนได้ ก็อาจจะมีความเป็นไปได้ที่จะทำให้ช่วยลดขนาดของพื้นที่การค้นหาลง ซึ่งจะส่งผลให้เวลาในการประมวลผลของเงินเนติกอัลกอริทึมลดลงได้เป็นอย่างมาก