


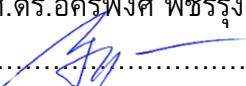


ข้อเสนอโครงการวิศวกรรมคอมพิวเตอร์

การควบคุมอากาศยานไร้คนขับเพื่อลอดผ่านช่องว่างที่กำลังเคลื่อนที่
ด้วยการเรียนรู้แบบเสริมกำลัง

**Control of Unmanned Aerial Vehicle for flight through moving gaps
with reinforcement learning**

นาย นภณัฐ ทองตัน รหัสนิต 6010502616

อาจารย์ที่ปรึกษา  วันที่ 27 เดือนสิงหาคม พ.ศ. 2563
(ผศ.ดร.อัชรพงศ์ พิชรรุ่งเรือง)
อาจารย์ที่ปรึกษาร่วม  วันที่ 27 เดือนสิงหาคม พ.ศ. 2563
(ผศ.ดร.กาญจน์พันธุ์ สุขวิชชัย)

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเกษตรศาสตร์
ปีการศึกษา 2562

1. ความสำคัญและที่มา

ในปัจจุบันมีการใช้อากาศยานไร้คนขับอย่างแพร่หลายในหลากหลายส่วนงานเช่น ส่วนของการทำการเกษตรกรรม โดยนำอากาศยานไร้คนขับมาช่วยในการฉีดปุ๋ยหรือมาช่วยในการเก็บข้อมูลต่าง ๆ เพื่อนำไปพัฒนาต่อยอดทางเกษตรกรรม โดยการนำอากาศยานไร้คนขับมาช่วยทำงานนั้นเป็นการเพิ่มประสิทธิภาพการทำงาน สามารถลดเวลาที่ใช้ในการทำงาน และลดจำนวนแรงงานที่จำเป็นต้องใช้ในการทำงานต่าง ๆ ภายในพื้นที่เกษตรกรรม หรือในส่วนของการบินส่งพัสดุหรือเอกสารต่าง ๆ ที่มีขนาดเล็กไม่ใหญ่มากเพื่อช่วยลดเวลาที่ใช้ในการขนส่งลงจากเดิม เนื่องจากการคมนาคมบนท้องถนนนั้นค่อนข้างแออัดมากในเมืองใหญ่ หรือในส่วนของช่วยเหลือผู้ประสบภัย โดยเป็นการนำอากาศยานไร้คนขับมาทำการสำรวจพื้นที่และค้นหาผู้ประสบภัยที่หลงเหลืออยู่ ก็ทำให้เกิดโจทย์ปัญหาที่ทำนายสำหรับนักบินมากมาย เพราะว่าพื้นที่ที่เกิดภัยพิบัตินั้นค่อนข้างมีพื้นที่บินจำกัด รวมถึงช่องว่างและทัศนวิสัยในการบินมีจำกัด ซึ่งอาจจะทำให้เกิดความเสียหายกับอากาศยานไร้คนขับได้ และอาจจะเกิดเหตุการณ์ไม่คาดคิด ทำให้นักบินนั้นไม่สามารถทำการตอบสนองได้ทันในช่วงเวลาสั้น ๆ จึงถือเป็นขีดจำกัดของความสามารถของมนุษย์ในการตอบสนองต่อสถานการณ์เช่นนี้

ในช่วงไม่กี่สิบปีที่ผ่านมาได้มีการนำการเรียนรู้แบบเสริมกำลังมาใช้งานในด้านต่าง ๆ อย่างแพร่หลาย ซึ่งการเรียนรู้แบบเสริมกำลังเป็นการทำการจำลองการเรียนรู้ของมนุษย์ตอนผ่านสถานการณ์ต่าง ๆ จนทำให้เกิดการเรียนรู้ผ่านประสบการณ์ จนได้มาเป็นพฤติกรรมต่าง ๆ ที่ตอบสนองสถานการณ์นั้น ๆ โดยในช่วงสองถึงสามปีที่ผ่านมาสิ่งที่ทำให้เกิดความสนใจไปทั่วโลกโดยทาง Google Deepmind ได้ทำการสร้าง Alpha Go ขึ้นมาเพื่อทำการแข่งขันหมากล้อมกับมนุษย์ และในที่สุด Alpha Go ก็สามารถชนะ มีอวสานอันดับหนึ่งหมากล้อมของโลกได้สำเร็จ ทำให้เห็นว่าการเรียนรู้แบบเสริมกำลังนั้นสามารถทำให้คอมพิวเตอร์นั้นมีความสามารถในการเรียนรู้เทียบเท่าหรือเก่งกว่ามนุษย์แล้ว

จากที่กล่าวมาข้างต้นทำให้ข้าพเจ้าได้เล็งเห็นว่าสามารถนำการเรียนรู้แบบเสริมกำลังมาทำการเรียนรู้ และ ทำงานที่เกินขีดจำกัดความสามารถของมนุษย์ จึงเลือกปัญหาของการบินในสถานที่ที่เกิดสถานการณ์ภัยพิบัติ ที่เป็นส่วนของการบินลอดช่องว่าง เพื่อนำมาทำการศึกษาและหาวิธีแก้ปัญหาสำหรับสถานการณ์เช่นนี้ เพื่อทำการเพิ่มประสิทธิภาพของการทำภารกิจของอากาศยานไร้คนขับในปัจจุบัน

2. วัตถุประสงค์

- 2.1. เพื่อศึกษาและประยุกต์ใช้การเรียนรู้แบบเสริมกำลังมาทำการนำร่องอากาศยานไร้คนขับเพื่อลอดช่องว่างที่กำลังเคลื่อนที่

3. ขอบเขตโครงการ

- 3.1. ใช้การเรียนรู้แบบเสริมกำลัง โดยใช้การเรียนรู้แบบเสริมกำลังแบบผกผัน กับ Deep Q-Learning
- 3.2. อากาศยานไร้คนขับสามารถทำการ take off และ landing ได้ด้วยตัวเอง

3.3. อากาศยานไร้คนขับสามารถบินลอดช่องว่างที่พบและสามารถเดินทางไปยังจุดที่กำหนดได้

4. ประโยชน์ที่คาดว่าจะได้รับ

4.1. ผู้อ่านสามารถนำไปต่อยอดในการสร้าง Autonomous Rescue UAV ได้

4.2. ผู้อ่านสามารถนำไปประยุกต์เพื่อทำการสร้าง Autonomous Racing Drone ได้

5. ทฤษฎีและหลักการที่เกี่ยวข้อง

5.1. ทฤษฎีพื้นฐาน

- Unmanned Aerial Vehicle

Unmanned Aerial Vehicle (UAV) หมายถึงอากาศยานที่ไม่มีนักบินมนุษย์ ซึ่งเป็นประเภทหนึ่งของยานพาหนะไร้คนขับ โดยเป็นส่วนหนึ่งของ UAS (Unmanned Aerial System) ซึ่งสามารถแบ่งประเภทจากระบบควบคุมและระบบสื่อสารได้เป็น 3 ประเภทคือ เป็นการควบคุมผ่าน remote control , เป็นการประมวลผลบนเครื่อง (On board computer control) , หรือเป็นการควบคุมโดยหุ่นยนต์อัตโนมัติ (Autonomous Robot control) [1] โดยในปัจจุบันมีรูปแบบของอากาศยานไร้คนขับมากมาย โดยแบ่งออกเป็น 4 ประเภทหลัก ๆ ก็คือ อากาศยานไร้คนขับหลายใบพัด (Multi Rotor Drones) , อากาศยานไร้คนขับปีกตรึง (Fixed wing Drones) , อากาศยานไร้คนขับใบพัดเดี่ยว (Single Rotor Helicopter) และ อากาศยานไร้คนขับปีกตรึงแบบลูกผสมขึ้นลงแนวตั้ง (Fixed wing Hybrid Vertical Take-Off and Landing) [2] โดยอากาศยานไร้คนขับหลายใบพัดนั้นจะใช้แรงยกจากใบพัดเท่านั้นในการลอยตัวและเคลื่อนที่ โดยสามารถควบคุมการเคลื่อนที่ต่าง ๆ ได้ด้วยการกำหนดความเร็วของใบพัดแต่ละใบ โดยจะมีข้อดีที่เป็นอากาศยานไร้คนขับที่ง่ายต่อการออกแบบและควบคุม , ต้องการพื้นที่ในการขึ้นและลงจอดน้อย สามารถทำการหยุดนิ่งขณะที่ลอยตัวอยู่กลางอากาศได้ และสามารถปฏิบัติงานในพื้นที่จำกัดได้ แต่มีข้อสังเกตคือสิ่งที่ต้องอาศัยแรงยกจากใบพัดในการลอยตัวและเคลื่อนที่นั้นทำให้ใบพัดทุกใบจำเป็นต้องทำงานตลอดเวลา ส่งผลให้สิ้นเปลืองเชื้อเพลิงหรือไฟฟ้าที่ใช้ในการขับเคลื่อนอากาศยานไร้คนขับ ต่อมาในรูปแบบอากาศยานไร้คนขับปีกตรึงนั้นจะอาศัยแรงยกจากปีกเป็นหลัก แต่การที่ปีกจะสามารถสร้างแรงยกได้นั้นจำเป็นต้องมีอากาศเคลื่อนที่ผ่าน ทำให้จะต้องอาศัยใบพัดหรือเครื่องยนต์เพื่อสร้างแรงดันออกไปให้อากาศยานไร้คนขับเคลื่อนที่ และอากาศยานไร้คนขับประเภทนี้ทำการควบคุมการเคลื่อนที่โดยใช้ พื้นที่บังคับอากาศยาน (Control surface) เพื่อทำการควบคุมท่าทาง และควบคุมความเร็วของอากาศยานด้วยความเร็วของใบพัด โดยข้อดีคือมีระยะเวลาในการปฏิบัติการยาวนาน เพราะหาแรงยกที่ได้มานั้นมาจากปีก ส่งผลให้ประหยัดเชื้อเพลิงหรือไฟฟ้ามากกว่าอากาศยานไร้คนขับแบบหลายใบพัด และมีความเร็วในการปฏิบัติงานมาก ข้อสังเกตคือต้องการพื้นที่ในการขึ้นและลงจอดมาก ต่อมาอากาศยานไร้คนขับแบบใบพัดเดี่ยว เป็นอากาศยานที่ใช้ใบพัดเพียงใบเดียวในการสร้างแรงยก และควบคุมท่าทางและการเคลื่อนที่ด้วยใบพัดอีกใบ ร่วมกับ การควบคุม

องศาที่ทำการปะทะกับอากาศของใบที่สร้างแรงยก โดยข้อดีคือ ต้องการพื้นที่ในการขึ้นและลงจอดน้อย ควบคุมง่าย และ สามารถปฏิบัติงานในพื้นที่จำกัดได้ ส่วนข้อสังเกตคือเสถียรภาพของเครื่อง เนื่องจากมีใบพัดเพียงใบเดียวในการสร้างแรงยกให้กับเครื่อง และ ไม่สามารถทำความเร็วได้เท่ากับอากาศยานไร้คนขับปีกตรึง และรูปแบบสุดท้ายคือ อากาศยานไร้คนขับปีกตรึงแบบลูกผสมขึ้นลงแนวตั้ง ซึ่งเป็นการรวมข้อดีของอากาศยานไร้คนขับหลายใบพัด กับอากาศยานไร้คนขับปีกตรึงเข้าด้วยกัน โดยการทำงานคือการใช้แรงยกที่สร้างจากใบพัดเพื่อทำการขึ้นและลงจอด ส่วนการเคลื่อนที่ส่วนใหญ่บนอากาศนั้นจะใช้แรงยกจากปีกเป็นหลัก ทำให้มีข้อดีที่ต้องการพื้นที่ในการขึ้นและลงจอดน้อย มีระยะเวลาในการปฏิบัติการยาวนาน แต่ข้อสังเกตคือการควบคุม และ การออกแบบค่อนข้างยาก



ภาพที่ 11 : อากาศยานไร้คนขับหลายใบพัด



ภาพที่ 22 : อากาศยานไร้คนขับปีกตรึง



ภาพที่ 33 : อากาศยานไร้คนขับใบพัดเดียว



ภาพที่ 44 : อากาศยานไร้คนขับปีกตรึงลูกผสมขึ้นลงแนวตั้ง

- Navigation

Navigation หรือการนำทางนั้นคือสิ่งที่เกี่ยวข้องการวัตถุที่เคลื่อนที่ ซึ่งโดยมากคือ ยานพาหนะ และเกี่ยวข้องกับการกำหนดและแนะนำเส้นทาง โดยการกำหนดเส้นทางนั้นขึ้นอยู่กับ การได้มาของเวกเตอร์สถานะ (State vector) ของวัตถุ ณ เวลานั้น ๆ โดยทั่วไปเวกเตอร์

1 แหล่งที่มา : <https://pxhere.com/th/photo/1446049>

2 แหล่งที่มา : [https://commons.wikimedia.org/wiki/File:MQ-9_Reaper_UAV_\(cropped\).jpg](https://commons.wikimedia.org/wiki/File:MQ-9_Reaper_UAV_(cropped).jpg)

3 แหล่งที่มา : https://commons.wikimedia.org/wiki/File:F-flite_Blade_mSR_Micro_Single_Rotor_Model_Helicopter_3.jpg

4 แหล่งที่มา : https://en.wikipedia.org/wiki/File:DeltaQuad_VTOL_surveillance_UAV.jpg

สถานะนั้นจะมีตำแหน่ง ความเร็ว และ ทำทาง ในขณะที่การกำหนดระยะทางนั้น คือการได้มาซึ่งลักษณะของการเคลื่อนที่ของวัตถุโดยไม่มีการตอบกลับ โดยคำแนะนำที่ได้นั้นจะบังคับวัตถุให้เคลื่อนที่ไปยังเส้นทางที่กำหนดไว้ล่วงหน้าเพื่อดำเนินไปถึงปลายทาง โดยนักเดินทางชาวอเมริกันได้ทำการอธิบายการนำทางไว้ว่าคือ วิธีการ วางแผน บันทึก และ ควบคุมการเคลื่อนที่ของยานพาหนะจากที่หนึ่งไปยังอีกที่หนึ่ง (อ้างอิงจาก National Imagery and Mapping Agency, 1995) โดยการนำทางนั้นส่วนใหญ่จะเป็นการนำทางในเชิง 3 มิติ โดยการนำทางนั้นเป็นการปฏิบัติการที่ต้องทำตลอดเวลา โดยการเปรียบเทียบตำแหน่งและคำแนะนำ โดยสามารถแบ่งเทคนิคของการนำทางออกเป็น 2 ประเภทคือ เทคนิคการนำทางอัตโนมัติ (Autonomous navigation technique) หรือการนำทางแบบ on board และ การนำทางแบบไม่อัตโนมัติ (Nonautonomous navigation technique) โดยเทคนิคการนำทางแบบอัตโนมัติ นั้นคือการที่ระบบสามารถทำการปฏิบัติการได้ด้วยตัวเอง โดยไม่ต้องมีการสนทนากับสิ่งอำนวยความสะดวกและช่วยเหลือการนำทาง (Navigation support facility) ในทางตรงกันข้าม การค้นหาตำแหน่งแบบอัตโนมัติ (Autonomous positioning) นั้นการนำทางแบบอัตโนมัติอาจจะเกี่ยวข้องกับทฤษฎีการค้นหาตำแหน่งแบบไม่อัตโนมัติ (Nonautonomous positioning method) ในส่วนของเทคนิคการนำทางแบบไม่อัตโนมัติ ฟังก์ชันการทำงานนั้นจำเป็นจะต้องทำการสนทนาสื่อสารกับการนำทางนอก เช่น วัตถุหรือพาหนะที่อาศัยข้อมูลที่ได้จากสิ่งอำนวยความสะดวกและช่วยเหลือการนำทาง [3]

- Machine Learning

เป็นการศึกษาอัลกอริทึมของคอมพิวเตอร์ที่มีเป้าหมายในการพัฒนาระบบอัตโนมัติผ่านประสบการณ์ บางครั้งถูกมองว่าเป็นส่วนหนึ่งของ artificial intelligence โดยอัลกอริทึมของ machine learning นั้นเกิดจากการสร้างแบบจำลองทางคณิตศาสตร์ขึ้นมาจากข้อมูลตัวอย่างที่เรียกว่า training data เพื่อที่จะทำให้ระบบสามารถทำการตัดสินใจ หรือ ทำการคาดการณ์ล่วงหน้าได้ โดยไม่จำเป็นต้องทำการเขียนโปรแกรมที่จะต้องทำเช่นนั้น โดย machine learning นั้นมีความเกี่ยวข้องอย่างใกล้ชิดกับสถิติเชิงคำนวณที่เน้นไปทางการคาดการณ์โดยใช้คอมพิวเตอร์ [4] โดย Machine learning นั้นสามารถแบ่งออกตามประเภทของการเรียนรู้ได้ 3 ประเภทหลัก ๆ คือ การเรียนรู้แบบมีผู้สอน (Supervised Learning) , การเรียนรู้แบบเสริมกำลัง (Reinforcement Learning) และ การเรียนรู้แบบไม่มีผู้สอน (Unsupervised Learning) โดยการเรียนรู้แบบมีผู้สอนนั้นจำเป็นจะต้องใช้ข้อมูลที่มีคำตอบที่ถูกต้องแน่นอนเข้าไปทำการสอน โดยการเรียนรู้แบบมีผู้สอนนั้นแบ่งออกเป็น 2 ประเภทคือการแบ่งแยกประเภท (Classification) และ การลดถอย (Regression) ส่วนการเรียนรู้แบบไม่มีผู้สอนนั้นคือการที่นำข้อมูลดิบใส่เข้าไปเพื่อให้เรียนรู้ โดยจะสามารถทำได้แค่การแบ่งแยกประเภท โดยแบ่งออกเป็น 2 ประเภทคือ การแบ่งกลุ่มข้อมูล (Clustering) และ การลดมิติของข้อมูล (Dimension)

Reduction) และในส่วนของการเรียนรู้แบบเสริมกำลังคือการใส่ข้อมูลดิบ และ ฟังก์ชันรางวัลลงไป เพื่อให้เรียนรู้

- Reinforcement Learning

เป็นส่วนหนึ่งของ machine learning โดยเป็นเรื่องของการดำเนินการที่เหมาะสมเพื่อที่จะได้รับรางวัลสูงสุดในสถานการณ์หนึ่ง ๆ โดยจะใช้ซอฟต์แวร์หรือเครื่องมือต่าง ๆ เพื่อทำการค้นหาพฤติกรรมที่เป็นไปได้ที่ทำให้ได้รับรางวัลมากที่สุด ในสถานการณ์นั้น ๆ โดย reinforcement learning นั้นแตกต่างจาก supervised learning ตรงที่ในข้อมูลของ supervised learning นั้นจะมีคำตอบมาให้อยู่แล้ว จึงทำให้ supervised learning นั้นได้ศึกษามาจากข้อมูลที่ถูกต้องแล้วเสมอ แต่ในส่วนของการเรียนรู้แบบเสริมกำลังนั้นข้อมูลที่เราใส่เข้าไปให้มันไม่มีคำตอบอยู่ในนั้น แต่ agent ของ reinforcement learning นั้นจะทำการตัดสินใจด้วยตนเองว่าจะทำงานนั้น ๆ อย่างไร โดยตัดสินใจจากประสบการณ์ที่มี โดยจุดสำคัญของการเรียนรู้แบบเสริมกำลังนั้นประกอบด้วย อินพุต , เอาต์พุต แบบการเรียนรู้ โดยอินพุตของการเรียนรู้แบบเสริมกำลังควรเป็นสถานะเริ่มต้นของโมเดลที่จะเริ่มทำงาน ในส่วนของเอาต์พุตนั้นมีความเป็นไปได้มากมาย ขึ้นกับปัญหาที่ต้องการแก้ไข และในส่วนของการเรียนนั้นจะอยู่บนพื้นฐานของอินพุตเป็นหลัก โดยโมเดลจะทำการคืนค่าสถานะและผู้ใช้จะต้องทำการตัดสินใจรางวัลหรือบทลงโทษที่จะให้กับโมเดลโดยขึ้นอยู่กับเอาต์พุต โดยโมเดลทำการเรียนรู้ตลอดเวลาและคำตอบหรือวิธีการแก้ปัญหาที่ดีที่สุดที่เกิดขึ้นนั้น จะตัดสินใจจากการที่ได้รับรางวัลมากที่สุดจากผู้ใช้งาน โดยการเรียนรู้แบบเสริมกำลังนั้นแบ่งออกเป็น 2 ประเภทใหญ่ ๆ คือ การเรียนรู้แบบเสริมกำลังเชิงบวก และการเรียนรู้แบบเสริมกำลังเชิงลบ โดยการเรียนรู้แบบเสริมกำลังเชิงบวกหมายถึงการที่เกิดเหตุการณ์จากพฤติกรรมใด ๆ นั้นจะส่งผลให้เพิ่มความแข็งแรง และความถี่ของพฤติกรรม หรือกล่าวว่าเป็นผลดีต่อพฤติกรรม โดยข้อดีคือจะได้ประสิทธิภาพที่สูงที่สุด และสามารถควบคุมการเปลี่ยนแปลงได้ในระยะยาว แต่ข้อเสียคือ การที่มีการเรียนรู้ที่มากเกินไปอาจนำไปสู่สถานะสถานะเกินพิกัด (Overload of states) ซึ่งจะส่งผลทำให้ประสิทธิภาพลดลง และในส่วนของการเรียนรู้แบบเสริมกำลังเชิงลบนั้นหมายถึง การเรียนรู้จากพฤติกรรมเชิงลบที่ถูกให้หยุด หรือ หลีกเลียง โดยข้อดีคือ สามารถทำการเพิ่มพฤติกรรมต่าง ๆ ได้และมีความสามารถในการขจัดขึ้นมาตรฐานต่ำ แต่ข้อเสียคือการเรียนรู้แบบเสริมกำลังประเภทนี้จะพบกับพฤติกรรมที่น้อยมาก ในปัจจุบันนั้นการเรียนรู้แบบเสริมกำลังนั้นถูกนำไปใช้ในหลากหลายวัตถุประสงค์ เช่นเป็นการนำไปใช้กับหุ่นยนต์อุตสาหกรรม , การนำไปใช้กับการเรียนรู้ของเครื่องจักร (Machine Learning) และการประมวลผลข้อมูล โดยการเรียนรู้แบบเสริมกำลังนั้นสามารถใช้กับสภาพแวดล้อมขนาดใหญ่ได้ในเงื่อนไขต่อไปนี้ คือต้องทราบโมเดลของสภาพแวดล้อมนั้น ๆ แต่ไม่มีข้อมูลทำการวิเคราะห์แล้ว และทางเดียวที่จะสามารถเก็บข้อมูลจากสภาพแวดล้อม คือผลการตอบสนองของสภาพแวดล้อมเท่านั้น [5]

5.2. การทบทวนวรรณกรรม (Literature Review)

- Stuart Russell (1998) [6] ได้ทำการกล่าวถึง baseline architecture สำหรับการเรียนรู้ของ agent ที่สามารถรับมือกับสภาพแวดล้อมแบบสุ่ม (Stochastic Environment) ได้ โดยสถาปัตยกรรมนี้ใช้การเรียนรู้แบบเสริมกำลังควบคู่กับ representing temporal processes ที่เปรียบเสมือน graphical models และ ได้อธิบายว่าการเรียนรู้แบบเสริมกำลังนั้นจะมีประสิทธิภาพมากเมื่อนำมาใช้กับ adaptive control โดยในตอนแรก reward function นั้นได้ถูกกำหนดไว้ก่อนล่วงหน้าแล้ว แต่ปัญหาก็คือเราต้องการ reward function ที่เหมาะสมที่สุด โดย Russell ได้กล่าวไว้ว่า reward function ควรจะถูกปรับให้เหมาะสมด้วยวิธีการทางธรรมชาติ ดังนั้นทางแก้ปัญหาก็คือ การเรียนรู้แบบเสริมกำลังแบบผกผัน (Inverse Reinforcement Learning) ซึ่งการเรียนรู้แบบเสริมกำลังแบบผกผันนั้นคือการที่ระบบทำการเรียนรู้ผ่านการสังเกตการณ์จากผู้เชี่ยวชาญ โดย reward function ที่ได้ควรจะเป็นฟังก์ชันง่าย ๆ ที่แตกต่างกันขึ้นกับข้อมูลที่ได้รับเข้ามา โดยการแสดงพฤติกรรมของผู้เชี่ยวชาญนั้นยิ่งดีมากเท่าไร ก็จะทำให้ reward function ที่ได้รับกลับมานั้นมีประสิทธิภาพมากขึ้นเท่านั้น

5.3. งานที่เกี่ยวข้อง (Related Work)

- Erli Lyu, Yuan Lin, Wei Liu และ Max Q. -H. Meng (2015) [7] นั้นได้ทำการนำภาพที่ได้จากกล้องมาทำการรวมกับการควบคุมด้วย PID เพื่อให้การควบคุมด้วย PID นั้นมีประสิทธิภาพมากยิ่งขึ้น โดยเขาต้องการสร้างอากาศยานไร้คนขับแบบสี่ใบพัด (quadrotor) ในราคาถูกจึงเลือกที่จะทำการออกแบบระบบ image processing เพื่อให้สามารถค้นหาช่องว่างนั้นจากภาพที่ได้จากกล้องที่ติดอยู่บนเครื่อง โดยเลือกใช้การควบคุมด้วย PID รวมกับ segmented coefficients , restricted integral saturation และ integral separation โดยการทำงานของระบบนี้จะเป็นการ execute คำสั่งบนแล็ปท็อปแล้วทำการสื่อสารกับเครื่องผ่าน ROS
- Davide Falanga, Elias Mueggler, Matthias Faessler และ Davide Scaramuzza (2017) [8] ได้เล็งเห็นปัญหาว่าการบินผ่านช่องแคบนั้นเป็นอีกหนึ่งโจทย์ยากในการบินอากาศยานไร้คนขับในสภาพแวดล้อมที่ซับซ้อน เนื่องด้วยขณะที่กำลังจะลอดช่องว่างนั้นนักบินจะไม่สามารถมองเห็นขอบของช่องว่างนั้นได้ก็มีโอกาสที่เครื่องจะบินชนกับขอบของช่องว่างนั้นได้ พวกเขาจึงสร้างระบบที่ใช้ trajectory planning ร่วมกับ state estimation ในการสร้างท่าทางและเส้นทางเพื่อให้อากาศยานสี่ใบพัดนั้นสามารถบินลอดช่องว่างนั้นได้ มีการ replan ตลอดเวลาเนื่องจากว่างานก่อนที่มีคนทำนั้นไม่ได้มีการ replan ทำให้ถ้าช่องว่างนั้นมีมุม pitch ขึ้นมา มุมมองแรกตอนที่สร้าง trajectory นั้นระบบอาจจะคิดว่าไปไม่ได้เพราะขนาดของช่องว่างนั้นจะดูเล็กกว่าขนาดจริง และสามารถที่จะทำการ recovery ท่าทางหลังจากบินลอดช่องว่างนั้นแล้ว

ได้ด้วย แต่ในโครงงานนี้จะนำการเรียนรู้แบบเสริมกำลังมาทำการแทนที่ trajectory planning และ state estimation

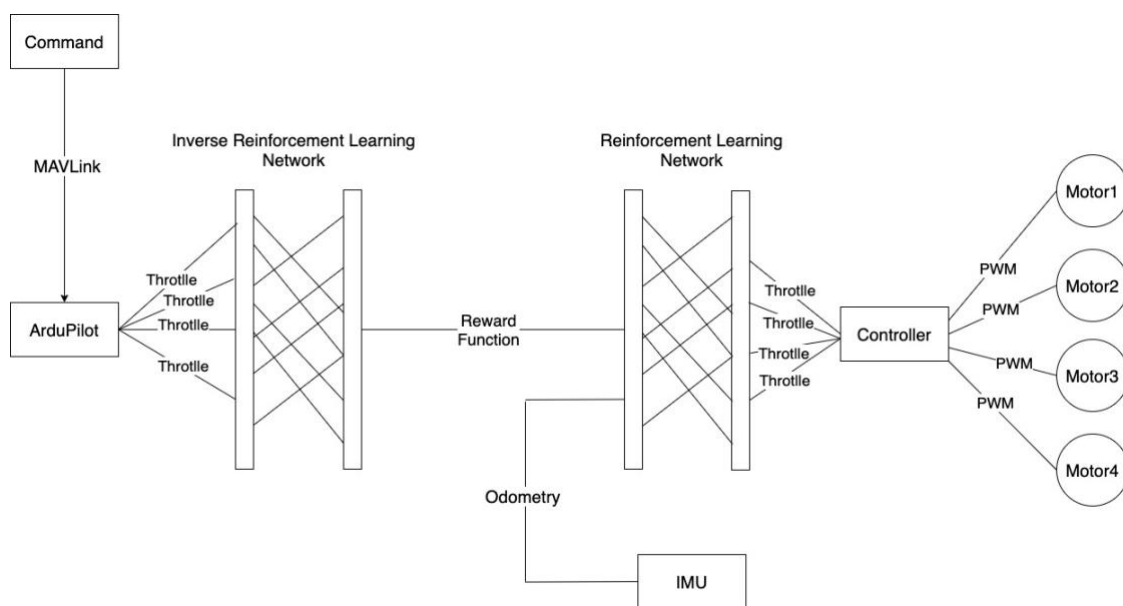
- Steven L. Waslander, Gabriel M. Hoffmann, Jung Soon Jang และ Claire J. Tomlin (2005) [9] นั้นทำการทดลองและเปรียบเทียบการทำงานของเทคนิคอย่าง Integral Sliding Mode กับ Reinforcement Learning ในการแก้ปัญหา nonlinear โดยทำการเลือกปัญหาที่เป็น Hovering มาทำการเปรียบเทียบโดยทำการแบ่งออกเป็น 2 การทดลองคือการรักษาความสูง และการรักษาความสูงกับตำแหน่ง โดย Reinforcement Control นั้นได้ทำการจับคู่สถานะล่าสุดและอินพุต เข้ากับสถานะถัดไป และได้รับการเรียนรู้เพียง 3 ชั่วโมงเท่านั้น ผลการทดลองสรุปได้ว่า response time ต่อการเปลี่ยนแปลงความสูงขึ้นไป 50 เซนติเมตรนั้นมีค่าโดยประมาณใกล้เคียงกันมาก แต่การนำการเรียนรู้แบบเสริมกำลังมาทำการรักษาตำแหน่งด้วยนั้นยังไม่สำเร็จ แต่การนำ Integral Sliding Mode มาทำนั้นมีความคลาดเคลื่อนของตำแหน่งที่อ่านค่ามาจาก GPS มีค่าไม่เกินวงกลมรัศมี 3 เมตรโดยมีจุดเริ่มต้นเป็นจุดกึ่งกลางของวงกลม สำหรับการบิน Hover เป็นเวลา 2 นาที
- Jemin Hwangbo , Inkyu Sa , Roland Siegwart and Marco Hutter (2017) [10] ได้ทำการนำเสนอทฤษฎีใหม่ในการควบคุม quadrotor โดยใช้โครงข่ายประสาทเทียมที่ได้รับการเรียนรู้โดยการเรียนรู้แบบเสริมกำลัง โดยปกติโครงข่ายประสาทเทียมนั้นจะถูกให้เรียนรู้ในการทำการจับคู่สถานะกับคำสั่งโดยตรง ทำให้การควบคุมที่กำหนดไว้ล่วงหน้า(Predefined control) นั้นล้าสมัยในการเรียนรู้แล้ว พวกเขาจึงทำการจับคู่ระหว่างสถานะกับแรงขับของมอเตอร์โดยตรง และมีอีกโครงข่ายประสาทเทียมคือการที่จับคู่ระหว่างข้อมูลที่ได้มาจากเซนเซอร์คือ ท่าทาง ตำแหน่ง ความเร็วเชิงเส้นและความเร็วเชิงมุม กับความเร็วของอากาศยาน โดยการทดลองนั้นเริ่มจากการทำงานในโลกเสมือนจริง ต่อมาก็นำ policy ที่ได้จากการเรียนรู้ในโลกเสมือนจริง มาใส่ใน quadrotor โดยโจทย์ที่ตั้งใจให้เครื่องทำงานคือการที่มีท่าทางเริ่มต้นที่แตกต่างไป ขณะที่ทดสอบในโลกเสมือนจริง เป็นการลุ่มท่าทางมา แต่ในโลกจริงเป็นการโยนออกไปแล้วทำให้เครื่องนั้นสามารถทำการปรับท่าทางให้ตรงและสามารถทำการรักษาท่าทางนั้นไว้ให้ได้

6. แนวทางการดำเนินการ (Methodology)

6.1. ศึกษาการควบคุมของอากาศยานไร้คนขับในปัจจุบัน

- 6.2. ศึกษาและเปรียบเทียบอัลกอริทึมที่จะใช้ในการทำงาน
- 6.3. เตรียม expert agent ที่จะใช้ในการ train
- 6.4. ทำการ train ในระบบเสมือนจริง
- 6.5. ทดสอบการทำงานในระบบเสมือนจริง
- 6.6. นำมาทดสอบในโลกจริง
- 6.7. ทำการปรับปรุงแก้ไขและทำการ train เพิ่มเติม

แผนผังระบบ



ภาพที่ 1 แผนผังการสอนและทำงานของระบบ

โดย ArduPilot คือแพลตฟอร์มที่เป็น open source ที่ใช้พัฒนาระบบอัตโนมัติ หรือ ระบบพาหนะไร้คนขับ โดยมีการพัฒนาอย่างต่อเนื่องโดยอาศัยการตอบรับอย่างรวดเร็วของชุมชนของผู้ใช้งานที่มีปริมาณมาก โดยทีมนักพัฒนานั้นทำงานร่วมกับชุมชนของผู้ใช้งานและพันธมิตรทางการค้า เพื่อทำการเพิ่มฟังก์ชันการใช้งานให้กับ ArduPilot เพื่อเป็นประโยชน์ต่อทุก ๆ คน [11] โดย ArduPilot นั้นแบ่งออกตามประเภทของพาหนะที่ต้องการควบคุม ในปัจจุบันแบ่งออกเป็น 5 ประเภทคือ ArduCopter , ArduPlane , ArduRover , ArduSub และ Antenna Tracker โดยในส่วนของ ArduCopter นั้นเป็นระบบในการควบคุมอากาศยานไร้คนขับหลายใบพัด และ อากาศยานไร้คนขับใบพัดเดี่ยว ในส่วนของ ArduPlane นั้นเป็นระบบในการควบคุมอากาศยานไร้คนขับปีกตรึง และ อากาศยานไร้คนขับปีกตรึงแบบลูกผสมขึ้นลงแนวตั้ง ในส่วนของ ArduRover นั้นเป็นระบบของการควบคุมยานพาหนะเคลื่อนที่บน

บกที่มีล้อ(Mobile Robot) และ เรือ ในส่วนของ ArduSub นั้นเป็นระบบของการควบคุมเรือดำน้ำ และ ชุดท้าย Antenna Tracker คือการควบคุมเสาสัญญาณเพื่อให้หมุนไปให้ตรงกับยานพาหนะที่เคลื่อนที่อยู่ให้ตรงกันที่สุด เพื่อเพิ่มระยะทางในการส่งสัญญาณควบคุมไปยังยานพาหนะนั้น ๆ

โดยในโครงการนี้ได้เลือกใช้ ArduCopter เพื่อมาเป็นผู้เชี่ยวชาญ ในการนำมาสอนให้กับการเรียนรู้แบบเสริมกำลังแบบผกผัน เพื่อให้ระบบนั้นสามารถเรียนรู้ได้มีประสิทธิภาพมากขึ้น

7. แผนการดำเนินการ (Project Timeline)

แบ่งการดำเนินการออกเป็น 3 ส่วน

ในส่วนแรก กรกฎาคม 2563 - กันยายน 2563 เป็นส่วนของการศึกษาและหาแนวทางของการทำโครงการ โดยเป็นการเน้นการปรึกษาอาจารย์ที่ปรึกษาโครงการเป็นหลัก รวมถึงพิสูจน์อัลกอริทึมต่าง ๆ ที่เลือกมาว่าเหมาะสมกับโครงการหรือไม่

ในส่วนที่สอง กันยายน 2563 - ธันวาคม 2563 เป็นส่วนของการนำอัลกอริทึมที่ได้เลือกมาทำการสร้าง network และ train agent ในระบบเสมือนจริง

ในช่วงที่สาม ธันวาคม 2563 – กุมภาพันธ์ 2564 เป็นส่วนของการนำ agent ที่ได้รับการ train แล้ว มาใช้ในโลกจริง

หัวข้อ	2563																2564			
	ภาคเรียนที่ 1				ภาคเรียนที่ 2				ภาคเรียนที่ 3				ภาคเรียนที่ 4				ภาคเรียนที่ 5			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1. Preparation																				
1.1. ศึกษาวิธีการทำโครงงาน																				
1.2. ประเมินเงื่อนไขของโครงงานก่อนการเริ่มใช้โครงงาน																				
1.3. ประเมินเงื่อนไขของโครงงานก่อนการเริ่มใช้โครงงาน																				
1.4. จัดทำข้อเสนอโครงงาน																				
1.5. นำเสนอโครงงาน																				
2. Study																				
2.1. ศึกษาการทำงานของ UAV controller																				
2.2. ศึกษาการทำงานของ Reinforcement learning																				
2.3. ศึกษาการทำงานของ Inverse reinforcement learning																				
2.4. ศึกษาการทำงานของ Inverse reinforcement learning																				
3. Implementation																				
3.1. ทำการออกแบบ network ของ reinforcement learning																				
3.2. ทำการออกแบบ network ของ Inverse reinforcement learning																				
3.3. ทำการออกแบบระบบการเรียนรู้ของ gazebo simulation																				
4. Development																				
4.1. ทำการตั้งค่าระบบการเรียนรู้ของ gazebo simulation																				
4.2. ทำการตั้งค่า network ของ reinforcement learning																				
4.3. ทำการตั้งค่า network ของ Inverse reinforcement learning																				
4.4. ทำการ train network																				
4.5. นำ network ที่ train แล้วมาใช้ในการจำลอง																				
5. Testing																				
5.1. ทำการ test network ที่ train แล้วในสถานการณ์จริง																				
5.2. ทำการแก้ไขปัญหานetwork ที่ train แล้วในสถานการณ์จริง																				
5.3. ทำการ test network ที่ train แล้วในสถานการณ์จริง																				
5.4. ทำการแก้ไขปัญหานetwork ที่ train แล้วในสถานการณ์จริง																				
6. Presentation																				
6.1. จัดทำรายงานโครงงาน																				
6.2. นำเสนอผลงาน																				
7. Conclusion																				
7.1. จัดทำสรุปโครงงาน																				

ตารางที่ 1 ตารางการดำเนินงาน

8. แนวทางการวัดและประเมินผลโครงการ

- 8.1. ประเมินโดยวัดจากเวลาที่อากาศยานไร้คนขับนั้นตอบสนองคำสั่งที่ได้รับเทียบกับจำนวน epoch
- 8.2. ได้ model การบังคับจาก expert agent ผ่าน invert reinforcement learning
- 8.3. อากาศยานจำลองสามารถ เรียนรู้ แบบ reinforcement ผ่าน reward model จาก expert agent
- 8.4. ทดสอบ model ควบคุมที่ได้ บน อากาศยานจำลอง และ จริง

9. เอกสารอ้างอิง

- [1] International Civil Aviation Organization, Unmanned Aircraft Systems (UAS), Canada: International Civil Aviation Organization, 2554.
- [2] Jojo, "Types of drones," Circuitstoday, 2561. [ออนไลน์]. Available: <https://www.circuitstoday.com/types-of-drones>. [วันที่เข้าถึง 15 สิงหาคม 2563].
- [3] B. Hofmann-Wellenhof, K. Legat และ M. Wieser, Principles of Positioning and Guidance, NewYork , USA: Springer, 2550.
- [4] "Machine learning," Wikipedia, 12 สิงหาคม 2563. [ออนไลน์]. Available: https://en.wikipedia.org/wiki/Machine_learning. [วันที่เข้าถึง 15 สิงหาคม 2563].
- [5] P. Bajaj, "Reinforcement learning," GeeksforGeeks, 17 พฤษภาคม 2563. [ออนไลน์]. Available: <https://www.geeksforgeeks.org/what-is-reinforcement-learning/>. [วันที่เข้าถึง 15 สิงหาคม 2563].
- [6] S. Russell, "Learning agents for uncertain environments (extended abstract)," ใน *Proceedings of the eleventh annual conference on Computational learning theory*, Madison Wisconsin, 2541.
- [7] L. Erli, L. Yuan, L. Wei และ M. M. Q.-H., "Vision based autonomous gap-flying-through using the micro unmanned aerial vehicle," ใน *Canadian Conference on Electrical and Computer Engineering (CCECE)*, Canada, 2558.
- [8] F. Davide, M. Elias, F. Matthias และ S. Davide, "Aggressive Quadrotor Flight through Narrow Gaps with Onboard Sensing and Computing using Active Vision," ใน *IEEE International Conference on Robotics and Automation (ICRA)*, Singapore, 2560.
- [9] S. Waslander, G. Hoffmann, J. S. Jang และ C. Tomlin, "Multi-agent quadrotor testbed control design: integral sliding mode vs. reinforcement learning," ใน *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Canada, 2548.

- [10] J. Hwangbo, I. Sa, R. Siegwart และ M. Hutter, "Control of a Quadrotor With Reinforcement Learning," *IEEE Robotics and Automation Letters*, เล่มที่ 2, ปัญหาที่ 4, pp. 2096 - 2103, 2560.
- [11] P. Barker, "Ardupilot," Ardupilot, 1 มิถุนายน 2563. [ออนไลน์]. Available: <https://ardupilot.org/ardupilot/>. วันที่เข้าถึง 15 สิงหาคม 2563].