# The Development of a Stable Four-Way Stop Coordination Algorithm for Self-Driving Vehicles


Thomas Brefeld*, James Golding, Mitchell Pleune, Giuseppe DeRose, Nicholas Paul, CJ Chung**

Department of Math and Computer Science Lawrence Technological University, Southfield, MI 48075

* tbrefeld@LTU.edu ** cchung@LTU.edu



## Abstract

This project aims to test the viability of the designed algorithm to perform in a simulated intersection. The objective of the algorithm is to ensure that the cars do not collide and determine the right of way for each vehicle. The vehicles should not be blocked indefinitely while waiting for each other to pass the cross-road. The design of the algorithm will start with small-scale EduBots to simulate a small intersection, then the creation of a simulation of the full-size vehicles to test a large number of times. Finally, after the MTBF is within an acceptable value we will test the algorithm with the real cars. Preliminary results show that the proposed algorithm is stable and deadlock-free.


## Introduction & Goals

Autonomous vehicles, when entering an intersection, need to process a lot of information to proceed safely and efficiently. Vehicles arriving at an intersection need to determine which vehicle has the right of way and if the intersection is clear to be processed safely. They need to quickly determine which vehicle has the right of way and clear the intersection quickly, so the next vehicle can traverse the intersection. Without a correct determination of right of way, vehicles can find themselves waiting indefinitely as other vehicles proceed through the intersection. These vehicles also need to determine when it is safe to enter the intersection. Creating an algorithm to handle this situation is vital for autonomous vehicles to become viable. The use of three different platforms helps to ensure the effectiveness of the algorithm as issues become readily apparent. Goals of the project are:

- Develop an algorithm to coordinate two vehicles at a simplified and simulated four-way stop.
- The algorithm must be stable - never cause a collision.
- The algorithm must be dead-lock free - no indefinite waiting for each other.
- Verify the algorithm using simulation as well as small robotics platforms.
- Test the algorithm using LTU ACTors with GPS and LIDAR.

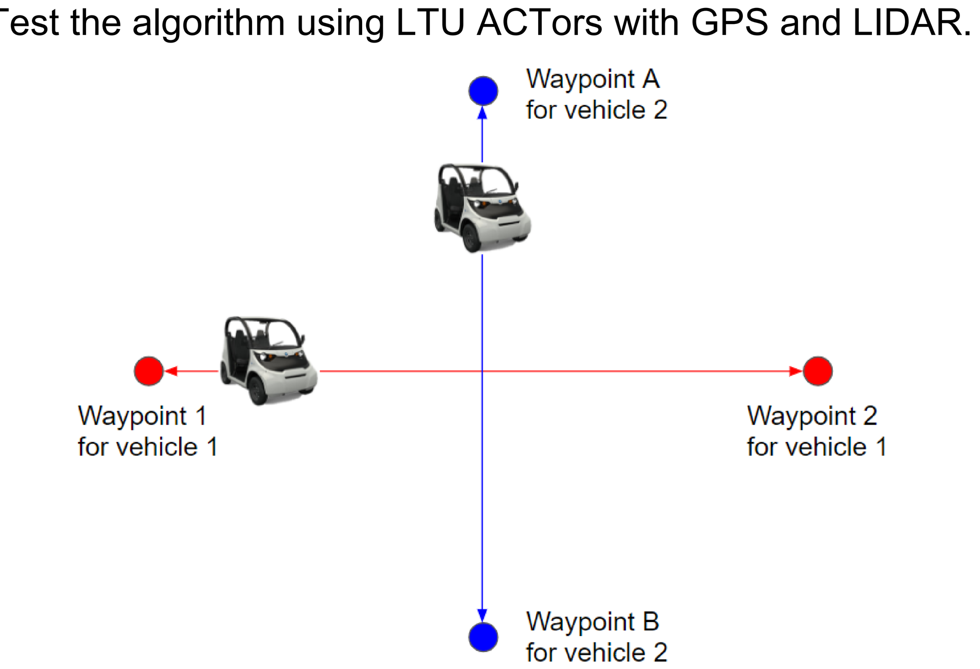


**Figure 1:** The conceptual model for the simplified four-way stop

## Algorithm Design

When two vehicles approach an intersection, the algorithm will behave based on the following rules. Vehicle one is using the algorithm; vehicle two is a secondary vehicle at the intersection (figure 1). Each step has priority over the next.

- If vehicle two directly ahead of vehicle one, vehicle one will yield to vehicle two.
- If vehicle one is within the intersection, it will continue through the intersection.
- If vehicle two approaches the intersection on the right-hand side of vehicle one, vehicle one will yield to vehicle two.
- If no vehicle is on the right-hand side of vehicle one, it will proceed through the intersection.

The algorithm in pseudo code is shown below.

```
Check Intersection(){
    Create integer variable countSafeStop set to 0
    Create integer variable countPriorityStop set to 0
    Create float variable range;

    For (every index of the lidar message) {
        The range is set to the distance of the object at the index.

        If (range falls between the max and min of possible ranges) {
            Create double variable x calculated  from index and range;
            Create double variable y calculated  from index and range;

            If (point(x,y) falls within the safe* box)
                Increment countSafeStop;
            If (point(x,y) falls within the Priority** box)
                Increment countPriorityStop
        }
    }
    If (countSafeStop plus countPriorityStop is larger than 0)
        Set car to wait for the other car
    else
        Set car to go through the intersection.
}
```

## Implementation & Test Method 1

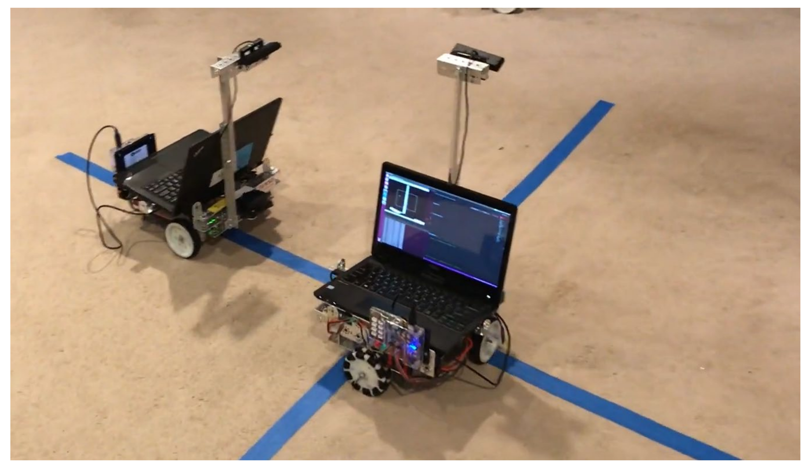

**Figure 2:** EduBots cooridition on cross-road made with blue tape

EduBots [3] are small-scale robots developed by the CS robotics lab for Robofest Vcc challenges and ROS classes using Tetrix kits and PRIZM controllers. We developed the initial code for the 2 EduBots to coordinate at a simplified cross-road using blue tape. The EduBots used vision sensors and 2D LIDARs for road and object detection, as shown below. A complete testing video can be found at [4].

## Implementation & Test Method 2

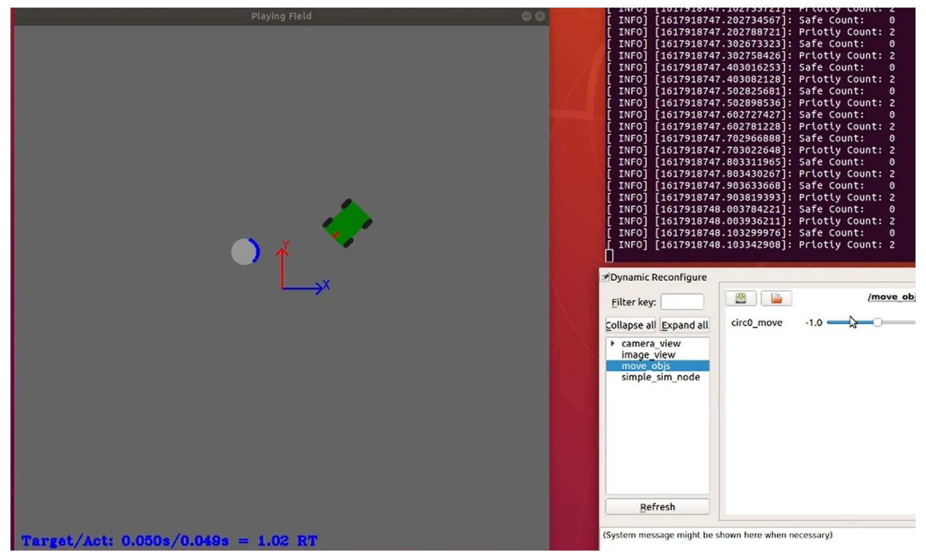

**Figure 3:** Simple Simulator using GPS and LIDAR

We created a simple simulator to simulate GPS coordinates and 2D LIDARs with a full-size Ackerman vehicle. Simulation was used to test a large number of scenarios. For simplicity, vehicle 2 is simulated as a circular object. This will allow us to focus on one vehicles performance with the algorithm. A demonstration can be found at [5].

## Implementation & Test Method 3

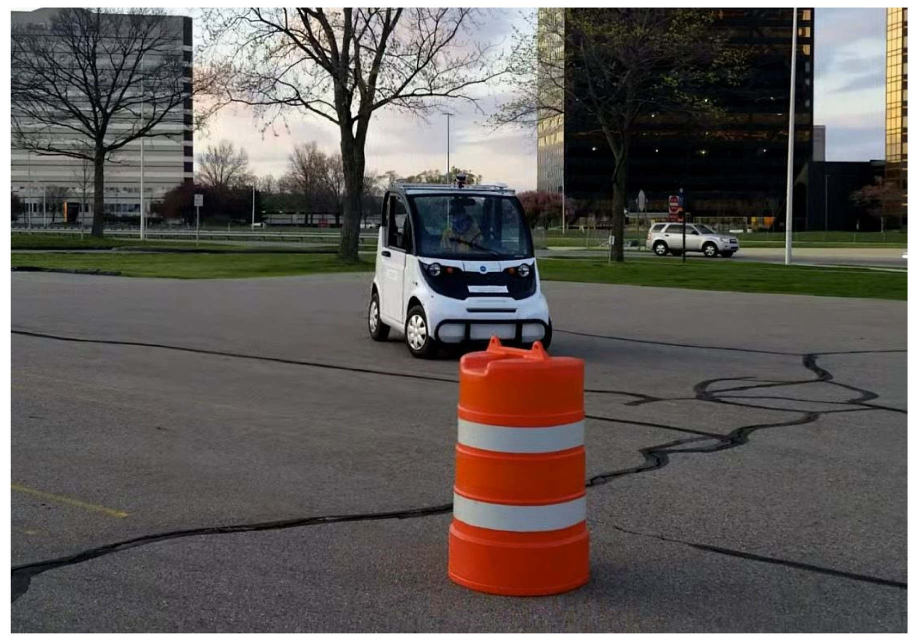

**Figure 4:** ACTor Vehicle Approaching a simulated vehicle as a construction barrel

The ACTor Vehicle is equipped with drive-by-wire system from Dataspeed Inc, LIDAR, and GPS. This allows the code used in the simple simulation to be used with little modification. The vehicle drove between two waypoints and had a construction cone placed where vehicles would enter the intersection. This allows real world testing without the chance of crashing. A demonstration can be found at [6].


## Acknowledgements

This project is supported by the US Army GVSC, GLS&T, NDIA-MI, Dataspeed Inc., DENSO, SoarTech, RealTime Technologies, and LTU reserve funds.


## Results

The three platforms were tested in succession, as each step would test the algorithm in a different environment. This allowed for observations to be made about the effectiveness of each step of the algorithm.

- Testing done on all 3 platforms supported that the algorithm is stable and effective at handling at an intersection with two vehicles.
- The algorithm was able to run indefinitely.
- Vehicles did not enter in a deadlock state.
- The algorithm is seen to be limited by only using GPS and LIDAR.

## Discussion

- The main limitation with the algorithm is the lack of object identification. The inclusion of computer vision would improve vehicle identification and subsequently improve the algorithms' efficiency. This is due to the fact that, the vehicle experienced false positives from objects near the intersection.
- The algorithm being deadlock free is critical to its success. if the vehicles are too cautious they will never enter the intersection. On the other hand if they are too aggressive they could cause a state where both vehicles determine it is not safe to continue.

## Future Work

- Testing with two ACTor Vehicles to simulate a 2 cars at an intersection.
- Testing with 4 vehicles. The current testing environment is a subset of a real 4 way stop environment.
- Improve algorithm efficiently. Minimize the sum of waiting times between all vehicles. (To find the optimized parameter values for the algorithm. The speed of each vehicle can be varying).
- Ensure that the algorithm is deadlock free.
- Integrate with Computer Vision. Ability to processes more complicated intersections.
- Inclusion of round-about coordination

## References


[1] Paul, N., Pleune, M., Chung, C., Faulkner, C., Warrick, B., Bleicher, S., A Practical, Modular, and Adaptable Autonomous Vehicle Research Platform, IEEE International Conference on Electro Information Technology 2018.

[2] Dhanasekar, R. et al. "Optimization of Four-Way Controlled Intersections with Autonomous and Human-Driven Vehicles." 2018 IEEE MIT Undergraduate Research Technology Conference (URTC) (2018): 1-4.

[3] Eric Liu, EduBot Construction Manual, 2019.

[4] EduBots Demo Video: https://youtu.be/lx-N6venD10

[5] Simple Sim Demo Video: https://youtu.be/oHLxOr8q7A4

[6] ACTor Vehicle Demo Video: https://youtu.be/effjmCoM1Ec