Intuitive Control Algorithm Development of 4WIS/4WID Using A SpaceMouse

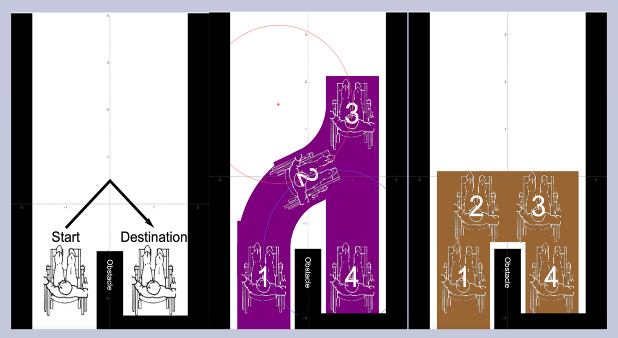
Tae wook (Tyler) Kim, Kent School'20 – kimt20@kent-school.edu

Lockheed Martin Physical Sciences Awards --- Finalist - Physical Sciences Senior High - CSF Medallion
Computer Science Awards --- Senior High Finalist - Medallion
Collins Aerospace Engineering Awards --- Finalist - Engineering High School - CSF Medallion
United Technologies Corporation Awards --- $500 in UTC Common Stock, Plaque, Backpack, and Annual Report

IEEE, Connecticut Section --- $250 Honors Award -Senior
Robotics And Beyond ---For Excellence in Engineering Design- $200 for Robotics And Beyond Academy in July
Barker Mohandas Awards for Transportation Research --- $200 Cash & Plaque for Excellence

Introduction

4WIS/4WID



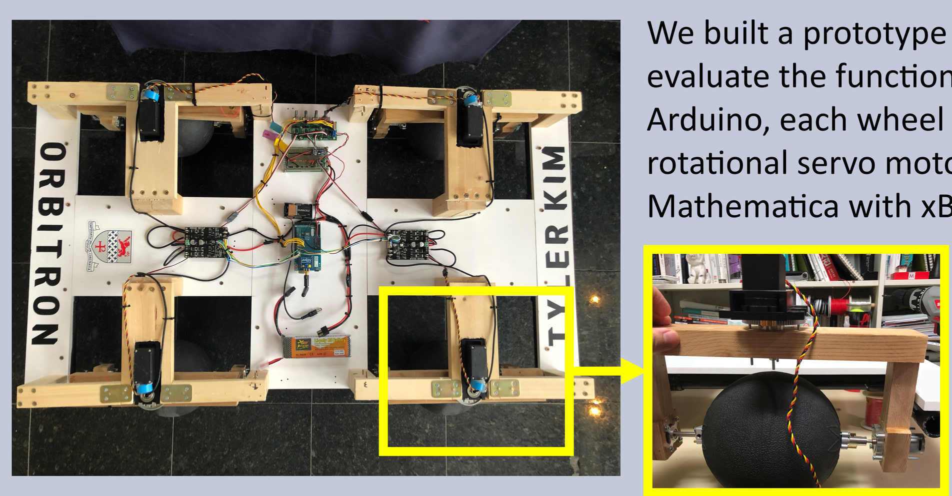
4WIS/4WIS (4 Wheel Independent Steering/Driving) is a steering system for a four-wheeled vehicle that allows for separate speed and direction controls for each wheel. These controls enable more versatile motion for vehicles that needs to navigate tight spaces, such as a wheelchair narrow subway cars.

In the left scenario, a 2 wheel steering system (2WS) requires much more space (colored purple) to get to the destination than a 4WIS/4WID (colored brown).

SpaceMouse

3dconnexion's SpaceMouse™ involves 3Dconnexion's patented 6 degrees of freedom (6DoF) sensor that is specifically designed for manipulating digital content or camera positions in industry-leading CAD applications. SpaceMouse converts an input from the user to a set of six numbers that range from −1 to +1. For example, one output created during the experiment was: { 0.234, −0.112, 0.066, −0.422, −0.208, 0.112 }.

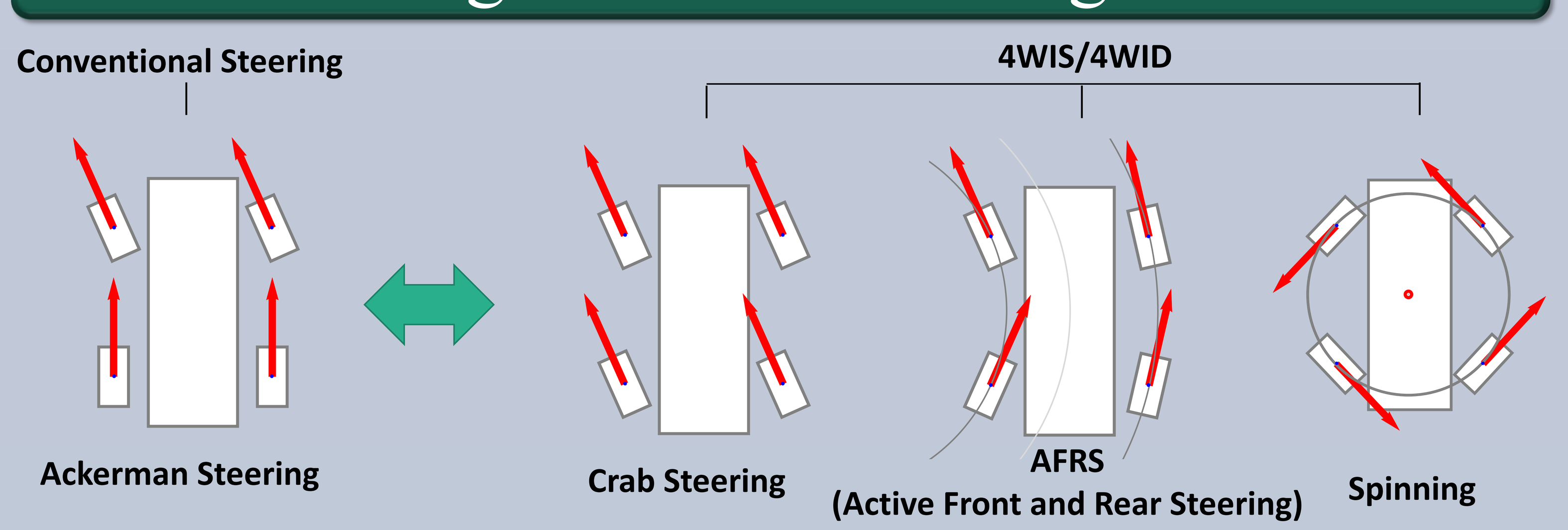
Prototype: Orbitron



We built a prototype that uses a 4WIS/4WID system to evaluate the functionality of our algorithm. Powered by Arduino, each wheel has a 12V DC motor and a multi-rotational servo motor which are controllable from Mathematica with xBee wireless modules.

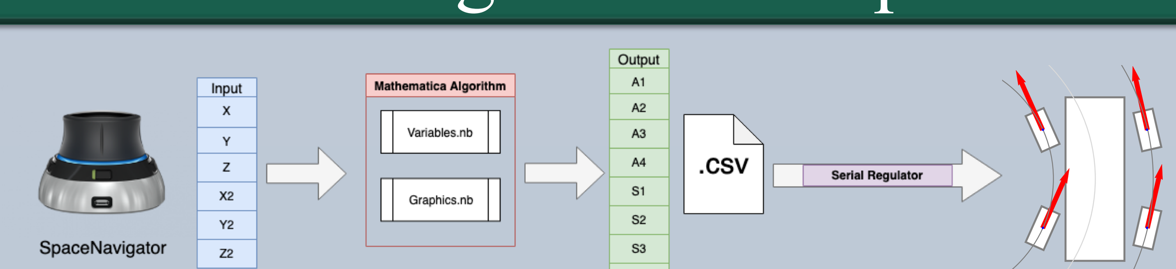
A PWM (Pulse Width Modulation) value between 0 and 255 controls the speed of each wheel, and a PWM value between 700 and 2300 controls the servo angle.

Steering Modes in Our Algorithm



While the conventional steering system only involves Ackerman Steering, 4WIS/4WID is capable of three different steering modes: AFRS, Crab Steering, and Spinning. Our algorithm supports all three steering modes and allows us to simultaneously control all four wheels without any danger of movement conflicts. We do so by computing both the wheel's direction and speed based on the wheel's angular velocity during a turn. This prevents any conflicts between the signals that can result in a vehicle slipping.

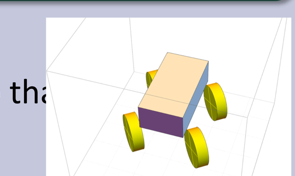
Algorithm Setup



Our algorithm's main job is to translate the six variables from the SpaceMouse to eight variables, each of which represents either angle or speed of a wheel. The algorithm is responsible for calculating variable transformations and records the sets of timestamped variables in a CSV file. We then use a serial regulator, a C# application that we've developed, to deliver the set of variables at a proper time without overfeeding data to the prototype.

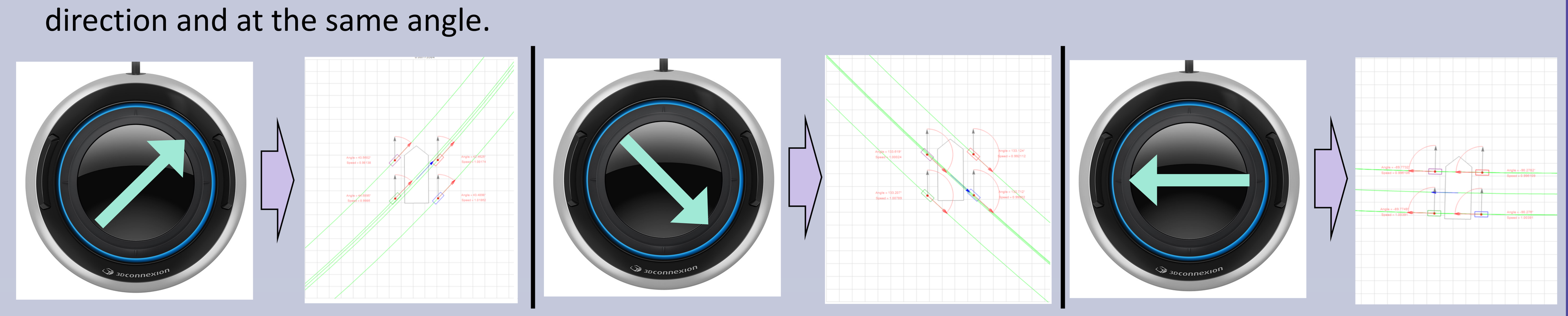
Development Procedure

In the beginning, we developed a 3D simulation with Mathematica to visualize how different steering modes actually work. Later on, we developed all simulations in 2D so that we calculate each wheel's speed and angle values more accurately.



Crab Steering

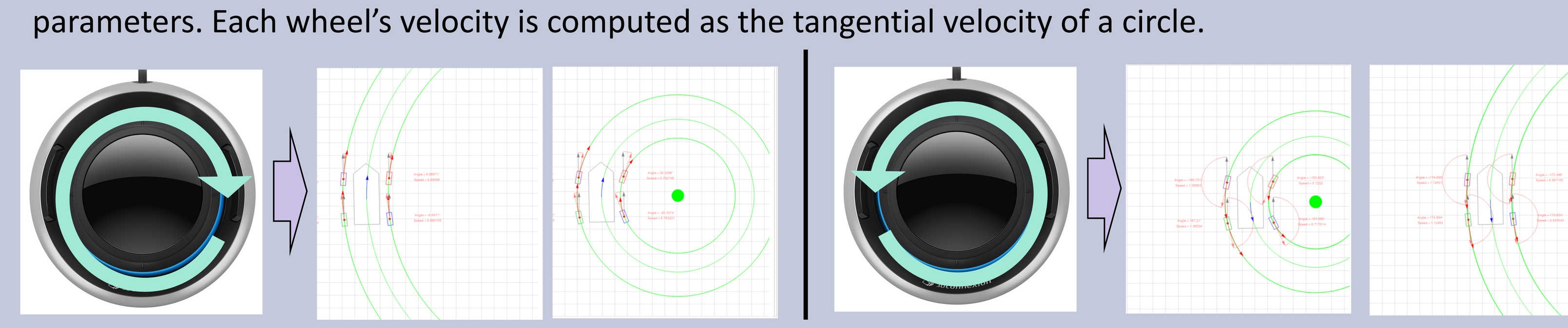
Crab steering is a special type of active four-wheel steering that operates by steering all wheels in the same direction and at the same angle.



Our algorithm uses the Crab steering mode whenever the user slides the SpaceMouse on the plane. In this specific example, when the mouse is facing upper right corner, all four wheels are angled in the direction the mouse is pointing at. All four wheels have the same speed which is computed as the tangential velocity of a circle with a very large radius.

AFRS

AFRS mechanism involves front and rear wheels turning independently for smaller turning radius and better cornering stability. In this steering mode, rear wheels change the way a vehicle turns based on driving parameters. Each wheel's velocity is computed as the tangential velocity of a circle.

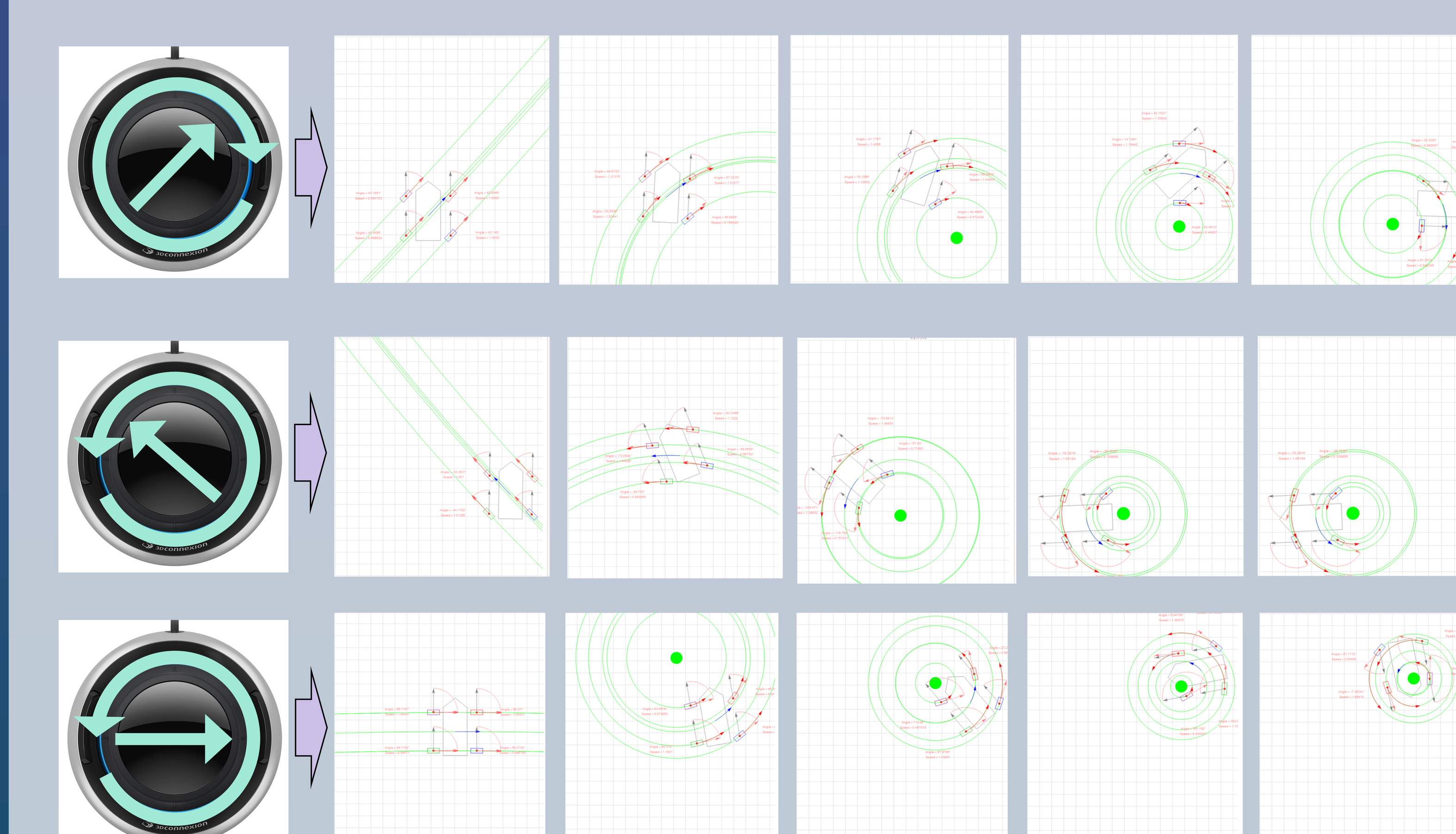


$$r \propto \frac{1}{\theta}$$

The twisting motion of the SpaceMouse is responsible for changing the radius of the vehicle's curvature. The core of our algorithm is that it considers every motion as a circular motion and computes each wheel's velocity and angle tangent to that circle. The radius of curvature (r) is calculated by the equation shown above, and the angle value (θ) is controlled with the twisting motion of the mouse.

When the mouse is twisted clockwise, θ is increased, resulting in a larger curvature radius. When the mouse is twisted counterclockwise, θ is decreased, resulting in a smaller curvature radius. Negative θ will make the circle move in the other side of the vehicle.

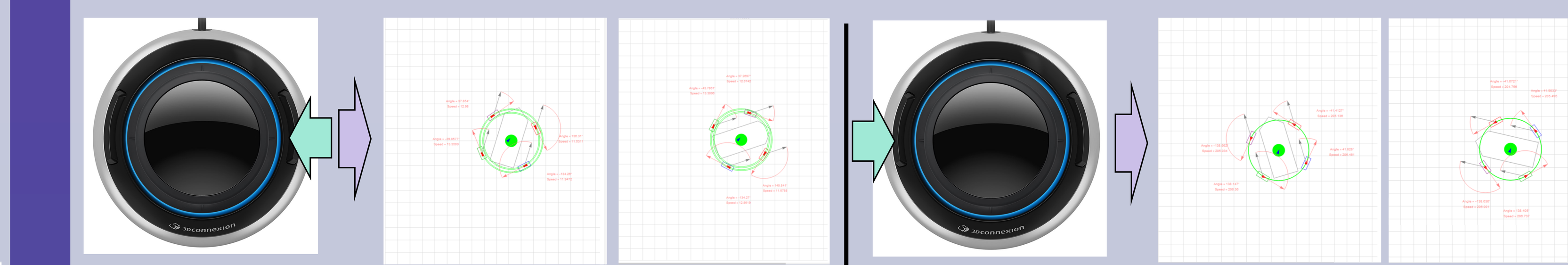
This principle also applies to straight motion in scenarios such as Crab Steering mode. Continuously twisting the mouse counterclockwise will make θ very small, leading the turning radius to be almost infinite. At this point, the vehicle's motion will be considered to be a straight motion.



In these example, straight motion and curving motion are accomplished simultaneously. When twisting and shifting the mouse occurs at the same time, the vehicle can perform more complicated motion, such as moving forward with a gradually decreasing turning radius.

Spinning

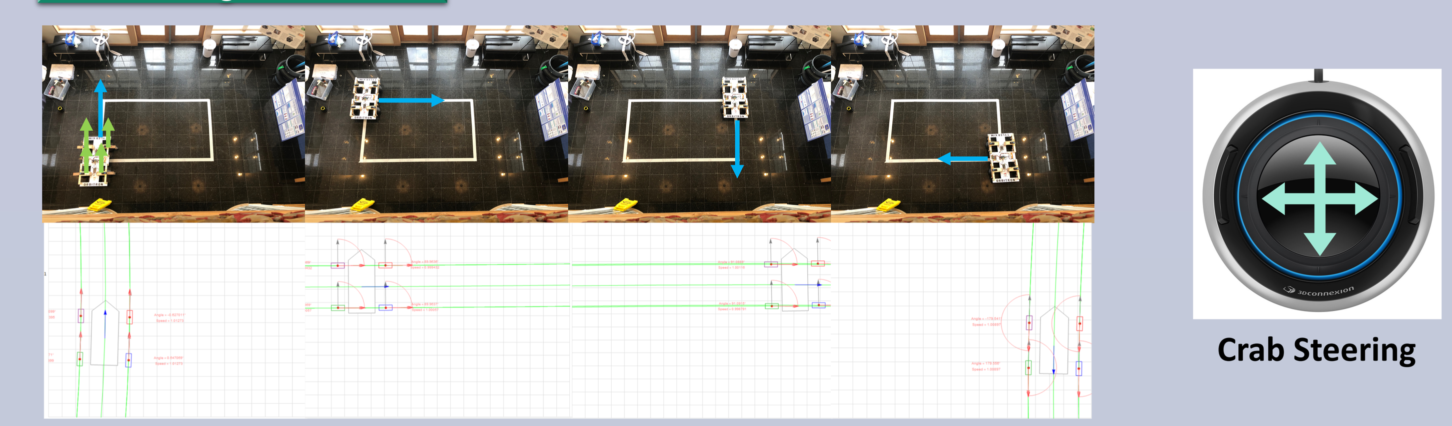
Also known as the Zero Turn mode, Spinning is the motion of a vehicle rotating with zero radius. It is easily accomplished by turning all wheels perpendicular to the center diagonal line and turning the wheels in the same direction.



When one of the two buttons on the side of the SpaceMouse is pressed, the vehicle rotates in the respective direction. As shown in the screenshots above, the vehicle turns in a clockwise direction when the right button is pressed, and the vehicle rotates in a counterclockwise direction when left button is pressed.

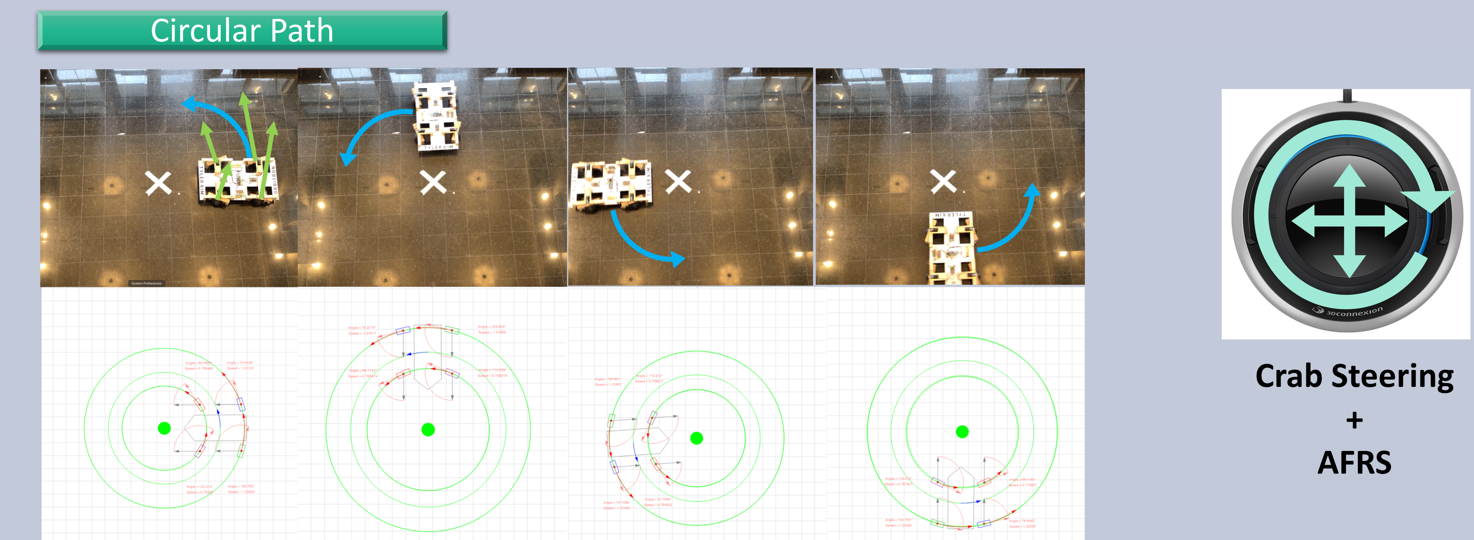
Results & Discussion

Rectangular Path



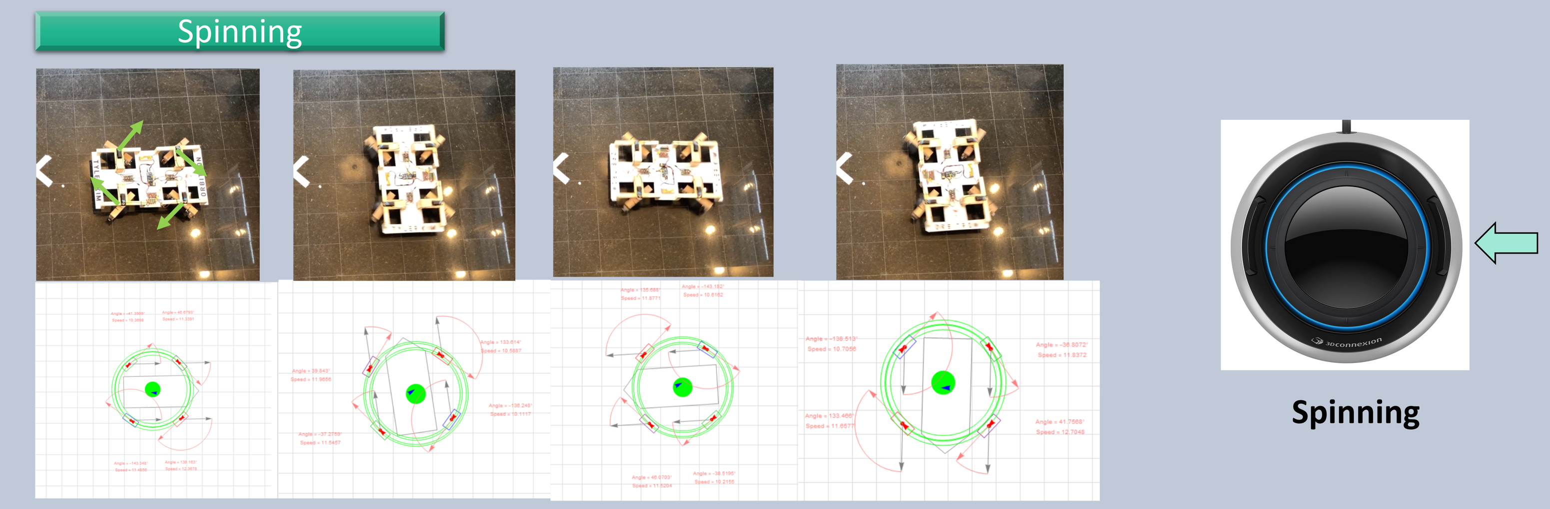
First experiment was to use the crab steering mode to make the vehicle move in a rectangular path without turning it. All four wheels always turned together in the corner of a rectangle and the speed was constant for all four wheels.

Circular Path



Second experiment was to use AFRS mode to drive the vehicle in a circular path, but facing the center at all times. Each wheel was fixed in an angle that is tangent to the turning direction, and different speed values were given for each wheels to account for the different distance from the center.

Spinning



Third experiment was to use the Spinning mode to rotate the vehicle with zero radius. It is similar to the circular path experiment, but the center of curvature was set to the center of the vehicle rather than outside of the vehicle. As a result, all wheels was set to be tangent to the turning circle and speed was constant while turning.

Conclusion

We developed an algorithm, implemented in Mathematica, that allows intuitive control of a 4WIS/4WID vehicle. With the assumption that all motions are circular motions, our algorithm abstracts away the complexity and allows for a full realization of the vehicle's capabilities without any special training on the operator's part. 4WIS/4WID system is a promising steering mechanism that can be applied in various vehicles/robots that require maneuverability in narrow spaces and more efficient control.