Hierarchical Rule-Base Reduction Fuzzy Control for Path Tracking Variable Linear Speed Differential Steer Vehicles

Samuel R. Dekhterman, William R. Norris, Ramavarapu Sreenivas, Dustin Nottage, Ahmet Soylemezoglu

***Abstract*—A novel waypoint navigation controller for a skid-steer vehicle is presented, where the controller is a multiple input-multiple output nonlinear angular velocity and linear speed controller. The membership functions of the fuzzy controller employed a trapezoidal structure with a completely symmetric rule-base. Notably, Hierarchical Rule-Base Reduction (HRBR) was incorporated into the controller. This entailed selecting inputs/outputs, determining the most globally influential inputs, generating a hierarchy relating inputs via a Fuzzy Relations Control Strategy (FRCS), and selecting only the rules corresponding to the hierarchy. Similar to a traditional fuzzy controller, the entire operating environment was covered. However, a rule for every possible combination of input states was no longer necessary. Thus, HRBR fuzzy controllers can increase both the number of inputs and their associated fidelity without the rule-base dramatically increasing. The stability analysis proves the asymptotic stability of the closed-loop controller-vehicle system. Additionally, a set of test courses were used to examine the effects of steering disturbance, phase lag, and overshoot as expressed in Root Mean Square Error (RMSE), Max Error (ME), and Course Completion Time (CCT). Outdoor experimental results for the controller's performance are contrasted with a comparable waypoint navigation controller, pure pursuit, and a simpler implementation with constant linear speed. The Variable Linear Speed Fuzzy was found to outperform the pure pursuit and Constant Linear Speed Fuzzy experimentally by 72% and 50% in RMSE, 71% and 40% in ME, and with moderately worse CCTs of 6.2% and 8.7%, on average. Thus, validating the controller's viability.**

***Index Terms*—Differential steer vehicle, fuzzy logic, pure pursuit, asymptotic stability, waypoint navigation.**

I. INTRODUCTION

WHEN designing path-tracking controllers for autonomous skid steer vehicles, considerations for position and velocity are paramount. Off-road environments pose a particular challenge to several traditional approaches.

A skid steer vehicle, when represented as a unicycle model [1], has no A matrix, while the B matrix directly maps the vehicle dynamics to the control action. This structure is directly at odds with model-based approaches, such as LQR, MPC, and H-infinity. These controllers use the vehicle's A matrix and control error signal to develop an optimal and/or robust target trajectory and a corresponding control action. [2] [3] [4].

In the case of frequency domain control on an ideal flat road, the stability and performance criteria can be guaranteed [5]. Although, better performance requires intense systems identification and tuning [6], [7]. In contrast, for off-road settings, the dynamics and thus, stability of these controllers vary greatly with velocity and terrain for a given set of gains [8].

Sliding mode controllers operate in a binary fashion by prescribing either a maximal or minimal control effort to drive the desired error state to a sliding manifold with a zero-error state [9]. Sliding mode controllers have been demonstrated to work to an extent on skid steer autonomous land vehicles [10], but the steady-state oscillations have kept them from widespread use.

Learning-based controllers, [11], that create a multi-variable, nonlinear, sensor-input control-output mapping have become popular for autonomous vehicle control applications [12]. However, vehicles can become unstable [13] when presented with disturbances outside of their training space, such as unexpected changes to the vehicle dynamics, ground contact physics, or unforeseen sensor measurements. As such, off-road environments pose a challenge.

Geometric controllers determine the optimal control action based on the manifold defined by the geometric-based constraints on the vehicle and its error state [14]. The most common form of geometric control for differential and skid steer vehicles is pure pursuit [15]. These controllers are robust, so long as the target point is far enough away from the vehicle to account for the maximum system time delays [16]. As such, pure pursuit was the baseline controller used in this paper.

The authors of this paper's previous work successfully solved this very control problem regarding a skid steer system using a fuzzy controller with Hierarchical Rule-Base Reduction (HRBR) [17]. In that paper, however, there was only one output to the proposed controller, angular velocity. As such, an additional output of linear speed was investigated in the hopes of seeing improved path-tracking performance.

Changing the linear velocity in path tracking is common in similar control efforts. Examples include:

In underwater environments, trajectory tracking using a

Samuel R Dekhterman is in the Department of Industrial and Enterprise Systems Engineering, University of Illinois Urbana-Champaign, IL, 61801, USA e-mail: srd2@illinois.edu.

William R Norris3 is a Clinical Associate Professor in the Department of Industrial and Enterprise Systems Engineering, University of Illinois Urbana-Champaign, IL, 61801, USA e-mail: wrnorris@illinois.edu.

Dustin Nottage and Ahmet Soylemezoglu are in the Construction Engineering Research Laboratory, U.S. Army Corps of Engineers Engineering Research and Development Center, IL, 61822, USA.

This research was supported by the U.S. Army Corps of Engineers Engineering Research and Development Center, Construction Engineering Research Laboratory.

Manuscript received April 19, 2005; revised August 26, 2015.

velocity-observer has been constructed for a line of unmanned underwater vehicles [18]. Similar work was seen using reinforcement learning in the context of car traffic [19]. In a non-highway heavy-duty ground vehicle setting, velocity control was implemented in [20] by solving an optimal control problem to reduce fuel usage.

Additionally, the HRBR method, used in [17] and further discussed in Section II, was employed to solve the ballooning rule-base issue associated with increasing the number of membership functions and linguistic variables in fuzzy systems. When paired with the use of trapezoidal membership functions and a symmetric rule-base due to HRBR, additional robustness and ease of implementation objectives were successfully achieved.

The remainder of the paper is organized as follows. Section II describes the design of the fuzzy logic controller and the HRBR. Section III proves the stability of the controller. Section IV discusses the outdoor experimental setup and results achieved using the skid steer Clearpath Jackal [21]. Lastly, Section VI concludes the paper.

II. CONTROL DESIGN

A. General Structure

In fuzzy logic control, crisp inputs, $z \in \mathcal{R}^n$, feed into the input linguistic variables I_n, which are then categorized into input linguistic values $A_{n,m}$ in a process called fuzzification [22]. Linguistic values describe their associated variable's performance with descriptors like "fast" and "slow". Furthermore, a level of membership between zero and one is assigned to a given linguistic value upon fuzzification. Membership functions, $\mu_{Z_{n,m}}$, determine what elements comprise the fuzzy set associated with each linguistic value [23].

After fuzzification, the levels of membership in the output values are determined using IF-THEN (t-norm) rules [24]. The antecedent of any given rule is the level of membership in the $A_{n,m}$ tied to I_n. While the consequent of any given rule is the level of membership in the output linguistic value $B_{n,m}$ of the output linguistic variable O_k.

Moreover, the AND (t-norm) and OR (t-conorm) operators can be used to integrate multiple linguistic values into a single antecedent/ consequent. The way that the AND, OR, and IF-THEN operators interact with the membership functions in the antecedents and consequents varies with implementation. Although, the two most common choices of t-norm/ t-conorm are the product/division and min/max operators [25].

An example of this structure is

$$\text{IF } I_1 \text{ is } A_{1,2} \text{ AND } I_2 \text{ is } A_{2,5} \text{ THEN } O_1 \text{ is } B_{1,1}$$

To calculate these crisp outputs, the Center of Mass (CoM) defuzzification, equation (1), is commonly used [25]. In equation (1), n is the total number of output membership functions, x_i is the amount of control output associated with output membership function i, and $\mu_c(x_i)$ is the degree of membership in output membership function i. Note, it is also common to defuzzify an aggregated membership function generated from the outputs of the rules.

$$x_{CoM} = \frac{\sum_{i=1}^{n} \mu_c(x_i)(x_i)}{\sum_{i=1}^{n} \mu_c(x_i)} \tag{1}$$

B. Proposed Controller

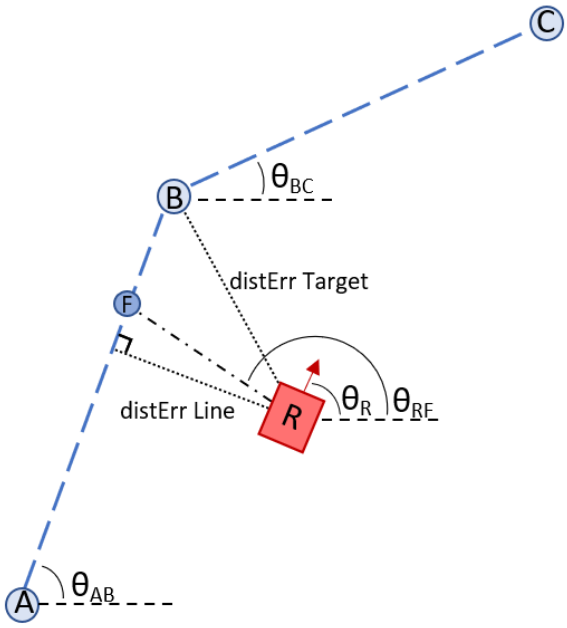

Fig. 1. Distances and Angles Relevant to the Robot

When initially developing the proposed fuzzy waypoint navigation controller, control objectives, outputs, and inputs were established. The control objectives sprung from the need to complete any given course with variably spaced waypoints, as quickly and accurately as possible. Thus, the corresponding objectives minimize the time required to complete a given course and the lateral distance error. The vehicle completed these objectives through movement. The movement command was decomposed into two outputs, namely an angular velocity and a linear speed.

The inputs to the control system were the path-related error functions that also followed from the control objectives such that the error functions were minimized when the vehicle was in a given state. Further, the span of all error states corresponded to all possible vehicle positions and orientations. For these errors, the relevant sections of the path at any given time are seen in Figure 1. In Figure 1, A is the waypoint the vehicle most recently passed, B is the current target waypoint, C is the next waypoint on the path after the current target waypoint, and R is the robot's current position.

Figure 1 also depicts the distance errors. Those distance errors were the "distErr Target", the distance from the vehicle to the target point, and "distErr Line", the lateral distance from the vehicle to the current path segment. The associated values and membership functions for both are presented on the pages that follow this one in Tables I and II.

In addition, Figure 1 depicts the angles used to generate the angular errors. In Figure 1, θ_R is the yaw angle of the vehicle, θ_{AB} is the orientation of the path segment formed by

TABLE I
DISTERR TARGET MEMBERSHIP FUNCTIONS

Linguistic Values	Membership Functions
close	*trap*(-100, -100, 0, 0.6418)
far	*trap*(0, 0.6418, 100, 100)

TABLE II
DISTERR LINE MEMBERSHIP FUNCTIONS

Linguistic Values	Membership Functions
far left	*trap*(-100, -100, -2, -1.6)
near left	*trap*(-2, -1.6, -1.398, -0.7981)
close left	*trap*(-1.3981, -0.7981, -0.3247, -0.01645)
zero	*trap*(-0.3247, -0.01645, 0.01645, 0.3247)
close right	*trap*(0.01645, 0.3247, 0.7981, 1.3981)
near right	*trap*(0.7981, 1.3981, 1.6, 2)
far right	*trap*(1.6, 2, 100, 100)

TABLE III
THETA NEAR MEMBERSHIP FUNCTIONS

Linguistic Values	Membership Functions
far left	*trap*(-3.142, -3.142, -1.46, -0.8556)
near left	*trap*(-1.46, -0.8556, -0.6118, -2.447e-05)
zero	*trap*(-0.6118, -2.447e-05, 2.447e-05, 0.6118)
near right	*trap*(2.447e-05, 0.6118, 0.8556, 1.46)
far right	*trap*(0.8556, 1.46, 3.142, 3.142)

waypoints A and B, θ_{BC} is the orientation of the path segment formed by waypoints B and C, F is a recovery point for when the vehicle is far from the path, and θ_{RF} is the orientation of the vector formed from points R and F.

From these path angles, the angular errors were calculated using equations (2)-(4); the angular errors were θ_{Near}, the angle between the vehicle's heading and the current path segment, $\theta_{Lookahead}$, the angle between the vehicle's heading and the next path segment, and θ_{Far}, the angle between the vehicle's heading and F. The angular errors' associated values and membership functions were presented on subsequent pages in Tables III-IV.

$$\theta_{near} = \theta_{AB} - \theta_R \tag{2}$$

$$\theta_{Lookahead} = \theta_{BC} - \theta_R \tag{3}$$

$$\theta_{Far} = \theta_{RF} - \theta_R \tag{4}$$

There were certain considerations regarding the set of input membership functions that further reflected the names of the linguistic values. Some of those considerations were: distErr Target could only output non-negative values as it measured the distance to the target waypoint. θ_{Near}, $\theta_{Lookahead}$, θ_{Far}, and distErr Line needed to output real values as they needed to distinguish if the orientation/position was to the right or left of the path. Another consideration was that the vehicle had to drive the real-valued inputs to and maintain some zero error state where no steering control action was needed. The purpose of the zero error state was to eliminate any potential bang-bang/oscillation in the control action when the errors were close to zero [26]. This prompted the creation of a zero linguistic value/ membership function for θ_{Near}, $\theta_{Lookahead}$, θ_{Far}, and "distErr Line". An assumption originating from the rule-base was a need for symmetry, similar to the rule-base presented in [27]. This assumption was used so the vehicle would act the same on both sides of the path as is typical of a steering control system. This came with the added benefit of simplifying membership function placement.

In addition, consideration was given to the form of the input membership functions. The fuzzy controller used trapezoidal input membership functions, as opposed to the traditional triangular or Gaussian membership functions. This choice is related to a human operator in [28]. The use of trapezoidal over triangular membership functions further reduced bang-bang and improved overall system stability as the flat regions provided a margin of acceptable error in the input, especially around the zero error region. Moreover, using a trapezoid allowed for some of the more desirable traits of a Gaussian function to be captured, [29], in a computationally efficient way. Furthermore, it was desirable to have shoulder functions that covered any remaining inputs. This ensured that the entire operating environment had an associated control effort.

DistErr Target, Table I, had two associated linguistic values, far and near. distErr Line, Table II, incorporated seven associated linguistic values to categorize how far away the vehicle was from the target trajectory: far left, near left, close left, zero, close right, near right, and far right. The remaining FRCVs, θ_{Near}, $\theta_{Lookahead}$, and θ_{Far}, Tables III - IV, all used similar linguistic values to qualify the orientation of the vehicle: far left, close/near left, zero, close/near right, and far right.

Having so many linguistic values with a Mamdani-type implementation and a product t-norm, while effective in testing, also introduced issues around the size of the rule-base. For a standard fuzzy controller, the rule-base associated with the above inputs would be exceedingly difficult to tune as a rule would be necessary for every combination of input variables and linguistic values (1750 rules). However, the fuzzy controller(s) incorporated all of the above inputs and linguistic values while keeping the rule-base fairly small (40 rules). This level of fidelity was achieved using the Fuzzy Relations Control Strategy (FRCS) introduced in [30].

Hierarchical Rule-Base Reduction (HRBR) starts with the establishment of an overarching control objective, inputs, and outputs for the system [30]. This control objective is then broken down into a set of sub-control objectives that enable the completion of the larger control objective. Each sub-control objective is assigned a Fuzzy Relations Control Variable (FRCV), i.e. an error tied to an input linguistic variable. These FRCVs are then put in a hierarchy of importance/influence. Next, a set of linguistic values are selected for each of the FRCVs. Each branch of the FRCV hierarchy then has a set of fuzzy values associated with it. Finally, a rule is assigned to each combination of linguistic values available in each of the complete branches [27].

Applying HRBR to the fuzzy controller was simple, as an overarching objective, inputs, and outputs, as discussed at the top of this section, were already established. From these, the sub-control objectives followed. Completing the path segments was unconditionally crucial for the entire duration of the control effort. Thus, distErr Target was a root node of the hierarchy. However, aligning the relative angle to the next

TABLE IV
THETA FAR MEMBERSHIP FUNCTIONS

Linguistic Values	Membership Functions
far left	*trap*(-3.142, -3.142, -2.45, -1.4)
close left	*trap*(-2.45, -1.4, -1.2, -0.2)
zero	*trap*(-1.2, -0.2, 0.2, 1.2)
close right	*trap*(0.2, 1.2, 1.4, 2.45)
far right	*trap*(1.4, 2.45, 3.142, 3.142)

TABLE V
THETA LOOKAHEAD MEMBERSHIP FUNCTIONS

Linguistic Values	Membership Functions
far left	*trap*(-3.142, -3.142, -2.391, -1.414)
close left	*trap*(-2.391, -1.414, -0.9409, -0.0034)
zero	*trap*(-0.9409, -0.0034, 0.0034, 0.9409)
close right	*trap*(0.0034, 0.9409, 1.414, 2.391)
far right	*trap*(1.414, 2.391, 3.142, 3.142)

path segment became necessary when the vehicle was close to completing the current path segment. As such, $\theta_{Lookahead}$ was a leaf node connected to distErr Target. Two additional leaf nodes of the hierarchy were present when the vehicle was far from completing the current path segment. Both leaf nodes were concerned with reducing the lateral distance to the path by minimizing distErr Line. The magnitude of the lateral distance from the path determined whether the vehicle needed to align itself toward the recovery point or parallel to the path. Thus, distErr Line was a child node of distErr Target and a parent node to θ_{Far} and θ_{Near} for the two respective cases. The hierarchy comprised of the single root node, single parent&child node, and three leaf nodes is summarized in Table VI. The linguistic values for the FRCVs can be found in Table I-V. With those in mind, the set of linguistic values associated with the case where the vehicle was close to the end of the path segment followed from distErr Target being near and $\theta_{Lookahead}$ being any of its linguistic values. Moreover, when the vehicle was far from the end of the path segment the values associated with the two remaining sets were distErr Target being far, distErr Line being far, and every value of θ_{Far} in the recovery case and distErr Target being far, distErr Line being zero/close/near, and every value of θ_{Near} in the close to the path case.

With the hierarchy established and the input linguistic values segmented along the branches of the hierarchy, the antecedents of the rules could be generated. For example, the leaf node for when the vehicle was close to the next path segment, distErr Target was near and $\theta_{Lookahead}$, had five associated rules, and thus needed five antecedents, as there were chosen to be five fuzzy values for $\theta_{Lookahead}$.

For the consequents of the rules, output linguistic values also needed to be assigned. For the angular velocity, its output was chosen to be in the set of real values in order to drive inputs to their zero error states. Conversely, the linear speed could only output non-negative values as the vehicle was only meant to drive forward. Thus, the angular velocity had the following nine linguistic values: left4, left3, left2, left1, zero, right1, right2, right3, and right4. Likewise, the linear speed had three linguistic values: slow, med[ium], and fast.

TABLE VI
SUMMARY OF APPLIED HIERARCHY (FRCS)

FRCVs	Metric Used for FRCS Classification		
	distErr Target Near	distErr Target Far	
		distErr Line Zero/ Close / Near	distErr Line Far
distErr Target	2	3	3
distErr Line		2	2
$\boldsymbol{\theta_{Lookahead}}$	1		
$\boldsymbol{\theta_{Near}}$		1	
$\boldsymbol{\theta_{Far}}$			1

Recalling that the rule-base was symmetric, the left half of the rule-base is finally presented in Table IX.

TABLE VII
PARAMETERS ASSOCIATED WITH THE STEERING MEMBERSHIP FUNCTIONS

Linguistic Values	Membership Functions
left 4	*tri*(-8.0704, -7.0704, -6.0704)
left 3	*tri*(-7.0561, -6.0561, -5.0561)
left 2	*tri*(-5.9934, -4.9934, -3.9934)
left 1	*tri*(-4.3981, -3.3981, -2.3981)
zero	*tri*(-1, 0, 1)
right 1	*tri*(2.3981, 3.3981, 4.3981)
right 2	*tri*(3.9934, 4.9934, 5.9934)
right 3	*tri*(5.0561, 6.0561, 7.0561)
right 4	*tri*(6.0704, 7.0704, 8.0704)

TABLE VIII
PARAMETERS ASSOCIATED WITH THE LINEAR SPEED MEMBERSHIP FUNCTIONS

Linguistic Values	Membership Functions
slow	*tri*(0, 0.2, 0.4)
med	*tri*(0.8, 1, 1.2)
fast	*tri*(1.8, 2, 2.2)

Having a complete rule-base, attention was turned toward defuzzifcation. Given that the level of membership in an output value could be determined from the crisp input values, input membership functions, and the rule-base, establishing the shapes of the output membership functions was necessary for the Mamdani-like fuzzy controller. Since bang-bang was only a concern for the inputs, the outputs used triangular as opposed to trapezoidal membership functions for simplicity. Table VII shows the membership functions for angular velocity, while Table VIII displays the membership functions for linear speed.

III. STABILITY ANALYSIS

A. Continuous System

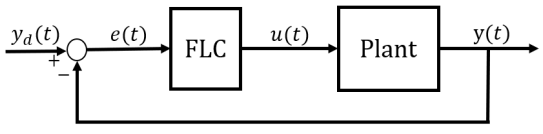

Fig. 2. Continuous Block Diagram I

TABLE IX
LEFT HALF OF THE SYMMETRIC FUZZY CONTROLLER RULE-BASE

IF					THEN	
distErr Target	**distErr Line**	$\boldsymbol{\theta_{Lookahead}}$	$\boldsymbol{\theta_{Far}}$	$\boldsymbol{\theta_{Near}}$	$\boldsymbol{\dot{\theta}_R}$	**Speed**
near		far left			right 4	slow
near		close left			right 2	med
near		zero			zero	fast
far	far left		far left		right 4	slow
far	far left		close left		right 1	fast
far	far left		zero		zero	fast
far	far left		close right		left 1	fast
far	far left		far right		left 4	slow
far	near left			far left	right 4	slow
far	near left			near left	right 3	med
far	near left			zero	right 2	med
far	near left			near right	right 1	fast
far	near left			far right	left 2	med
far	close left			far left	right 4	slow
far	close left			near left	right 2	med
far	close left			zero	right 1	fast
far	close left			near right	zero	fast
far	close left			far right	left 3	med
far	zero			far left	right 3	med
far	zero			near left	right 1	fast
far	zero			zero	zero	fast
far	zero			near right	left 1	fast
far	zero			far right	left 3	med

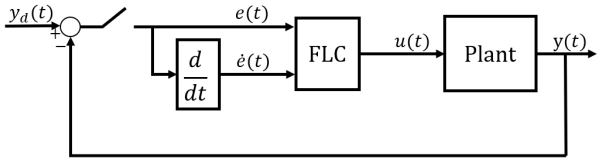

Fig. 3. Continuous Block Diagram II

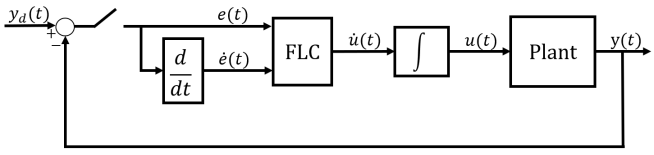

Fig. 4. Continuous Block Diagram III

$$e(t) = \begin{bmatrix} \theta_{near} \\ \theta_{far} \\ \theta_{lookahead} \\ \text{distError Line} \\ \text{distError Target} \end{bmatrix}, \qquad u(t) = \begin{bmatrix} \dot{\theta}_R \\ speed \end{bmatrix} \tag{5}$$

$$y(t) = \begin{bmatrix} \theta_R \\ \theta_R \\ \theta_R \\ -\text{distError Line} \\ -\text{distError Target} \end{bmatrix}, \qquad y_d(t) = \begin{bmatrix} \theta_{AB} \\ \theta_{RF} \\ \theta_{BC} \\ 0 \\ 0 \end{bmatrix} \tag{6}$$

In order to prove the stability of the proposed controller, Figure 2, an equivalent controller was constructed to match a popular passively stable fuzzy structure in the literature [31], [32], [33]. In Figure 2, the error input into the controller was $e(t)$, and the output of the controller was $u(t)$ as seen in equation (5). $u(t)$ then entered the plant and resulted in $y(t)$, which was combined with $y_d(t)$, equation (6), to make the new $e(t)$.

In Figure 3, an additional input of $\dot{e}(t)$ was included. Furthermore, each of the previous rule's antecedents were t-normed with the singular linguistic value of $\dot{e}(t)$. This singular linguistic value Dummy was assumed to have a membership value of one regardless of input. As such, the controllers in Figure 2 and Figure 3 are equivalent.

In Figure 4, the output of the controller was taken to be $\dot{u}(t)$, which was then integrated so the plant still took in $u(t)$. Thus, from the plant's perspective, the controllers in Figures 2, 3, and 4 are all equivalent.

Next, the $e(t)$ and $\dot{e}(t)$ terms were transformed into $e_1 = \bar{e}(t, \theta_{near}(t), \cdots, \text{distError Target}(t))$ and $e_2 = \dot{\bar{e}}(t, \dot{\theta}_{near}(t), \cdots, \text{distError Target}(t))$. e_1 was the representative singular input error generated using the elements composing $e(t)$, the calculation of which will be explored below. A similar simplification was seen for e_2 and u_1, with u_1 being the output trajectory of the vehicle. This output was made by combining the rule-base's two outputs of angular velocity and linear speed into a single Cartesian velocity.

Consequently, for each rule of the original fuzzy controller, each antecedent and consequent combination of linguistic values described above became its own linguistic value. For example, if distError Target was far, distError Line was far left, and θ_{far} was close left, Angular Velocity would be set to right1 and Linear Speed would be set to fast. Here, e_1 would have the linguistic value $TF, LFL, \theta_f CL$ (Target Far, Line Far Left, θ_{far} Close Left) whose level of membership would be determined by taking the t-norm of the level of membership in distError Target in Far, distError Line in Far Left, and θ_{far} in Close Left. e_2 would have a membership of 1 in Dummy and $\dot{u}(t)$ would have the linguistic value $\dot{AR}1, \dot{LF}$ (Derivative of Angular Right1, Derivative of Linear Fast).

Note, Definitions 1 - 3, and any other quoted text below in

this section was taken from [33] with:

$$\begin{aligned}\dot{e_1} &= e_2\\ u_1 &= \Phi(e_1, e_2)\\ -e_1 &= \bar{e}(t, -\theta_{near}(t), \cdots, -\text{ dE Line}(t),\ \text{dE Target}(t))\\ -e_2 &= \dot{\bar{e}}(t, -\dot{\theta}_{near}(t), \cdots, -\text{ dE }\dot{\text{L}}\text{ine}(t),\ \text{dE }\dot{\text{T}}\text{arget}(t))\end{aligned}$$

An additional core assumption of [33] that will initially be carried forward is that there are no delays in either the continuous or discrete systems. However, in most systems, such assumptions are not true. As such, there has been a great deal of work done showing both the passivity and robustness of time-delayed fuzzy systems [34], [35], [36].

"**Definition 1.** For any fuzzy controller with two inputs and one output, if the input-output non-linear mapping can be described by a continuous bounded Lipschitz function $\Phi(\cdot,\cdot)$ with the following properties, the fuzzy controller is called an SFC

1.$|\Phi(e_1, e_2)| \leq u_M$ and $M = \max_j x_j$;

2.$\Phi(0,0) = 0$ *(steady state condition)*;

3.$\Phi(e_1, e_2) = -\Phi(-e_1, -e_2)$ *(odd symmetry)*;

4.$\Phi(e_1, 0) = 0 \Rightarrow e_1 = 0$;

5.$\Phi(\cdot,\cdot)$ is a sectional function, in the sense that for every (e_1, e_2), there exist $\lambda', \gamma' > 0$ such that:

$$0 \leq e_1 \cdot (\Phi(e_1, e_2) - \Phi(0, e_2)) \leq \lambda' e_1^2$$
$$0 \leq e_2 \cdot (\Phi(e_1, e_2) - \Phi(e_1, 0)) \leq \gamma' e_2^2$$ "

Since every input that $e_2(k)$ could take had a membership value of one in Dummy, $\Phi(e_1, e_2) = \Phi(e_1, 0)$ and $\Phi(0, e_2) = 0$. As a result 5. simplifies to

$$0 \leq e_1 \cdot (\Phi(e_1, e_2) - 0) \leq \lambda' e_1^2 \tag{7}$$
$$0 \leq e_2 \cdot (\Phi(e_1, e_2) - \Phi(e_1, e_2)) \leq \gamma' e_2^2 \tag{8}$$

$$0 \leq e_1 \cdot \Phi(e_1, e_2) \leq \lambda' e_1^2 \tag{9}$$
$$0 \leq 0 \leq \gamma' e_2^2 \tag{10}$$

which is true for $\forall \lambda' \geq u_M/|e_1|$, $\forall \gamma' > 0$, and for every (e_1, e_2).

θ_{near} had membership functions $\mu_n(x_n)$, θ_{far} had membership functions $\mu_f(x_f)$, $\theta_{lookahead}$ had membership functions $\mu_l(x_l)$, distError Line had membership functions $\mu_L(x_L)$, and distError Target had membership functions $\mu_T(x_T)$ such that $x_n, x_f, x_l \in \{-\pi, \pi\}$ and $x_T, x_L \in \mathcal{R}$. Both the Aggregation operation on the output membership functions and the t-norm were a product of the inputs. Note that the following structure would be valid for any combination of aggregation operation and t-norm types.

For rules i = 1-10, θ_{far}, distError Line, and distError Target determined the membership in the output as follows

$$\mu_{1_{ia}} = \text{AND}\{\mu_{f_i}(x_f), \mu_{L_i}(x_L)\}$$
$$\mu_{1_i} = \text{AND}\{\mu_{1_{ia}}, \mu_{T_i}(x_T)\}$$

with the output membership for this set of rules determined by

$$\mu_1 = \text{Agg}\{\mu_{1_1}, \mu_{1_2}, \cdots, \mu_{1_{10}}\}$$

For rules i = 11-35, θ_{near}, distError Line, and distError Target determined the membership in the output as follows

$$\mu_{2_{ia}} = \text{AND}\{\mu_{n_i}(x_n), \mu_{L_i}(x_L)\}$$

$$\mu_{2_i} = \text{AND}\{\mu_{2_{ia}}, \mu_{T_i}(x_T)\}$$

with the output membership for this set of rules determined by

$$\mu_2 = \text{Agg}\{\mu_{2_{11}}, \mu_{2_{12}}, \cdots, \mu_{2_{35}}\}$$

For rules i = 36-40, $\theta_{lookahead}$ and distError Target determined the membership in the output as follows

$$\mu_{3_i} = \text{AND}\{\mu_{l_i}(x_l), \mu_{T_i}(x_T)\}$$

with the output membership for this set of rules determined by

$$\mu_3 = \text{Agg}\{\mu_{3_{36}}, \mu_{3_{37}}, \cdots, \mu_{3_{40}}\}$$

Aggregating everything together gave:

$$\mu = \text{Agg}\{\mu_1, \mu_2, \mu_3\}$$

The defuzzified output function was thus given by

$$\Phi = \frac{\sum_{j=1}^{N} (x_j * \mu(x_j))}{\sum_{j=1}^{N} \mu(x_j)} \tag{11}$$

where x_j was the Cartesian acceleration associated with rule j.

"Consider a continuous system in the state-space form as

$$\dot{x} = f(x, u), \quad x \in \mathcal{R}^n, u \in \mathcal{R}^m \tag{12}$$

$$y = h(x, u), \quad y \in \mathcal{R}^m \tag{13}$$

where $f(\cdot,\cdot)$ and $h(\cdot,\cdot)$ are smooth functions. $\cdots$

Definition 2. System (12) with a properly chosen output (13) is said to be passive with respect to the supply rate $s(u, y) = u^T y \in \mathcal{R}$, if there exists a positive definite function V with $V(0) = 0$, regarded as the storage function, such that the following inequality is satisfied for all $x(t_0)$.

$$V(x(t_f)) - V(x(t_0)) \leq \int_{t_0}^{t_f} s(u(\sigma), y(\sigma)) d\sigma, \quad \forall x, \forall u$$

$\cdots$ the fuzzy controller can be considered as a single-input single-output (SISO) non-linear system with internal dynamics, where e_2 is the input, u_1 is the output and e_1 is the state variable."

Using **Definition 2.** and the above variable declarations, the passivity of the controller was shown with:

$$\begin{aligned}\int_{t_0}^{t_f} e_2(\sigma)u_1(\sigma)d\sigma &= \int_0^t e_2(\tau)\Phi(e_1(\tau), e_2(\tau))d\tau \\ &= \int_0^t e_2(\tau)\Phi(e_1(\tau), 0)d\tau \\ &+ \int_0^t e_2(\tau)(\Phi(e_1(\tau), e_2(\tau)) \\ &- \Phi(e_1(\tau), 0))d\tau \\ &\geq \int_0^t e_2(\tau)\Phi(e_1(\tau), 0)d\tau \\ &= \int_0^t \dot{e}_1(\tau)\Phi(e_1(\tau), 0)d\tau \\ &= \int_{e_1(0)}^{e_1(t)} \Phi(e_1, 0)de_1 \\ &= V(e_1(t)) - V(e_1(0))\end{aligned}$$

Corollary 1. The memoryless function $y = h(t, u(t))$ where $h : [0, \infty) \times \mathcal{R} \rightarrow \mathcal{R}$ is said to be input-output strictly passive with respect to the supply rate $s(u(t), y(t)) = u(t)^T y(t) - \epsilon_1 u(t)^2 - \epsilon_2 y(t)^2$ for $\epsilon_1, \epsilon_2 > 0$, if $s(u(t), y(t)) > 0, \quad \forall t, \forall u(\cdot)$ [37] [32].

Using **Corollary 1.** with the plant, $y(t) = u(t)$, input-output strict passivity was determined via:

$$\begin{aligned}u(t)^T y(t) - \epsilon_1 u(t)^2 - \epsilon_2 y(t)^2 > 0 \\ u(t)^2 - \epsilon_1 u(t)^2 - \epsilon_2 u(t)^2 > \\ (1 - \epsilon_1 - \epsilon_2)u(t)^2 >\end{aligned}$$

Theorem 1. A sufficient condition for the asymptotic stability of an SFC, see **Definition 1.**, fuzzy control closed-loop is the input–output passivity of the controller and the input–output strict passivity of the plant such that the two in series have a well defined negative feedback connection [32] [33].

From the passivity of the controller and the input-output strict passivity of the plant, the sufficient condition of **Theorem 1.** for the asymptotic stability of the continuous fuzzy control closed-loop was met.

Including delays the state space form of the controller became:

$$\dot{x}(t) = f(x(t), u(t)) + f(x(t - T_n), u(t)) \tag{14}$$
$$y(t) = h(x(t), u(t)) + h(x(t - T_n), u(t)) \tag{15}$$

where $x(t) \in \mathcal{R}^n$, $u(t) \in \mathcal{R}^m$, $y(t) \in \mathcal{R}^m$, and T_n was some delay s.t. $0 \leq T_n < c$ for $c \in \mathcal{R}^+$. The supply rate from **Definition 2.** remained $s(u, y) = u^T y$. However, $\dot{x}(t)$ and $y(t)$ changed in accordance with equations (14) and (15) respectively. As a result, the passivity of the controller was expressed using,

$$\begin{aligned}&\int_{t_0}^{t_f} e_2(\sigma)u_1(\sigma)d\sigma \\ &= \int_0^t e_2(\tau) \\ &* (\Phi(e_1(\tau), e_2(\tau)) + \Phi(e_1(\tau - T_1), e_2(\tau)))d\tau \\ &= \int_0^t e_2(\tau)(\Phi(e_1(\tau), 0) + \Phi(e_1(\tau) - T_1, 0))d\tau \\ &+ \int_0^t e_2(\tau)(\Phi(e_1(\tau), e_2(\tau)) + \Phi(e_1(\tau - T_1), e_2(\tau)) \\ &- \Phi(e_1(\tau), 0) - \Phi(e_1(\tau - T_1), 0))d\tau \\ &\geq \int_0^t e_2(\tau)(\Phi(e_1(\tau), 0) + \Phi(e_1(\tau - T_1, 0))d\tau \\ &= \int_0^t (\dot{e}_1(\tau) + \dot{e}_1(\tau - T_1)) \\ &* (\Phi(e_1(\tau), 0) + \Phi(e_1(\tau - T_1), 0))d\tau \\ &= \int_0^t (\dot{e}_1(\tau)\Phi(e_1(\tau), 0))d\tau + \int_0^t (\dot{e}_1(\tau)\Phi(e_1(\tau - T_1), 0)))d\tau \\ &+ \int_0^t (\dot{e}_1(\tau - T_1)\Phi(e_1(\tau), 0))d\tau \\ &+ \int_0^t (\dot{e}_1(\tau - T_1)\Phi(e_1(\tau - T_1), 0)))d\tau \\ &= A_1 + A_2 + A_3 + A_4 \\ &\geq 4C_n \int_{e_1(0)}^{e_1(t)} \Phi(e_1, 0)de_1 \\ &= V(e_1(t)) - V(e_1(0))\end{aligned}$$

Such that $A_1 = \int_0^t (\dot{e}_1(\tau)\Phi(e_1(\tau), 0))d\tau$, $A_2 = \int_0^t (\dot{e}_1(\tau)\Phi(e_1(\tau - T_1), 0)))d\tau$, $A_3 = \int_0^t (\dot{e}_1(\tau - T_1)\Phi(e_1(\tau), 0))d\tau$, and $A_4 = \int_0^t (\dot{e}_1(\tau - T_1)\Phi(e_1(\tau - T_1), 0)))d\tau$. There were then four possibilities at any time t. Rather, $min(A_1, A_2, A_3, A_4) = A_1$. Then,

$$C_n = A_1/A_1 = 1$$

$min(A_1, A_2, A_3, A_4) = A_2$. Then,

$$C_n = \min(A_2/A_1) \text{ from 0 to t}$$

$min(A_1, A_2, A_3, A_4) = A_3$. Then,

$$C_n = \min(A_3/A_1) \text{ from 0 to t}$$

$min(A_1, A_2, A_3, A_4) = A_4$. Then,

$$C_n = A_2/A_1$$

The continuous plant could not integrate a delay as it was a memoryless function and thus could only act on the current input. As such, the input-out strict passivity of the plant shown in the delay-free case would carry over. Given the four possible C_ns, the sufficient condition of **Theorem 1.** for the asymptotic stability of the continuous fuzzy control closed-loop with delays was met.

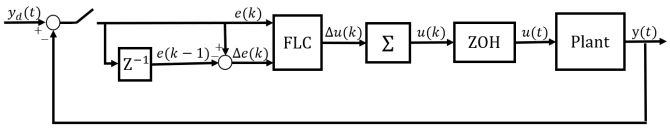

Fig. 5. Discrete Block Diagram III

B. Discrete System

Figure 5 is the discrete equivalents of Figures 4 with $u(k) = u(k-1)+\Delta u(k)$ and similarly $\Delta e(k) = e(k)-e(k-1)$. The transition from discrete to continuous time for this controller was achieved via a Zero Order Hold (ZOH), which held the controller output/ plant input for time Δt. Additionally, the assumptions from the continuous case carried over except for,

$$\Delta e_1 = e_2$$
$$-e_2 = \Delta\bar{e}(t, -\Delta\theta_{near}(t), \cdots, -\Delta \text{ dE Line}(t), \Delta \text{ dE Target}(t))$$

Now, examining a discrete version of passivity:

"Consider a discrete system in the state-space form as

$$x(k+1) = f(x,u), \quad x \in \mathcal{R}^n, u \in \mathcal{R}^m \tag{16}$$

$$y = h(x(k), u(k)), \quad y \in \mathcal{R}^m \tag{17}$$

where $f(\cdot,\cdot)$ and $h(\cdot,\cdot)$ are smooth functions. $\cdots$

Definition 3. System (16) with a properly chosen output (17) is said to be passive with respect to the supply rate $s(u(k), y(k)) = u(k)^T y(k) \in R \quad (k \geq 0)$, if there exists a positive definite function V with $V(0) = 0$, regarded as the storage function, such that the following inequality is satisfied for all $x(0)$, and $\forall k \in \mathcal{Z}^+ := 0, 1, 2, ...$

$$V(x(k+1)) - V(x(k)) \leq s(u(k), y(k)),$$
$$\forall x(k), \forall u(k), \forall k"$$

Applying the definition of passivity in the discrete case to the controller:

$$\begin{aligned} e_2(k)u_1(k) &= e_2(k)\Phi_1(e_1(k), e_2(k)) \\ &= e_2(k)\Phi(e_1(k), 0) \\ &+ e_2(k)(\Phi(e_1(k), e_2(k)) - \Phi(e_1(k), 0)) \\ &\geq e_2(k)\Phi(e_1(k), 0) \\ &= (e_1(k) - e_1(k-1))\Phi(e_1(k), 0) \\ &= e_1(k)\Phi(e_1(k), 0) - e_1(k-1)\Phi(e_1(k), 0) \\ &= V(e_1(k+1)) - V(e_1(k)) \end{aligned}$$

Corollary 2. The memoryless function $y = h(k, u(k))$ where $h : [0, \infty) \times \mathcal{R} \rightarrow \mathcal{R}$ is said to be input-output strictly passive with respect to the supply rate $s(u(k), y(k)) = u(k)^T y(k) - \epsilon_1 u(k)^2 - \epsilon_2 y(k)^2$ for $\epsilon_1, \epsilon_2 > 0$, if $s(u(k), y(k)) > 0, \quad \forall k, \forall u(\cdot)$ [37] [32].

Applying **Corollary 2.** with the discrete plant, $y(k) = u(k)$, input-output strict passivity was determined via:

$$\begin{aligned} u(k)^T y(k) - \epsilon_1 u(k)^2 - \epsilon_2 y(k)^2 > 0 \\ u(k)^2 - \epsilon_1 u(k)^2 - \epsilon_2 u(k)^2 > \\ (1 - \epsilon_1 - \epsilon_2) u(k)^2 > \end{aligned}$$

Hence, the sufficient condition of **Theorem 1.** for the asymptotic stability of the discrete fuzzy control closed-loop was met.

Including delays in the discrete system, the state space form of the controller became

$$x(k+1) = f(x(k), u(k)) + f(x(k - T_m), u(k)) \tag{18}$$
$$y(k) = h(x(k), u(k)) + h(x(k - T_m), u(k)) \tag{19}$$

where
$x(k) \in \mathcal{R}^n$, $u(k) \in \mathcal{R}^m$, $y(k) \in \mathcal{R}^m$, and T_m was some delay s.t. $0 \leq T_m < d$ for $d \in \mathcal{Z}^+$. The supply rate from **Definition 3.** remained $s(u, y) = u(k)^T y(k)$. However, $x(k+1)$ and $y(k)$ changed in accordance with equations (18) and (19) respectively. As a result, the passivity of the controller was demonstrated via,

$$\begin{aligned} &e_2(k)u_1(k) \\ &= e_2(k)(\Phi_1(e_1(k), e_2(k)) + \Phi_1(e_1(k - T_3), e_2(k))) \\ &= e_2(k)(\Phi(e_1(k), 0) + \Phi(e_1(k - T_3), 0)) \\ &+ e_2(k)(\Phi(e_1(k), e_2(k)) + \Phi(e_1(k - T_3), e_2(k)) \\ &- \Phi(e_1(k), 0) - \Phi(e_1(k - T_3), 0)) \\ &\geq e_2(k)(\Phi(e_1(k), 0) + \Phi(e_1(k - T_3), 0)) \\ &= ((e_1(k) - e_1(k-1) + (e_1(k - T_3) - e_1(k - T_3 - 1) \\ &* (\Phi(e_1(k), 0) + \Phi(e_1(k - T_3), 0)) \\ &= (e_1(k) + e_1(k - T_3) - e_1(k-1) - e_1(k - T_3 - 1)) \\ &* (\Phi(e_1(k), 0) + \Phi(e_1(k - T_3), 0)) \\ &= V(e_1(k+1)) - V(e_1(k)) \end{aligned}$$

The discrete plant could not integrate a delay as it was a memoryless function and thus could only act on the current input. As such, the input-out-strict passivity of the plant shown in the delay-free case would carry over. Thus, the sufficient condition of **Theorem 1.** for the asymptotic stability of the discrete fuzzy control closed-loop with delays was met.

IV. EXPERIMENTAL STUDIES

A. Test Courses

The methodology presented in [38] for validating controller performance was used for both the fuzzy and pure pursuit controllers. The approach validated controller performance by testing controllers under a set of path conditions that emphasized the effects of disturbance rejection, phase lag, overshoot, etc.

There are no figures in this section associated with blank versions of each of these test courses. However, each course is shown in Section V with plots of the vehicle performance overlayed.

Test Course 1 incorporated above minimum radius turns that were still relatively sharp. These above minimum radius turns allowed for a more accurate assessment of RMSE as a vehicle that could not turn in place could still have zero error. Accordingly, straightaways were paired with these above minimum radius turns to evaluate overshoot. This distinction is more important than it would initially seem, as squaring the

error term amplifies the errors associated with overshooting. Test Courses 1 also had both right and left-handed turns, thus verifying that the vehicle operated identically in both directions.

Test Course 2 was a figure-eight-like path. The at or above minimal turn radii of the circles were used to evaluate the ability of a controller to accommodate the associated steering disturbances as present in the Maximum Error (ME). Meanwhile, the curvature of this design was useful in evaluating path phase lag about the curves which could lead to distance error and thus higher Root Mean Squared Error (RMSE).

Test Course 3 incorporated both oscillatory turning that invoked phase lag similar to Test Course 2 and the above minimum radius turns paired with straightaways of Test Course 1. Thus, the course allowed for a more holistic examination of the controllers, given the factors associated with RMSE and ME discussed above.

B. Pure Pursuit

On each course, the fuzzy controller's performance was compared to that of the classical pure pursuit algorithm implemented in [39]. This choice was made as pure pursuit is one of the most commonly used waypoint navigation control algorithms and thus made for an ideal baseline controller. For that reason and those discussed in the Introduction, the controller used was the default MATLAB pure pursuit control block [40]. The geometric structure of the controller is presented in Figure 6 with: D being the lookahead distance from the vehicle to the path, α being the angle between the vehicle's orientation and D, and r being the radius of the curve that the vehicle R travels along. Moreover, r is computed using D and α with the equation $r = \frac{D}{2\sin(\alpha)}$.

The lookahead distance was chosen to be 0.5 meters by sweeping through potential lookahead values while converging to a straight line with an initial offset. The results of this experiment were run in simulation. The lookahead value of 0.5 meters was selected because it appeared to have reasonably small overshoots while also having acceptably fast rise times and settling distances.

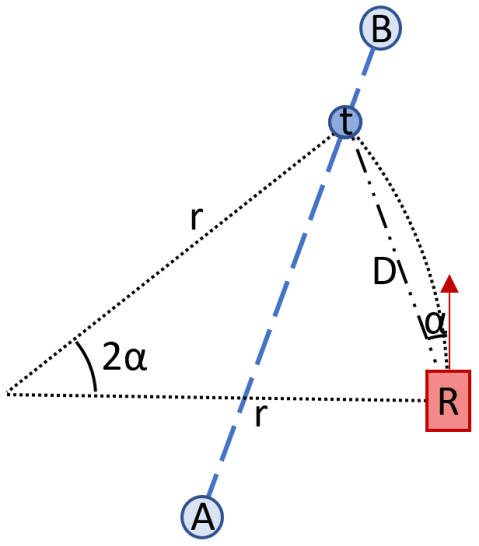

Fig. 6. Geometry Used by Pure Pursuit to Determinate Trajectories

C. Experimental Setup

The experimental results were acquired by running the Clearpath Jackal on a lightly worn concrete parking lot. An instance of the Robot Operating System (ROS) ran on the Clearpath Jackal. Using ROS allowed for both sensor data to be sent to and commands to be received from an external laptop. To enable such communication/control, the laptop ran MATLAB, the MATLAB ROS Toolbox [41], the MATLAB Fuzzy Logic Toolbox [42], and Simulink. In Simulink, a subscriber block subscribed to the position and orientation wheel encoder dead reckoning data from the ROS topic 'jackal_velocity_controller/odom'. Next, these inputs were converted into vehicle states and a target trajectory. Those were then fed into the fuzzy controller. The controller proceeded to determine the angular velocity and linear speed setpoints. Both the angular velocity and linear speed setpoints were then published to the ROS topic '/cmd_vel' using a publish block. At the same time, the x position, y position, angular velocity, and distErr Line were saved to a matrix in MATLAB.

Due to variability in the ROS time, the Course Completion Times (CCT) were calculated by taking the sum of each time step's distance traveled divided by its overall Cartesian velocity.

The Clearpath Jackal ran at an angular velocity setpoint ranging between -4 rad/s and 4 rad/s across all three courses. Meanwhile, the linear speed setpoint ranged between 0 m/s and 2 m/s on all courses. This gave the vehicle a theoretical minimum turn radius of 0.5 m at its top speed.

V. RESULTS

A tabulated set of results for all test courses can be found in Table X and Table XI. For brevity, exact RMSEs, MEs, and CCTs are not discussed as the Percent Change (PC) is most relevant when comparing controllers.

Test Course 1, as seen in Figure 7, and as discussed in Section IV-A, was designed to examine overshoot on turns when the vehicle's turn radius was no longer a factor. In Figure 7, the VLSF controller outperformed the pure pursuit controller in terms of overshoot for the first turn while seeing more overshoot than the pure pursuit on the second turn.

The associated angular control efforts are in Figure 8. The angular velocity control efforts of the VLSF were remarkably similar to the CLSF with a fraction of a second more time used to settle on the end waypoint. Meanwhile, the linear speed control effort, Figure 8, of the VLSF had dips in speed associated with each turn and the return to the endpoint.

Numerically, the controller saw an experimental RMSE PC of -77.6865%. Likewise, the ME PC came out to -74.2644%, and the CCT PC saw a larger increase than other courses to 7.59544% when compared to the pure pursuit.

Similar trends were seen against the CLSF. Experimentally, values of RMSE PC were at -70.8799%, ME PC at -68.6296%, and CCT PC at 11.242%.

TABLE X
CONTROLLER PERFORMANCE RESULTS: PURE PURSUIT AND VARIABLE LINEAR SPEED FUZZY

Course	Pure Pursuit			Variable Linear Speed Fuzzy			Percent Change		
	RMSE (m)	Max Error (m)	Time (s)	RMSE (m)	Max Error (m)	Time (s)	RMSE (%)	Max Error (%)	Time (%)
1	0.2922	0.5506	3.5021	0.0652	0.1417	3.7681	-77.687	-74.264	7.5954
2	1.6373	2.6778	14.3836	0.1367	0.2568	10.3042	-91.651	-90.410	N/A
3	0.0901	0.2765	30.2438	0.0736	0.1423	31.6682	-18.313	-48.535	4.7097

TABLE XI
CONTROLLER PERFORMANCE RESULTS: CONSTANT AND VARIABLE LINEAR SPEED FUZZY

Course	Constant Linear Speed Fuzzy			Variable Linear Speed Fuzzy			Percent Change		
	RMSE (m)	Max Error (m)	Time (s)	RMSE (m)	Max Error (m)	Time (s)	RMSE (%)	Max Error (%)	Time (%)
1	0.2239	0.4517	3.3873	0.0652	0.1417	3.768	-70.880	-68.630	11.242
2	0.2289	0.3039	9.4269	0.1367	0.2568	10.304	-40.280	-15.499	9.3064
3	0.1205	0.2235	29.9869	0.0736	0.1423	31.668	-38.921	-36.331	5.6068

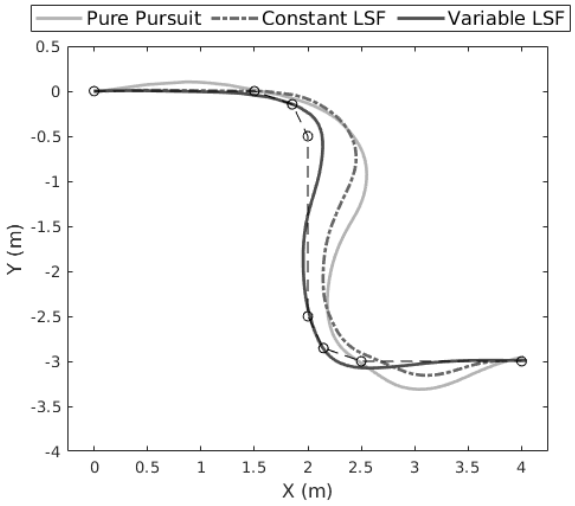

Fig. 7. Vehicle Trajectories on Test Course 1

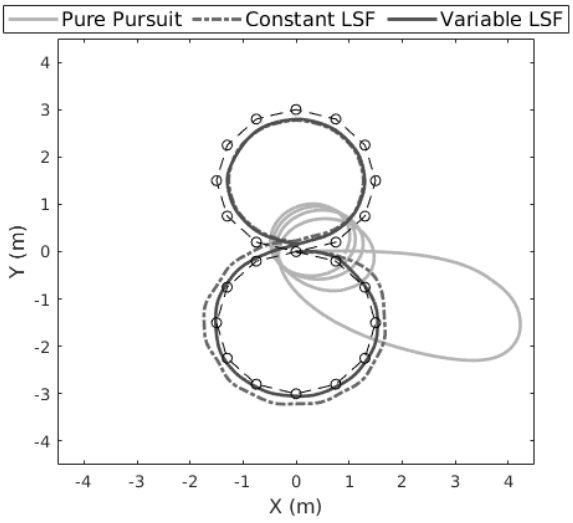

Fig. 9. Vehicle Trajectories on Test Course 2

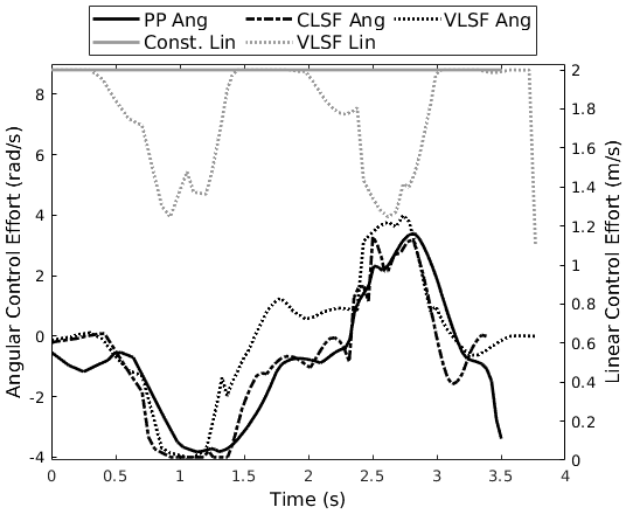

Fig. 8. Controller Angular Velocity (Bottom, Left Y-Axis) and Linear Speed (Top, Right Y-Axis) Setpoints on Test Course 1

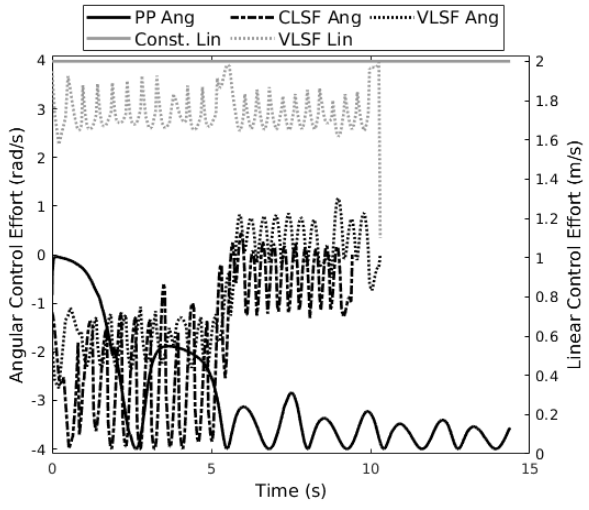

Fig. 10. Controller Angular Velocity (Bottom, Left Y-Axis) and Linear Speed (Top, Right Y-Axis) Setpoints on Test Course 2

Test Course 2, as depicted in Figure 9, and as discussed in Section IV-A, was designed to assess whether the controllers experienced non-minimum phase lag due to layout-related steering disturbance.

The associated angular control efforts can be seen in Figure 10 with the VLSF besting both the pure pursuit and CLSF in terms of overshoot. As in Test Course 1, the angular velocity control effort of the VLSF mimicked that of the constant linear speed fuzzy controller. However, in Figure 10, a moderate phase lag from the CLSF to the VLSF is present. This was due to the VLSF slowing down to account for the tight turns of the course.

In terms of metrics, the VLSF experimentally demonstrated a large improvement against the pure pursuit. The results were much larger than Test Course 1 with an RMSE PC of -91.6509% and an ME PC of -90.41%. The CCT PC was N/A as the experimental pure pursuit was stopped as it appeared to be in a loop and thus would be unable to complete the course in a timely manner.

The experimental errors were smaller when compared to the CLSF with an RMSE PC of -40.2796%, ME PC of -15.4985%, and CCT PC of 9.30635%.

A downside of this controller was visible in the poor experimental CCTs mentioned above; that being that on courses near the scale of the minimum turn radius of a vehicle, the VLSF controller would see large delays due to the controller's goal of improving accuracy at the cost of linear speed, predominately when turning. As such, this course presented the worst-case scenario for the controller as the VLSF was encouraged to reduce its linear speed for the duration of the test course.

Test Course 3, as seen in Figure 11, and as discussed in Section IV-A, was designed to examine both phase lag and overshoot, as well as if the two combine to create instability in the controller.

The associated angular control efforts can be seen in Figure 12 with the variable linear speed fuzzy controller outperforming the pure pursuit controller in terms of both phase lag and overshoot.

In terms of metrics, the variable linear speed fuzzy controller again had a favorable showing in experiment with an RMSE PC of -18.313%. The ME PC was also favorable at -48.5353%, and the CCT PC of 4.70973% remained at a small positive value.

The variable linear speed fuzzy controller similarly outperformed the constant linear speed fuzzy controller. The experimental values were still commendable with an RMSE PC at -38.9212%, ME PC at -36.3311%, and CCT PC at 5.60678%.

For the pure pursuit controller, the phase lag appeared to compound linearly on top of the overshoot as compared to the constant speed fuzzy controller. There was a similar degree of overshoot observed between the pure pursuit and CLSF. Although, as demonstrated in Figure 12, the VLSF's angular control effort was both lower on turns, 0s - 15s, and less volatile. As well, all three controllers remained stable despite the increased stress on the pure pursuit, and they all could be said to perform adequately.

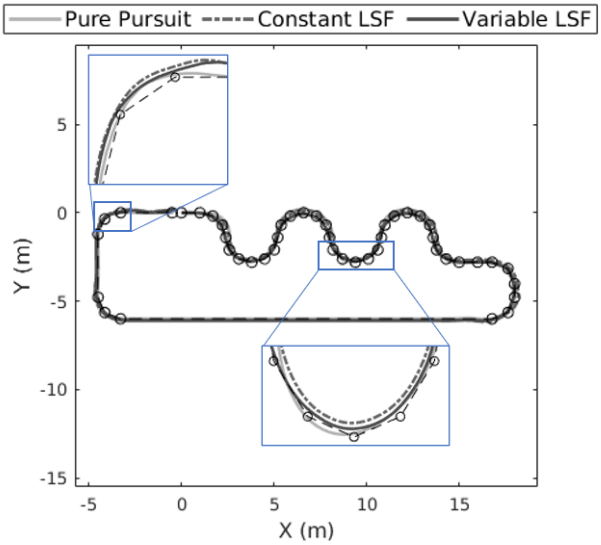

Fig. 11. Vehicle Trajectories on Test Course 3

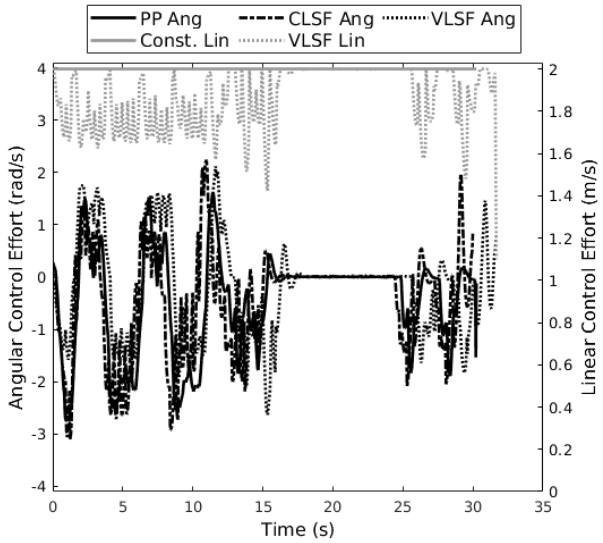

Fig. 12. Controller Angular Velocity (Bottom, Left Y-Axis) and Linear Speed (Top, Right Y-Axis) Setpoints on Test Course 3

VI. CONCLUSION

In this paper, a novel waypoint navigation control algorithm for a skid-steer vehicle that used an HRBR methodology with an FRCS was presented. This methodology was used to reduce the rule-base of the controller and, as a result, the algorithm's computational complexity, while simultaneously increasing the number of input membership functions that the controller could reasonably accept from two or three to five plus. This was a substantial improvement that better allowed the controller to emulate an expert human operator as it more accurately modeled the complex nonlinear decision-making that a human uses. The controller also used trapezoidal membership functions to mitigate the bang-bang effects that traditional controllers, especially fuzzy controllers, can encounter near the zero error state. The symmetric nature of the rule-base further simplified both the rule-base and tuning. Additionally, the asymptotic stability of the controller

and vehicle in series with a well-defined negative feedback connection was proven in both the continuous and discrete cases. As a result of all of the above, the Variable Linear Speed Fuzzy controller was seen in the Results section to have outperformed the pure pursuit and Constant Linear Speed Fuzzy controllers in terms of RMSE (avg. 72% and 50%), ME (avg. 71% and 40%), overshoot, and phase lag with minimal increases in CCT (6.2% and 8.7%). This is noteworthy as pure pursuit controllers are a common baseline for waypoint path following [43].

Future work should include examining optimal tuning methods while including obstacle avoidance protocols and dynamically adjusting the K values depending on the current speed of the vehicle.

REFERENCES

[1] T. Hellström, "Kinematics equations for differential drive and articulated steering," Umeå University, Department of Computing Science, Tech. Rep. 11.19, 2011.

[2] J. J. Vinodh Kumar. (2013) Robust lqr controller design for stabilizing and trajectory tracking of inverted pendulum. [Online]. Available: https://doi.org/10.1016/j.proeng.2013.09.088.

[3] T. Tashiro, "Vehicle steering control with mpc for target trajectory tracking of autonomous reverse parking," in *2013 IEEE International Conference on Control Applications (CCA)*, 2013, pp. 247–251.

[4] M. Khelfi and A. Abdessameud, "Robust h-infinity trajectory tracking controller for a 6 d.o.f puma 560 robot manipulator," in *Proceedings of IEEE International Electric Machines and Drives Conference, IEMDC 2007*, Oct. 2007, pp. 88–94.

[5] J. Pentzer, S. Brennan, and K. Reichard, "The use of unicycle robot control strategies for skid-steer robots through the icr kinematic mapping," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014, pp. 3201–3206.

[6] N. Majid, Z. Mohamed, and A. Basri, "Velocity control of a unicycle type of mobile robot using optimal pid controller," *Jurnal Teknologi*, vol. 78, 07 2016.

[7] T. Q. Khai and Y. . Ryoo, "Design of adaptive kinematic controller using radial basis function neural network for trajectory tracking control of differential-drive mobile robot," *International Journal of Fuzzy Logic and Intelligent Systems*, vol. 19, no. 4, pp. 349–359, 2019.

[8] S. F. Campbell, "Steering control of an autonomous ground vehicle with application to the darpa urban challenge," Master's thesis, Massachusetts Institute of Technology, Cambridge, MA, September 2007.

[9] K. D. Young, V. I. Utkin, and U. Ozguner, "A control engineer's guide to sliding mode control," *IEEE Transactions on Control Systems Technology*, vol. 7, no. 3, pp. 328–342, 1999.

[10] Jong-Min Yang and Jong-Hwan Kim, "Sliding mode control for trajectory tracking of nonholonomic wheeled mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 3, pp. 578–587, 1999.

[11] S. Kuutti, R. Bowden, Y. Jin, P. Barber, and S. Fallah, "A survey of deep learning applications to autonomous vehicle control," 2019.

[12] S. Ferrari and R. F. Stengel, "Smooth function approximation using neural networks," *IEEE Transactions on Neural Networks*, vol. 16, no. 1, pp. 24–38, 2005.

[13] J. Yi, D. Song, J. Zhang, and Z. Goodwin, "Adaptive trajectory tracking control of skid-steered mobile robots," in *Proc. IEEE Int. Conf. Robot. Autom. '07*, 05 2007, pp. 2605 – 2610.

[14] Y. Sachkov, "Introduction to geometric control," 2019.

[15] M. Samuel, M. Hussein, and M. Mohamad, "A review of some pure-pursuit based path tracking techniques for control of autonomous vehicle," *International Journal of Computer Applications*, vol. 135, pp. 35–38, 2016.

[16] K. N. Murphy, "Analysis of robotic vehicle steering and controller delay," in *Yuh, Volume 5, Robotics and Manufacturing, pg 631, Wailea*, 1994, pp. 631–636.

[17] S. R. Dekhterman, M. Cichon, W. R. Norris, D. Nottage, and A. Soylemezoglu, "Hierarchical rule-base reduction fuzzy control for constant velocity path tracking of a differential steer vehicle," 2023.

[18] H. Liang, Y. Fu, J. Gao, and H. Cao, "Finite-time velocity-observed based adaptive output-feedback trajectory tracking formation control for underactuated unmanned underwater vehicles with prescribed transient performance." *Ocean Engineering*, vol. 233, May 2021.

[19] Z. Xu, S. Liu, Z. Wu, X. Chen, K. Zeng, K. Zheng, and H. Su, "Patrol: A velocity control framework for autonomous vehicle via spatial-temporal reinforcement learning," in *CIKM '21*, Gold Coast, Australia, Nov. 1–5, 2021, pp. 2271–2280.

[20] M. Held, O. Flärdh, and J. Mårtensson, "Experimental evaluation of a look-ahead controller for a heavy-duty vehicle with varying velocity demands," *Control Engineering Practice*, vol. 114, Mar. 2021.

[21] "Jackal 2020 data sheet," Clearpath Robotics, Kitchener, Canada.

[22] Y. Shan, W. Yang, C. Chen, J. Zhou, L. Zheng, and B. Li, "Cf-pursuit: A pursuit method with a clothoid fitting and a fuzzy controller for autonomous vehicles," *International Journal of Advanced Robotic Systems*, vol. 12, no. 9, p. 134, 2015.

[23] S. Hanumanthakari, G. C. Sekhar, H. S. Behera, J. Nayak, B. Naik, and D. Pelusi, *Comparative Analysis of Different Types of Membership Functions for Fuzzy Logic Controller in Direct Torque Control of Induction Motor*, 1st ed., ser. Intelligent Computing in Control and Communication. Singapore :: Springer,, 2021, vol. 702.

[24] E. Mamdani and S. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller," *International Journal of Man-Machine Studies*, vol. 7, no. 1, pp. 1–13, 1975-1.

[25] I. Eker and Y. Torun, "Fuzzy logic control to be conventional method," *Energy conversion and management*, vol. 47, no. 4, pp. 377–394, 2006.

[26] R. Bellman, I. Glicksberg, and O. Gross, "On the "bang-bang" control problem," *Quarterly of Applied Mathematics*, vol. 14, no. 1, pp. 11–18, 1956.

[27] W. R. Norris, Q. Zhang, and R. S. Sreenivas, "Rule-base reduction for a fuzzy human operator performance model," *Applied engineering in agriculture*, vol. 22, no. 4, pp. 611–618, 2006.

[28] W. Norris, Q. Zhang, R. Sreenivas, and J. Lopez-Dominguez, "A design tool for operator–adaptive steering controllers." *Transactions of the ASAE*, vol. 46, no. 3, p. 883, May/Jun. 2003.

[29] S. H. Khairuddin, M. H. Hasan, and M. A. Hashmani, "Integration of cluster centers and gaussian distributions in fuzzy c-means for the construction of trapezoidal membership function," *Mathematics and Statistics*, vol. 8, no. 5, pp. 559–565, 2020.

[30] W. R. Norris, "A design framework for qualitative human-in-the-loop system development," Ph.D. dissertation, Univ. of Illinois, Urbana-Champaign, Jul. 2001. [Online]. Available: http://www.ece.udel.edu/~qli

[31] G. Calcev, "Passivity result for fuzzy control systems," in *Proceedings of the IEEE Conference on Decision and Control*, vol. 3, Kobe, Japan, Dec. 1996, pp. 2727–2728.

[32] ——, "Some remarks on the stability of mamdani fuzzy control systems," *IEEE Transactions on Fuzzy Systems*, vol. 6, no. 3, pp. 436–442, 1998.

[33] C. Xu and Y. C. Shin, "A stable hierarchical fuzzy control design for certain non-linear systems based on input-output passivity theory," *Control and Intelligent Systems*, vol. 37, no. 2, pp. 103–113, 2009.

[34] C. Li, H. Zhang, and X. Liao, "Passivity and passification of fuzzy systems with time delays," *Computers and Mathematics with Applications*, vol. 52, no. 6-7, pp. 1067–1078, 2006.

[35] T. Lei, Q. Song, and Z. Zhao, "Further result on passivity for discrete-time stochastic t-s fuzzy systems with time-varying delays," *Discrete Dynamics in Nature and Society*, vol. 2014, 2014.

[36] J. Liang, Z. Wang, and X. Liu, "Robust passivity and passification of stochastic fuzzy time-delay systems," *Information Sciences*, vol. 180, no. 9, pp. 1725–1737, 2010.

[37] H. K. Khalil, *Nonlinear systems 3rd Edition*. Pearson, 2002, vol. 115.

[38] W. Norris and A. Patterson, "System-level testing and evaluation plan for field robots: A tutorial with test course layouts," *Robotics*, vol. 8, p. 83, 09 2019.

[39] M. Samuel, M. Hussein, and M. Mohamad, "Implementation of the pure pursuit path tracking algorithm," *International Journal of Computer Applications*, vol. 135, pp. 35–38, 2016.

[40] I. The MathWorks, *Pure Pursuit*, Natick, Massachusetts, United State, 2021. [Online]. Available: https://www.mathworks.com/help/nav/ref/purepursuit.html

[41] ——, *ROS Toolbox*, Natick, Massachusetts, United State, 2021. [Online]. Available: https://www.mathworks.com/help/ros/index.html

[42] ——, *Fuzzy Logic Toolbox*, Natick, Massachusetts, United State, 2021. [Online]. Available: https://www.mathworks.com/help/fuzzy/index.html

[43] H. Ohta, N. Akai, E. Takeuchi, S. Kato, and M. Edahiro, "Pure pursuit revisited: Field testing of autonomous vehicles in urban areas," in *2016 IEEE 4th International Conference on Cyber-Physical Systems, Networks, and Applications (CPSNA)*, 2016, pp. 7–12.