2017 IEEE Conference on Control Technology and Applications (CCTA)
August 27-30, 2017. Kohala Coast, Hawai'i, USA

Modeling, Design and Control of Low-Cost Differential-Drive Robotic Ground Vehicles: Part I - Single Vehicle Study

Armando A. Rodriguez, Karan Puttannaiah, ZhenYu Lin, Jesus Aldaco, Zhichao Li, Xianglong Lu, Kaustav Mondal, Shubham D. Sonawani, Nikhilesh Ravishankar, Nirangkush Das and Pragyan A. Pradhan

Abstract— **Toward the ambitious long-term goal of a fleet of cooperating Flexible Autonomous Machines operating in an uncertain Environment (FAME), this two part paper addresses several critical modeling, design and control objectives for ground vehicles. One central objective was to show how off-the-shelf (low-cost) remote-control (RC) toy vehicles can be converted into intelligent multi-capability robotic-platforms for conducting FAME research. This was done for 13 differential drive RC vehicles called Thunder Tumbler (DDT^2). Each DDT^2-vehicle was augmented with a suite of sensor-computing-communication devices in order to provide a substantive suite of capabilities. Part I of this two part paper, focusing on a single vehicle, examines the associated non-holonomic dynamical model (including motor dynamics) for the DDT^2 vehicle under consideration. We shed light on how vehicle coupling impacts control design a topic not well addressed within the robotics community. Because our vehicle exhibits little coupling, we are able to use classical decentralized control to design a wheel speed inner-loop controller. This controller is used for all of our outer-loop control modes: (speed-direction) cruise control along a curve, planar Cartesian (x,y) stabilization and minimum-time optimal-control around an oval race track. Empirically collected data is shown to agree well with simulation results. Reasons for observed differences are provided. Within Part II, focus is on control laws for the coordination of multiple vehicles. In short, many capabilities that are critical for reaching the longer-term FAME goal are demonstrated within this two part paper.**

I. Introduction and Overview

As the evolution of electromechanical and computing technologies continue to accelerate, the possible applications continue to grow. This accelerated growth is observed within the robotics research and hobbyist communities [1–8]. New technologies (e.g. Arduino, Raspberry Pi with compatible interfaces, software and actuators/sensors) now permit young hobbyists and researchers to perform very complicated tasks with little effort - tasks that would have taken many person-hours of cumbersome development (e.g. hardware interfacing and/or tedious programming) just a few years ago. Within this paper, current off-the-shelf technologies (e.g. Arduino, Raspberry Pi, commercially available RC cars) are exploited to develop low-cost ground vehicles that can be used for multi-vehicle robotics research. Potential applications can include: remote sensing, mapping, intelligence gathering, intelligence-surveillance-reconnaissance (ISR), search, rescue and much more. It is this vast application arena as well as the ongoing accelerating technological revolution that continues to fuel robotic vehicle research.

This paper is the first part of a two part paper (Parts I and II [9]). Within this paper, the focus is on single vehicle applications. For the vehicle type considered (i.e. differential-drive), both kinematic and dynamical (planar $x-y-\theta$) models are examined. Here, differential-drive means that the speed of each of the rear wheels are controlled independently by separate dc motors. Such a vehicle is said to be non-holonomic; i.e. the 2 (x,y) or (v,θ) controllable degrees of freedom is less than the 3 total (x,y,θ) degrees of freedom. This fundamentally limits the ability of a single continuous (non-switching) control law to "precisely park the vehicle" [10–13]. Despite this, it is shown how continuous-time linear control theory can be used to develop suitable control laws that are essential for achieving various critical capabilities.

Within this paper, we discuss the non-holonomic nature of the vehicle as well as nonlinear controllability (based on Lie brackets), classic loss of linear controllability and the need for a discontinuous control law in order to solve the "classic (x,y,θ) parking problem" [10–13]. While the associated kinematic model can be used to gain great insight, a dynamical model with motor dynamics is considered in order to better understand fundamental limitations when implementing control laws in hardware. Empirical model parameter estimation as well as parameter uncertainty are also addressed. Suitable nonlinear/linear-models are used to develop inner/outer-loop control laws [5, 14].

While some coupling ideas are addressed, they are not deeply explored. When the vehicle aspect ratio (width over length) of the vehicle is large (fat vehicle), the coupling is large and negative; i.e. when the right wheel rotates, the left wheel rotates in the opposite direction. When the vehicle aspect ratio is small (thin vehicle), the coupling is large and positive. These ideas are used in order to show that an aspect ratio of $\frac{1}{\sqrt{5}}$ results in zero-coupling in the transfer function matrix from motor voltages to wheel angular velocities [5]. To our knowledge, this concept of a zero-coupling aspect ratio is not well known in the literature. This is important because it provides a foundation for vehicle as well as control design. Our inner-loop controller relies on the ideas within [5, 14]. It is used for all outer-loop control modes. The following specific outer-loop control laws are considered, developed and demonstrated: (1)(speed-direction) cruise con-

Dr.A.A.Rodriguez is a Prof. in School of Elect., Computer & Energy Eng. (ECEE), Arizona State University (ASU), Tempe, AZ, aar@asu.edu; K.Puttannaiah, K.Mondal are Ph.D. students in ECEE, ASU; S.D.Sonawani, N.Ravishankar, N.Das, P.A.Pradhan are MS students in ECEE, ASU; Z.Lin is Ph.D student in Dept. of Elect. Eng., Univ. Maryland; J.Aldaco is with Delphi, IN. Z.Li is a Ph.D. student in Mech. & Aerospace Eng., Univ. of California, San Diego; X.Lu is with Changan US R&D Center, MI. This work has been supported, in part, by NSF Grant No. 1565177. Any opinion, findings, and conclusions or recommendations expressed in this material are those of the authors(s) and do not necessarily reflect the views of the NSF.

trol along a curve (both proportional and proportional-plus-derivative: P & PD), (2)planar Cartesian (x, y) stabilization (via proportional gain control) and (3)minimum-time optimal-control around an oval race track. Our outer-loop designs rely on the ideas within [10–13]. Empirically collected data is shown to agree well with simulation results. Reasons for observed differences are provided. **Contributions.** Within this paper, focused on single vehicle applications, answers are provided to the following fundamental questions:

- How can off-the-shelf (low-cost) remote-control toy vehicles be converted into intelligent multi-capability robotic-platforms for conducting *FAME* research?
- What are the fundamental limitations imposed by off-the- shelf hardware? How to account for the difference between simulation and empirical results? What dynamic and parametric uncertainty should be considered?
- How can continuous-time linear control theory be used to develop suitable critical capabilities such as (1) wheel/vehicle velocity inner-loop control, (2) speed-direction cruise control along a curve, (3) planar Cartesian (x, y) stabilization and (4) minimum-time optimal-control around an oval race track?
- How can the vehicle be designed so that coupling is small and simple decentralized control suffices? When is a multivariable controller essential? Are there simple guidelines that to answer these? The answer, here, is yes and vehicle aspect ratio coupling ideas we present are fairly unknown within robotics community [5].

The remainder of the paper is organized as follows: Section II describes the hardware on each vehicle. Section III summarizes critical modeling concepts. Section IV summarizes the inner-loop control law that was implemented. Section V summarizes each of the outer-loop control laws. Section VI provides the summary and directions for future research.

II. Hardware: Enhanced Differential-Drive Thunder-Tumbler (DDT^2) Vehicle

One central objective of the MS theses [1–4] was to show how to take off-the-shelf (low-cost) remote control "toy" vehicles and convert them into "intelligent" multi-capability robotic platforms that can be used for conducting robotics/*FAME* research. In this section, we will describe the hardware on each of our low-cost DDT^2 robots. An enhanced Thunder Tumbler is shown in Fig. 1.

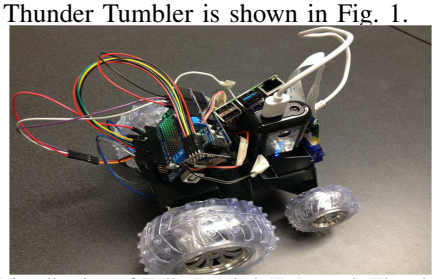

Fig. 1. Visualization of Fully-Loaded (Enhanced) Thunder Tumbler

Each original (un-enhanced) Thunder Tumbler is a $10 "toy" vehicle which can be found at Walmart, Target, etc. It is a differential-drive vehicle with two dc motors - one on the left wheel; another on the right wheel. Each DDT^2 vehicle was augmented/enhanced to provide a suite of substantive capabilities. While each enhanced DDT^2 costs less than $225, it offers the capability of commercially available vehicles costing over $500. DDT^2 vehicles were augmented with the following components: (1) Arduino Uno microcontroller board to provide low-level (inner-loop) control and some high-level (outer-loop) control, (2) Arduino Motor Shield to provide PWM control signals to vehicle's dc motors, (3) Magnetic wheel encoders to provide dead-reckoning velocity, angular velocity and (x, y) displacement information, (4) Inertia Measurement Unit (IMU), (5) Raspberry Pi 2/3 to provide heavy-duty computing power for more demanding (e.g. vision based) outer-loop computations, (6) Raspberry 5MP camera module (7) Ultrasonic Distance Sensor, (8) spread-spectrum 2.4GHz Nrf24L01 transceivers to permit reliable remote control, and (9) Edimax Wifi adapter and router to facilitate a Wifi-based video link to computer.

Use of Arduino and Raspberry Pi Computing Units. Arduino was used to implement our (ω_r, ω_l) inner-loop control law (see Section IV). Raspberry Pi was used to implement the M^{-1} matrix (see Section III) to translate (v_{ref}, ω_{ref}) commands into $(\omega_{r_{ref}}, \omega_{l_{ref}})$ commands for the inner-loop. It thus follows that the Pi was used to implement some inner-loop functionality. Arduino was used to implement outer-loop functions as well as low-level inner-loop feedback control functions. Outer-loops 1-6 are discussed within Sec. V. (1) Raspberry Pi was used to implement (v, θ) Cruise Control System. (2) Arduino was used in Planar (x, y) Cartesian Stabilization. (3) To implement solution to Minimum Time around a Track, Raspberry Pi and Arduino were used when camera-based and IMU-based approach were taken respectively. (4) $(\Delta x, \theta)$ Separation-Direction Control along a line was implemented using Arduino to maintain constant separation from an object. (5) Obstacle Avoidance involves switching between Cartesian Stabilization and Separation-Direction Control. Arduino was used to implement this. (6) For Platooning, the Raspberry Pi was used for leader vehicle cruise control, whereas Arduino in followers for maintain constant separation from leader.

Video Demos. Video demonstrations can be found at: http://aar.faculty.asu.edu/research/mosart/mosart-fame.html.

III. Modeling of DDT^2 Vehicle

Many mobile robots use a so-called differential-drive mechanism, which involves two rear wheels that are independently controlled via torque-generating dc motors. Within this paper, the motors are assumed to be identical in order to simplify the presentation. In practice, motor differences must be accounted for. This, in part, is addressed by the motor control laws being employed. Within this section, we examine two models for our DDT^2 robot: (1) a kinematic model and (2) a two-input two-output (TITO) linear time invariant (LTI) model that was presented within [14]. It shall also serve as the basis for developing inner-loop control designs for our DDT^2 vehicles. **Kinematic Model.** Figure 2 can be used to understand the kinematics of a DDT^2 ground robot [15]. The point that robot rotates about at a given

instant in time is called the instantaneous center of curvature (ICC) [15]. If (x, y) denotes the planar inertial coordinate of the robot and θ denotes the direction of the robot's longitudinal body axis with respect to the x-axis, then we obtain the following nonlinear kinematic model:

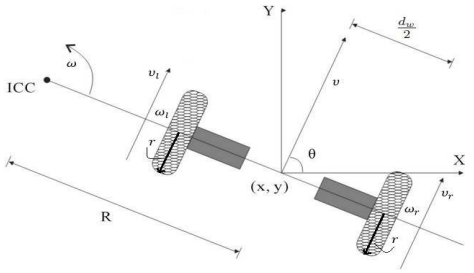

Fig. 2. Visualization of Differential-Drive Mobile Robot

$$\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{\theta}\end{bmatrix} = \begin{bmatrix}\cos\theta\\ \sin\theta\\ 0\end{bmatrix} v + \begin{bmatrix}0\\ 0\\ 1\end{bmatrix} \omega \quad (1)$$

$v = \sqrt{\dot{x}^2 + \dot{y}^2}$ denotes the translational velocity and $\omega = \dot{\theta}$ denotes its angular velocity. Within the above very simple (and intuitive) model, v can ω can be thought of as inputs (or controls). This is not intuitive - especially to a controls person because v and ω cannot be instantaneously generated because of real-world mass-inertia effects. In practice, v and ω are generated by applying voltages to the two dc motors.

Controllability. The above 4-state system is controllable. When we include position and angle as states, the linear model becomes uncontrollable. Brockett's theorem [10, 11] gives necessary conditions for smooth feedback stabilizability. Necessary and sufficient conditions for controllability using feedback linearizaiton [12] are derived using Lie brackets. Within [13], the authors show that a (smooth) linear control law involving longitudinal distance to the target can be used to get arbitrarily ϵ-close to a desired parking target (posture). Again, based on the work of [10], one must switch control laws in order to reach the desired $(x_{ref}, z_{ref}, \theta_{ref})$ parking target. The paper demonstrates that linear control laws suffice for the operational modes considered despite the fact that we have a non-holonomic robotic system

Robot-Wheel Speed Relationships. It is instructive to relate the (v, ω) to the angular velocities (ω_l, ω_r) of the left and right rear wheels. The idea here, is that if we can precisely control (ω_l, ω_r), then we will be able to precisely control (v, ω). The desired relationships are as follows: $v = \left[\frac{r(\omega_r+\omega_l)}{2}\right], \omega = \left[\frac{r(\omega_r-\omega_l)}{d_w}\right]$ where r denotes the wheel radius and $d_w = L$ denotes the distance between the rear wheels. It is convenient to rewrite the above relations in vector-matrix form as follows: $\begin{bmatrix}v\\ \omega\end{bmatrix} = M \begin{bmatrix}\omega_R\\ \omega_L\end{bmatrix}, M = \begin{bmatrix}\frac{r}{2} & \frac{r}{2}\\ \frac{r}{d_w} & -\frac{r}{d_w}\end{bmatrix}$.

Nominal DDT^2 Vehicle-Motor System Parameters. Nominal plant parameters are given in Table I [2].

(ω_l, ω_r) System State Space Representation. A state space representation for the TITO LTI model for our DDT^2 vehicle is taken from [14], which was also used within [5] for inner-loop control design. The associated fourth order TITO LTI state space representation is given by $\dot{x} = Ax + Bu, y = Cx + Du$, where $x = [\ v\ \ \omega\ \ i_{ar}\ \ i_{al}\ ]^T$, $y = [\ \omega_r\ \ \omega_l\ ]^T$, $u = [\ e_r\ \ e_l\ ]^T$,

Params.	Definition	Nominal Value
m	Mass (Fully Loaded, Enhanced Vehicle)	0.89 kg
I	Moment of Inertia (Esti. using Cube)	0.0051 kg m^2
r	Wheel Radius	0.05 m
d_w	Distance between Two Rear Wheels	0.14 m
L_a	Armature Inductance	0 H
R_a	Armature Resistance	0.79 Ω
K_b	Back EMF Constant	0.0032 V/rad/s
K_t	Torque Constant	0.0032 Nm/A
β	Speed Damping Constant	70.4 μNms

TABLE I

DDT^2 NOMINAL PARAMETER VALUES AND CHARACTERISTICS

$$A = \begin{bmatrix} \frac{-2\beta K_g^2}{mr^2} & 0 & \frac{K_g K_t}{L_a mr} & \frac{K_g K_t}{L_a mr} \\ 0 & \frac{-\beta K_g^2 d_w^2}{2Ir^2} & \frac{K_g K_t d_w}{2Ir} & \frac{-K_g K_t d_w}{2Ir} \\ \frac{-K_b K_g}{L_a r} & \frac{-K_b K_g d_w}{2L_a r} & \frac{-R_a}{L_a} & 0 \\ \frac{-K_b K_g}{L_a r} & \frac{K_b K_g d_w}{2L_a r} & 0 & \frac{-R_a}{L_a} \end{bmatrix}$$

$$B = \begin{bmatrix} 0_{2\times 2} \\ \frac{1}{L_a} I_{2\times 2} \end{bmatrix} \quad C = [\ M^{-1} \quad 0_{2\times 2}\ ] \quad D = 0_{2\times 2}$$

Here, (i_{al}, i_{ar}) represent left and right motor armature currents, (e_l, e_r) represent left and right motor voltage inputs.

Coupling. A natural question to ask is when is vehicle coupling small? To answer this, we examine the transfer function matrix from input voltages (e_l, e_r) to wheel velocities (ω_l, ω_r). We consider the coupling ratio $\frac{|P_{12}(j\omega)|}{|P_{11}(j\omega)|}$. It can be shown that this ratio takes the form $\frac{bs}{s+z}$ where z is a zero of P_{11} element and $b = \frac{w^2+l^2-6d^2}{w^2+l^2+6d^2}$ where w is vehicle width, l the vehicle length and d the distance between wheels. If $d = w$, then $b = \frac{l^2-5w^2}{l^2+7w^2} = \frac{1-5AR^2}{1+7AR^2}$ where $AR \stackrel{\text{def}}{=} \frac{w}{l}$ is the car aspect ratio. It follows that the coupling ratio is zero at the zero-coupling aspect ratio: $AR_{zero-coup} = \frac{w}{l} = \frac{1}{\sqrt{5}} \approx 0.45$. Finally, the coupling ratio (magnitude) is less than 0.1 when $0.866 < AR < 0.9574$. Our vehicle has an aspect ratio of $AR \approx 0.95$, hence coupling ratio will be small. This implies that classical decentralized SISO control will suffice.

(v, ω) Plant. The above has focused on the (ω_r, ω_l) system. The inner-loop plant, strictly speaking, has (v, ω) as outputs Given this, it is important to note that the (v, ω) plant $P = P_{(v,\omega)}$ is related to the "angular wheel velocity plant" $P_{(\omega_r,\omega_l)}$ according to the relationship: $P = P_{(v,\omega)} = MP_{(\omega_r,\omega_l)}$. The following transfer function approximates the hardware measured response in Fig. 3 (singular values in Fig. 4): $P_{[\omega_r,\omega_l]} \approx 11.268 \left[\frac{1.159}{s+1.159}\right] \times I_{2\times 2}$. Unlike $P_{(\omega_r,\omega_l)}$, $P_{(v,\omega)}$ is not decoupled at low frequencies.

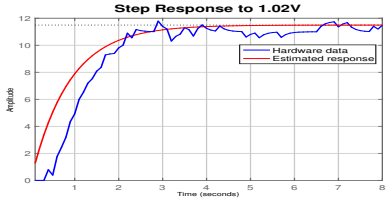

Fig. 3. ω_r response to step $(e_r, e_l) = (1.02, 1.02)V$

Response to Motor Step Voltage. Fig. 3 shows the output

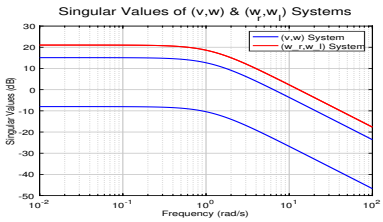

Fig. 4. Plant Singular Values

(wheel speed) response to step input. How can we explain the difference between hardware and simulated step responses?

Transient Differences: PWM LPF Effect. The hardware response fig. 3 is a bit more rate limited than simulated response. This difference can be explained by a higher fidelity model, that incorporates additional low pass filtering and/or a rate limiter. Also, there is inherent zero order hold associated with A-to-D conversion. For control actuation, we use PWM-based A-to-D conversion.

Steady State Differences: PWM 8-Bit Resolution. In steady state, we see that the hardware response goes up to about 11.73 (or 0.21 above the ideal steady state of 11.5) and down to about 11.02 (or 0.48 below ideal). We claim that this is due to the associated 8 bit PWM resolution. The 8 bits give us $2^8 = 256$ levels. This implies a motor input command resolution of $\frac{5.25}{256} \approx 0.021$ V. Noting that our voltage to wheel plant has a dc gain of 11.268, this input command resolution translates into an output angular velocity resolution of $\omega_{res} = 11.268\left(\frac{6}{256}\right) \approx 0.23$. Given this, the hardware oscillation is between $11.5 + \omega_{res}$ and $11.5 - 4\omega_{res}$. Reasons to adequately explain this asymmetry will be examined in future work. Using hardware simulation, we obtain eqn. III, whereas using state-space model, we get, $P_{[\omega_r,\omega_l]} \approx 16.268\left[\frac{3.9}{s+3.9}\right] \times I_{2\times 2}$. Reasons for mismatch:

- Deadzone Effect from Friction. When vehicle starts from rest, static friction is much higher than the rolling friction, resulting in slow pole.
- Battery internal resistance is high when high voltages are applied, making back EMF constant K_b high.
- The output impedance of Arduino Motor shield board is found to be high when motor current is high, resulting in degradation of the DC gain of the plant.

Vehicle Group V-1. Group of 6 vehicles, each augmented with components described in section II, that have approximate transfer function as in eqn. III. Used to implement Longitudinal Platooning. **Vehicle Group V-2**. Group of 6 vehicles, each augmented with components similar to V-1, but Raspberry Pi 3 is used instead of Raspberry Pi 2. The approximate transfer function is $P_{[\omega_r,\omega_l]} \approx 11.333\left[\frac{2}{s+2}\right] \times I_{2\times 2}$. As expected, the model is close to that of V-1. Outer-loops $((v,\theta)$ Cruise Control Along a Path, Planar (x,y) Cartesian Stabilization Control Law, and Two-Vehicle Separation/Direction, Collision Avoidance) are implemented using V-2. **Vehicle Group V-3**. A single vehicle augmented with components similar to V-2, but a caster wheel that can rotate in any direction is used to replace the two front wheels. The approximate transfer function is $P_{[\omega_r,\omega_l]} \approx 5.50\left[\frac{1.73}{s+1.73}\right] \times I_{2\times 2}$. Though we expect V-3 to have higher DC-gain (wheel can rotate in any direction), the friction is much more than expected resulting in lower DC gain. V-3 is used to implement Minimum Lap Time Around a Track.

IV. Inner-Loop Control Law

The controller used for the (ω_r,ω_l) inner-loop has the (PI plus roll-off) decentralized (diagonal) structure: $K_{(\omega_r,\omega_l)} = \frac{g(s+z)}{s}\left[\frac{100}{s+100}\right] \times I_{2\times 2}$ where $g \approx 0.677$, $z \approx 3.77$. A reference command pre-filter $W = \frac{z}{s+z}$ ensures that (using standard second order theory) the overshoot is 0.43% (effectively 0), and settling time is 5sec. In other words, the unity gain crossover bandwidth is 1rad/s. A PI control structure suffices provided that the performance criteria are not too aggressive; For a higher bandwidth objective, a multivariable controller may be helpful to address vehicle coupling. It can be shown that if $K_{(\omega_r,\omega_l)}$ works well for $P_{(\omega_r,\omega_l)}$, then $K = K_{(\omega_r,\omega_l)}M^{-1}$ will work well for $P = MP_{(\omega_r,\omega_l)}$ [18–21]. With $K = K_{(\omega_r,\omega_l)}M^{-1}$ as the controller for P, the new open loop transfer function matrix for the (v,ω) system is $PK = PK_{(\omega_r,\omega_l)}M^{-1} = MP_{(\omega_r,\omega_l)}K_{(\omega_r,\omega_l)}M^{-1}$. Both $P_{(\omega_r,\omega_l)}$ and P have identical open loop singular values (fig. 5). This can also be shown algebraically by first showing that P & $MP_{(\omega_r,\omega_l)}$ have identical singular values.

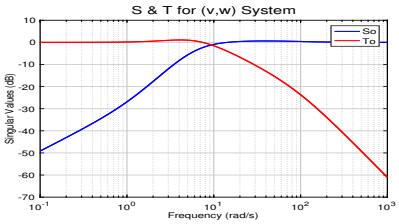

Fig. 5. Sensitivity S and Complementary Sensitivity T

The two designs $((P,K)$ and $(P_{(\omega_r,\omega_l)},K_{(\omega_r,\omega_l)}))$ will possess same singular values for the output/error open loop transfer function matrix $L_o = PK$, output sensitivity $S_o = (I + L_o)^{-1}$, and output complementary sensitivity $T_o = I - S_o = L_o(I + L_o)^{-1}$. Since $KP = K_{(\omega_r,\omega_l)}M^{-1}P = K_{(\omega_r,\omega_l)}M^{-1}MP_{(\omega_r,\omega_l)} = K_{(\omega_r,\omega_l)}P_{(\omega_r,\omega_l)}$, it follows that two designs $((P,K)$ and $(P_{(\omega_r,\omega_l)},K_{(\omega_r,\omega_l)}))$ will possess the same singular values for the input/control open loop transfer function matrix $L_i = PK$, input sensitivity $S_i = (I + L_i)^{-1}$, and input complementary sensitivity $T_i = I - S_i = L_i(I + L_i)^{-1}$. Moreover, these will be the same as those for L_o, S_o, and T_o, respectively.

The above ("almost") implies that if $K_{(\omega_r,\omega_l)}$ works well for $P_{(\omega_r,\omega_l)}$, then $K = K_{(\omega_r,\omega_l)}M^{-1}$ will work well for $P = MP_{(\omega_r,\omega_l)}$. Why do we say "almost?" The reason is that there are still a couple of singular value plots that differ for the two designs. the singular values for the transfer function matrices T_{ru} (reference to controls) and T_{d_iy} (input disturbance to outputs) may differ for the two designs: $(P_{(\omega_r,\omega_l)},K_{((\omega_r,\omega_l))})$ and (P,K) [18–21].

- T_{ru} **Differences for** (ω_r,ω_l) **and** (v,ω) **Systems.** Simple multivariable system algebra shows that

$$\begin{aligned} T_{ru} &= K(I+PK)^{-1} && (2)\\ &= K_{(\omega_r,\omega_l)}M^{-1}(I+MP_{(\omega_r,\omega_l)}K_{(\omega_r,\omega_l)}M^{-1})^{-1} && (3)\\ &= T_{ru_{(\omega_r,\omega_l)}}M^{-1} && (4)\end{aligned}$$

This shows that the control response to reference commands can be different for the two systems.

- $T_{d_i y}$ **Differences for** (ω_r, ω_l) **and** (v, ω) **Systems.** $T_{d_i y} = (I + PK)^{-1}P = (I + MP_{(\omega_r,\omega_l)}K_{(\omega_r,\omega_l)}M^{-1})^{-1}MP_{(\omega_r,\omega_l)} = MT_{d_i y_{(\omega_r,\omega_l)}}$ This shows that the output response to input disturbances can be different for the two loops. The associated T_{ru} and $T_{d_i y}$ singular value plots are shown in Fig. 6.

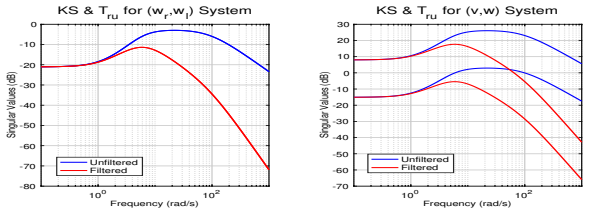

Fig. 6. Reference to Controls Singular Values

Fig. 6 suggests that (v, ω) plant will generally exhibit larger controls when reference commands are issued.

V. Outer-Loop Control Laws: Part-I

We now demonstrate (via simulations and hardware implementation) the following outer-loop control laws:

1. (v, θ) Cruise Control Along a Path. The objective is to follow commanded velocity and angle (v_{ref}, θ_{ref}). The outer-loop control law can be visualized as in Fig. 7. It uses the encoders to measure v and the IMU to measure θ [1–4]. Here, θ is calculated based on ω information from IMU (i.e. $\theta = \theta_{previous} + \omega T$, $T = 0.1$sec), v is estimated based on (w_r, w_l) obtained using wheel encoders. Here, we use

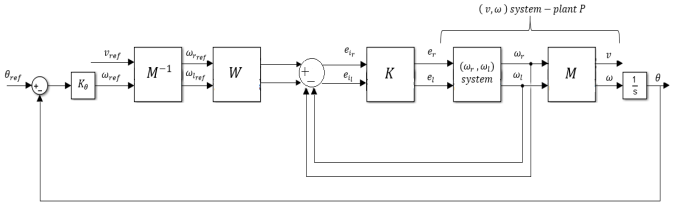

Fig. 7. Visualization of Cruise Control Along a Line/Curve

proportional gain control law. This is justified because the closed loop control map from the references (v_{ref}, θ_{ref}) to the actual speed/angle (v, θ) looks like a diagonal system $diag\left(\frac{a}{s+a}, \frac{b}{s+b}\right)$ (at low frequencies). This is a consequence of a well-designed inner-loop (see above). The outer-loop θ controller therefore sees $\frac{b}{s(s+b)}$, and position controller sees $\left[\frac{b}{s(s+b)}\right]$ (since v_y is so small, integrating v results in x position). From root locus ideas, a proportional controller is therefore justified. When the gain increased to too large values, it results in oscillations becase of high frequency dynamics in the cruise control system (e.g. armature inductance). A PD controller with roll off helps overcome this problem. Figs. 8-9 show simulation responses to an initial condition $\theta_o = 0.5$ rad. The responses corroborate that a sufficiently large gain proportional control law can result in peaking while a PD control law has little peaking. Fig. 10

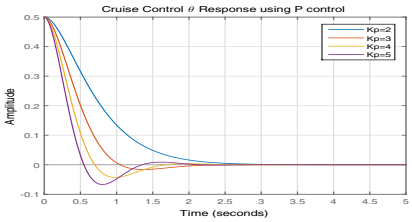

Fig. 8. Cruise Control θ Response Using P Control

shows that as for lower speed, the robot follows the curve

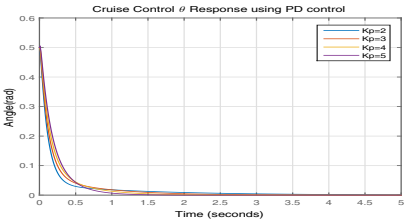

Fig. 9. Cruise Control θ Response Using PD Control

well. For high speeds, the robot was found (not shown) to depart from the curve. This also holds if radius of curvature of the path to be followed decreases (not shown).

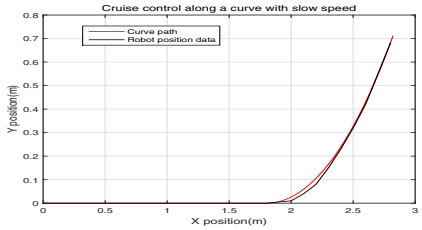

Fig. 10. Cruise Control Along Curve - Slow Speed (Follows Curve Well)

2. Planar (x, y) Cartesian Stabilization Control Law. The objective is to reach a desired target (x_{ref}, y_{ref}). The control law can be thought of as cruise control law with position control augmentation. As was the case in the cruise control system, this control law uses the encoders to measure v and the IMU to measure θ Here, X and Y position is estimated based on (w_r, w_l) obtained using wheel encoders. A simple control law $v = k_s e_s$, $\omega = k_\theta e_\theta$ results in an error dynamics matrix (after linearization) that is Hurwitz when $k_\theta > k_s > 0$ [13]. It is shown in Fig. 11 that the path followed by the robot depends heavily on the directional gain k_θ. For large (more directionally aggressive) k_θ, the robot moves much more directly towards the target. For small k_θ (less directionally aggressive), the robot follows a curved concave up path. Fig. 12 shows that experimentally observed

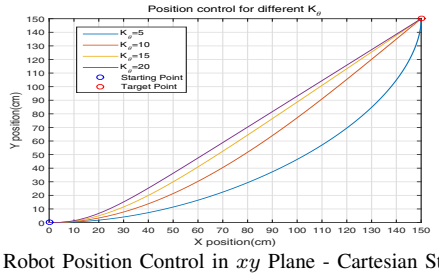

Fig. 11. Robot Position Control in xy Plane - Cartesian Stabilization

data is close to the simulation data. Differences can be explained by considering the following possibilities: stiction in wheels, backlash in gears (due to some spacing between gears), dead-zone of motors (minimum voltage required to move the robot), wheel structure (e.g. smooth and soft for maximum contact on hard track vs dimpled and hard for better gripping on soft track).

3. Minimum Lap Time Around a Track [3]. The objective is to obtain minimum time solution for a known geometry of the track in the presence of physical constraints imposed by the mobile robot. This outer-loop uses ideas from optimal control in order to compute near-optimal (v, ω) trajectories. Such commands can be fed directly to the inner-loop control law or (if ω is integrated to get θ) to the (v, θ) outer-loop control law presented above. The optimization is based on the kinematic model of the robot, along with a simple diagonal model $\left(\frac{a}{s+a}, \frac{b}{s+b}\right)$, with $a, b > 0$), robot's initial position,

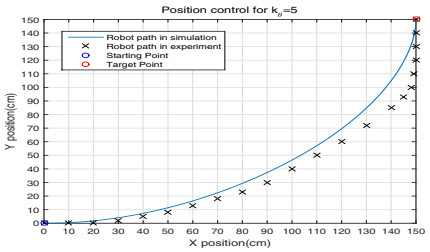

Fig. 12. Robot xy Cartesian Stabilization - Small K_θ

upper and lower bounds on v and ω. We assume a oval track geometry with a circumference of 10.282m (Fig. 13). Also, it is desired that the robot stays within 5cm from the track. The maximum reliable linear velocity of was measured to be 0.5m/sec. A modification taken from [16, 17] is used to transform the optimization problem from an independent variable time t dependence to a distance s dependence. The latter variable is easier to parameterize the track. Doing so makes the final distance s_f a known constant.

$$\begin{aligned} \min_{\mathbf{u}(s)} \quad & J = x_{p+1}(S) \\ \text{subject to:} \quad & \frac{d\mathbf{x}}{ds} = \bar{\mathbf{f}}(\mathbf{x}(s), \mathbf{u}(s)) \quad \mathbf{x}(0) = \mathbf{x_0} \\ & c_i \leq 0 \quad \forall i, \mathbf{u}_L \leq \mathbf{u}(s) \leq \mathbf{u}_U, s \in [0, S] \end{aligned} \tag{5}$$

where x_{p+1} represents elapsed time; S is length of the track to be traversed; c_i represents physical constraint; $\mathbf{x}$ is state vector of plant; $\mathbf{u}(s) = [v_{ref} \quad \theta_{ref}]^T$, $\mathbf{u}_L = [0 \quad -1]^T$, $\mathbf{u}_U = [0.5 \quad 1]^T$. Case-I Camera-based measurement: Raspberry Pi camera was used to compute e_θ. Case-II IMU-based measurement: e_θ is obtained by directly computing $\theta_{ref} - \theta$, while θ is computed using ω based on IMU measurements. Using IMU-based approach, the vehicle is unable to follow the track well. This behavior is due to (somewhat periodic) dead reckoning error accumulation (visualized in Fig. 13). In contrast (but not surprisingly) to IMU-based approach, when the camera is used, the above dead reckoning issue is not an issue and the vehicle is able to follow the track well. Using IMU-based approach, simulated minimum time is 21.14sec, and that in hardware is 21.3sec. Using camera-based approach, simulated minimum time is 25.02sec, and that in hardware is 41sec. As one might expect, camera-based solution is slower than IMU-based solution because of the extra computational constraint imposed by image processing.

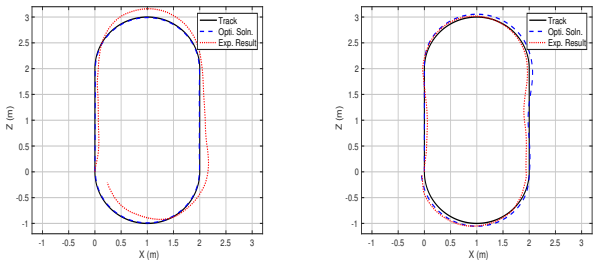

Fig. 13. Left-IMU-based; Right-Camera-based

VI. Summary & Future Directions

In this paper (part I), the focus was on a single differential drive robotic ground vehicle. A TITO LTI differential-drive vehicle-motor model was used as the basis for inner-loop wheel velocity control. Outer-loop control was facilitated by having a well-designed inner-loop - thus permitting simple classical outer-loop designs. The following outer-loop control laws were demonstrated: (speed-direction) cruise control, planar Cartesian (x,y) stabilization and minimum-time optimal-control around an oval race track.

Future single-vehicle work will examine more complex models (e.g. longitudinal-lateral coupling [5], rear-wheel drive [4]) vehicle design (e.g. coupling and performance tradeoffs), and performance objectives (e.g. following a prescribed path in minimum time or with a prescribed speed and accuracy). Future control law design will examine when precisely decentralized controllers suffice and when multivariable controllers are needed [5]. The latter is expected when performance specifications are aggressive; e.g. high bandwidth vis-a-vis vehicle coupling/aspect ratio. The work will also include the development of lab-based localization system using a variety of technologies [6], addition of multiple onboard sensors, use of advanced image processing and optimization algorithms, The above work will impact proposed multiple-vehicle work associated with Part II.

References

[1] Z. Lin, *Modeling, Design and Control of Multiple Low-Cost Robotic Ground Vehicles,* MS Thesis, Arizona State Univ., Tempe, AZ, 2015.

[2] Z. Li, *Modeling and Control of a Longitudinal Platoon of Ground Robotic Vehicles,* MS Thesis, Arizona State Univ., Tempe, AZ, 2016.

[3] J. Aldaco, *Image Processing Based Control of Mobile Robotics,* MS Thesis, Dept. Elect. Eng., Arizona State Univ., Tempe, AZ, 2016.

[4] X. Lu, *Modeling and Control for Vision Based Rear Wheel Drive Robot and Solving Indoor SLAM Problem Using LIDAR*, MS Thesis, Dept. Elect. Eng., Arizona State Univ., Tempe, AZ, 2016.

[5] I. Anvari, *Non-holonomic Differential-Drive Mobile Robot Control & Design: Critical Dynamics and Coupling Constraints*, MS Thesis, Dept. Elect. Eng., Arizona State Univ., Tempe, AZ, 2013.

[6] N. Ravishankar, *Utilization of single image perspective cues for autonomous navigation in indoor envi.*, MS Thesis, ASU, 2017.

[7] N. Das, *Modeling, Design and & Control of a Robotic Ground Vehicle with a Ball Catching Capability*, MS Thesis, ASU, 2017.

[8] S. Sonawani, *Modeling, Design and & Control of Multiple Ground Vehicles with a Ball Catching Capability*, MS Thesis, ASU, 2017.

[9] A. A. Rodriguez et al., "Modeling, Design and Control of Low-Cost Differential-Drive Robotic Ground Vehicles Part-II: Multiple Vehicle Study," in *1st IEEE Conf. on Control Tech. & Applications*, 2017.

[10] R. W. Brockett, "Asymptotic stability and feedback stabilization," in *Diff. Geometric Control Theory*. Birkhauser, pp. 181-191, 1983.

[11] A. Astolfi, "On the Stabilization of Nonholonomic Systems," *Proc. 33rd IEEE Conf. Decision and Control*, vol.4, pp. 3481-3486, 1994.

[12] A. Isidori. *Nonlinear Control Systems,* 3rd ed., Springer-Verlag, 1995.

[13] F.C. Vieira et al. "Position and Orientation Control of a 2-Wheeled Diff. Driven Nonholonomic Mobile Robot," *ICINCO Proc.*, 2004.

[14] R. Dhaouadi and A. A. Hatab, "Dynamic Modeling of Differential-Drive Mobile Robots using Lagrange and Newton-Euler Methodologies: A Unified Framework," *Advances in Robot and Auto.*, 2013.

[15] M.I Ribeiro, P. Lima, "Kinematic models of mobile robots," em Instituto de Sistemas e Robotica, pp. 1000-1049, 2002.

[16] D. Casanova, "On Minimum Time Vehicle Manoeuvring: The Theoretical Optimal Lap," PhD Thesis, Cranfield University, England, 2000.

[17] T. Gustafsson, "Computing the Ideal Racing Line Using Optimal Control," MS Thesis, Linkoping Univ., Linkoping, Sweden, 2008.

[18] K. Puttannaiah et al., "A Generalized Mixed-Sensitivity Convex Approach to Hierarchical Multivariable Inner-Outer Loop Control Design Subject to Simultaneous Input and Output Loop Breaking Specifications," in *American Control Conf.*, pp. 5632-5637, 2016.

[19] K. Puttannaiah et al., "Analysis and Use of Several Generalized $\mathcal{H}^\infty$ Mixed Sensitivity Frameworks for Stable Multivariable Plants subject to Simultaneous Output and Input Loop Breaking Specifications," in *Conf. on Decition & Control, IEEE*, pp. 6617-6622 2015.

[20] K. Puttannaiah et al., "A Generalized $\mathcal{H}^\infty$ Control Design Framework for Stable Multivariable Plants subject to Simultaneous Output and Input Loop Breaking Specifications," in *ACC, IEEE*, 2015.

[21] J. A. Echols et al., "Fundamental Vehicle and Control System Design Issues for Scramjet-Powered Hypersonic Vehicles," *AIAA Guidance, Navigation & Control Conf.*, Kissimmee, FL, 2015.