

Derivation of a new drive/coast motor driver model for real-time brushed DC motor control, and validation on a MIP robot

Eric N. Sihite¹, Daniel J. Yang², and Thomas R. Bewley³

Abstract—Brushed DC motors are usually driven with PWM forcing in one of two modes: drive/brake or drive/coast. That is, at the low state of the PWM forcing profile, the motor driver will either “brake” the motor with its own back EMF, or allow the motor to “coast” (i.e., spin freely). Drive/brake motor drivers, which are by far the most common, may be represented by a Multilevel Four-Quadrant DC Chopper model, while drive/coast motor drivers may be represented by two independent Bipolar Two-Quadrant DC Chopper models. Conveniently, when averaged over the PWM duty cycle, drive/brake motor drivers are accurately modeled as linear systems over their entire operational range. On the other hand, drive/coast motor drivers, when averaged over the PWM duty cycle, exhibit significant nonlinear behaviors that are dependent on factors such as inductance, PWM frequency, and rotor speed. Though there are some existing partial derivations of drive/coast motor driver models, no comprehensive, experimentally-validated modeling approaches appropriate for feedback control applications over the full dynamic range of the motor could be readily found in the literature. In this paper, we derive a practical nonlinear model of a drive/coast motor driver, validate this model using a motor dynamometer, and demonstrate a real-time implementation of this model on a Mobile Inverted Pendulum (MIP) robot.

I. INTRODUCTION

Agile dynamic UxVs require accurate physical models to inform mechanical designs and to develop controllers that achieve maximum performance. Mass distributions and other physical parameters can generally be obtained via CAD and simple experiments, but accurate dynamic motor models are often much more difficult to develop. Least-square fits to experiments can be used to identify the parameters of simple linear dynamic models of motors if they are of the correct structure, but getting the (nonlinear) structure of these models correct in the drive/coast case is delicate.

Brushed DC motors are commonly driven by modulating the input voltage via pulse width modulation (PWM), at frequencies from 500Hz to 20kHz, together with one or two logic signals to indicate the forward or reverse direction. The MOSFETs in an H-Bridge are then opened or closed in pairs to allow current to pass through the circuit in the appropriate direction. There are two types of motor driver modes discussed in this paper, depending on the MOSFET settings during the low state of the PWM forcing profile: braking or coasting. Braking occurs when both of the upper (or, lower) MOSFETs are closed, and the other MOSFETs are open; in this case, current circulates in the upper (or,

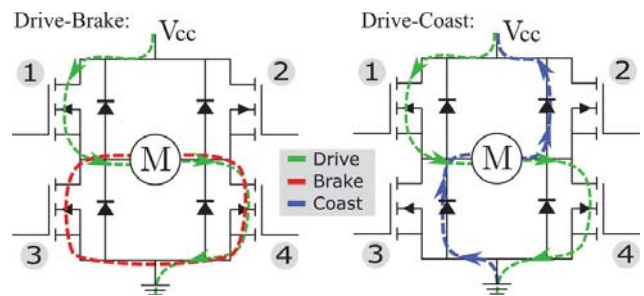


Fig. 1. Full H-bridge circuit demonstrating the current paths during forward drive, brake, and coast. To drive the motor forward (green), MOSFETs 1 and 4 are closed. To brake (red), either MOSFETs 3 and 4 are closed (low side) or 1 and 2 (high side) which shorts the motor and allows the current to circulate and brake the motor. Finally, to coast (blue), all of the MOSFETs are left open, allowing the current to flow through the diodes as necessary, and causing the motor to coast when this current decays to zero.

lower) part of the H-bridge circuit, and the back EMF of the motor itself drives the current in the opposite direction of the rotor speed, slowing the motor. Coasting occurs when all four MOSFETs in the H-bridge are open, and current flows from ground to V_{cc} through two of the flyback diodes when necessary. The coasting steady state current applies a zero torque on the motor, allowing the motor to spin freely. Figure 1 shows the driving, braking and coasting states of an H-bridge.

Drive/brake motor drivers may be represented by a Multilevel Four-Quadrant DC Chopper, and drive/coast motor drivers may be represented by two independent Bipolar Two-Quadrant DC Chopper [1][2]. Drive/coast motor drivers in constant forward drive use the I-IV quadrants, and in reverse drive use the II-III quadrants [1]. Averaged over the PWM duty cycle, such drive/brake motor drivers may be accurately modeled with a simple linear model [3][4][5][6]. Averaged over the PWM duty cycle, the drive/coast motor driver, on the other hand, exhibits a significant nonlinear behavior especially near zero duty cycle. Partial derivations of electrical current models for such drivers exist [1]; however, this work does not extend to a full range of forward and reverse driving, and no real-time implementation suitable for control applications is provided.

There are several advantages of using drive/coast motor drivers. Coasting engages the natural (unforced) dynamic behavior of the vehicle, which is at times preferred. There might also be a good incentive to use drive/coast instead of drive/brake drivers from an energy efficiency standpoint. There has been much research on regenerative braking with brushless DC motors in, e.g., hybrid/electrical vehicles

*University of California San Diego, Department of Mechanical and Aerospace Engineering, Coordinated Robotics Lab.

¹ esihite@eng.ucsd.edu, ² djiang@ucsd.edu,

³ bewley@eng.ucsd.edu

[7][8][9][10]. However, to the best of our knowledge, the energy efficiency of drive/coast vs. drive/brake systems for vehicles that often operate near zero speed and/or zero torque has not been extensively studied. A drive/brake system can produce regenerative current during the braking sequence, transforming the kinetic energy into electrical energy [7]. A drive/coast system, on the other hand, regenerates the battery during coasting by flowing current back into the Vcc through the flyback diodes [1][9]. In a motor that operates near zero torque and/or zero speed, the drive/brake regenerative method is ineffective due to the lack of kinetic energy. The drive/coast approach might have better efficiency in this range, but conclusively establishing this would require further study. Recent advanced TI motor drivers, such as the DRV8881 in the so-called “fast decay” mode, detect the regenerative current during coasting and close the respective gates such that the current flows through the MOSFETs instead of the flyback diodes, resulting in reduced voltage drop (and, thus preventing the flyback diodes from overheating). If this is utilized well, drive/coast systems might show better energy efficiency than drive/brake systems near zero torque in speed control applications.

Brushed DC motors are cheap and often used in low cost robots which generally have a high amount of noise. The challenge of accurately modeling such motors is exacerbated in a drive/coast implementation due to the nonlinearities near zero torque and speed, which is a primary mode of operation for a balancing robot such as a Mobile Inverted Pendulum (MIP). Modeling the system dynamics as accurately as possible can increase the performance of the controller. For these reasons, it is important to develop an accurate drive/coast model that can be used in a real-time feedback controller.

In this paper, we present a new dynamic model for a system with a brushed DC motor using a drive/coast motor driver, validate the model using a low cost dynamometer, propose a method of real-time implementation of the drive/coast model, and demonstrate the performance of our model on an educational Mobile Inverted Pendulum (eduMIP) robot.

II. DRIVE MODES

An H-bridge circuit allows for directional control of a brushed DC motor. It typically consists of four MOSFETs and diodes arranged as letter H with a load at the center [11]. By switching the diagonally opposite MOSFETs on and off, the motor driving direction (forward or reverse) can be controlled. During the off duty of the PWM signal, the motor will either brake or coast based on the states of the four MOSFETs, as shown in Fig. 1. The choice of coast or brake mode has a significant effect on motor’s dynamics, especially at low duty cycles. A quick survey of common motor drivers used in robotics shows that the drivers are either drive/coast only (DRV8881E, MC33926), drive/brake only (TB6612FNG), or configurable in either mode (DRV8881P, DRV8871). In practice, there is no “standard” mode of operation for the motor drivers and it is often left to the user’s discretion. Table I lists the parameters and the variables used in the equations and derivations in this paper. The following

TABLE I
LIST OF PARAMETERS AND TIME VARYING VARIABLES

Constant Parameters, all values are > 0 .			
k	torque constant.	V	supply voltage.
R	motor resistance.	i_s	stall current at $u = 1$, $\omega = 0$.
L	motor inductance.	ω_{nl}	maximum rotor no load speed.
T_{pwm}	PWM period.	T_e	electrical time constant.
f_{pwm}	PWM frequency.	T_r	ratio of T_{pwm} over T_e .
Time Varying Variables			
i	electrical current.	u	motor command $\in [-1, 1]$.
τ	motor torque.	v	PWM duty cycle, $v = u \in [0, 1]$.
ω	rotor velocity.	ω_r	scaled rotor velocity $\in [-1, 1]$.

subsections derive and discuss the electrical equation model for the drive/brake and drive/coast motor drivers.

A. Drive/Brake

In drive/brake mode, the current path through the H-bridge can be seen in Fig. 1. During a high PWM signal, the appropriate MOSFETs pair is closed to drive the motor forward or reverse [11]. During low PWM signal, either the high side or low side MOSFETs are closed, shorting the motor terminals and allowing the back EMF to circulate and brake the motor [12]. The drive/brake system is very well understood and commonly modeled as follows: let u be the motor command, where $v = |u|$ is the PWM duty cycle and $\text{sign}(u)$ is the motor driving direction where $u > 0$ and $u < 0$ are forward and reverse driving respectively. The electrical circuit equation during the high and low PWM signals can be seen in (1) and (2).

$$\text{PWM high:} \quad \text{sign}(u)V = Ri + L \frac{di}{dt} + k\omega \quad (1)$$

$$\text{PWM low:} \quad 0 = Ri + L \frac{di}{dt} + k\omega \quad (2)$$

$$uV = Ri_{avg} + k\omega. \quad (3)$$

Both models can be combined into the linear system shown in (3) where i_{avg} is the average current of one PWM pulse [6]. The inductance can be ignored because the controller and measurement update period are significantly longer than the electrical time constant. This linear behavior and simplicity is a significant advantage for controller design and is the most common motor model used for PWM based motor controls.

B. Drive/Coast

The drive/coast model is much more complex. The current path for drive/coast mode can be seen in Fig. 1. During the low PWM signal, all of the MOSFETs are opened, forcing the current of the spinning motor to pass through the diodes into the battery positive terminal. The current quickly decays to zero but does not change direction, allowing the motor to spin unimpeded. The electrical circuit equation during the high and low PWM signals can be seen in (4) and (5).

$$\text{PWM high:} \quad \text{sign}(u)V = Ri + L \frac{di}{dt} + k\omega \quad (4)$$

$$\text{PWM low:} \quad -\text{sign}(i)(V + 2V_d) = Ri + L \frac{di}{dt} + k\omega, \quad (5)$$

where V_d is the voltage drop across the diodes during the low PWM signal. As the motor coasts, the current drains down

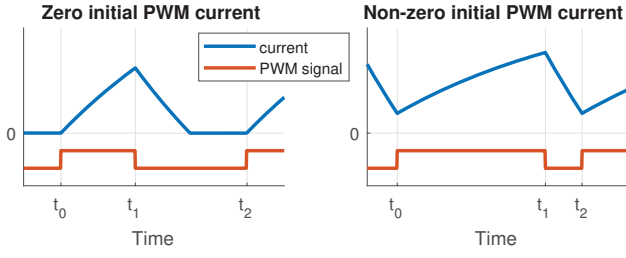


Fig. 2. Plot of the current and PWM signal (shown at an offset) during one time-periodic PWM pulse in a drive/coast motor driver. t_0 , t_1 , and t_2 are the pulse start time, low PWM signal start time and the final pulse time respectively. The initial current during each time-periodic PWM pulse is assumed to be constant ($i(t_0) = i(t_2)$).

to zero and stays at zero for as long as the MOSFETs are opened, resulting in nonlinear dynamics. Additionally, di/dt is nonzero at $i = 0$, so this system must be described as a hybrid system, making it difficult to determine an averaged system model for a single PWM pulse.

III. DRIVE/COAST DYNAMIC MODEL

Although the drive/coast system is a hybrid system, $i(t)$ can be solved from both ODEs in (4) and (5), as done by [1] on a First-Quadrant and a Bipolar Two-Quadrant DC Choppers. The averaged system model such as in (3) can't be derived. However, the average current i_{avg} during one PWM pulse can then be derived from the solution of $i(t)$. Deriving the i_{avg} is useful for controls application because the applied motor torque is a linear function of the motor current ($\tau = ki$). The following assumptions are used in order to simplify the $i(t)$ derivations:

- Electrical time constant is much smaller than mechanical time constant, controller update and sampling time.
- Time-periodic PWM pulse, as shown in Fig. 2, where the starting and final current in one PWM pulse are equal.
- Voltage drop across the diodes is negligible ($V_d \approx 0$).
- Instantaneous MOSFETs switching time.
- No external force to the mechanical system that can push the rotor speed above ω_{nl} .

The first assumption simplifies the ODEs in (4) and (5) by setting u and ω approximately constant compared to the i within one PWM pulse. The second assumption simplifies the problem further by constraining the initial and final current in one PWM pulse. The equations can be simplified even further by applying change of variables below:

$$\begin{aligned} i_s &= V/R & T_e &= L/R & \omega_{nl} &= V/k \\ T_r &= T_{pwm}/T_e & \omega_r &= \omega/\omega_{nl}, \end{aligned} \quad (6)$$

where ω_r is the ratio of between the rotor velocity and the maximum no load speed while T_r is the ratio between the PWM period and the electrical time constant. The final assumption above guarantees that $\omega_r \in [-1, 1]$. Using (6) and the assumptions above into (4) and (5) results in the

following simplified equations:

$$\text{PWM high: } \frac{di}{dt}(t) = (-i(t) + i_s(\text{sign}(u) - \omega_r))/T_e \quad (7)$$

$$\text{PWM low: } \frac{di}{dt}(t) = (-i(t) - i_s(\text{sign}(i(t)) + \omega_r))/T_e. \quad (8)$$

Let t_0 be the time at the beginning of the PWM pulse, $t_1 = t_0 + vT_{pwm}$ is the starting time of the low PWM signal, and $t_2 = t_0 + T_{pwm}$ is the end time of the pulse, as shown in Fig. 2. There are several statements that can be made from using $\omega_r \in [-1, 1]$, (7) and (8) that will be useful during the derivation of $i(t)$.

Lemma 1. *The current is always being driven to $i_{hi} = i_s(\text{sign}(u) - \omega_r)$ during high PWM signal. Also, $i_{hi} \geq 0$ if $u > 0$ and $i_{hi} \leq 0$ if $u < 0$.*

Proof: From (7), it can be easily shown that $\frac{di}{dt} < 0$ if $i > i_{hi}$ and $\frac{di}{dt} > 0$ if $i < i_{hi}$. Using $\omega_r \in [-1, 1]$, it can also be shown that if $\text{sign}(u) > 0$, then $i_{hi} \geq 0$ and if $\text{sign}(u) < 0$, then $i_{hi} \leq 0$. $\square$

Lemma 2. *The current is always being drained to zero during low PWM signal.*

Proof: Use $\omega_r \in [-1, 1]$ in the (8), then it can be shown that $\frac{di}{dt} < 0$ if $i > 0$ and $\frac{di}{dt} > 0$ if $i < 0$. $\square$

Proposition 1. *Assuming time-periodic PWM pulse, if $u > 0$, then $i(t) \geq 0$ within one pulse ($t \in [t_0, t_0 + T_{pwm}]$). Conversely, if $u < 0$, then $i(t) \leq 0$.*

Proof: During high PWM signal, the current is being driven to $i_{hi} \geq 0$ if $u > 0$ using Lemma 1. Then during low PWM signal, the current is being driven to zero using Lemma 2. Then that means the current during both high and low PWM signal, or one PWM pulse, is always ≥ 0 for $u > 0$. Case $u < 0$ can be proven the same way. $\square$

Solve the ODE in (7) and (8) for $i(t)$ using the assumptions and statements listed above. There are two different cases that need to be explored: $i(t_0) \neq 0$ and $i(t_0) = 0$, which is also shown in Fig. 2.

A. Case $i(t_0) \neq 0$

Solve the ODE in (7) and (8) for $u > 0$ and $i(t_0) > 0$. Using Proposition 1 and time-periodic PWM signal, $\text{sign}(i)$ is inferred to be constant in (8). This transforms (7) and (8) into easier to solve linear equations:

$$\text{PWM high: } \frac{di}{dt}(t) = -(i(t) + i_s(-1 + \omega_r))/T_e \quad (9)$$

$$\text{PWM low: } \frac{di}{dt}(t) = -(i(t) + i_s(1 + \omega_r))/T_e. \quad (10)$$

Solve the ODE in (9) and (10) for $i(t)$:

$$i_H(t_H) = e^{-t_H/T_e} \left(i(t_0) + i_s(1 - e^{t_H/T_e})(\omega_r - 1) \right) \quad (11)$$

$$i_L(t_L) = e^{-t_L/T_e} \left(i(t_1) + i_s(1 - e^{t_L/T_e})(\omega_r + 1) \right), \quad (12)$$

where $i_H(t_H)$ and $i_L(t_L)$ are the current equation during high and low PWM signal respectively. $t_H = t - t_0$ and $t_L = t - t_1$ are the time shift such that t_H and t_L are zero at the start of the high and low PWM signal respectively. Using (11) and

(12), solve for $i(t_2)$ which should be equal to $i(t_0)$ by using the time-periodic PWM pulse assumption as shown below:

$$i(t_1) = i_H(vT_{pwm}) = e^{-vT_r} [i(t_0) + i_s(1 - e^{vT_r})(\omega_r - 1)] \quad (13)$$

$$i(t_2) = i_L((1 - v)T_{pwm}) = i(t_0) \quad (14)$$

$$i(t_0) = e^{-T_r} [i(t_0) - i_s(1 - \omega_r - 2e^{vT_r} + e^{T_r}(1 + \omega_r))] \quad (15)$$

Then solve for $i(t_0)$ from (15):

$$i(t_0) = \frac{-i_s(1 - \omega_r - 2e^{vT_r} + e^{T_r}(1 + \omega_r))}{e^{T_r} - 1} > 0. \quad (16)$$

Using the knowledge that $T_r > 0$ and $e^{T_r} > 1$, derive the domain for $i(t_0) > 0$:

$$e^{vT_r} > (e^{T_r}(1 + \omega_r) + 1 - \omega_r)/2. \quad (17)$$

By plugging in the initial current $i(t_0)$ in (16) into (11), (13) and (12), the average current during one time-periodic PWM pulse can be solved for $u > 0$ and $i(t_0) > 0$ case:

$$i_{avg,i \neq 0}^+ = \left(\int_0^{vT_{pwm}} i_H(t_H) dt_H + \int_0^{(1-v)T_{pwm}} i_L(t_L) dt_L \right) / T_{pwm} \\ = i_s(2v - 1 - \omega_r). \quad (18)$$

Solving the equation for $u < 0$ and $i(t_0) < 0$ case using the same methodology yields the following average current:

$$i_{avg,i \neq 0}^- = -i_s(2v - 1 + \omega_r), \quad (19)$$

with the domain for where $i(t_0) < 0$ being:

$$e^{vT_r} > (e^{T_r}(1 - \omega_r) + 1 + \omega_r)/2. \quad (20)$$

B. Case $i(t_0) = 0$

In this case, the current is fully drained during the low PWM signal, and the draining time must be derived in order to solve for the $i_L(t)$. Solve for the $u > 0$ case first by using (11), (12) and (13) with $i(t_0) = 0$. Then solve for the time where the current is fully drained (t_d):

$$i_L(t_d) = i_s e^{-t_d/T_e} (2 + e^{-vT_r}(\omega_r - 1)) - i_s(1 + \omega_r) = 0 \quad (21)$$

$$t_d = -T_e \log \left(\frac{e^{vT_r}(1 + \omega_r)}{2e^{vT_r} - 1 + \omega_r} \right). \quad (22)$$

This log function must have a real solution for the domain where $i(t_0) = 0$. Using $v \in (0, 1]$, $\omega_r \in [-1, 1]$ and $T_r > 0$, the log function is valid everywhere except for $\omega_r = -1$. However $\omega_r = -1$ is always within the domain for $i(t_0) > 0$, so the log function is always valid. The current is fully drained during the low PWM signal, so $t_d \leq (1 - v)T_{pwm}$ is set as a constraint. This constraint can be solved further into:

$$e^{vT_r} \leq (e^{T_r}(1 + \omega_r) + 1 - \omega_r)/2, \quad (23)$$

which is the complete opposite of the domain for $i(t_0) > 0$ in (17). This means that the initial current $i(t_0) \geq 0$ covers for all $u > 0$ and $\omega_r \in [-1, 1]$. Finally, solve for the average

current in one time-periodic PWM pulse for the $u > 0$ and $i(t_0) = 0$ case:

$$i_{avg,i_0=0}^+ = \left(\int_0^{vT_{pwm}} i_H(t_H) dt_H + \int_0^{t_d} i_L(t_L) dt_L \right) / T_{pwm} \\ = \frac{i_s}{T_r} \left(vT_r(1 - \omega_r) + (1 + \omega_r) \log \left(\frac{e^{vT_r}(1 + \omega_r)}{2e^{vT_r} - 1 + \omega_r} \right) \right). \quad (24)$$

Using the same methodology to solve for the $u < 0$ and $i(t_0) = 0$ case yields the following average current:

$$i_{avg,i_0=0}^- = -\frac{i_s}{T_r} (vT_r(1 + \omega_r) \\ + (1 - \omega_r) \log \left(\frac{e^{vT_r}(1 - \omega_r)}{2e^{vT_r} - 1 - \omega_r} \right)), \quad (25)$$

and the following domain where $i(t_0) = 0$:

$$e^{vT_r} \leq (e^{T_r}(1 - \omega_r) + 1 + \omega_r)/2. \quad (26)$$

C. Summary

Let $v = |u| \in [0, 1]$, $\omega_s = \text{sign}(u) \omega_r$, $\omega_r \in [-1, 1]$, and the domain for $i(t_0) \neq 0$ is:

$$\mathbf{V} = \left\{ v \mid v > \frac{1}{T_r} \log \left(\frac{1}{2} (e^{T_r}(1 + \omega_s) + 1 - \omega_s) \right) \right\}. \quad (27)$$

Then average current i_{avg} of the time-periodic PWM pulse as a function of motor command u , scaled rotor speed ω_r , and a constant T_r can be calculated with the following algorithm:

if $v = 0$ **then**

$$i_{avg,v=0} = 0 \quad (28)$$

else if $v \in \mathbf{V}$ **then**

$$i_{avg,i_0 \neq 0} = i_s(2u - \text{sign}(u) - \omega_r) \quad (29)$$

else

$$i_{avg,i_0=0} = i_s(u(1 - \omega_s) \\ + \left(\frac{\text{sign}(u) + \omega_r}{T_r} \right) \log \left(\frac{e^{vT_r}(1 + \omega_s)}{2e^{vT_r} - 1 + \omega_s} \right)) \quad (30)$$

end if

As shown above, the average current is nonlinear especially when $i(t_0) = 0$. The average current equations are defined using the i_s and unitless parameters ω_r , T_r , and u .

IV. IMPLEMENTATION STRATEGY

In order to be used in a controller, the motor command u must be solved given the target average current i_t and the measured rotor speed ω . Solving for u in the $i_0 \neq 0$ case from (29) is easy. However, as shown in the (30), the drive/coast model is nonlinear in the $i_0 = 0$ case, making it difficult to solve for $u(i_t, \omega_r)$ directly. Solving for $v = |u|$ in the $i(t_0) = 0$ case can be done by using scalar iterative methods such as the Newton-Raphson method [14]. Our tests have shown that this algorithm converges within 3 to 5 iterations which is quick enough to be used in real-time computations. The

TABLE II
EXPERIMENT MOTOR AND MOTOR DRIVER LIST

No.	Motors	Motor Drivers
1	Maxon A-Max 22 110160, 14:1	DRV8871 (3A i_{max})
2	Maxon 273688, 13mm, 17:1	DRV8881P (2.5A i_{max})
3	eduMIP motor, 6.6:1	MC33926 (5A i_{max})

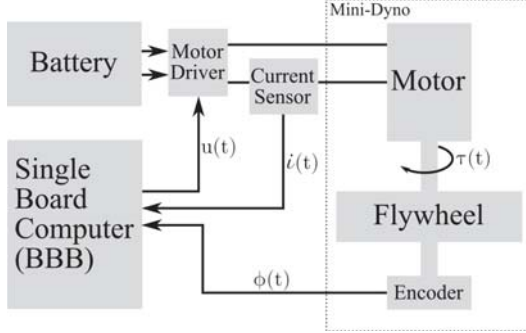


Fig. 3. Experimental setup diagram of the dynamometer for the drive/coast model validation.

equations to be used in the Newton-Raphson method can be seen below:

$$f(v) = i_{avg, i_0=0} - i_t = 0, \quad (31)$$

$$f'(v) = \frac{2i_s(e^{vT_r} - 1)(\text{sign}(i_t) - \omega_r)}{2e^{vT_r} + \omega_s - 1} \quad (32)$$

$$v_{n+1} = v_n - f(v_n)/f'(v_n). \quad (33)$$

Then the method to solve for v can be outlined below:

- 1) $\text{sign}(u) = \text{sign}(i_t)$. If $i_t = 0$, then $v = 0$.
- 2) else, assume $i(t_0) \neq 0$ and solve for v using (29) then check if $v \in \mathbf{V}$ and $v \in (0, 1]$.
- 3) If $v \notin \mathbf{V}$ or $v \notin (0, 1]$, then we have the $i(t_0) = 0$ case and solve for v with Newton-Raphson method using (30) to (33). Set the initial guess v_0 to be the middle of the range of v for $i(t_0) = 0$:

$$v_0 = \frac{1}{2T_r} \log\left(\frac{1}{2}(e^{T_r}(1 + \omega_s) + 1 - \omega_s)\right). \quad (34)$$

In addition, there are two edge cases for the model in (27) to (30): the case of $\lim T_r \rightarrow 0$ and $\lim T_r \rightarrow \infty$ which represent very high and very low PWM frequency respectively. Solving the model using the limits and L'Hospital Rule yields the following results:

$$\text{Case } \lim T_r \rightarrow 0: \quad u = (i_t/i_s + \text{sign}(i_t) + \omega_r)/2 \quad (35)$$

$$\text{Case } \lim T_r \rightarrow \infty: \quad u = (i_t/i_s)/(1 - \text{sign}(i_t)\omega_r). \quad (36)$$

V. DRIVE/COAST MODEL VALIDATION

To validate our drive/coast model, we compared the estimated and actual current measurements of a motor-flywheel system (Mini-Dyno [13]). First we identify the motor parameters, shown in Table III, using the procedure outlined in [13]. Using these parameters and a known input signal we were able to estimate the drive/coast model and compare it to the actual output data.

TABLE III
IDENTIFIED MOTOR PARAMETERS

Parameters	Motor No.		
	1	2	3
Resistance, R (Ω)	6.49	15.4	9.06
Inductance, L (mH)	0.362	0.0494	2.36
Motor gain, k (N.m/A)	0.133	0.161	0.127

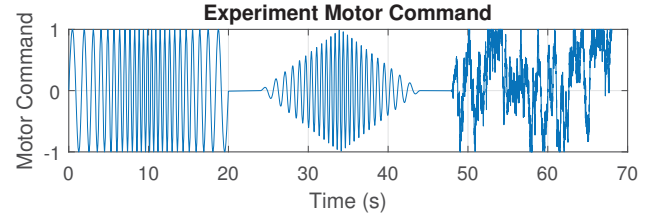


Fig. 4. Plot of the motor command u signal for the experiment in the following sequence: sine chirp, ramping sine chirp, and random walk.

A. Experimental Setup

The Mini-Dyno [13] is a low cost motor dynamometer that consists of a weighted flywheel of known inertia attached to a motor and an optical encoder (US Digital E6-2500-250). The motor is attached to a motor driver, a current sensor (INA219), and driven by a single board computer (Beaglebone Black), as shown in Fig. 3.

We conducted this experiment with three different motors and motor drivers which are listed in Table II. The motor drivers were selected because they are commonly used in small ground robots, have off-the-shelf breakout boards, and feature drive/coast operating modes. Two different high quality Maxon motors with datasheets were selected to provide a benchmark of performance. The last motor is a low-cost toy motor used in the eduMIP robots with unknown parameters. We selected this motor to demonstrate that the drive/coast model is accurate with low quality motors that have higher inductance and friction. The motors' resistance and torque constant were identified using the methodology outlined in [13] but with one important note: the identification must be done with a drive/brake motor driver to avoid the nonlinear drive/coast dynamics. We estimated the motor's inductance by measuring the rise time of the voltage across the shunt resistor on the current sensor with an oscilloscope. The identified motor parameters are shown in Table III.

To excite the motor across a range of frequencies and amplitudes, the motor was driven by a sequence of open loop input signals: sine chirp, ramping sine chirp, and random walk, as shown in Fig. 4. The sine chirp is a sine signal with frequency ramping up from 0.5 Hz to 2 Hz and back down to 0.5 Hz. The ramping sine chirp is a sine chirp with varying amplitude. The random walk is a sum of a uniform random variable, $u_{k+1} = u_k + \text{unif}(-0.3, 0.3)$, and the same random walk signal was used for each experiment. Each motor and motor driver pairs were tested at PWM frequencies of 500 Hz, 1 kHz, 5 kHz, 10 kHz and 20 kHz. The low range

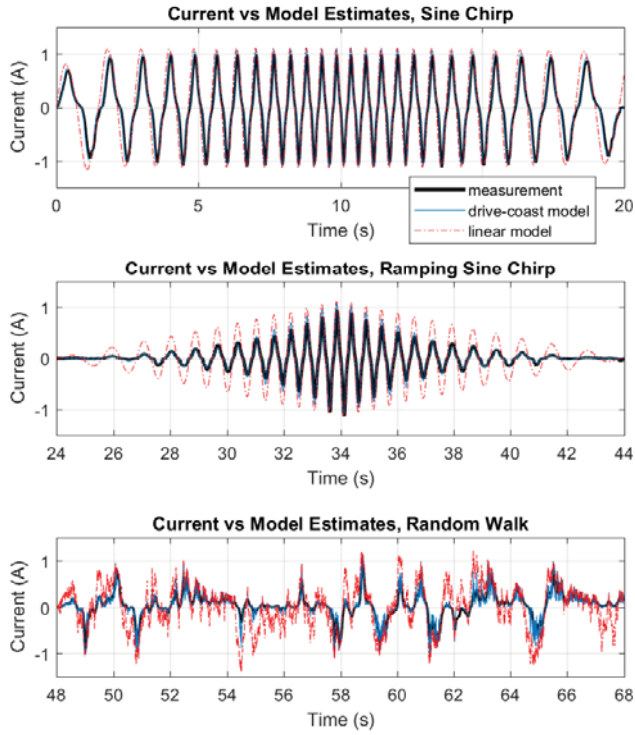


Fig. 5. Plot of measured current vs. current estimate from the drive/coast model and the linear motor model. The data shown is of motor 1 & driver 1 at 20kHz PWM frequency.

was chosen because 500 Hz is the default PWM frequency of certain microcontrollers (e.g. Arduino) and 20 kHz is ultrasonic and inaudible. During this experiment, the motor command u , current i , battery voltage V and the flywheel angle ϕ were recorded. From the input and output data, we estimated the current of the drive/coast system using the model from (27) to (30). In order to show the inaccuracy of using the drive/brake model with a coasting motor driver, we also estimated the current by using the linear model in (3) for motors with driver 1, such that:

$$i_{avg,lin} = (uV - k\omega)/R. \quad (37)$$

R^2 and RMSE values of the estimated drive/coast current with respect to the current measurement are used to compare model accuracy. The RMSE values are represented as a percentage of the motor stall current i_s .

B. Model Validation Results

The drive/coast model has high R^2 values and low RMSE values, as shown in and Table IV and V respectively, which indicates an accurate model. The drive/coast model has an average RMSE error of 6.5% i_s while the linear model has an average RMSE error of 22.5% i_s . From Fig. 5, we can see how much more accurate the drive/coast model is compared to the linear model especially at low duty cycles. As shown in Table IV and V, the drive/coast model is accurate and consistent for all PWM frequencies, motor, and motor driver combinations. However, the low quality eduMIP

TABLE IV
 R^2 VALUES OF THE CURRENT MEASUREMENT VS. MODEL ESTIMATE

Motor & Mtr. Driver	PWM Frequency (Hz)				
	500	1000	5000	10000	20000
vs. coast model:					
Motor 1 & driver 1	0.964	0.969	0.980	0.964	0.962
Motor 1 & driver 2	0.971	0.973	0.973	0.650	0.937
Motor 1 & driver 3	0.987	0.986	0.984	0.972	0.964
Motor 2 & driver 1	0.982	0.985	0.982	0.980	0.973
Motor 2 & driver 2	0.979	0.979	0.973	0.968	0.959
Motor 2 & driver 3	0.986	0.985	0.984	0.982	0.973
Motor 3 & driver 1	0.962	0.967	0.949	0.944	0.929
Motor 3 & driver 2	0.965	0.959	0.929	0.902	0.865
Motor 3 & driver 3	0.982	0.974	0.965	0.954	0.913
vs. linear model:					
Motor 1 & driver 1	0.753	0.746	0.642	0.470	0.413
Motor 2 & driver 1	0.841	0.844	0.740	0.628	0.516
Motor 3 & driver 1	0.783	0.713	0.358	0.315	0.244

TABLE V
RMSE VALUES OF THE CURRENT MEASUREMENT VS. MODEL ESTIMATE AS A PERCENTAGE OF STALL CURRENT i_s

Motor & Mtr. Drv.	PWM Frequency (Hz)				
	500	1000	5000	10000	20000
vs. coast model:					
Motor 1 & driver 1	7.72%	6.70%	5.12%	5.44%	6.48%
Motor 1 & driver 2	6.95%	6.61%	5.76%	6.16%	7.69%
Motor 1 & driver 3	4.58%	4.65%	4.38%	5.47%	5.98%
Motor 2 & driver 1	6.16%	5.56%	5.39%	5.33%	5.93%
Motor 2 & driver 2	6.33%	6.41%	6.46%	6.50%	6.92%
Motor 2 & driver 3	5.16%	5.07%	4.85%	4.75%	5.65%
Motor 3 & driver 1	7.69%	6.73%	7.57%	7.89%	8.78%
Motor 3 & driver 2	5.25%	5.70%	6.27%	7.62%	11.4%
Motor 3 & driver 3	6.88%	6.82%	7.88%	8.89%	10.1%
vs. linear model:					
Motor 1 & driver 1	20.3%	20.1%	21.9%	23.9%	25.6%
Motor 2 & driver 1	18.5%	17.9%	20.6%	22.9%	25.2%
Motor 3 & driver 1	18.4%	20.0%	26.7%	27.6%	28.6%

motor (motor 3) has an increased error as PWM frequency increases. We believe that this might have been caused by inaccuracies in some parameter values, or nonlinearities such as static friction and backlash which are commonly seen problems in low cost motors. However, the drive/coast model still yielded a significantly more accurate estimates than the linear model.

VI. REAL-TIME DRIVE/COAST EXPERIMENT

We used an eduMIP to demonstrate the viability of our drive/coast model on an unstable, real-time system. A MIP balancing about it's equilibrium point requires small amount of torque to stabilize the system. Since a drive/coast system is highly nonlinear about low duty cycles, a MIP robot is a perfect test platform.

A. Hardware and Setup

We fitted an eduMIP with two DRV8871 motor drivers which can toggle between drive/brake and drive/coast mode. We used a state feedback controller designed for MIP with

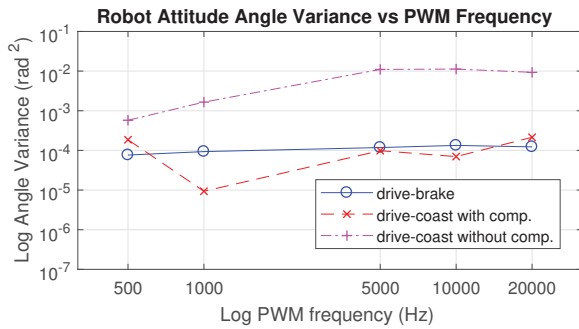


Fig. 6. Plot of the robot attitude angle variance between drive/brake mode, and drive/coast mode with and without compensation.

drive/brake motor drivers for this experiment. Using the same balancing controller, we allowed the robot to balance in place under three modes: drive/brake, and drive/coast with and without coasting model compensation. The coasting model compensation adjusts the PWM duty cycle given by the drive/brake controller to the equivalent duty cycle for drive/coast systems, as discussed in Section IV. Assuming that the coasting model is correct, then the control torque of the drive/coast with compensation should be the same as the drive/brake, resulting in a similar balancing performance. The balancing performance under each mode was evaluated by calculating the variance of the robot's attitude angle during the position hold.

B. Coasting Compensation Results

The drive/coast system performance is significantly improved by implementing the coasting compensation as shown in Fig. 6. With compensation, the drive/coast system has approximately the same performance as the drive/brake system across all tested PWM frequencies, except at PWM frequency of 1000 Hz. We concluded that the compensated PWM duty cycle in this particular frequency hit the sweet spot for small amplitude but high frequency limit cycle for the robot. Without any compensation, the drive/coast system is significantly less stable especially as the PWM frequency increases. This demonstrates that our drive/coast model is accurate and we can effectively compensate the control signal of a drive/coast system in real-time.

VII. CONCLUSION AND FUTURE WORK

In this paper we mathematically derived a novel model for drive/coast systems and demonstrated its accuracy with real-world experiments. In addition, we proposed a real-time implementation method of the model by using the target current and rotor speed to calculate a motor command

which compensates for the nonlinearities of a drive/coast system. Finally, we demonstrated the performance of the drive/coast model and coasting compensation in real time on an eduMIP which showed a significant improvement in stability. This new model opens up the possibility of designing more accurate system dynamic models when using drive/coast motor drivers to offer different system behaviors and advantages compared to its drive/brake counterpart.

Although we showed the effectiveness of the model on a MIP robot, we suggest to experiment further with different types of robots or controls applications that can take advantage of free-spinning motors. In addition, although coasting motor drivers can recharge the battery, we believe that more experimentation should be done to compare overall energetic efficiency between drive/brake and drive/coast motor drivers in low torque and speed applications. Regardless of efficiency, in practice, some robotic systems are bound to use drive/coast motor drivers, and our work provides an accurate method of modeling the dynamics of such drivers.

REFERENCES

- [1] B. Williams, "Power Electronics: Devices, Drivers, Applications, and Passive Components," University of Strathclyde, Glasgow, pp. 348-411, 2013.
- [2] G. Ch. Ioannidis, C. S. Psomopoulos, et. al., "AC-DC & DC-DC Converters for DC Motor Drives," International Conference on Electronics and Communication Systems, 2013.
- [3] K. M. Lynch and F. C. Park, "Modern Robotics: Mechanics, Planning and Control," Cambridge University Press, 2017.
- [4] J. Braun, "Formulae Handbook," Maxon Academy, Sachseln, 2012, pp. 39 - 46.
- [5] U. Kafader, "The selection of high-precision microdrives," Maxon Academy, Sachseln, 2012.
- [6] R. J. Schilling, "Fundamental of Robotics: Analysis and Controls," pp. 268-271, Prentice Hall, 1990.
- [7] X. Nian, F. Peng, and H. Zhang, "Regenerative braking system of electric vehicle driven by brushless DC motor," IEEE Transactions on Industrial Electronics 61.10, pp. 5798-5808, 2014
- [8] H. Wu, S. Cheng, and S. Cui, "A controller of brushless DC motor for electric vehicle," IEEE Transactions on magnetics 41.1, pp. 509-513, 2005.
- [9] M. K. Yoong, et al. "Studies of regenerative braking in electric vehicle," IEEE Conference on Sustainable Utilization and Development in Engineering and Technology, 2010.
- [10] M. Yang, H. Zhou, B. Ma, and K. Shyu, "A cost-effective method of electric brake with energy regeneration for electric vehicles," IEEE Transactions on Industrial Electronics 56.6, pp. 2203-2212, 2009.
- [11] Braga, Newton C, "Robotics, Mechatronics, and Artificial Intelligence: Experimental Circuit Blocks for Engineers," Newnes, 2002.
- [12] J. E. Carryer, R. M. Ohline, and T. W. Kenny, "Introduction to Mechatronic Design," Pearson, 2011.
- [13] N. Morozovsky, R. Moroto, T. Bewley, "RAPID: An inexpensive open source dynamometer for robotics applications," IEEE/ASME Transactions on Mechatronics vol. 18, no. 6, pp. 1855-1860, 2013.
- [14] Bewley, Thomas R. "Numerical renaissance: simulation, optimization, and control," pp. 475, Renaissance Press, San Diego, 2012.