High Precision Point-To-Point Maneuvering of an Experimental Structure Subject To Friction Via Adaptive Impulsive Control.

Rajaey Kased and Tarunraj Singh
kased@eng.buffalo.edu tsingh@eng.buffalo.edu
State University of New York at Buffalo, Buffalo, NY 14260

Abstract— Two adaptive impulsive control techniques designed to eliminate steady state error for a rigid body system subject to friction, and undergoing a point-to-point maneuver are implemented. Pulse width and pulse amplitude pulse width modulations are explored. It is shown that the results of the pulse amplitude modulation never generates limit cycles and has lower steady-state error than the pulse width modulation.

I. INTRODUCTION

The significance of friction to the control community is in its effects on positioning systems and velocity tracking operation. Positioning apparatuses include telescopes, antennas, machine tools, disk drives and robot arm positioning. Velocity control is also relevant in machine tool, disk drive and robot arm industrial applications which require the accurate tracking of a pre-determined trajectory. The effect of friction becomes magnified in the low velocity region near the reference position.

The majority of work done on control of frictional systems is on rigid body systems. Yang and Tomizuka [1] exploited a simple relationship between a pulse input width and the displacement of the rigid body. This utilizes the fact that the rigid body subject to a pulse input never changes the direction of the velocity and thus the Coulomb friction acts like a bias to the input. This scheme, known as Pulse Width Control (PWC), is presented in an adaptive control setting where an estimate of the friction is provided. Wijdeven and Singh [2], modified the PWC approach to increase accuracy in actual discrete implementation of the input. Their technique modulates the pulse height to compensate for a rounded up pulse width and is called Pulse Amplitude Pulse Width Control (PAPWC).

Additional schemes developed for rigid body systems include internal-model following error control [3], PID and state feedback linearization control [4] and variable structure control in order to try to handle qualitatively different friction regimes [5], [6]. Nonlinear PID control has also been developed to overcome the stick-slip behavior of friction [7].

In this paper two techniques, the Adaptive Pulse Width Control (PWC) and the Adaptive Pulse Amplitude Pulse Width Control (PAPWC) proposed by [1] and [2] respectively, are presented for the rigid body system, subject to stiction and Coulomb friction. These techniques are implemented experimentally on the setup presented in section III-A.

Experiments and simulations are presented for qualitative comparison only. This is due to the fact that the actual friction in the system is unknown and an assumed actual value must be used for the simulation. Generally, the assumed value used for the simulation is close to the final estimation of friction obtained during the corresponding experiment run. This accounts for some quantitative comparison obtained from the simulation.

II. MATHEMATICAL FORMULATION

The equation of motion of the rigid body in consideration subject to a positive pulse input, is given as:

$$\ddot{\theta} = \begin{cases} \dfrac{1}{J_1}\left(u_H - f_c - c_1\dot{\theta}\right) & if \ \dot{\theta} \neq 0 \\ 0 & if \ \dot{\theta} = 0 \ \& \ |u_H| < f_s \\ \dfrac{1}{J_1}\left(u_H - sgn(u_H)f_s\right) & if \ \dot{\theta} = 0 \ \& \ |u_H| > f_s \end{cases} \tag{1}$$

The input to the system u_H can be expressed as:

$$u_H = f_{pm}\mathcal{H}(t - T_1) - f_{pm}\mathcal{H}(t - T_2),$$

where f_{pm} is the pulse height magnitude and $\mathcal{H}(t-T)$ is the Heavy-side function which is equal to one for time greater than T.

The Coulomb friction only appears as a bias force since the direction of the rigid body will never change during a single pulse. This eliminates the $sgn\left(\dot{\theta}\right)$ term in the conventional Coulomb friction model permitting linear analysis.

The total distance travelled by the rigid body due to u_H can be found by solving equation (1). Ignoring damping, the distance travelled is given as [1]:

$$\theta(t_{end}) = \pm\frac{f_{pm}\left(f_{pm} - f_c\right)t_p^2}{2J_1 f_c} \tag{2}$$

Equation (2) is the basis of the development of the control schemes presented in this paper.

A. Pulse Width Control Formulation (PWC)

The essential idea of the Pulse Width Control (PWC), is to provide a single pulse to the system near the reference point, such that the total energy (from inertia, damping, control, and friction) is zero at the reference point. This exploits the ability of the friction force to slow down the system with no added control effort (coasting).

The total distance travelled by the inertia subject to viscous damping is

$$d_{damp} = \frac{f_{pm}t_p}{c_1} - \frac{J_1 f_c}{c_1^2} \ln \left[\frac{f_{pm}}{f_c} \left(\exp \left(c_1 \frac{t_p}{J_1} \right) - 1 \right) + 1 \right].$$
(3)

Yang and Tomizuka [1], were the first to use this relationship to come up with an adaptive means of controlling rigid body systems subject to stiction and Coulomb friction. They prove that, for small t_p, the predominant variation of the distance travelled, d, is due to variations in the Coulomb friction. This provides a justification for the use of (2) in all further discussions involving PWC and its variations. As stated, the condition for this assumption to be valid is that the pulse width, t_p, is short. This is practical due to the intended use of these techniques, i.e. to be used near the reference point.

It is assumed that the inertia is known and due to the fluctuation of friction, f_c is considered as an unknown variable. Equation (2) can be rewritten as:

$$d = bt_p^2 sgn(f_p)$$
(4)

where b comprises all of the coefficients of t_p^2 and f_p is the pulse height shown in equation (5).

$$f_p = f_{pm}sgn(e),$$
(5)

where e is the difference between the desired reference point and the current position. The $sgn(f_p)$ is present in order to account for the $\pm$ term in equation (2) due to a positive or negative desired displacement respectively.

It is seen from equation (5) that $sgn(f_p) = sgn(e)$ and therefore equation (4) can be written as:

$$d = bt_p^2 sgn(e)$$
(6)

It is useful to lump all of the coefficients, known and unknown, into one parameter, b, due to the simplicity of the resulting relationship between the pulse width and displacement. Another advantage of this definition is that the coefficient to be estimated now comprises of terms that are known. This allows the adaptation algorithm, which estimates b, to be more robust to the large unknown variations in the friction coefficient, f_c.

The adaptive algorithm, proposed in [1], estimates the coefficient b in equation (6) using a standard recursive identification algorithm. An estimation based on this type of algorithm along with equation (6) is called a self-tuning regulator (STR).

Since b is unknown and varying, it will generally take more than one pulse to reach the desired reference point. Subsequently it is necessary to formulate recursive discrete time equations that describe the motion of the system. Rewriting equation (6) in discrete time results in

$$d(k+1) = bu_p(k)$$
(7)
$$u_p(k) = t_p^2(k)sgn(e(k)),$$
(8)

where $d(k+1)$ is the distance traveled after the kth pulse has been applied. The variable k should not be viewed as the kth sample. It is the kth stoppage of the inertia.

Since equation (7) has one degree of freedom, namely the pulse width, a feedback control law is formulated for $u_p(k)$ to ensure stability of the controller. Assuming b to be known, then $u_p(k)$ is written as

$$u_p(k) = \frac{1}{b} K_c e(k),$$
(9)

If K_c is chosen to be 1, the distance traveled, $d(k+1)$, will be the current error and thus the system will reach the desired position. This is how the STR is formulated: "assuming the unknown parameters to be known, what controller will force the error to zero."

Since b is actually unknown, (9) is replaced by

$$u_p(k) = \frac{1}{\hat{b}(k)} K_c e(k)$$
(10)

and the STR proposed for $\hat{b}(k)$ is

$$\epsilon_0^0(k+1) = d(k+1) - \hat{b}(k)u_p(k)$$
(11)
$$F^{-1}(k+1) = \lambda_1 F^{-1}(k) + \lambda_2 u_p^2(k)$$
(12)
$$\hat{b}(k+1) = \hat{b}(k) + F(k)u_p(k)\epsilon_0^0(k+1)$$
(13)
$$0 < \lambda_1 \leq 1, \ 0 \leq \lambda_2 < 2, \ F(0) > 0,$$
(14)

where $d(k+1)$ is the actual displacement after the kth pulse. F is referred to as the time-varying gain matrix and λ_1 and λ_2 are related to the forgetting of previous data. F, is also known as the adaptation gain if it is a scaler, as it is in this development.

B. Pulse Amplitude Pulse Width Control

The inherent flaw of the PWC, introduced in Section II-A, is that during implementation, the pulse width cannot be exactly what is calculated from equation (8). This is due to the necessary discretization of the pulse width to be a multiple of the sampling time. This is the motivation for the PAPWC scheme.

This has the same principle idea and origins as PWC, except that it creates an extra degree of freedom by varying the pulse height $f_p(k)$, while constraining the pulse width to be a multiple of the sampling time. The expectation is an increase in accuracy.

It is still desired to have the inertia move to a desired distance assuming knowledge of all parameters. This makes equation (6) valid ignoring the effects of damping for small maneuvers. It is now, however, a function of $T_p = nT$. This is depicted as

$$d(t_p) = d(T_p)$$
(15)
$$b \ t_p^2 sgn(e) = b^* \ T_p^2 sgn(e)$$
(16)
$$\frac{f_{pm}(f_{pm} - f_c)}{2J_1 f_c} t_p^2 sgn(e) =$$
$$\frac{f_{pm}^*(f_{pm}^* - f_c)}{2J_1 f_c} T_p^2 sgn(e)$$
(17)

In order to ensure that the energy input into the system is such that equation (15) is satisfied, the constant b must become b^*. From the definition of b, this suggests that the

pulse height, f_{pm}, must be modulated to f_{pm}^*. The pulse width is chosen to be rounded towards infinity to ensure that $f_{pm}^* \le f_{pm}$ (ie. the maximum pulse height).

Solving equation (17) for f_{pm}^* results in

$$f_{pm}^* = .5f_c \pm .5\sqrt{f_c^2 + 4f_{pm}\left(f_{pm} - f_c\right)\frac{t_p^2}{T_p^2}} \qquad (18)$$

Since the term under the square root is always larger than f_c, the $+$ will yield a feasible value of f_{pm}^*, as opposed to a minus sign.

As in equation (9), b must be estimated because the friction coefficient is unknown. It is desired that this be done using Recursive Least-Squares (RLS) Adaptive Filters due to its simplicity. In order to formulate the adaptive algorithm, the output of the system, equation (7), is first written as:

$$d(k+1) = a^T u(k), \qquad (19)$$

where

$$a = \left[\begin{array}{cc} \dfrac{1}{2J_1 f_c} & -\dfrac{1}{2J_1} \end{array}\right]^T$$

and

$$u(k) = \left[\begin{array}{cc} f_p^2(k)t_p^2(k) & f_p(k)t_p^2(k) \end{array}\right]^T.$$

The adaptation algorithm will estimate the vector of unknowns, a, which in-turn will yield an estimate for f_c. It should be noted that $f_p^*(k)$ and $T_p(k)$ can also be used for the calculation of $u(k)$ in equation 19 instead of $f_p(k)$ and $t_p(k)$.

The adaptation algorithm proposed in [8] is:

$$\pi(k) = P(k-1)u(k) \qquad (20)$$

$$K(k) = \frac{\pi(k)}{\lambda + u^H(k)\pi(k)} \qquad (21)$$

$$\xi(k) = d(k) - \hat{a}^H(k-1)u(k) \qquad (22)$$

$$\hat{a}(k) = \hat{a}(k-1) + K(k)\xi^*(k) \qquad (23)$$

$$P(k) = \left[\mathbf{I} - K(k)u^H(k)\right]\lambda^{-1}P(k-1), \qquad (24)$$

where H is the Hermitian transpose and $(\ )^*$ means 'the complex conjugate of'. $K(k)$ and $P(k)$ are known as the *gain vector* and *inverse correlation matrix* respectively. The purpose of calculating the gain vector in two steps instead of one is that it is advantageous from a finite-precision arithmetic point of view [8]. λ is the *forgetting factor*, and $\lambda = 1$ weights all previous data equally.

III. EXPERIMENTAL SETUP AND RESULTS

A. Hardware, Software, System Identification

An experimental testbed of a rotational rigid body subject to friction, is built and the proposed controllers are implemented. A picture of the experiment is shown in Figure 1.

The motor that is used is the *MircoMo 4490 024B* model, with a recommended no load torque operation of 0.192 [N-m]. The nominal torque that is used throughout this work is approximately 3.5×10^{-3} [N-m]. This was calculated for an inertial load of 2.75×10^{-4} [kg-m^2] and a max velocity of approximately 4π [rad/s] reached in 1 [s] (ie. the angular

acceleration: $\alpha = 4\pi$ [rad/s^2]. This torque value is used as a reference point for nominal torque values and is not a restrictive bound.
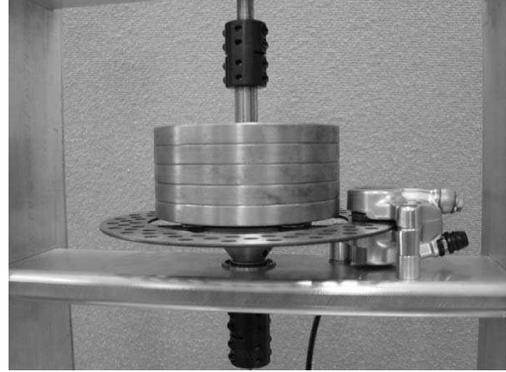

Fig. 1. Rotational Rigid Body Subject to friction

The corresponding amplifier that is used is a *MircoMo MVP2001 A01 Driver Electronics* (*MVP*). The motor is commanded via serial commands to the *MVP*. The dynamics of the electronics of motor (*MVP*, magnetic fields, etc.) are assumed to be negligible. An identification of the Amplifier and Motor shows that a model of a rigid body system subject to damping can be fit reasonably well. Therefore, the inertia and damping of the motor and amplifier can be lumped with that of the rotating mass. A *USDigital E6S* series encoders with 2048 cycles per revolution quadrature (4 pulses per cycle) output is used to query the position of the masse. A *Formula Evoluzione 9.5 Disk Brake Set* was used to mount the inertias and apply the desired friction. In order to translate

System Variable	Estimated Value
$\hat{J}$ [N-m-s^2]	1.35×10^{-5}
$\hat{c}$ $\left[\frac{\text{kg-m}}{\text{s}}\right]$	1.35×10^{-5}
$\hat{f}_c$ [N-m]	4.32×10^{-4}
$\hat{f}_s$ [N-m]	1.58×10^{-3}

TABLE I

EXPERIMENTAL PARAMETERS OF THE RIGID BODY EXPERIMENT.

from physical units to voltage that the hardware understands, equation (25) is used.

$$1[\text{N-M}] = 0.0063V \qquad (25)$$

The units throughout this paper are interchanged depending on the appropriateness in the application. LabVIEW[1] is used as the real time data processing software for all the experiments ([9]). Through extensive system identification, the system parameters are estimated and are shown in Table I.

---

[1]LabVIEW is a registered trademark of National Instruments, Inc.

B. Controller Implementation Results

Ten experiments were performed for both control schemes, the PWC and the PAPWC. The final estimation of the friction coefficient, $b(k)$, is set as the initial estimate for the subsequent run. In order to maintain a relatively low variation in the friction coefficient, the position of the mass is reset after each experiment.

The sampling time is taken to be $T_s = 0.005$ [s]. The desired reference position is assumed to be .2 [rad]. The tolerance values for convergence are taken to be ± 0.005 [rad] and are shown in the figures. The pulse height, f_{pm} is taken to be 0.5 [V], corresponding to .0032 [N-m]. The initial friction value is estimated to be $f_c(0) = .075$ [V] corresponding to 4.725×10^{-4}. The algorithm stops when the inertia is stuck and within the chosen tolerances.

For the experiments, the inertia was considered stuck when $q = 3$ consecutive queries of the encoder are the same. This is an accurate assumption when the encoder resolution is high relative to the sampling of LabVIEW.

The learning gain K_c was taken to be 0.85 for the PWC and 1.2 for the PAPWC. This was chosen to achieve good performance across all estimates of the friction coefficient. The reason for choosing different learning gains for the two control schemes is due to the fact that the pulse height is a variable in PAPWC and the development is such that it is always less than f_{pm} as seen in equation (18). For a control gain of 0.85, the calculated pulse height f_{pm}^* tends to be too small to achieve satisfactory results. In regards to the effect of the validity of the comparison of the two methods; the learning gain will effect to speed of convergence only and not the accuracy of the algorithms. Despite that, a more valid comparison is made in Section III-C.

Figures 2 and 3 show the first experimental and simulation results of the PWC and PAPWC algorithms respectively. The fact that the friction coefficient is uncertain and varying, multiple pulses are required to converge within the desired tolerance.

Due to the lower pulse height values, f_{pm}^*, as compared to f_{pm}, the convergence time for the PAPWC is longer than that of the PWC. The undershoot in the experimental results indicates that the initial friction estimate, along with the current control gain K_c, is underestimated.

The discrepancies between the simulations and experiments are due to friction uncertainties. The true friction value, f_c, doesn't actually exist. In order to perform the simulation, however, a true friction value must be chosen. f_c for the simulation was determined from the friction estimation in the experiment. Assuming that the friction estimate at the end of the experiment is closest to the true friction value. Since friction will vary depending on position of the mass as well as the wear of the friction pad, a different f_c for each simulation run is used.

Figures 4 and 5 shows the experimental and simulation results of both the PWC and PAPWC respectively for the 10th iteration. Both techniques show improvement, showing the effectiveness of the adaptation algorithm. The results also indicate that despite the fact that the friction is varying and
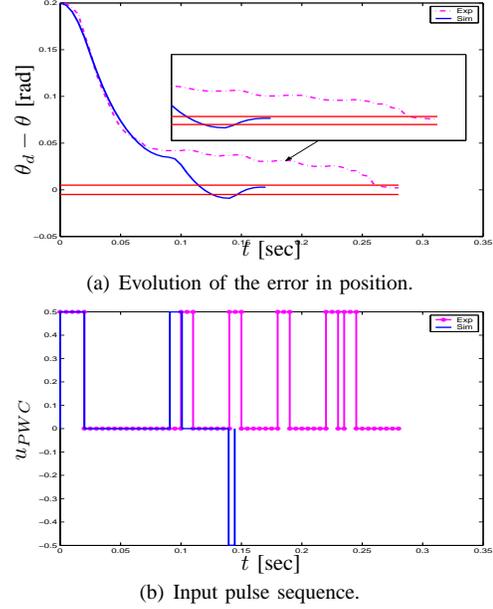

(a) Evolution of the error in position.

(b) Input pulse sequence.

Fig. 2. PWC: Experimental and Simulation results (1st iteration).

(a) Evolution of the error in position.
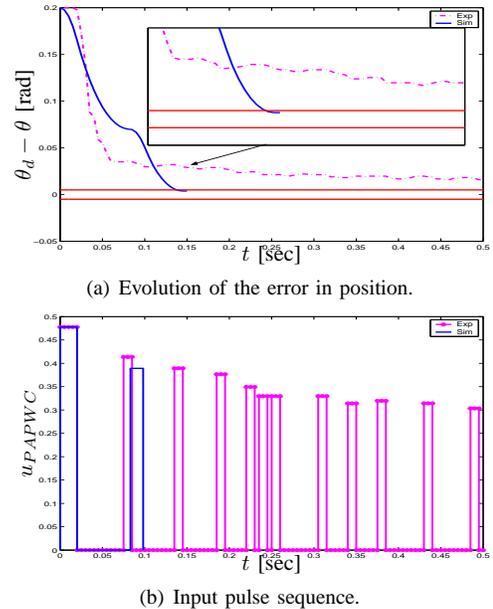

(b) Input pulse sequence.

Fig. 3. PAPWC: Experimental and Simulation results (1st iteration).

uncertain, experimental conditions can be setup such that the variation is controlled.

The pulse sequence of the PWC indicates that a single pulse was required to achieve the desired position tolerance. This is an improvement as compared to the first iteration (Figure 2) The PAPWC has also improved performance as compared to Figure 3. A single pulse, however, is not something which is guaranteed after a certain amount of iterations. If 10 more experiments are performed the same results might not be obtained due to the variation of friction and other un-modelled nonlinearities. Despite that, there still should be improvement from the first iteration.
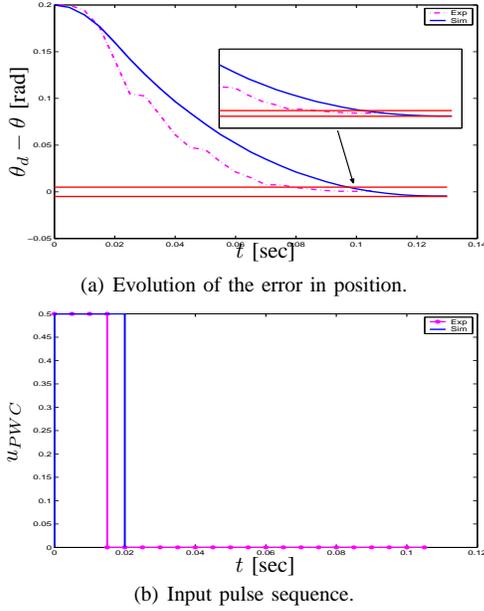
(a) Evolution of the error in position.

(b) Input pulse sequence.

Fig. 4. PWC: Experimental and Simulation results (10th iteration).

(a) Evolution of the error in position.
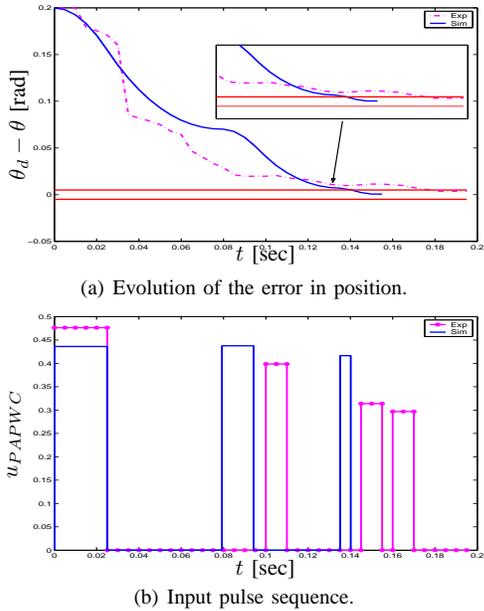

(b) Input pulse sequence.

Fig. 5. PAPWC: Experimental and Simulation results (10th iteration).

C. PWC and PAPWC Comparison Results

Thus far, all of the experimental algorithms have had a convergence tolerance of .005 [rad]. This tolerance is chosen to ensure that a limit cycle does not occur. Due to the nature of PWC, the possibility of a limit cycle around the reference point will occur if the final error is less than the minimum distance d_{min}. The minimum distance, d_{min}, is defined as the minimum distance that the actual inertia can move due to a single pulse of width T, where T is the sampling time.

From equation (3), d_{min} for the PWC is calculated as

$$d_{min} = \frac{f_{pm}T}{c} - \frac{mf_c}{c^2} \ln\left[\frac{f_{pm}}{f_c}\left(\exp\left(c\frac{T}{m}\right) - 1\right) + 1\right] \tag{26}$$

If the error is less than this minimum distance, the inertia will enter a limit cycle.

Since in the PAPWC, the pulse height can take a value anywhere between f_{pm} and f_s, where f_s is the stiction value, the minimum displacement possible in PAPWC is therefore

$$d_{min} = \frac{f_sT}{c} - \frac{mf_c}{c^2} \ln\left[\frac{f_s}{f_c}\left(\exp\left(c\frac{T}{m}\right) - 1\right) + 1\right] \tag{27}$$

That is, the smallest pulse width T and the smallest pulse height f_s.

This is the reason for the higher accuracy as compared to PWC. Furthermore, due to the development of the PAPWC, when the minimum pulse height is reached, the inertia will simply not move because of stiction and a limit cycle does not occur. These results are verified in the subsequent section.

Two experiments are performed in order to compare qualitative behavior between PWC and PAPWC. For the first experiment, the pulse height, f_{pm} is taken to be .55 [V]. The learning gain is set to $K_c = 0.725$ and θ_d remains unchanged at 0.2 [rad]. The sampling time remains at $T = 0.005$. The convergence tolerance is chosen to be .0005. The different values are used to get satisfactory results for the new tolerance level.

As in the previous sections, due to the nature of experimental data and the nonlinear behavior of friction, the simulations are shown along with the experiments to illustrate qualitative behaviors only. It is possible, however, to compare them numerically to some extent.

As is mentioned in [2], the difference between PAPWC and PWC, is that PAPWC should considerably reduce the steady state error. In simulation this can be seen easily by reducing the convergence tolerance to past the PWC tolerance limit but before the PAPWC convergence limit. This results in convergence of the PAPWC to a tolerance that the PWC cannot meet. This is not possible in the experimental setups, however, due to the resolution of the encoder being the limiting agent.

As is mentioned in Section III-A, the encoder counter resolution is 2048 cycles per encoder revolution, corresponding to approximately $7.67 \times 10^{-4} \left[\frac{\text{rad}}{\text{count}}\right]$. The most accurate that the encoder can get to the desired position of .2 [rad] is after 261 counts where it is at .200187 [rad]. The final error of either algorithm will be no better than at this encoder count. It should be noted that due to what has been mentioned with regards to the encoder resolution, there will be a different tolerance levels for different desired reference points.

If the convergence tolerance is set to greater than the encoder resolution then both of the algorithms will eventually converge due to the encoder resolution, otherwise there will never be convergence. This is independent of the value of d_{min}. Meaning, despite the possibility that d_{min} is greater

than the final error defined, and thus cause an overshoot, eventually there will be a coasting period where the final error is within the given tolerance. This is because the coasting period will vary in the experiment, making ~~it just~~ a matter of time that the system converges to within the encoder resolution limits.

In the simulation, however, if the system starts operating in a limit cycle then it will remain in one. This is because the coasting period for each subsequent pulse will be the same and thus the system will never be able to coast into the tolerance range. The coasting period doesn't change in the simulation because the 'true' friction coefficient doesn't change throughout one simulation run. This is seen ~~in Figure~~ 6(a) where the tolerance for the PWC is at $\pm 5 \times 10^{-4}$ [rad] and $d_{min} = 1.034 \times 10^{-2}$ [rad]. However, the experimental results illustrate that the system eventually converges after some initial oscillations since the variation of friction results in some pulses driving the states into the convergence region.

Figure 6(b) is the corresponding results of the PAPWC. The system reaches the tolerance zone quicker than the PWC due to the flexibility provided by modifying both the pulse width and the pulse amplitude. This is consistent with the analytical derivation of d_{min} for the PAPWC which is 1.64×10^{-3} [rad].

It should be noted that it is possible that the PAPWC calculates $|f_p^*(k)|$ such that it is less than f_s, resulting in the system being stuck outside the tolerance limits, whereas if the PWC stops operating due to convergence, it is within the desired tolerances. This will depend on values of K_c and $P(0)$.

Figures 7(a) and 7(b) are compared for an error tolerance of $\pm 1E - 4$. Since this is less than the encoder resolution, which can only get as accurate as $\approx 2E - 4$ [rad], neither of the two algorithms can converge. It is seen, however, that PWC enters a limit cycle, whereas PAPWC gets stuck near the reference trajectory after some overshoot.
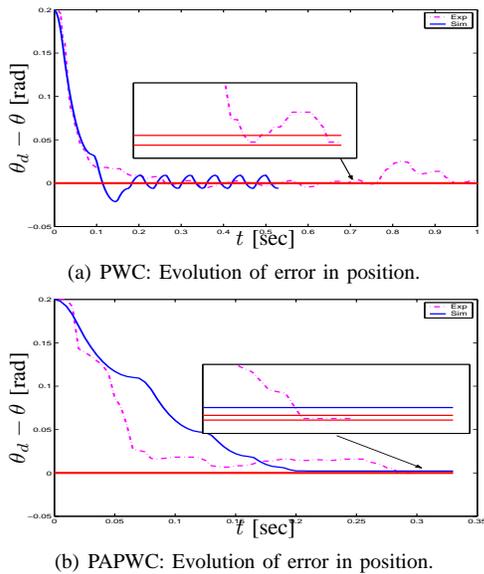

(a) PWC: Evolution of error in position.

(b) PAPWC: Evolution of error in position.

Fig. 6.   PWC/PAPWC: Results of 1st comparison.

(a) PWC: Evolution of error in position.

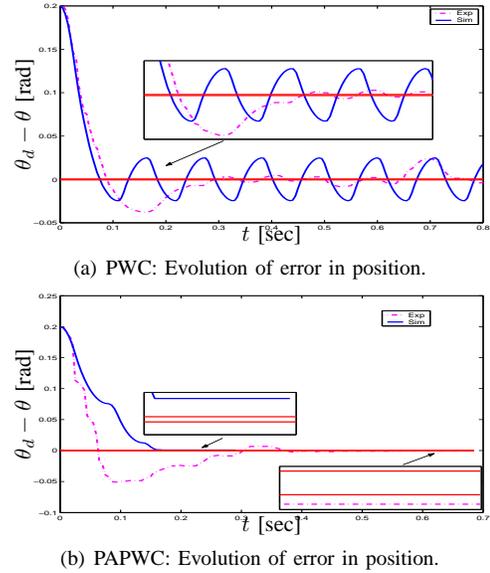(b) PAPWC: Evolution of error in position.

Fig. 7.   PWC/PAPWC: Results of 2nd comparison.

IV. CONCLUSIONS

Two adaptive pulse control schemes, PWC and PAPWC, are implemented on the rigid body subject to friction. Results show that the PAPWC control yields more accurate results than the PWC. It is further shown that the convergence of the experimental setup depends on the encoder resolution and both techniques will eventually converge. If the convergence tolerance is set smaller than the encoder resolution, the PWC will result in an infinite limit cycle, whereas the PAPWC will cause the system to get stuck outside the tolerance level.

REFERENCES

[1] S. Yang and M. Tomizuka. Adaptive pulse width control for precise positioning under the influence of stiction and coulomb friction. *ASME Journal of Dynamic Systems, Measurements, and Control*, 110:221–227, September 1988.

[2] J.J.M. van de Wijdeven and Tarunraj Singh. Adaptive pulse amplitude pulse width control of systems subject to coulomb and viscous friction. In *American Control Conference*, June 2003. Denver, Colorado.

[3] Bong Jeun Kim and Wan Kyun Chung. Motion Control of Precision Positiong Systems Using Adaptive Compensation. In *Proceedings of the American Control Conference*, pages 4589–4594, 2002.

[4] F. Altpeter, P. Myszkorowski, M. Kocher, and R. Longchamp. Friction compensation: Pid synthesis and state control. In *European Control Conference*, page 1, 1997.

[5] Ruh-Hua Wu and Pi-Cheng Tung. Fast Pointing Control for Systems with Stick-Slip Function. *ASME Journal of Dynamic Systems, Measurements, and Control*, 126:614–626, 2004.

[6] H.G. Kwatny, C. Teolis, and M. Mattice. Variable structure control of systems with uncertain nonlinear friction. *Automatica*, 38:1251–1256, 2002.

[7] D. Neevel, B. Armstrong, and T. Jusik. New Results in NPID Control: Tracking, Integral Control, Friction Compensation and Experimental Results. In *IEEE Trans. on Control Systems Technology*, volume 9, pages 399–406, 2001.

[8] S. Haykin. *Adaptive Filter Theory*, chapter Recursive Least-Squares Adaptive Filters, pages 436–444. Prentice Hall Information and System Sciences Series. Prentice Hall, One Lake Street, Upper Saddle River, NJ, 4th edition, 2002.

[9] J. Travis. *LabVIEW For Everyone*. National Instruments virtual instrumentation series. Prentice Hall, One Lake Street, Upper Saddle River, NJ, 2nd edition, 2002.