Control Science and Engineering
2021; 5(1): 13-19
http://www.sciencepublishinggroup.com/j/cse
doi: 10.11648/j.cse.20210501.12

Critical Commentary on Deterministic Artificial Intelligence Applied to Oscillatory Circuits

Eric Miller[1], Timothy Sands[2, *]

[1]Systems Engineering Program, Cornell University, Ithaca, USA

[2]Sibley School of Mechanical and Aerospace Engineering, Cornell University, Ithaca, USA

Email address:
em828@cornell.edu (E. Miller), tas297@cornell.edu (T. Sands)

[*]Corresponding author

To cite this article:
Eric Miller, Timothy Sands. Critical Commentary on Deterministic Artificial Intelligence Applied to Oscillatory Circuits. *Control Science and Engineering*. Vol. 5, No. 1, 2021, pp. 13-19. doi: 10.11648/j.cse.20210501.12

Received: August 12, 2021; **Accepted**: August 21, 2021; **Published**: August 31, 2021

Abstract: With heritage in nonlinear adaptive control (as proposed by Slotine) and physics-based control (as proposed by Lorenz), recently proposed methods referred to as deterministic artificial intelligence (D.A.I.) claim slight performance improvement over the parent methods. This brief communication firstly validates claims of slight improvement, but furthermore highlights a key feature: indications that improvements in observer implementations are the proper path for subsequent development in the field. The manuscript validates the recently published 97% performance improvement over classical methods using nonlinear adaptive methods, with an addition 0.23% performance improvement using D.A.I. compared to nonlinear adaptive control. Furthermore, the work also identifies strong correlation between system performance and observer performance, which is significant since D.A.I. eliminates controller tuning. Thus, observer improvement is recommended for future developments. The recently published 2-norm optimal learning scheme (of Smeresky) is recommended as the next step in the lineage of research in the discipline assuming augmentation with nonlinear state observers.

Keywords: Deterministic Artificial Intelligence, D.A.I., Van Der Pol, Adaptive Control, Physics-Based Controls, State Observers, Luenberger Observers

1. Introduction

This short communication highlights controversial and diverging hypotheses focusing on the differences between a nonlinear adaptive controller and a D.A.I.-based controller in forcing a van der Pol oscillatory system to a prescribed circular trajectory common in the literature, where D.A.I. refers to deterministic artificial intelligence as proposed by Smerseky et al., [1] and by Lobo et al., applied to spacecraft attitude control [2]; by Sands applied to unmanned underwater vehicles [3] as an improvement to classical methods [4], and by Cooper and Heidlauf applied to oscillatory timing circuits [5], which is the focus of this manuscript.

The strengths and challenges of deterministic artificial intelligence include:

1. Strength: broad applicability to disparate scientific disciplines;
2. Strength: elimination of control tuning while maintaining optimality;
3. Strength: combines strength of optimal methods and nonlinear adaptive approaches;
4. Strength: analytic reparameterization enables use of simple regression solutions;
5. Weakness: utilizes physics-based methods to ensure optimality, so the method is clearly more strongly applied to systems determined by known physics phenomena;
6. Weakness: necessitates autonomous trajectory generation;
7. Weakness: tuning of observer becomes more paramount.

While the method of D.A.I. appears attractive since it eliminates controller tuning, this manuscript asserts the fact that rather than eliminating tuning altogether, tuning of the (required) observer becomes relatively more paramount, since variations of the forcing function strongly correlated to the observer error (as a matter of fact the plots of each are hardly distinguishable). The deterministic artificial intelligence (D.A.I.) method stems from a combination of nonlinear adaptive control [6] as taught by Slotine and physics-based controls methods as taught by Lorenz [7], whose combination

is recently proposed in a unifying narrative referred to as feedforward statements of self-awareness in D.A.I., while the self-awareness statements are enhanced by optimal feedback by reparametrizing the problem into a standard regression form [1].

The foundation of D.A.I. lies in its foundational notion of self-awareness statements stemming from physics-based control, which has been applied to a wide variety of disciplines particularly relevant to this manuscript: Kang et al., applied to self-sensing machines, [8] and also high frequency inductance estimation [9]; Alvi applied to three-phase current sensing [10, 11]; Polom applied to transient heat transfer's thermal frequency response [12, 13] and also health monitoring systems [14]; Sheng applied to integrated current sensing of power modules [15]; Van der Broeck applied to power electronic modules [16]; Zhu applied to inductive wireless power transfer [17, 18]; Imamura applied to stator windings for magnetomotive force [19]; Xu applied to torque control accuracy [20]; Flieh applied to rapid servo dynamics [21]; Petit applied to spatial deadbeat control for permanent magnet synchronous machines [22], and injector-based self-sensing of electrical machines [23, 24]; Slininger applied to pulsating voltage injection self-sensing [25].

Assertion 1. The importance of the applicability of the physics-based method of Lorenz is manifest and furthermore establish the basis for assertion of self-awareness statements in D.A.I.

After some arithmetic efforts, the physics-based methods [26, 27] can be seen inherent in Slotine's nonlinear adaptive method [28] as improved by firstly by Fossen [29] and subsequently by Sands [30]. The method of D.A.I. only became easily implementable with codification by Baker et al., of autonomous trajectory generation [31] resulting in the eventual publication of the first book on D.A.I. [32]

This short manuscript very briefly reviews the forced van der Pol oscillator and its use (per the physics-based methods) to formulate D.A.I. self-awareness statements. Three categories of simulations are presented in direct comparison: classical methods, nonlinear adaptive methods, and D.A.I. Simulations firstly validate the superior performance of nonlinear adaptive control and D.A.I. compared to baseline classical control, and the slight improvement of D.A.I. over nonlinear adaptive methods. *The simulations reveal a strong correlation between the performance of D.A.I. and estimation performance of requisite state observers*, and this assertion comprise the main novel contribution in the short communication manuscript leading to recommendations for the next increments of developments in the lineage of D.A.I. research. A secondary contribution is validation of the slight performance improvement in the literature using D.A.I. compared to nonlinear adaptive control.

Research motivations and resulting novelties:

1. Brief re-introduction to burgeoning field of deterministic artificial intelligence;
2. Validation attempt of recently published comparison to classical methods as well as disparate state-of-the art methods (not previously published) to discern efficacy of deterministic artificial intelligence;
3. Critical evaluation of claim in the literature of elimination of tuning producing results that reveal elimination of control tuning is replaced by the paramountcy of tuning state observers.

2. Materials and Methods

The Van der Pol oscillator can be described by position x, velocity $\dot{x}$ and acceleration $\ddot{x}$ is illustrated in equation (1) from reference [5].

$$\ddot{x} + x + \mu(1 - x^2)\,\dot{x} = F(t) \tag{1}$$

where μ is the damping coefficient of the system and $F(t)$ is a driving function that can be used to control the system dynamics. An initial estimate of μ (labeled $\hat{\mu}$) is made and updated by a learning feature as depicted in figure 1 necessitating inclusion of a full state observer.

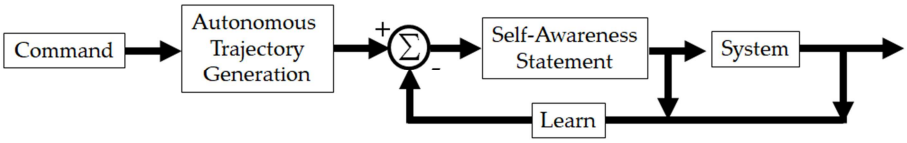


Figure 1. *Topology of deterministic artificial intelligence, where the "learn" function mandates inclusion of a full state observer.*

Assertion 2. Deterministic artificial intelligence self-awareness statements: Self-assertion statement embody the physics-based design methodology of Lorenz in that nonlinear function $\dot{x} = f(x, u, t)$ constitute the forcing function in equation (2) where the subscript 'd' indicates "desired" states provided by autonomous trajectory generation (e.g. per Baker [31])

$$u_{ff} \equiv \ddot{x}_d + x_d + \mu(1 - {x_d}^2)\,\dot{x}_d \tag{2}$$

Assertion 3. Nonlinear adaptive design methodology of Slotine in that nonlinear function $\dot{x} = f(x, u, t)$ constitute the forcing function in equation (3) where the suffixed two terms represent classical proportional + derivative augmentations to the feedforward prequel three terms.

$$F(t) = u \equiv u_{ff} + u_{fb} = \ddot{x}_d + x_d + \mu(1 - {x_d}^2)\,\dot{x}_d + K_p(x_d - x) + K_d(\dot{x}_d - \dot{x}) \tag{3}$$

Assertion 4. Alternative instantiations express the unknown estimate of μ as equation (4) and comparison of this approach to the 2-norm optimal estimate proposed by Smeresky [1] is offered later as likely future research. Smeresky's approach remains heavily reliant upon the performance of the observer.

$$\hat{\mu} = K_p(x_d - x) + K_d(\dot{x}_d - \dot{x}) \tag{4}$$

The two asserted methods are well-compared in the literature, but section 3 will simulate both methods (in MATLAB/SIMULINK R2020a with Runge-Kutta integration and a fixed step-size of 0.0001 seconds) on a Windows-based PC to reveal a strong correlation between performance deducing forcing functions and observer accuracy. Simulations use a classical proportional, integral, derivative (PID) observer topology akin the observer used in [6] to estimate spacecraft attitude is employed rather than the nonlinear observer attempted by Cooper in [5] directly applied to the van der Pol system, theorizing the linear observer should be easily exacerbated by the nonlinearities of the highly nonlinear van der Pol equation. The classical PID form is presumed to be well understood by the reader.

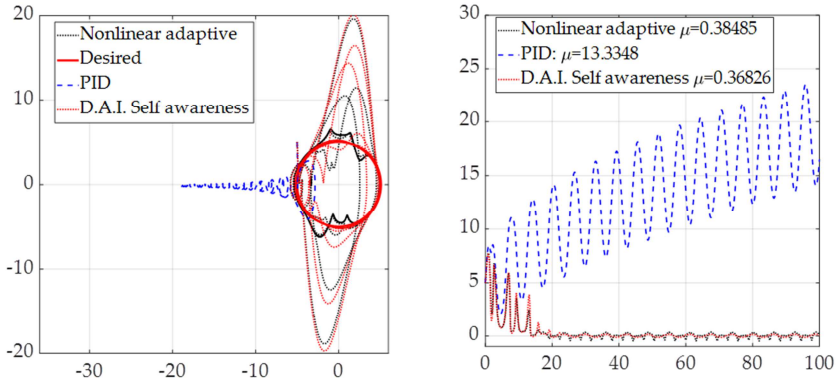
3. Results

A Van der Pol system, due to its inherent dynamics, cannot follow a circular trajectory [5]. To fairly assess the effectiveness of the three control methods (classical control, nonlinear adaptive control, and deterministic artificial intelligence) the desired trajectory is specified to be a circle of radius equal to five as used in the predominant literature [1-3, 5]. The system is initialized with both position and velocity equal to 0.1. The system is given a damping coefficient μ=1 with an initial estimate, ($\hat{\mu} = 0.9$) assumed for the unity true value. Thus, the controllers must overcome the incorrect system parameter μ in addition to forcing the system onto the unnatural desired trajectory, adding an extra level of difficulty to the task of each controller.

3.1. Comparing Classical Control, Nonlinear Adaptive Control, and Deterministic Aritifical Intelligence

Classical proportional plus integral plus derivative feedback controls were designed in accordance with [5], where $K_p = 2.6818$, $K_d = 0.4142$, $K_i = 0$. Nonlinear adaptive control is implemented by [6] as modified by [29] and improved in [30] where [5] illustrates the repeatable control design procedure resulting in adaption gains $K_{pA} = 0.2$, $K_{dA} = 0.01$, $K_{iA} = 0.1$. Deterministic artificial intelligence is implemented with the physics-based feedforward proposed in [5] where van der Pol states are observed with Luenberger topologies per [33] and gains $K_{pO} = 1$, $K_{dO} = 35$, $K_{iO} = 0$. Figures 2 and 3 display simulations with results, while Table 1 shows summary information and metrics for the three trajectories:

1. Classical proportional + integral + derivative control;
2. Nonlinear adaptive control (per Slotine, et al. [26]);
3. D.A.I. self-awareness statements (per Cooper et al. [5]).



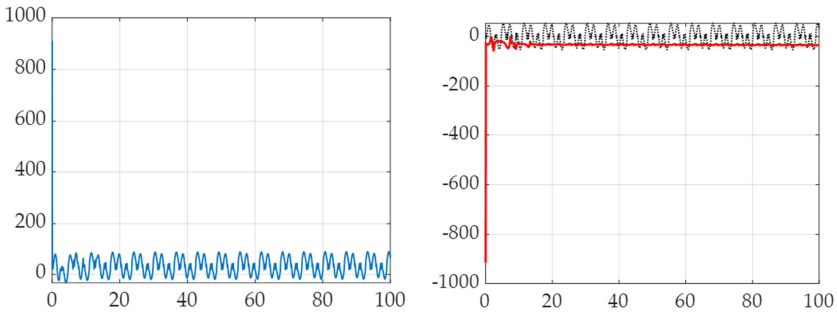
(a) Forcing van der Pol into circular trajectory (b) Tracking Error

Figure 2. *(a) Phase space portrait of desired trajectory, nonlinear adaptively controlled trajectory, DAI controlled trajectory and classical proportional, integral, derivative control (PID) from (0.1, 0.1). (b) Position tracking error plotted over time for adaptively controlled trajectory, D.A.I. and classical PID. In particular, notice the difficulty of control using PID amidst the seeming relative success of both nonlinear adaptive control and D.A.I.*

Figure 2(a) shows the phase space trajectories examined and Figure 2(b) shows the positional tracking error for each relevant trajectory compared to the desired state. Notice classical PID case is firstly completely unable to approach or regulate on the set defined by the desired circular (red solid line) trajectory. Secondly, the classical form diverges exactly as experience by Cooper et. al, [5]. Both the nonlinear adaptive method and deterministic artificial intelligence exhibit a startup transient followed by steady-state regulation on the desired circular trajectory. Tracking error in figure 2b validates the performance improvement achieved in [5] using nonlinear adaptive control, and the 97.01% improvement is listed in table 1. Furthermore, the simulations in figure 2 reveal (in figure 2b) a slight additional improvement using D.A.I (97.24%) compared to nonlinear adaptive control (97.01%).

Table 1. *Validating results of the literature: D.A.I. achieves slightly improved tracking over nonlinear adaptive control which is dramatically superior to classical methods.*

Method	Tracking error	Percent improvement
Classical proportional + integral +derivative control	13.3348	--
Nonlinear adaptive control	0.38485	97.01%
Deterministic artificial intelligence	0.36826	97.24%



a) Observer errors on ordinate vs. time (seconds) on abscissa (b) Forces on ordinate vs. time (seconds) on abscissa

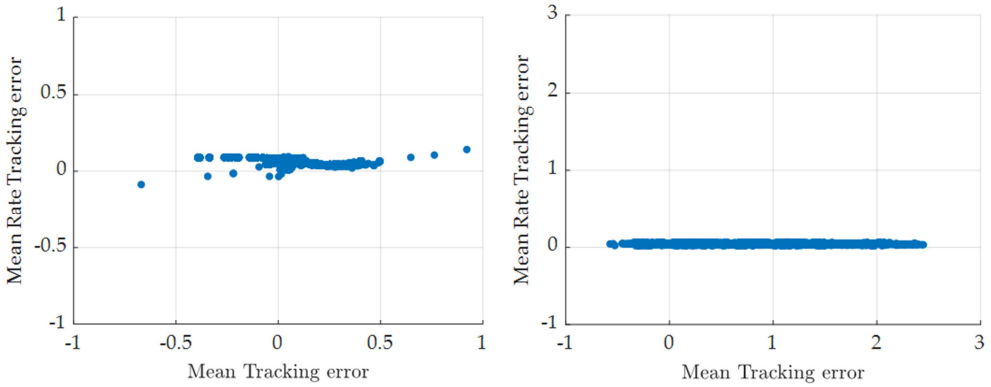
Figure 3. *Illustration of strong correlation between D.A.I. forcing function oscillations to observer errors compared to establishment of steady-state force using nonlinear adaptive control. Compare the observer errors in figure 3(a) and the calculated forcing function utilizing D.A.I. (the black dotted lines in figure 3(b), while noticing the forcing function using nonlinear adaptive control (the solid-red line in figure 3(b)) is relatively insensitive to observer errors.*

Figure 3 shows the estimation of μ over time for the control methods simulated and the resulting forcing functions reveal in the key correlation. Figure 3(a) displays observer errors, while subfigure (b) displays the calculated forcing functions comparing nonlinear adaptive control and deterministic artificial intelligence. While the forcing function using nonlinear adaptive control achieves non-oscillatory steady-state, the forcing function using deterministic artificial intelligence exhibits oscillatory steady-state behavior that very closely mimics the performance of the Luenberger observer of the van der Pol states.

Illustration of strong correlation between D.A.I. forcing function oscillations to observer errors compared to establishment of steady-state force using nonlinear adaptive control. Notice the forcing function using nonlinear adaptive control (the solid-red line in figure 3(b)) is relatively insensitive to observer errors.

3.2. Statistical Analysis of Sensitivity to Variations

This section reveals the results of Monte Carlo simulations varying the assumed initial values of μ and assumed values of $\hat{\mu}$. More than ten thousand simulations runs (respectively) were performed varying assumed values $0.7 < \hat{\mu} < 1.1$ and $0.9 < \mu < 1.1$. Figure 4a. and 4b. respectively display the results revealing slight positive biases.



a) Scatter plot varying actual coefficient μ (b) Scatter plot varying initial coefficient estimate $\hat{\mu}(0)$

Figure 4. *Rate versus voltage errors for 10,000 simulation runs with various values of actual coefficient μ and assumed (designed) initial estimate $\hat{\mu}(0)$. In the context of circuitry applications, the rate and position represent current and voltage, respectively. The observed positive bias in the two subfigures reflects a positive bias in tracking the voltage of the desired trajectory of Figure 1(a).*

Monte Carlo simulations reveal relatively greater sensitivity to initial estimates of $\hat{\mu}$ than variations in the actual system parameter μ. Both system variations and coefficient estimates variations lead to relatively more rate tracking errors than voltage tracking errors as displayed in figures 4a and 4b.

4. Conclusions

This brief communication validates recently published results utilizing nonlinear adaptive control and also deterministic artificial intelligence (D.A.I.) to force a van der Pol oscillatory circuit to track a regular circular trajectory amplifying its use as for indigenous timing. In doing so, simulations reveal the strong correlation of the D.A.I. method's performance and the performance of the state observer used to estimate the control. While simulation of both methods reveal roughly 97% performance improvement (with D.A.I. doing slightly better than nonlinear adaptive control), the clear correlation of D.A.I. forcing function to the accuracy of estimation by the observer lead to obvious paths for future research.

5. Future Research Recommendations

Despite relatively superior performance (compared to nonlinear adaptive control), the newer methodology of deterministic artificial intelligence, the feedback mechanism is shown in this manuscript to be worthy of future consideration. Two avenues of pursuit seem clearly manifest: 1) improvements in estimation (e.g. nonlinear observers), and 2) implementation of the formerly proposed optimal learning as feedback. The 2-norm optimal learning of the self-awareness proposed by Smeresky et al., [1] is given by a typical least-squares formulation illustrated in equations (5) – (7).

$$u = \ddot{x}_d + x_d + \hat{\mu}(1 - x_d^2)\,\dot{x}_d = [\Phi]\{\Theta\} = [\ddot{x}_d \quad x_d \quad (1 - x_d^2)\,\dot{x}_d][1 \quad 1 \quad \mu]^T \tag{5}$$

where d subscripts indicate the desired states. Combing equations (1) and (2) gives

$$\ddot{x} + x + \mu(1 - x^2)\,\dot{x} = u = [\ddot{x}_d \quad x_d \quad (1 - x_d^2)\,\dot{x}_d][1 \quad 1 \quad \mu]^T \tag{6}$$

Equation (4) shows the pseudoinverse solution for $[\Theta] = [1 \quad 1 \quad \mu]^T$ from which an estimate of μ can be extracted and used to update $F(t)$

$$[1 \quad 1 \quad \mu]^T = (\Phi^T\Phi)^{-1}\Phi\, u \tag{7}$$

The value of u is found by feeding the position error through a potentially nonlinear observer. The product of u and $(\Phi^T\Phi)^{-1}$ provides the estimate of μ.

Secondly, future elimination of the linear observer (of PID form or alternatively Luenberger form [33]) in favor of the nonlinear observer should be investigated to enhance performance in accordance with the correlation demonstrated here between performance and observer accuracy. This second line of investigation stems naturally from the conclusions asserted in this manuscript about the strong correlation between observer performance and tracking errors.

Author Contributions

Conceptualization, E. M. and T. S.; methodology, E. M. and T. S.; software, E. M. and T. S.; supervision, T. S.; funding acquisition, T. S. All authors have read and agreed to the published version of the manuscript. Please turn to the CRediT taxonomy for the term explanation. Authorship has been limited to those who have contributed substantially to the work reported.

Conflicts of Interest

The author declares no conflict of interest.

Appendix

The following MATLAB code was used to call a SIMULINK model comprised of figure 1's implementation of equations (1)-(4) to produce the results presented in the manuscript.

```
clear all; close all; clc; warning('off','all');

% COMMENT OUT "clear all; close all; clc" in InitFcn CALLBACKS
% "COMMENT OUT" THESES GAINS IN mdl InitFcn CALLBACKS

KpO=[];KdO=[];KiO=[]; Gains=[]; GoodGains=[]; Count=0;

  for KpO=1:1:2
  for KdO=1:1:2
  for KiO=1:1:2
  sim('IterateObserver_eric_2020a');
  if mean(TrackingError)<0.01; Gains=[KpO;KdO;KiO]; GoodGains=[GoodGains Gains]; end
  clc; Count=Count+1
  end
  end
  end

  % x=out.Actual(:,1);
  % xdot=out.Actual(:,2);
  %
  % xD=out.Desired(:,1);
  % xDdot=out.Desired(:,2);
  %
  % TrackingError=xD-x;
  %
  % t=out.tout(:,1);
  % MU=ones(1,max(size(out.tout(:,1))));
  % uff=out.uff(:,1);
  % u=out.u(:,1);
  %
  % figure(1); subplot(1,2,1);
  % plot(x,xdot,'k:','linewidth',2); hold on; plot(xD,xDdot,'r','linewidth',3); hold off; axis([-13,13,-13,13]);
  % xlabel('x(t)','fontsize',16,'fontname','Palatino Linotype');
  % ylabel('xdot(t)','fontsize',16,'fontname','Palatino Linotype');
  % set(gca,'fontsize',16); set(gca,'fontname','Palatino Linotype'); grid;
  %
  % subplot(1,2,2);
  % plot(out.tout,TrackingError,'k:','linewidth',2);
  % xlabel('time, t(seconds)','fontsize',16,'fontname','Palatino Linotype');
```

```
% ylabel('Tracking Error','fontsize',16,'fontname','Palatino Linotype');
% set(gca,'fontsize',16); set(gca,'fontname','Palatino Linotype'); grid;
% legend(['\mu=' num2str(mean(TrackingError)) '; \sigma=' num2str(std(TrackingError)) ])
%
% figure(2);
% plot(out.tout,out.u,'linewidth',2);
% xlabel('time, t (seconds)','fontsize',16,'fontname','Palatino Linotype');
% ylabel('F^*(t)','fontsize',16,'fontname','Palatino Linotype');
% set(gca,'fontsize',16); set(gca,'fontname','Palatino Linotype'); grid;
% legend(['\mu=' num2str(mean(out.u))])
%
% figure (3);
% plot(out.tout,MU,'r','linewidth',3); hold on; plot(out.tout,out.muHatAdapted,'k:','linewidth',2); hold off;
% xlabel('time, t (seconds)','fontsize',16,'fontname','Palatino Linotype');
% ylabel('\mu Adapted','fontsize',16,'fontname','Palatino Linotype');
% set(gca,'fontsize',16); set(gca,'fontname','Palatino Linotype'); grid;
% legend(['\mu_\mu_(_t_)=' num2str(mean(out.muHatAdapted)) '; \sigma_\mu_(_t_)=' num2str(std(out.muHatAdapted)) '; K_p=' num2str(KpA) '; K_d=' num2str(KdA) '; Ki=' num2str(KiA) ],'fontsize',16,'fontname','Palatino Linotype')
%
% figure(4);
% subplot(1,2,1); plot(out.tout,MU,'r','linewidth',3); hold on; plot(out.tout,out.muHatLearned,'k:','linewidth',2); hold off;
% xlabel('time, t (seconds)','fontsize',16,'fontname','Palatino Linotype');
% ylabel('Learned \mu(t) Estimate','fontsize',16,'fontname','Palatino Linotype');
% set(gca,'fontsize',16); set(gca,'fontname','Palatino Linotype'); grid;
% legend(['\mu_\mu=' num2str(mean(out.muHatLearned)) '; \sigma_\mu=' num2str(std(out.muHatLearned))])
%
% subplot(1,2,2); plot(out.tout,out.uff,'r','linewidth',2);
% hold on; plot(out.tout,out.uHatLearned,'k:','linewidth',2); hold off;
% xlabel('time, t (sec)','fontsize',16,'fontname','Palatino Linotype');
% ylabel('Learned & Actual Forces, f(t)','fontsize',16,'fontname','Palatino Linotype');
% set(gca,'fontsize',16); set(gca,'fontname','Palatino Linotype'); grid;
% legend(['\mu_F=' num2str(mean(out.uHatLearned))],['\mu_u=' num2str(mean(out.uff))])
```

The following MATLAB code was used to call a SIMULINK model comprised of figure 1's implementation of equations (1)-(4) to produce the Monte Carlo results presented in the manuscript.

```
clear all; close all; clc; warning('off','all');
R=[]; MUHAT0=[]; ERROR=[]; RATEERROR=[];

% COMMENT OUT "clear all; close all; clc" in InitFcn CALLBACKS
% "COMMENT OUT" muhat0 mdl InitFcn CALLBACKS

muhat0=0.9;

for i=1:10000
i
muhat0=muhat0+0.2*randn(1); MUHAT0=[MUHAT0 muhat0];
sim('Iterate_vanderPol_eric_2020a');
ERROR=[ERROR mean(ans.Actual(:,1)-ans.Desired(:,1))];
RATEERROR=[RATEERROR mean(ans.Actual(:,2)-ans.Desired(:,2))];
end

scatter(ERROR,RATEERROR,'filled');
axis([-3,3,-0.3,0.3])
set(gca,'fontsize',16); set(gca,'fontname','Palatino Linotype'); grid;
ylabel('Mean Rate Tracking error','fontsize',16,'fontname','Palatino Linotype','interpreter','tex');
xlabel('Mean Tracking error','fontsize',16,'fontname','Palatino Linotype','interpreter','latex');
```

References

[1] Smeresky, B.; Rizzo, A.; Sands, T. Optimal Learning and Self-awareness versus PDI. *Algorithms* 2020, *13* (1), 23.

[2] Lobo, K.; Lang, J.; Starks, A.; Sands, T. Analysis of Deterministic Artificial Intelligence for Inertia Modifications and Orbital Disturbance. *Int. J. Control Sci. Eng.* 2018, 8, 53.

[3] Sands, T. Development of deterministic artificial intelligence for unmanned underwater vehicles (UUV). *J. Mar. Sci. Eng.* 2020, *8* (8), 578.

[4] Sands, T.; Bollino, K.; Kaminer, I.; Healey, A. Autonomous Minimum Safe Distance Maintenance from Submersed Obstacles in Ocean Currents. *J. Mar. Sci. Eng.* 2018, *6* (3), 98.

[5] Cooper, M.; Heidlauf, P.; Sands, T. Controlling Chaos—Forced van der Pol Equation. *Mathematics* 2017, *5* (4), 70.

[6] Slotine, J., Weiping Li. *Applied Nonlinear Control*. Prentice Hall: Englewood Cliffs, U.S.A, 1991; pp. 422-433.

[7] Sands, T.; Lorenz, R. Physics-Based Automated Control of Spacecraft. In Proceedings of the AIAA Space Conference & Exposition, Pasadena, CA, USA, 14–17 September 2009.

[8] Kang, Ye Gu & Reigosa, David & Lorenz, Robert. (2020). SPMSMs HFI Based Self-Sensing Using Intentional Magnetic Saturation. IEEE Access. PP. 1-1. 10.1109/ACCESS.2020.3045275.

[9] Kang, Ye Gu & Reigosa, David & Sarlioglu, Bulent & Lorenz, Robert. (2020). D and Q-axes Inductance Estimation and Self-Sensing Condition Monitoring using 45 Angle High-Frequency Injection. IEEE Transactions on Industry Applications. PP. 1-1. 10.1109/TIA.2020.3029993.

[10] Alvi, Muhammad & Sheng, Minhao & Lorenz, Robert & Jahns, Thomas. (2020). Power Module Design for Integrated Three-Phase Current Sensing Using a Single 3-D Point Field Detector. 3328-3335. 10.1109/ECCE44975.2020.9235335.

[11] Alvi, Muhammad & Sheng, Minhao & Lorenz, Robert & Jahns, Thomas. (2020). SiC Power Module Design for High Bandwidth Integrated Current Sensing using a Magnetoresistive Point Field Detector. 1506-1512. 10.1109/APEC39645.2020.9124383.

[12] Polom, Timothy & Lorenz, Robert. (2020). Expediting Transient Thermal Frequency Response Characterization and Sensitivity Analysis. 10.1109/ECCE44975.2020.9235671.

[13] Polom, Timothy & Lorenz, Robert. (2020). Spatial Thermal Frequency Response Measurement of Power Semiconductor Equipment.

[14] Polom, Timothy & Van der Broeck, Christoph & Doncker, Rik & Lorenz, Robert. (2020). Designing Power Module Health Monitoring Systems Based on Converter Load Profile. IEEE Transactions on Industry Applications. PP. 1-1. 10.1109/TIA.2020.3023070.

[15] Sheng, Minhao & Alvi, Muhammad & Lorenz, Robert. (2020). GMR-based Integrated Current Sensing in SiC Power Modules with Phase Shift Error Reduction. IEEE Journal of Emerging and Selected Topics in Power Electronics. PP. 1-1. 10.1109/JESTPE.2020.3028275.

[16] Van der Broeck, Christoph & Polom, Timothy & Lorenz, Robert & Doncker, Rik. (2020). Real-Time Monitoring of Thermal Response and Life-Time Varying Parameters in Power Modules. IEEE Transactions on Industry Applications. PP. 1-1. 10.1109/TIA.2020.3001524.

[17] Zhu, Guangqi & Lorenz, Robert & Wan, Cheng. (2020). Optimization of "I" Type Shielding for Low Air-Gap Magnetic and Electric Fields Inductive Wireless Power Transfer. 1208-1213. 10.1109/ITEC48692.2020.9161730.

[18] Zhu, Guangqi & Lorenz, Robert & Diao, Fei & Zhao, Yue. (2020). Low Loss Online Capacitor Tuning Method for Reactive Power Reduction of Inductive Wireless Power Transfer System Under Misalignment. 961-965. 10.1109/ITEC48692.2020.9161382.

[19] Imamura, Ryoko & Lorenz, Robert. (2020). Stator Winding MMF Analysis for Variable Flux and Variable Magnetization Pattern-PMSMs. IEEE Transactions on Industry Applications. PP. 1-1. 10.1109/TIA.2020.2981605.

[20] Xu, Yang & Morito, Chikara & Lorenz, Robert. (2020). Accurate Discrete-Time Modeling for Improved Torque Control Accuracy for Induction Machine Drives at Very Low Sampling-to-Fundamental Frequency Ratios. IEEE Transactions on Transportation Electrification. PP. 1-1. 10.1109/TTE.2020.2977204.

[21] Flieh, Huthaifa & Slininger, Timothy & Lorenz, Robert & Totoki, Eigo. (2020). Self-Sensing via Flux Injection With Rapid Servo Dynamics Including a Smooth Transition to Back-EMF Tracking Self-Sensing. IEEE Transactions on Industry Applications. PP. 1-1. 10.1109/TIA.2020.2970150.

[22] Petit, Marc & Lorenz, Robert & Gagas, Brent & Secrest, Caleb & Sarlioglu, Bulent. (2019). Spatial Deadbeat Torque Control for Six-Step Operation. 10.1109/ECCE.2019.8912507.

[23] Petit, Marc & Sarlioglu, Bulent & Lorenz, Robert & Gagas, Brent & Secrest, Caleb. (2019). Using Flux and Current for Robust Wide-Speed Operation of IPMSMs. 10.23919/EPE.2019.8915510.

[24] Petit, Marc & Sarlioglu, Bulent & Lorenz, Robert & Van der Broeck, Christoph. (2019). Carrier Separation Techniques for Improved Disturbance Rejection of Injection-Based Self-Sensing Control. 10.1109/SLED.2019.8896264.

[25] Slininger, Timothy & Petit, Marc & Flieh, Huthaifa & Chien, Shao-Chuan & Ku, Li-Hsing & Lorenz, R.. (2019). Full Order Discrete-Time Modeling for Accurate and Speed-Independent Pulsating Voltage Injection Self-Sensing. 1-6. 10.1109/SLED.2019.8896346.

[26] Polom, T.; van der Broeck, C.; De Doncker, R.; Lorenz, R. "Designing Power Module Health Monitoring Systems Based on Converter Load Profile," in IEEE Transactions on Industry Applications, vol. 56, no. 6, pp. 6711-6721, Nov.-Dec. 2020, doi: 10.1109/TIA.2020.3023070.

[27] Polom, T.; Andresen, M.; Liserre, M.; Lorenz, R. "Experimentally Extracting Multiple Spatial Thermal Models that Accurately Capture Slow and Fast Properties of Assembled Power Semiconductor Converter Systems," 2018 IEEE Energy Conversion Congress and Exposition (ECCE), Portland, OR, USA, 2018, pp. 7391-7398, doi: 10.1109/ECCE.2018.8557855.

[28] Boffi, N.; Slotine, J. Implicit regularization and momentum algorithms in nonlinear adaptive control and prediction. arXiv: 1912.13154v6 [math. OC], available online (accessed 29 March 2021): https://arxiv.org/abs/1912.13154.

[29] Fossen, T. Comments on Hamiltonian adaptive control of spacecraft by J. J. E. Slotine and M. D. Di Benedetto. *IEEE Trans. Auto. Con.* 1993, *38* (4) 671.

[30] Sands, T.; Kim, J. J.; Agrawal, B. N. Improved Hamiltonian adaptive control of spacecraft. In Proceedings of the IEEE Aerospace, Big Sky, MT, USA, 7–14 March 2009; IEEE Publishing: Piscataway, NJ, USA, 2009; pp. 1–10.

[31] Baker, K.; Cooper, M.; Heidlauf, P.; Sands, T. Autonomous trajectory generation for deterministic artificial intelligence. *Electr. Electron. Eng.* 2018, *8* (3), 59.

[32] Sands, T. (Ed.). *Deterministic Artificial Intelligence.* IntechOpen: London, 2020. ISBN: 978-1-78984-111-4.

[33] Cooper M., Heidlauf, P. Nonlinear Feed Forward Control of a Perturbed Satellite using Extended Least Squares Adaptation and a Luenberger Observer. *J Aeronaut Aerospace Eng* 2018, *7* (1), 205. doi: 10.4172/2168-9792.1000205.