A Demo for "Data-driven Fault Detection for Lipschitz Nonlinear Systems: From Open to Closed-loop Systems"

Authors: Hongtian Chen, Zhongyan Liu, and Biao Huang*

Part A: Descriptions of System Models

Consider the direct-current motor control system shown in Fig. 1, where two proportional-integral (PI) controllers are used and named as the current controller $\mathbf{K}^{\mathrm{c}}$ and the speed controller $\mathbf{K}^{\mathrm{s}}$ [1*]. In Fig. 1, SU represents the sub-unit. Its actual system is typically nonlinear, and its parameters are presented in Table I.

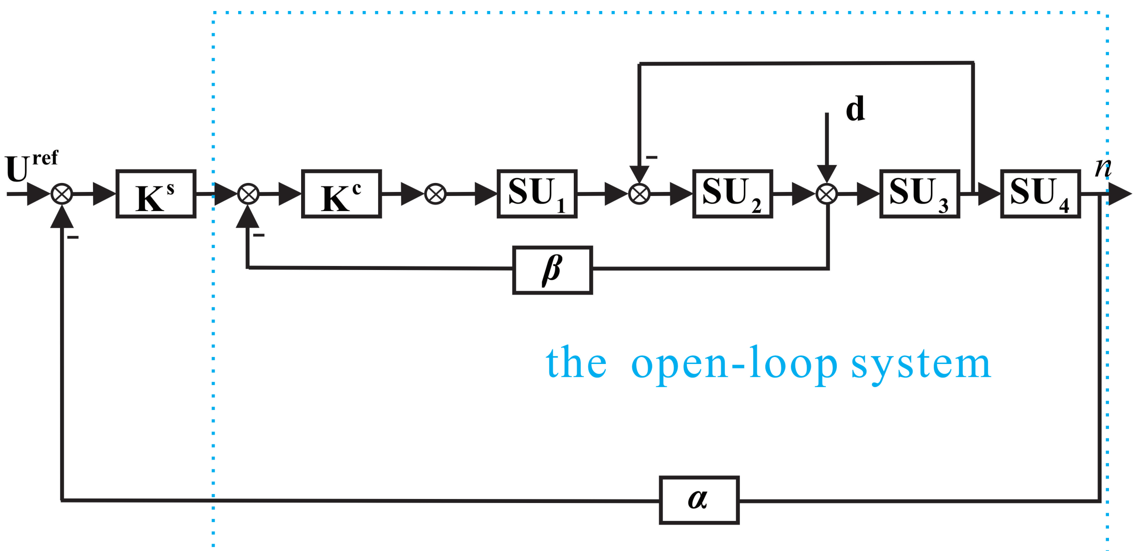

Fig. 1: A structure of the direct-current motor control system.

Part B: Off-line Training

In the demo, the open-loop system is chosen as the part highlighted by the blue box; $\mathbf{K}^{\mathrm{s}}$ and $\mathbf{U}^{\mathrm{ref}}$ are treated as the controller and external input $\mathbf{u}$, respectively. Note that only $\mathbf{u}$ and $\mathbf{y}$ are available for training. In order to have an insightful observation about the proposed fault detection (FD) scheme, a schematic is depicted in Fig. 2.

In order to model the unknown nonlinearity of $\mathcal{K}$, off-line samples are collected from $\mathbf{u}$ and

TABLE I: Parameters used in the direct-current motor control system

tab1

Description	Definition	Value or Transfer function
$\mathbf{U}^{\mathbf{ref}}$	Reference voltage	[5 25]V (*varying*)
$\mathbf{K}^{\mathbf{s}}$	Controller in the outer loop	$9.29 \times \left\{\mathbf{e}^{\mathbf{s}}(t) + \frac{1}{0.087}\int_0^t \mathbf{e}^{\mathbf{s}}(\tau)d\tau\right\}$
$\mathbf{K}^{\mathbf{c}}$	Controller in the inner loop	$0.545 \times \left\{\mathbf{e}^{\mathbf{c}}(t) + \frac{1}{0.0313}\int_0^t \mathbf{e}^{\mathbf{c}}(\tau)d\tau\right\}$
$\mathbf{e}^{\mathbf{s}}$	Feedback error in the outer loop	– (*varying*)
$\mathbf{e}^{\mathbf{c}}$	Feedback error in the inner loop	– (*varying*)
$\mathbf{SU_1}$	sub-unit 1	$\frac{36}{0.0017s+1}$
$\mathbf{SU_2}$	sub-unit 2	$\frac{1}{0.58(0.313s+1)}$
$\mathbf{SU_3}$	sub-unit 3	$\frac{0.58}{0.063s+1}$
$\mathbf{SU_4}$	sub-unit 4	$\frac{1}{0.11}$ min/r
α	Feedback gain in the outer loop	0.067V·min/r
β	Feedback gain in the inner loop	0.125V/A

where "–" represents the variable with a varying value.

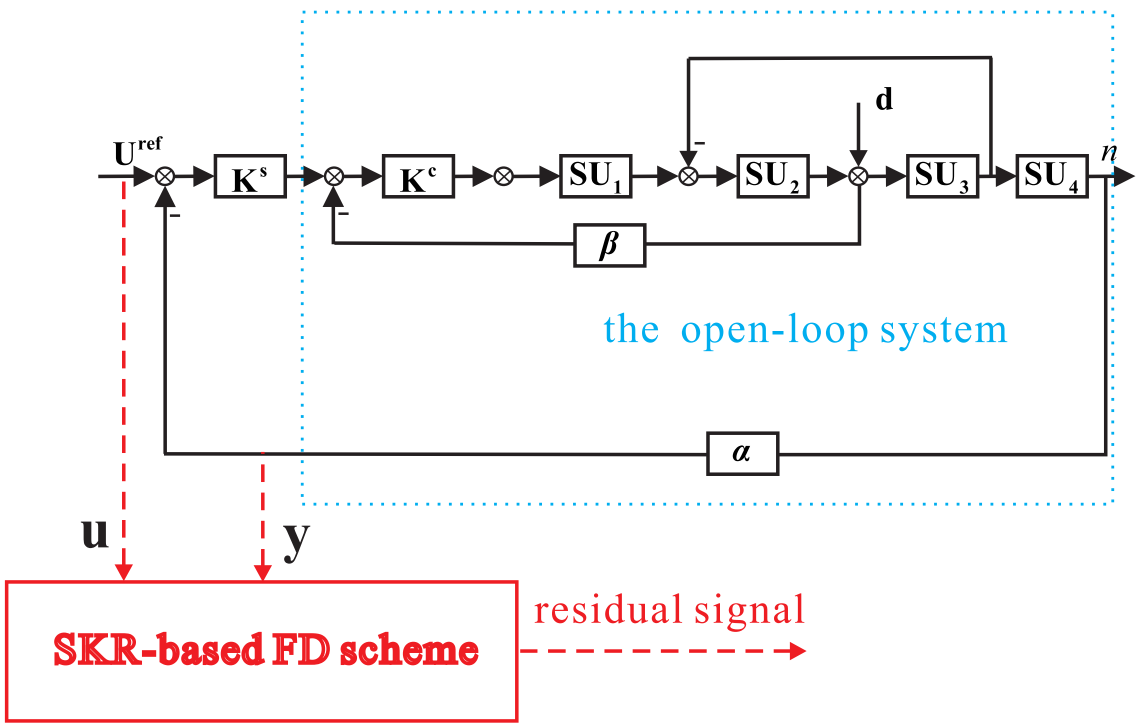

Fig. 2: A structure of the direct-current motor control system.

fig2

y, where the reference signal $\mathbf{U}^{\mathbf{ref}}$ is chosen to follow a uniform distribution ranging from 5V to 25V. In the following simulation studies, we assume that the controller $\mathbf{K}^{\mathbf{s}}$ is unknown, i.e., the information of $\mathbf{K^s}$ is unavailable in both the off-line training and online tests. In the demo, $s_p = 3$ is used as the past horizon and $s_f = 1$ is chosen as the future horizon.

Without loss of generality, $\mathbf{U^{ref}}$ and $n \cdot \alpha$ can be regarded as the system input $\mathbf{u}$ and output $\mathbf{y}$ of the feedback control system, respectively. Corresponding to the definitions used in the article (entitled "Data-driven Fault Detection for Lipschitz Nonlinear Systems: From Open to Closed-loop Systems"), the variables used in parameter identification and FD are given as follows.

$$\mathbf{u_{k,s_p}} = \begin{bmatrix} \mathbf{U^{ref}}(k+1) \\ \vdots \\ \mathbf{U^{ref}}(k+s_p) \end{bmatrix} \in R^3, \ \mathbf{y_{k,s_p}} = \begin{bmatrix} n(k+1)\cdot\alpha \\ \vdots \\ n(k+s_p)\cdot\alpha \end{bmatrix} \in R^3. \tag{Dm1} \quad \boxed{\texttt{eq1}}$$

Therefore, $\hat{\mathcal{K}}$ for the whole feedback control system is the following projection:

$$\hat{\mathcal{K}} : U^4 \times Y^3 \to Y \tag{Dm2} \quad \boxed{\texttt{eq2}}$$

such that

$$\hat{\mathcal{K}} \begin{bmatrix} \mathbf{u_{k,s_p}} \\ \mathbf{y_{k,s_p}} \\ \mathbf{u}(k+s_p+1) \end{bmatrix} - \mathbf{y}(k+s_p+1) \to 0 \tag{Dm3} \quad \boxed{\texttt{eq3}}$$

for fault-free cases. Assume that $N = 4 \times 10^5$ samples are collected in the off-line phase. As suggested in [2*], the division of the data set could be achieved via other manners for reducing the heavy data burden. For instance, $\mathbf{U^{ref}}(k+1)$ and $\mathbf{U^{ref}}(k+2)$ fall into the same range in the k time instant, allowing for a simplified version of Python codes in the demo.

In the off-line phase, the training data together with the off-line modeling is shown in Fig. 3. Based on the total modeling error, squared prediction error (SPE) is chosen as the test statistic to perform the FD task. In this demo, the threshold is given, which can also be determined via Theorem 5 in the manuscript.

<u>Part C</u>: Online test

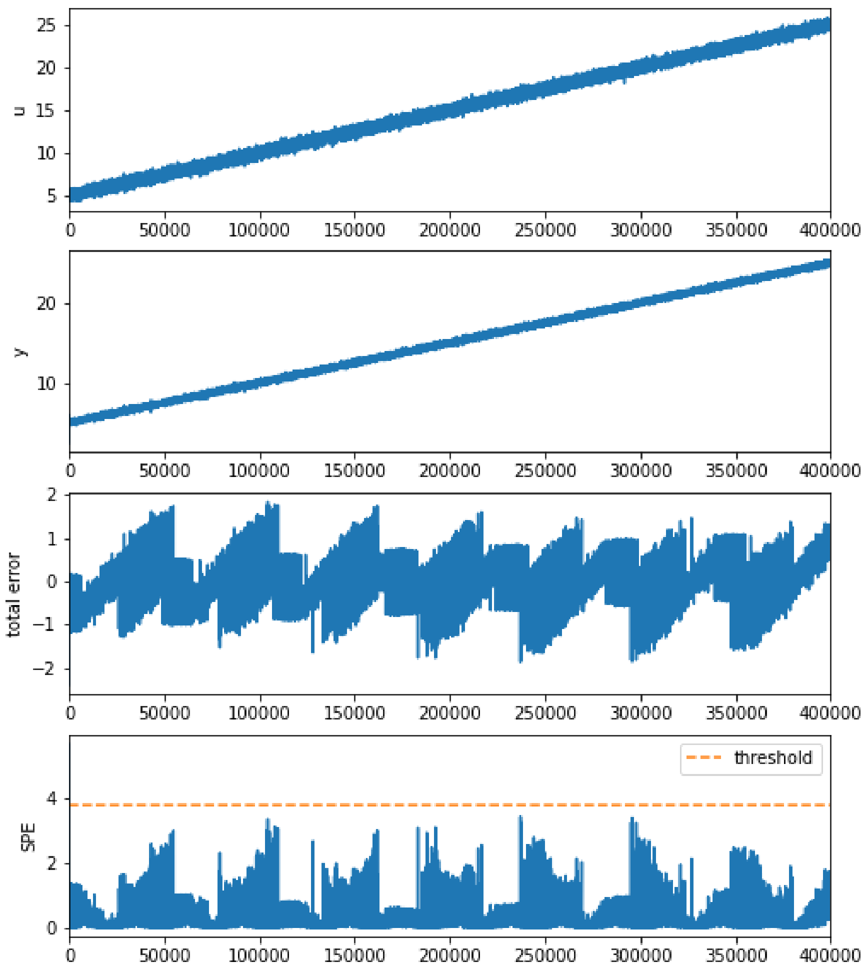

fig3

Fig. 3: Off-line training in the closed-loop case.

According to the time instant k, u in the online phase is given as follows.

$$\mathbf{u}(k) = \begin{cases} 10, \ k = 1, \cdots, 4000; \\ 15, \ k = 4001, \cdots, 6000; \\ 20, \ k = 6001, \cdots, 10000. \end{cases} \tag{Dm4}$$

eq4

In addition, a sensor fault $\mathbf{f}$ in the feedback control system is set to

$$\mathbf{f}(k) = \begin{cases} 0, \ \ k = 1, \cdots, 8000; \\ 5, \ \ k = 8001, \cdots, 10000. \end{cases} \tag{Dm5}$$

eq4

where the effect caused by $\mathbf{f}$ propagates along with the control loop in the demo.

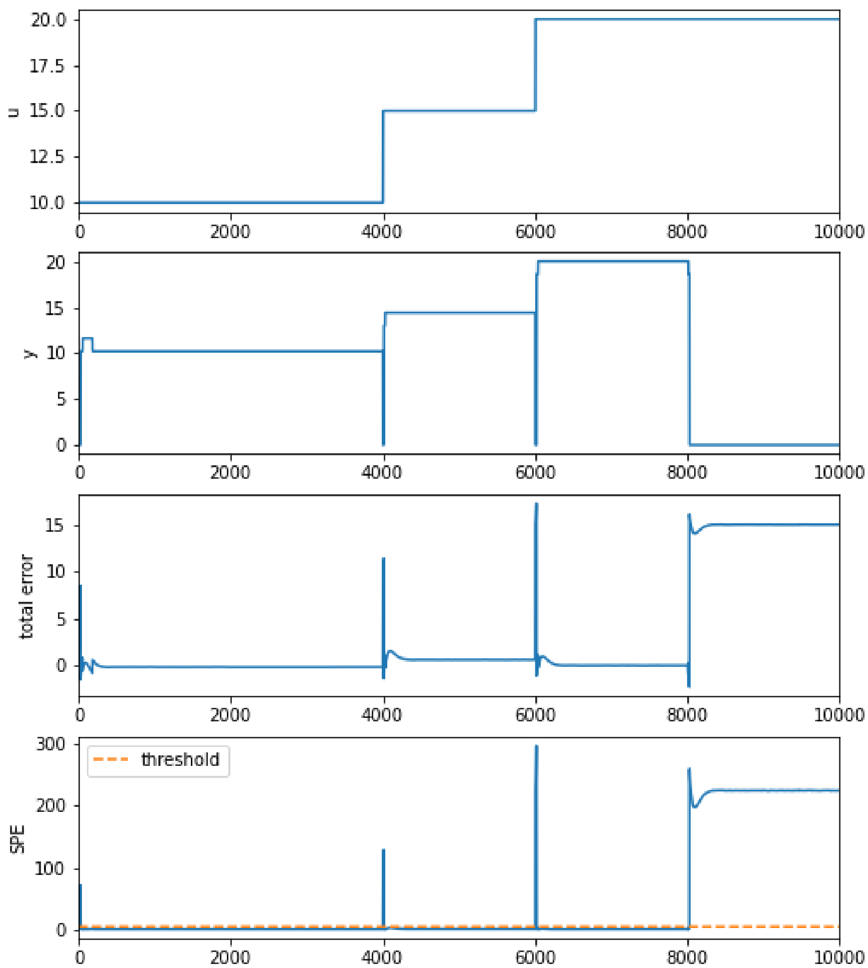

fig4

Fig. 4: Online tests in the closed-loop case.

By using the obtained $\hat{\mathcal{K}}$, an observer-driven residual generator can be constructed. In the online phase, it works well for the dynamic nonlinear systems, i.e., changes of system inputs do not affect residual signals. As shown in Fig. 4, the total error and SPE are independent of the varying $\mathbf{u}$ before 8000-th step. In addition, the spike is a common phenomenon in data-driven dynamic FD methods. For example, these similar spikes also arise in the studies [4*,5*]. When the fault $\mathbf{f}$ occurs, $\mathbf{f}$ affects the total error as well as SPE. It can be readily observed from Fig. 4 that the test statistic show excellent FD power.

Part D: A Note on Codes

- Off-line training:
  (1) Run *ininormal.m* to initialize the parameter used in the Simulink model.
  (2) Run *motornormal.mdl* to generate normal data for training.
  (3) Run the first part of *stackdata.m* to obtain the stack data.
  (4) Run the first part of *Examples.ipynb* for training the FD algorithm.
- Online test:
  (1) Run *inifaulty.m* to initialize the parameter used in the Simulink model.
  (2) Run *motorfaulty.mdl* to generate faulty data for online tests.
  (3) Run the second part of *stackdata.m* to obtain the stack faulty data.
  (4) Run the rest of *Examples.ipynb* for online FD.

We would like to express our gratitude to Prof. Hao Luo for his help in the completion of the direct-current motor control model. This is just a demo for ‘Data-driven Fault Detection for Lipschitz Nonlinear Systems: From Open to Closed-loop Systems.” Any questions about the Python code, please contact Ms. Liu (zl32@illinois.edu).

References

[1*] Luo, H. (2017). Plug-and-play monitoring and performance optimization for industrial

automation processes. Springer Vieweg.

[2*] Jin, Z., Khajenejad, M., and Yong, S. Z. (2020). Data-driven model invalidation for unknown Lipschitz continuous systems via abstraction. *American Control Conference* pp. 2975–2980.

[3*] Chen, H., Jiang, B., Ding, S. X., and Huang, B. (2022). Data-driven fault diagnosis for traction systems in high-speed trains: A survey, challenges, and perspectives. *IEEE Transactions on Intelligent Transportation Systems* **23(3)**, 1700–1716.

[4*] Jiang, Y., Yin, S., and Kaynak, O. (2021). Optimized design of parity relation based residual generator for fault detection: data-driven approaches. *IEEE Transactions on Industrial Informatics* **17(2)**, 1449–1458.

[5*] Wang, X., Yang, G., and Zhang, D. (2020). Data-driven fault detection for linear systems: A q-step residual iteration approach. *International Journal of Robust and Nonlinear Control* **30(14)**, 5341–5355.