Implementation of Nonlinear Optimal Control of Two-wheel Robot with Extended Kalman Filter

KOKKRATHOKE, Surapong and XU, Xu <http://orcid.org/0000-0002-9721-9054>

Published version

KOKKRATHOKE, Surapong and XU, Xu (2021). Implementation of Nonlinear Optimal Control of Two-wheel Robot with Extended Kalman Filter. In: 2021 IEEE International Conference on Automatic Control & Intelligent Systems. IEEE.

Implementation of Nonlinear Optimal Control of Two-wheel Robot with Extended Kalman Filter

Surapong Kokkrathoke
Department of Engineering and Mathematics
Sheffield Hallam University
Sheffield, UK
surapong.kokkrathoke@student.shu.ac.uk

Xu Xu
Department of Engineering and Mathematics
Sheffield Hallam University
Sheffield, UK
xu.xu@shu.ac.uk

***Abstract*— This paper presents a nonlinear freezing optimal control (NFOC) technique combined with an extended Kalman filter (EKF) for stabilising a two-wheel robot (TWR). The balancing LEGO EV3 Robot is utilised as a prototype for simulation and practical implementation to test the performance of the NFOC with EKF, compared against the well-known linear optimal control, i.e., the linear quadratic regulator (LQR) and the stand-alone NFOC. The stabilisation of the TWR system when starting from various ranges of initial pitch angles with different types of controllers are investigated and discussed. The MATLAB simulation result demonstrates wider operation ranges from both nonlinear optimal controllers over the linear one when simulated with a high-performance motor. In the case of implementation, the two nonlinear methods also displayed slightly more comprehensive initial pitch angle ranges than the linear control. Significantly, the precision of state variable estimation from the EKF technique removes the signal drift problem in the gyro sensor, which is used to measure the pitch angle of the TWR. The effectiveness of the NFOC controller combined with EKF is demonstrated by results from MATLAB simulation and implementation on the LEGO TWR.**

Keywords—Two-wheel robot, optimal control, nonlinear control, extended Kalman filter, balancing robot.

I. Introduction

The Two-Wheel Robots (TWR) have been recognised as a benchmark tool in various research to study and test control theory on underactuated systems. The design of TWR is based on the inverted pendulum on a cart introduced by Grasser and coworkers in [1]. They presented a mobile inverted pendulum robot known as JOE and the pole placement control was applied to balance JOE. More recently, several other control techniques have been applied to the TWR, producing appropriate performances of the stabilising system. For instance, the implementations of the TWR using a PID controller was presented in servicing robot [2], transportation robot [3] and cart-inverted pendulum on a test bench [4]. Furthermore, a Linear Quadratic Regulator (LQR) was applied to the TWR system, as demonstrated in the self-balancing robot [5], walker-assisted robot [6] and inverted pendulum on a cart [7]. Fuzzy logic control is another controller utilised on the TWR as introduced in the inverted pendulum on a rail [8], [9] and rotary inverted pendulum [10], which have similar system models as the TWR. However, these linear controllers [1]-[10] can only provide stabilisation of the TWR system within restricted operating ranges of pitch angles as the linearisation is applied around the equilibrium point in its upright position.

In this paper, a nonlinear control technique, namely, the Nonlinear Freezing Optimal Control (NFOC), is used to improve the operational range of stabilised TWR as the linearisation is not necessary, therefore the operating range of pitch angles is much wider. For instance, the implementation in [11] introduced the balancing of inverted pendulum on a cart using the NFOC technique and showed excellent results. In a previous paper [12], simulation work demonstrated that NFOC could achieve much wider operating ranges of pitch angle in the TWR model comparing against the classical LQR control.

In this work, the NFOC designed in [12] is implemented to the prototype of the TWR by using the LEGO Mindstorms EV3 robot. This is because the LEGO Mindstorms EV3 kit is inexpensive, reprogrammable, and versatile. Moreover, it has been widely used in education and research purposes with various elements such as [5] and [13].

However, the LEGO EV3 gyro sensor produces drift signal due to noise corruption, resulting inaccurate pitch angle measurement during implementation. Hence, the extended Kalman filter (EKF) is utilised to remove the sensor's drifting issue by estimating the pitch angle state variable more accurately, for the NFOC to utilise for feedback. The approach of combining the NFOC with EKF have been used in other application previously, for instance, the simulation of missile guidance by estimating missile trajectory in [14]. Moreover, the NFOC with EKF also refers to the state-dependent Riccati equation with an extended Kalman filter applied to estimate the number of normal cells for achieving the drug regimens in cancer treatment in [15], and also ultilised to approximate the dynamic system of flexible-joint of robotic arm with external disturbance and noise [16].

The research is organised as follows: Section II analyses the system models of the TWR, representing both linear and nonlinear state-space models. In Section III, control design procedures, including NOC, NFOC and NFOC with EKF, are described. Section IV and V demonstrate simulation results using MATLAB and implementation results using a LEGO EV3 Root, respectively. Finally, the paper is concluded in Section VI.

II. Mathematical Model

The system model of LEGO EV3 Robot is constructed from Fig.1, divided into a side view and a top view. Moreover, the generalised coordinates of the TWR consist of

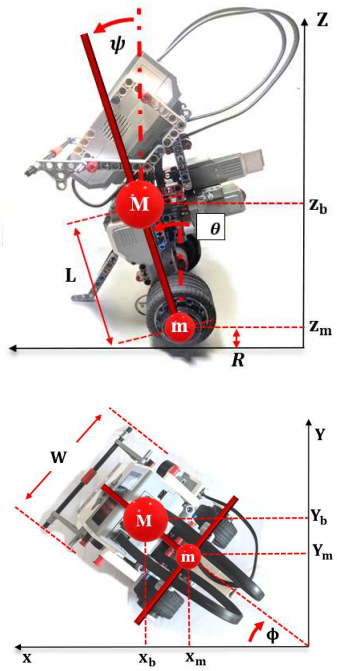

Fig. 1. Side view and top view of a two-wheel robot created with LEGO EV3.

wheel angle $\boldsymbol{\theta}$, robot pitch angle $\boldsymbol{\psi}$, and robot yaw angle $\boldsymbol{\phi}$ [5]. To establish the state-space model of the TWR, the Lagrangian technique is applied to analyse the system model based on the kinetic energy and the potential energy equations. Furthermore, a tracking system design is selected to enhance the accuracy of following setpoint changes by reducing steady-state errors in the horizontal displacement of the TWR, related to $\boldsymbol{x_1}$. By doing this, the wheel angle $\boldsymbol{x_1}$ is tracked by adding an integrator, which is presented as $\dot{\boldsymbol{x}}_{\boldsymbol{5}} = \boldsymbol{x_1}$. Thus, the nonlinear state-space model of the TWR with tracking feature is written as [12]:

$$\begin{pmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\\ \dot{x}_4\\ \dot{x}_5\end{pmatrix} = \begin{pmatrix}0 & 1 & 0 & 0\\ 0 & \frac{e_{m22}(x_3)}{a+b(x_3)} & \frac{e_{23}(x_3)}{[a+b(x_3)]x_3} & \frac{e_{m24}(x_3,x_4)}{a+b(x_3)}\\ 0 & 0 & 0 & 1\\ 0 & \frac{e_{m42}(x_3)}{a+b(x_3)} & \frac{e_{43}(x_3)}{[a+b(x_3)]x_3} & \frac{e_{m44}(x_3,x_4)}{a+b(x_3)}\\ 1 & 0 & 0 & 0\end{pmatrix} \times \begin{pmatrix}x_1\\ x_2\\ x_3\\ x_4\\ x_5\end{pmatrix} + \begin{pmatrix}0 & 0\\ \frac{f_{m21}(x_3)}{a+b(x_3)} & \frac{f_{m22}(x_3)}{a+b(x_3)}\\ 0 & 0\\ \frac{f_{m41}(x_3)}{a+b(x_3)} & \frac{f_{m42}(x_3)}{a+b(x_3)}\\ 0 & 0\end{pmatrix}\begin{pmatrix}v_1\\ v_2\end{pmatrix}. \quad (1)$$

where the generalised coordinates are defined as:

$$x_1 = \theta,\ \ x_2 = \dot{\theta} \Rightarrow \dot{x}_2 = \ddot{\theta},\ x_3 = \psi,\ x_4 = \dot{\psi} \Rightarrow \dot{x}_4 = \ddot{\psi},$$

Noticeably, the yaw angle is neglected in this research. The $\boldsymbol{v_1}$ and $\boldsymbol{v_2}$ are left and right motor voltage and other parameters [12] are given in Appendix.

Furthermore, the standard NFOC and NFOC with EKF methods are compared against the linear optimal control (LOC) technique in this research; hence, (1) can be linearised by approximate $\boldsymbol{x_3}$ and $\boldsymbol{x_4}$ to be operating around an equilibrium point with small variations as follows:

$$\boldsymbol{sin(x_3) \approx x_3, cos(x_3) \approx 1,\ x_4\, sin(x_3) \approx 0} \text{ and } \boldsymbol{sin(x_3)^2 \approx 0}$$

Therefore, the linear state-space representation of the TWR from (1) can be presented as follow:

$$\begin{pmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\\ \dot{x}_4\\ \dot{x}_5\end{pmatrix} = \begin{pmatrix}0 & 1 & 0 & 0\\ 0 & -228.11 & -493.22 & 228.11\\ 0 & 0 & 0 & 1\\ 0 & 48.58 & 163.79 & -48.58\\ 1 & 0 & 0 & 0\end{pmatrix}\begin{pmatrix}x_1\\ x_2\\ x_3\\ x_4\\ x_5\end{pmatrix} + \begin{pmatrix}0 & 0\\ 221.71 & 221.71\\ 0 & 0\\ -47.22 & -47.22\\ 0 & 0\end{pmatrix}\begin{pmatrix}v_1\\ v_2\end{pmatrix}. \quad (2)$$

III. Controller Design

A. Linear Optimal Control

The linear optimal control (LOC) is also known as linear quadratic regulator (LQR) [12], which is a widely used controller for unstable systems. The state-space system is given as:

$$\dot{\boldsymbol{x}} = \boldsymbol{Ax} + \boldsymbol{Bu}\,, \quad (3)$$

where $\boldsymbol{A}$ is the matrix of system, $\boldsymbol{B}$ is the matrix of control, $\boldsymbol{x}$ is state variables vector, and $\boldsymbol{u}$ is control variables vector which the linear feedback control is presented by [17]:

$$\boldsymbol{u} = -\boldsymbol{K_l x} = -\boldsymbol{R^{-1}B^T P x}\,, \quad (4)$$

where $\boldsymbol{K_l}$ is linear feedback gain, $\boldsymbol{Q}$ and $\boldsymbol{R}$ are weighting matrices, and $\boldsymbol{P}$ is an algebraic matrix Riccati equation solution, shown in the form as:

$$\boldsymbol{A^T P + PA + Q - PBR^{-1}B^T P = 0}\,. \quad (5)$$

B. Nonlinear Freezing Optimal Control

In this subsection, the LOC is extended by 'freezing' and optimising the nonlinear control system in every time step, known as the freezing technique first introduced in [18]. It is also referred to as nonlinear freezing optimal control (NFOC).

Therefore, the nonlinear system can be constructed in the form [18]:

$$\dot{\boldsymbol{x}} = \boldsymbol{Ax} + \boldsymbol{Bu}, \quad (6)$$

and the nonlinear feedback control equation is given by:

$$\boldsymbol{u} = -\boldsymbol{K_n}(\boldsymbol{x})\boldsymbol{x} = -\boldsymbol{R^{-1}}(\boldsymbol{x})\boldsymbol{B^T}(\boldsymbol{x})\boldsymbol{P}(\boldsymbol{x})\boldsymbol{x}\,, \quad (7)$$

where $\boldsymbol{K_n}$ is nonlinear feedback gain and the $\boldsymbol{P(x)}$ is presented by an algebraic matrix Riccati equation solution in the form as:

$$\boldsymbol{A^T(x)P(x) + P(x)A(x) + Q(x)} \\ \boldsymbol{-P(x)B(x)R^{-1}(x)B^T(x)P(x) = 0}\,. \quad (8)$$

Thus, these parameters, $\boldsymbol{A(x)}$, $\boldsymbol{B(x)}$, $\boldsymbol{B^T(x)}$, $\boldsymbol{P(x)}$ and $\boldsymbol{R^{-1}(x)}$, are fixed in every time step of $\boldsymbol{x}$ iteratively by applying an integration method in (6). Moreover, the advantage of this method is that linearisation is no longer

necessary as the freezing technique can control the system globally, wherever controllability criterion is met.

C. Nonlinear Freezing Optimal Control with Extended Kalman Filter

The extended Kalman filter (EKF) is a well-known technique to improve the control signal in the nonlinear controller design by estimating the state variables. The benefits of this method are noise filtering and signal drift reduction. In this research, the signal drift problem in the gyro sensor is improved by using the state estimation from the EKF. Thus, the nonlinear continuous time-invariant systems can be represented as [19]:

$$\dot{\boldsymbol{x}} = \boldsymbol{a}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{t}) + \boldsymbol{G}(\boldsymbol{t})\boldsymbol{w}, \tag{9}$$

$$\boldsymbol{y} = \boldsymbol{c}(\boldsymbol{x}, \boldsymbol{t}) + \boldsymbol{v}, \tag{10}$$

where $\boldsymbol{w}$ is the process noise, $\boldsymbol{v}$ is the measurement noise, with $\boldsymbol{w} \sim (\boldsymbol{0}, \boldsymbol{Q_k})$, $\boldsymbol{v} \sim (\boldsymbol{0}, \boldsymbol{R_K})$, which $\boldsymbol{Q_k}$ and $\boldsymbol{R_k}$ are weighting matrices of noise filtering, and $\boldsymbol{G}$ is the process noise matrix, set as $\boldsymbol{G} = \boldsymbol{I_{5\times5}}$.

The state estimator of nonlinear system is given as:

$$\dot{\hat{\boldsymbol{x}}} = \boldsymbol{a}(\hat{\boldsymbol{x}}, \boldsymbol{u}, \boldsymbol{t}) + \boldsymbol{K_k}(\hat{\boldsymbol{x}}, \boldsymbol{t})(\boldsymbol{y} - \boldsymbol{c}(\hat{\boldsymbol{x}}, \boldsymbol{t}), \tag{11}$$

where $\boldsymbol{K_k}$ is the Kalman filter gain. The Jacobian matrices are represented as follows:

$$\boldsymbol{A}(\boldsymbol{x}, t) = \frac{\partial \boldsymbol{a}(\boldsymbol{x}, \boldsymbol{u}, t)}{\partial \boldsymbol{x}}, \boldsymbol{A}(\hat{\boldsymbol{x}}, t) = \frac{\partial \boldsymbol{a}(\hat{\boldsymbol{x}}, \boldsymbol{u}, t)}{\partial \hat{\boldsymbol{x}}}, \tag{12}$$

$$\boldsymbol{C}(\boldsymbol{x}, t) = \frac{\partial \boldsymbol{c}(\boldsymbol{x}, t)}{\partial \boldsymbol{x}}, \boldsymbol{C}(\hat{\boldsymbol{x}}, t) = \frac{\partial \boldsymbol{c}(\hat{\boldsymbol{x}}, t)}{\partial \hat{\boldsymbol{x}}}. \tag{13}$$

Consequently, $\boldsymbol{K_k}$ is represented as:

$$\boldsymbol{K_k}(\hat{\boldsymbol{x}}, t) = \boldsymbol{P_k}(\hat{\boldsymbol{x}}, t)\boldsymbol{C^T}(\hat{\boldsymbol{x}}, t)\boldsymbol{R_k^{-1}}(\hat{\boldsymbol{x}}, t), \tag{14}$$

where the solution of algebraic Riccati equation $\boldsymbol{P_k}$ is demonstrated as:

$$\begin{aligned}&\boldsymbol{A}(\hat{\boldsymbol{x}}, t)\boldsymbol{P_k}(\hat{\boldsymbol{x}}, t) + \boldsymbol{P_k}(\hat{\boldsymbol{x}}, t)\boldsymbol{A^T}(\hat{\boldsymbol{x}}, t)\\ &- \boldsymbol{P_k}(\hat{\boldsymbol{x}}, t)\boldsymbol{C^T}(\hat{\boldsymbol{x}}, t)\boldsymbol{R_k^{-1}}(\hat{\boldsymbol{x}}, t)\boldsymbol{C}(\hat{\boldsymbol{x}}, t)\boldsymbol{P_k}(\hat{\boldsymbol{x}}, t) + \boldsymbol{Q_k}(\hat{\boldsymbol{x}}, t) = 0\end{aligned} \tag{15}$$

Additionally, the nonlinear feedback gain $\boldsymbol{K_n}$ of the NFOC with EKF is the same parameter as the stand-alone NFOC technique.

IV. Simulation Results

Simulation results of the LEGO EV3 robot are obtained using MATLAB as follows. The controllers are designed to stabilise the mathematical model of TWR from different initial pitch angles x_3. Moreover, the weight of matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$ in this simulation are selected as $\boldsymbol{Q}$ = diag{20,1,1,5} and $\boldsymbol{R}$ = diag{10,10} following from [9]. Therefore, the linear matrix gain is obtained as $\boldsymbol{K_l}$= [-1.391, -1.449, -59.848, -7.168, -0.5]. In terms of the EKF, the process noise is set to $Q_k = \text{diag}\{1, 1, 1, 1, 1\}$, as we concentrate only on the output signal, which is the gyro sensor. In the case of the output signal, the measurement noise is chosen from trial and error tests as $\boldsymbol{R_k} = \text{diag}\{0.2, 0.2\}$.

In this research, the power supply of LEGO EV3 is limited to 8.3 V (See more details of LEGO EV3 Robot hardware in Section V). Therefore, a hard constraint is applied to restrict motor voltage in the MATLAB simulation, as follows:

$$\boldsymbol{v} = \begin{cases} \boldsymbol{8.3} & \boldsymbol{v} \geq \boldsymbol{8.3} \\ \boldsymbol{v} & |\boldsymbol{v}| < \boldsymbol{8.3} \\ \boldsymbol{-8.3} & \boldsymbol{v} \leq \boldsymbol{-8.3}. \end{cases} \tag{16}$$

We simulate from different initial pitch angles $\boldsymbol{x_3}$ and investigate the maximum initial pitch angle that TWR model can remain in stabilisation at the vertical upright position under the voltage constraint condition by using three controllers: LOC, NFOC and NFOC with EKF.

Results in Fig. 2 indicate that the maximum initial pitch angle $\boldsymbol{x_3}$ achieved by three controllers is the same: at 20.9°, when simulating with this voltage constraint. Furthermore, the behaviours of the TWR are very similar. Beyond this angle, the robot model will be unstable and topple over. Moreover, outputs of the three controllers resemble each other because the optimal feedback control gains $\boldsymbol{K}$ of the linear and nonlinear controls have only very minor differences at small pitch angles. Details of different control gains $\boldsymbol{K}$ from two methods will be presented in Section V.

It is possible to introduce higher initial pitch angles in the simulation, however, the TWR will demand higher power supply voltage from a high-performance motor. To test this, the performance of power supply and motor parameters in MATLAB simulation are improved to demonstrate the capabilities of the three controllers. The Maxon motor series EC 45 Flat 30W [20] is selected which has a higher voltage at 36V and a nominal torque of 66 mNm. As a result, the voltage saturation in MATLAB simulation is raised to 36V from 8.3V and the motor torque parameter is also changed.

It is clear from Fig. 3 that there are noticeable differences between linear and nonlinear control when the voltage constraint and motor are improved. The LOC presents significant oscillations when the $\boldsymbol{x_3}$ reaches a new maximum at 60.7°; by contrast, the graphs of NFOC and NFOC with EKF (same as NFOC) are smoother with less oscillations.

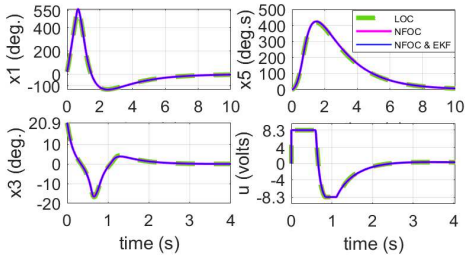

Fig. 2. Stabilisation of the TWR using LOC, NFOC and NFOC with EKF, with a voltage constraint 8.3V at the initial $\boldsymbol{x_3}$ = 20.9°

Furthermore, both nonlinear controllers go beyond the 60.7° capability of LOC, stabilising the TWR up to a maximum $\boldsymbol{x_3}$ = 72.4° when simulating with this motor, as shown in Fig.4.

V. Practical Results

As discussed in Section IV, the TWR was stabilised in the vertical upright position in simulation, with three different control designs. Next, these control designs will be implemented on a practical TWR for investigation and analysis. Drift problems from the gyro sensor will also be examined.

The TWR communicates with the laptop via Wi-Fi for real-time monitoring of all state variables during the implementation using Simulink programs for stabilisation and tracking design introduced by [13] but significantly revised here and shown in Fig.5-6.

Furthermore, the TWR hardware consists of the following components:

- CPU processor 32-bit (ARM9) with Linux operating system
- Two Large EV3 Motors with Rotary Encoders
- Ultrasonic Sensor
- Gyro sensor
- External USB Wi-Fi dongle (Edimax N150)
- DC battery 2200 mAh 7.4V

However, the actual voltage of the DC battery was measured before implementation by a multimeter, which showed approximately 8.3V. Therefore, we used this value as the voltage constraint when the TWR controls were simulated in Section IV. Additionally, this saturation parameter has also been applied in the Simulink programme for the implementation here.

In terms of programming designed, the nonlinear control needs to update the gain matrix $\boldsymbol{K_n}$ by applying the LQR function in MATLAB to solve the algebraic Riccati equation at every time step. In simulations, the LQR function in MATLAB was applied to solve the Riccati equation. However, this function cannot be downloaded to the LEGO EV3 in Simulink programme as it is not supported. Therefore, to overcome this issue, a lookup table is applied to store the pre-calculated gain matrix $\boldsymbol{K_n}$ instead.

It can be seen in (1) that the states x_3 and x_4 both have affects on the stability of the TWR system. As a result, the lookup table comprises variables x_3 and x_4, and the matrix gain $\boldsymbol{K_n}$. For instance, a partial lookup table is presented in Table I.

Noticeably, the matrix gain $\boldsymbol{K_n}$ of the nonlinear control at x_3= 0° and x_4 = 0° shown in the bold text in Table I is equal to the fixed matrix gain $\boldsymbol{K_l}$ of the linear controller. This is because the linearisation is applied at this equilibrium point of the TWR system. Moreover, Table I also presents the varying nonlinear control gain $\boldsymbol{K_n}$ based on the state variables x_3 and x_4, which supports the stabilisation of the nonlinear TWR at wider pitch angles than using the linear method. For a more detailed look-up table, see Appendix Table A2.

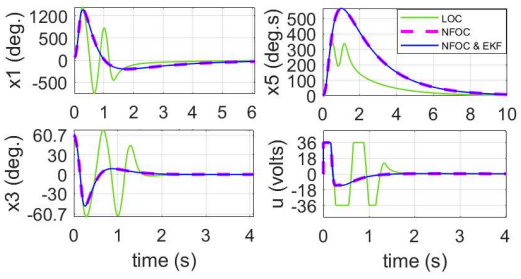

Fig.3. Stabilisation of the TWR using LOC, NFOC and NFOC with EKF, with a voltage constraint 36V of Maxon Motor EC45 Flat at the initial $\boldsymbol{x_3}$ = 60.7°

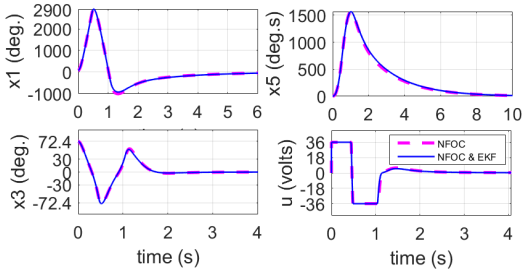

Fig.4. Stabilisation of the TWR using NFOC and NFOC with EKF, with a voltage constraint 36V of Maxon Motor EC45 Flat at the initial $\boldsymbol{x_3}$ = 72.4°

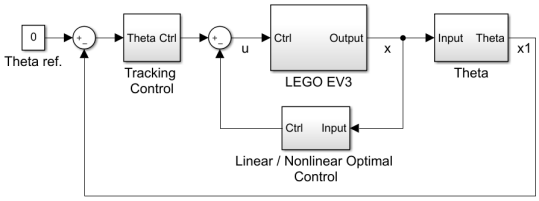

Fig.5. Simulink block diagram for stabilising the LEGO EV3, using Linear or Nonlinear Optimal Control (separated program)

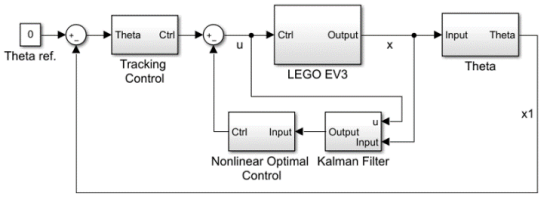

Fig.6. Simulink block diagram for stabilising the LEGO EV3, using Nonlinear Optimal Control and Extended Kalman Filter

TABLE I. THE PARTIAL LOOKUP TABLE OF NONLINEAR MATRIX GAIN $\boldsymbol{K_n}$

$x_3 \backslash x_4$	-120°/s	-20°/s	0°/s	20°/s	120°/s
20°	[-1.392, -1.453, -62.76, -7.490, -0.5]	[-1.392, -1.451, -61.907, -7.481, -0.5]	[-1.391, -1.451, -61.834, -7.479, -0.5]	[-1.391, -1.450, -61.761, -7.477, -0.5]	[-1.392, -1.448, -61.398, -7.467, -0.5]
10°	[-1.391, -1.450, -60.546, -7.249, -0.5]	[-1.391, -1.449, -60.368, -7.245, -0.5]	[-1.391, -1.449, -60.435, -7.260, -0.5]	[-1.391, -1.449, -60.297, -7.243, -0.5]	[-1.391, -1.448, -60.120, -7.239, -0.5]
0°	[-1.391, -1.449, -59.848, -7.168, -0.5]	[-1.391, -1.449, -59.848, -7.168, -0.5]	**[-1.391, -1.449, -59.848, -7.168, -0.5]**	[-1.391, -1.449, -59.848, -7.168, -0.5]	[-1.391, -1.449, -59.848, -7.168, -0.5]

1) Small Initial Pitch Angle Implementation

In the first implementation experiment, the LEGO EV3 robot is controlled by three different controllers at small initial pitch angle x_3=8°.

To begin with, in Fig.7, the TWR is stabilised by the LOC technique. The practical robot outputs represented by the blue curve are compared against the simulation result (black curve). It can be seen that the wheel angle $x_1$and the robot pitch angle x_3 do not converge to their reference positions. This means the robot deviates from the central position, and the robot body is not vertically upright. The error in x_1 also affects the x_5 variable which does not produce successful tracking.

Similarly, the stand-alone NFOC is applied to the TWR at the same initial pitch angle, shown in Fig. 8. The red curves show similar results as the LOC one in Fig. 7.

However, the state variables x_1 and x_3 do converge to the reference positions after stabilising at the initial pitch angle x_3=8°, when applying NFOC with EKF, as shown by the green curve in Fig. 9. There is some deviation in x_5 which is caused by the oscillations in the x_1 signal.

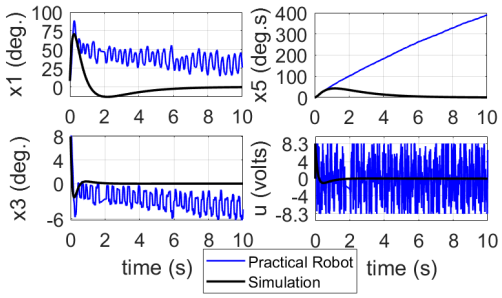

Fig.7. The stabilisation at initial pitch angles x_3 = 8° between simulation and hardware implementation by LOC technique

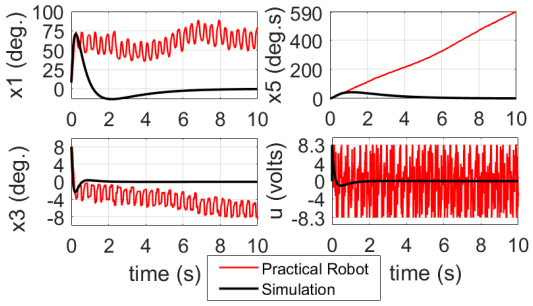

Fig.8. The stabilisation at initial pitch angles x_3 = 8° between simulation and hardware implementation by NFOC technique

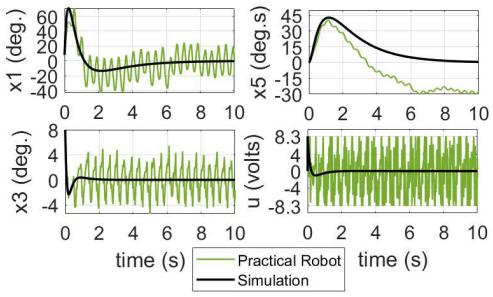

Fig.9. The stabilisation at initial pitch angles x_3 = 8° between simulation and hardware implementation by NFOC with EKF technique

Noticeably, in Fig. 9, the robot pitch angle and wheel angle do not deviate from the reference positions after stabilisation is completed. This is because EKF provided more accurate estimation of the state variable x_3, which is used instead of the drifting signal produced by the gyro sensor. The more accurate signal of x_3 is forwarded to NFOC for stabilising the TWR appropriately.

2) Maximum Initial Pitch Angle Implementation

After the small initial pitch angle is implemented, the maximum initial pitch angles of each controller are investigated in this subsection.

Firstly, the LOC is applied to stabilise the practical TWR with the maximum initial pitch angle x_3=16°, shown in Fig.10. The state variables x_1, x_3 and x_5 demonstrate non-converging trends, similar to the experiment run at small initial pitch angle by LOC (Fig. 7), also caused by the gyro sensor drift problem. Moreover, comparing against the simulation results, the overshoots in x_1 (twice in amplitude) and x_3 are both higher.

Furthermore, the stand-alone NFOC and NFOC with EKF demonstrate similar maximum initial pitch angles at x_3=18°, both shown in Fig. 11. This represents a marginally more comprehensive operating range than the LOC as the matrix gain $\boldsymbol{K_n}$ of the nonlinear feedback controls varied depending on the varying state variables and provided more effective control, but the linear control gain $\boldsymbol{K_l}$ was fixed.

It is clear from Fig. 11 that the comparison of results between the stand-alone NFOC and NFOC with EKF methods demonstrate the gyro sensor problem has been solved by applying the EKF. In particular, the state variables x_1, x_3 and x_5 of the normal NFOC deviate from the referent positions represented by the red curve. In contrast, these state variables converge to their reference positions when using the NFOC with EKF technique.

It can also be noted that the overshoots in x_1 and x_3 graphs are similar to each other from the two controllers; for example, both peaks of x_1 are approximately 250°, and both overshoots are approximately 8° in the x_3 graph. The gyro sensor drift issue has been removed by applying the NFOC with EKF shown in the blue curve because the more accurate state variable x_3 has been estimated by the EKF and then provided to the control system for feeding back.

Importantly, the TWR is also stabilised in the vertical upright position by using NFOC with EKF. The wheel angle x_1 of TWR rotates forward by approximately 240° and then

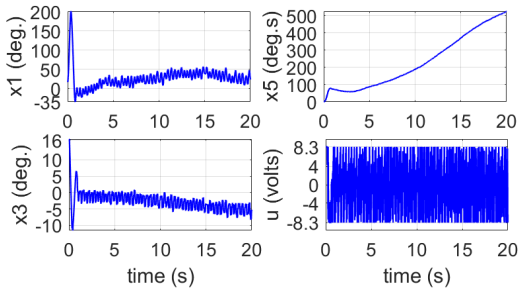

Fig.10. The maximum initial pitch angles of LOC technique at x_3 = 16°, implemented by LEGO EV3 robot

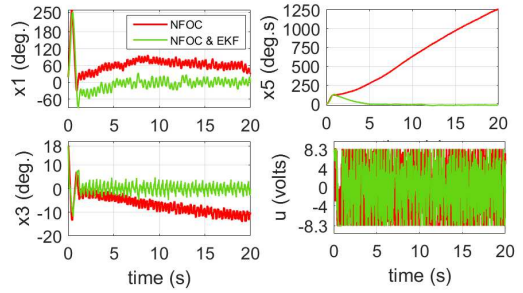

Fig.11. The stabilisation of LEGO EV3 robot at initial pitch angles x_3 = 18°, by NFOC technique and NFOC with EKF (sensor drift removed)

moves backward by approximately 90° to the reference position before stablilising in ~8 seconds. Because of the fluctuation in x_1 , the amplitude of integral of wheel angle x_1 reaches approximately 120 deg·s and then drops to the reference point.

The maximum initial pitch angles of the LEGO EV3 robot with stabilisation are summarised in Table II. The results between the physical implementation and computer simulation with motor voltage constraint are similar. Moreover, both NFOC techniques present more accurate results than the LOC method with the lower error between A and B column, at 2.9°.

Significantly, the maximum initial pitch angles of both nonlinear controllers are slightly larger (by 2°) than the linear method (LOC) in the practical implementations. This is because the difference of matrix gain $\boldsymbol{K}$ between nonlinear and linear controllers is small for the chosen x_3 and x_4, as shown in Table III.

TABLE II. THE SUMMARISE OF MAXIMUM INITIAL PITCH ANGLES FROM THREE CONTROLLERS

Controllers	Motor Voltage Constraint 8.3V Simulation (A)	LEGO EV3 Robot 8.3V Implementation (B)
LOC	20.9°	16°
NFOC	20.9°	18°
NFOC with EKF	20.9°	18°

TABLE III. THE DIFFERENCE OF LINEAR AND NONLINEAR MATRIX GAINS K AT X_3=20° AND X_4=0°

	Matrix Gain $\boldsymbol{K}$
Nonlinear Control	[-1.391, -1.451, -61.834, -7.479, -0.5]
Linear Control	[-1.391, -1.449, -59.848, -7.168, -0.5]
Difference	[0, 0.002, 1.986, 0.311 ,0]

VI. CONCLUSIONS

In this research, the LEGO EV3 two-wheel robot was simulated and implemented using the NFOC technique with EKF, compared against the stand-alone NFOC and the LOC methods. Results of NFOC with EKF showed that sensor drift problem was resolved, which the normal NFOC and LOC were unable to achieve. Furthermore, both nonlinear controllers satisfied the stabilisation requirement of the practical TWR, controlling it to stay in the vertical upright position, starting from different initial pitch angles. The nonlinear control methods presented a slightly broader operational range than the linear controller.

Future research will focus on the controllability of the nonlinear system, which depends on the non-unique forms of matrices A and B. Different state-space models from the same physical system will be investigated and analysed using NFOC and NFOC with EKF techniques.

ACKNOWLEDGMENT

The first author would like to thank the Royal Thai Government and the Synchrotron Light Research Institute (SLRI) (public organisation), Thailand for the scholarship funding.

REFERENCES

[1] F. Grasser, A. D. Arrigo, S. Colombi, and A. . Rufer, "JOE: a mobile, inverted pendulum," *IEEE transactions on industrial electronics*, vol. 49, no. 1, pp. 107–114, Feb. 2002.

[2] J. K. Ahn and S. Jung, "Development of a two-wheel mobile manipulator: balancing and interaction control," *Robotica*, vol. 32, no. 7, pp. 1135–1152, January 2014.

[3] H. W. Kim and S. Jung, "Control of a two-wheel robotic vehicle for personal transportation," *Robotica*, vol. 34, no. 5, pp. 1186–1208, May 2016.

[4] R. Mondal and J. Dey, "Performance Analysis and Implementation of Fractional Order 2-DOF Control on Cart-Inverted Pendulum System," *IEEE transactions on industry applications*, vol. 56, no. 6, pp. 7055–7066, November 2020.

[5] Y. Yamamoto, "NXTway-GS (Self-Balancing Two-Wheeled Robot) Controller Design)." Mathwork.com. https://www.mathworks.com/matlabcentral/fileexchange/19147-nxtway-gs-self-balancing-two-wheeled-robot-controller-design (accessed Feb. 1, 2021).

[6] A. R. da Silva Jr, and F. C. Sup IV, "A Robotic Walker Based on a Two-Wheeled Inverted Pendulum*," Journal of intelligent & robotic systems*, vol. 86, no. 1, pp. 17–34, April 2017.

[7] S. Maity and G. R. Luecke, "Stabilization and Optimization of Design Parameters for Control of Inverted Pendulum," *Journal of dynamic systems, measurement, and control*, vol. 141, no. 8, August 2019.

[8] C. Tao, J. Taur, Tzuen Wuu Hsieh, and C. Tsai, "Design of a Fuzzy Controller With Fuzzy Swing-Up and Parallel Distributed Pole Assignment Schemes for an Inverted Pendulum and Cart System," *IEEE transactions on control systems technology*, vol. 16, no. 6, pp. 1277–1288, November 2008.

[9] S. Bicakci, "On the Implementation of Fuzzy VMC for an Under Actuated System," *IEEE access*, vol. 7, pp. 163578–163588, November 2019.

[10] E. Aranda-Escolástico, M. Guinaldo, M. Santos, and S. Dormido, "Control of a Chain Pendulum: A fuzzy logic approach," *International journal of computational intelligence systems*, vol. 9, no. 2, pp. 281–295, March 2016.

[11] X. Xu, H. Zhang, & G. Carbone. "Case studies on nonlinear control theory of the inverted pendulum," *In Inverted pendulum: from theory to new innovations in control and robotics*. O. Boubaker and R. Iriarte, eds. London, UK: IET, 2017, pp 225-262.

[12] S. Kokkrathoke, A. Rawsthorne, H. Zhang, and X. Xu, "Nonlinear Optimal Stabilising Control of a Two-wheel Robot," *International Journal of Modelling, Identification and Control.*, in press.

[13] P. Roslovets. "Gyroboy - self-balancing two-wheel robot based on Lego EV3." Mathworks.com. https://uk.mathworks.com/matlabcentral/fileexchange/60322-gyroboy-self-balancing-two-wheel-robot-based-on-lego-ev3?s_tid=srchtitle (accessed Feb. 1, 2021).

[14] T. Çimen and A. . Merttopçuoğlu, "Asymptotically Optimal Nonlinear Filtering: Theory and Examples with Application to Target State Estimation," *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 8611–8617, July 2008.

[15] Y. Batmani and H. Khaloozadeh, "Optimal chemotherapy in cancer treatment: state dependent Riccati equation control and extended Kalman filter: Optimal Chemotherapy in Cancer Treatment," *Optimal control applications & methods,* vol. 34, no. 5, pp. 562–577, September 2013.

[16] M. H. Korayem, N. Y. Lademakhi, and S. R. Nekoo, "Application of the state - dependent Riccati equation for flexible - joint arms: Controller and estimator design," *Optimal control applications & methods*, vol. 39, no. 2, pp. 792‐808, March 2018.

[17] K. Ogata, *Modern control engineering*, 5th ed., London, UK: Pearson, 2009.

[18] S. Banks, & K. Mhana, "Optimal control and stabilisation for nonlinear system*," IMA Journal of Mathematical Control and Information*, vol. 9, no. 2, pp 179-196, June 1992

[19] L. Frank, L. Xie, & D. Popa, *Optimal and Robust Estimation With an Introduction to Stochastic Control*. Florida, USA: CRC Press, 2007.

[20] Maxon. "Maxon Product Page Range 2020/21."Online.flippingbook.com. https://online.flippingbook.com/view/1042987/285 (accessed Feb. 1, 2021).

APPENDIX

The parameters of TWR system model are defined as in [12]:

$a = 2J_wJ_\psi + 2mR^2ML^2 + 4mR^2n^2J_m + 2MR^2n^2J_m$
$+2n^2J_mML^2 + 2mR^2J_\psi + MR^2J_\psi + 2J_wML^2$
$+4J_wn^2J_m + 2n^2J_mJ_\psi,$
$b(x_3) = M^2R^2L^2\,sin(x_3)^2 + 4MLRcos(x_3)n^2J_m,$
$e_{23}(x_3) = 2n^2J_mMgLsin(x_3) - M^2RL^2cos(x_3)g\,sin(x_3),$
$e_{43}(x_3) = MgLsin(x_3)(2n^2J_m + 2J_w + 2mR^2 + MR^2),$
$e_{m22}(x_3) = 2df_{22}(x_3) - 2(d + f_w)f_{21},$
$e_{m24}(x_3, x_4) = e_{24}(x_3, x_4) + 2d[f_{21} - f_{22}(x_3)],$
$e_{m42}(x_3) = 2df_{42} - 2(d + f_w)f_{41}(x_3),$
$e_{m44}(x_3, x_4) = e_{44}(x_3, x_4) + 2d[f_{41}(x_3) - f_{42}],$
$f_{m21}(x_3) = f_{m22}(x_3) = c[f_{21} - f_{22}(x_3)],$
$f_{m41}(x_3) = f_{m42}(x_3) = c[f_{41}(x_3) - f_{42}],$
$f_{21} = ML^2 + 2n^2J_m + J_\psi,$
$f_{22}(x_3) = f_{41}(x_3) = 2n^2J_m - MRLcos(x_3),$
$f_{42} = 2n^2J_m + 2J_w + 2mR^2 + MR^2,$

TABLE A1. PHYSICAL PARAMETERS OF LEGO EV3 ROBOT

No.	Parameters	Description	Value
1	m	Wheel mass	0.05 kg
2	M	Robot body mass	0.64 kg
3	R	Wheel radius	0.027 m
4	W	Robot's body width	0.105 m
5	D	Robot's body depth	0.1 m
6	H	Robot's body height	0.21 m
7	L	Distance between centre of robot and wheel axle	0.105 m
8	g	Gravitational acceleration	9.81 m^2/s
9	J_w	Wheel inertia moment	0.0000162 kgm^2
10	J_ψ	Pitch inertia moment	0.002352 kgm^2

No.	Parameters	Description	Value
11	J_ϕ	Yaw inertia moment	0.001121kgm^2
12	J_m	DC motor inertia moment	1×10^{-5} kgm^2
13	R_m	DC motor Resistance	6.69 Ω
14	K_b	DC motor back EMF constant	0.468 $V \cdot Sec/rad$
15	K_t	DC motor torque constant	0.317 Nm/A
16	n	Gear ratio	1
17	f_m	Coefficient of friction between robot and motor	0.0022
18	f_w	Coefficient of friction between wheel and floor	0

Note: The parameters number 12-18 were introduced in [5].

TABLE A2. THE PARTIAL LOOKUP TABLE OF NONLINEAR MATRIX GAIN $\boldsymbol{K_n}$

x_3(deg) \ x_4(deg/s)	-120	-100	-60	-20	0	20	60	100	120
-20	[-1.391, -1.448, -61.398, -7.468, -0.5]	[-1.391, -1.449, -61.470, -7.470, -0.5]	[-1.391, -1.450, -61.615, -7.474, -0.5]	[-1.391, -1.450, -61.761, -7.77, -0.5]	[-1.391, -1.451, -61.834, -7.479, -0.5]	[-1.391, -1.451, -61.834, -7.479, -0.5]	[-1.392, -1.452, -62.054, -7.485, -0.5]	[-1.392, -1.453, -62.202, -7.488, -0.5]	[-1.392, -1.453, -62.276, -7.490, -0.5]
-15	[-1.391, -1.448, -60.600, -7.332, -0.5]	[-1.391, -1.448, -60.680, -7.334, -0.5]	[-1.391, -1.449, -60.788, -7.337, -0.5]	[-1.391, -1.450, -60.895, -7.339, -0.5]	[-1.391, -1.450, -60.949, -7.340, -0.5]	[-1.391, -1.450, -61.003, -7.342, -0.5]	[-1.391, -1.451, -61.111, -7.344, -0.5]	[-1.392, -1.451, -61.219, -7.347, -0.5]	[-1.392, -1.452, -61.274, -7.348, -0.5]
-10	[-1.391, -1.448, -61.120, -7.239, -0.5]	[-1.391, -1.448, -60.155, -7.240, -0.5]	[-1.391, -1.449, -60.226, -7.241, -0.5]	[-1.391, -1.449, -61.297, -7.243, -0.5]	[-1.391, -1.449, -60.332, -7.244, -0.5]	[-1.391, -1.449, -60.368, -7.245, -0.5]	[-1.391, -1.450, -60.439, -7.247, -0.5]	[-1.391, -1.450, -60.510, -7.248, -0.5]	[-1.391, -1.450, -60.346, -7.249, -0.5]
-5	[-1.391, -1.448, -59.854, -7.184, -0.5]	[-1.391, -1.448, -59.880, -7.185, -0.5]	[-1.391, -1.449, -59.915, -7.186, -0.5]	[-1.391, -1.449, -59.950, -7.186, -0.5]	[-1.391, -1.449, -59.968, -7.187, -0.5]	[-1.391, -1.449, -59.986, -7.187, -0.5]	[-1.391, -1.449, -60.021, -7.188, -0.5]	[-1.391, -1.449, -60.056, -7.189, -0.5]	[-1.391, -1.449, -60.074, -7.190, -0.5]
0	[-1.391, -1.449, -59.848, -7.168, -0.5]	[-1.391, -1.449, -59.848, -7.168, -0.5]	[-1.391, -1.449, -59.848, -7.168, -0.5]	[-1.391, -1.449, -59.848, -7.168, -0.5]	[-1.391, -1.449, -59.848, -7.168, -0.5]	[-1.391, -1.449, -59.848, -7.168, -0.5]	[-1.391, -1.449, -59.848, -7.168, -0.5]	[-1.391, -1.449, -59.848, -7.168, -0.5]	[-1.391, -1.449, -59.848, -7.168, -0.5]
5	[-1.391, -1.449, -60.083, -7.190, -0.5]	[-1.391, -1.449, -60.056, -7.189, -0.5]	[-1.391, -1.449, -60.021, -7.188, -0.5]	[-1.391, -1.449, -59.986, -7.187, -0.5]	[-1.391, -1.449, -59.968, -7.187, -0.5]	[-1.391, -1.449, -59.950, -7.186, -0.5]	[-1.391, -1.449, -59.915, -7.186, -0.5]	[-1.391, -1.448, -59.880, -7.185, -0.5]	[-1.391, -1.448, -59.863, -7.184, -0.5]
10	[-1.391, -1.450, -60.563, -7.250, -0.5]	[-1.391, -1.450, -60.510, -7.248, -0.5]	[-1.391, -1.450, -60.439, -7.247, -0.5]	[-1.391, -1.449, -60.368, -7.245, -0.5]	[-1.391, -1.449, -60.332, -7.244, -0.5]	[-1.391, -1.449, -60.297, -7.243, -0.5]	[-1.391, -1.449, -60.226, -7.241, -0.5]	[-1.391, -1.448, -60.155, -7.240, -0.5]	[-1.391, -1.448, -60.120, -7.239, -0.5]
15	[-1.392, -1.452, -61.301, -7.349, -0.5]	[-1.392, -1.451, -61.219, -7.347, -0.5]	[-1.391, -1.451, -61.111, -7.344, -0.5]	[-1.391 -1.450, -61.003, -7.342, -0.5]	[-1.391, -1.450, -60.949, -7.340, -0.5]	[-1.391, -1.450, -60.895, -7.339, -0.5]	[-1.391, -1.449, -60.788, -7.337, -0.5]	[-1.391, -1.448, -60.680, -7.334, -0.5]	[-1.391, -1.448, -60.627, -7.333, -0.5]
20	[-1.392, -1.453, -62.312, -7.491, -0.5]	[-1.392, -1.453, -62.202, -7.488, -0.5]	[-1.392, -1.452, -62.054, -7.485, -0.5]	[-1.392 -1.451, -61.907, -7.481, -0.5]	[-1.391, -1.451, -61.834, -7.479, -0.5]	[-1.391, -1.450, -61.761, -7.477, -0.5]	[-1.391, -1.450, -61.615, -7.474, -0.5]	[-1.391, -1.449, -61.470, -7.470, -0.5]	[-1.391, -1.448, -61.398, -7.468, -0.5]