Noname manuscript No.
(will be inserted by the editor)

The Furuta Pendulum

Technical Report

**Tabea A. Wilke · Yannik P. Frisch ·
Maximilian A. Gehrke**

Received: date / Accepted: date

Abstract The Furuta Pendulum is a complex non-linear system and therefore of big interest in control system theory. It consists of one controllable arm rotating in the horizontal plane and one pendulum, which is attached to the end of this arm, moving uncontrollably in the vertical plane.
The non-linearities result from an interplay between gravitational, coriolis, centrifugal forces and friction. They can be modeled through equations of motion through the Euler-Lagrange method.
We present an overview of the systems technical details and discuss algorithms to solve the control problem.

Keywords Furuta Pendulum · Reinforcement Learning · Equations of motion

1 Introduction

Many examples in the field of control engineering like aircraft landing or aircraft stabilizing can be well modeled by an inverted pendulum (Akhtaruzzaman and Shafie 2010). As a reaction to problems with the limited movement of the cart from the inverted pendulum, the Furuta pendulum (also called rotary inverted pendulum) has been developed by Furuta et al. (1992). The advantages of the Furuta pendulum are its smaller need for space and one moving arm is directly linked to the motor. Therefore, the dynamics is less unmodeled thanks to a power transmission mechanism (Furuta

Tabea A. Wilke
TU Darmstadt, Germany
E-mail: tabeaalina.wilke@stud.tu-darmstadt.de

Yannik P. Frisch
TU Darmstadt, Germany
E-mail: yannik_phil.frisch@stud.tu-darmstadt.de

Maximilian A. Gehrke
TU Darmstadt, Germany
E-mail: maximilian_alexander.gehrke@stud.tu-darmstadt.de

et al. 1992).
The Furuta pendulum is an underactuated system which is characterized by the difference of the degrees of freedom and the number of actuators. If the degrees of freedom dominate, the system is trivially underactuated (Tedrake 2009). Hence, it is not possible to accelerate the system in every direction. In case of the Furuta pendulum, this means that there are two degrees of freedom (ϕ, θ), but only one arm is directly controlled by the motor. This is the one which changes the angle ϕ in the horizontal plane. The other arm, called pendulum, is attached to the end of the controlled arm and therefore is moved indirectly by it in the vertical plane with angle θ (Spong 1998; Tedrake 2009). The whole system is highly non-linear as a result of an interplay between gravitational, coriolis, centrifugal forces and friction (Izutsu et al. 2008).

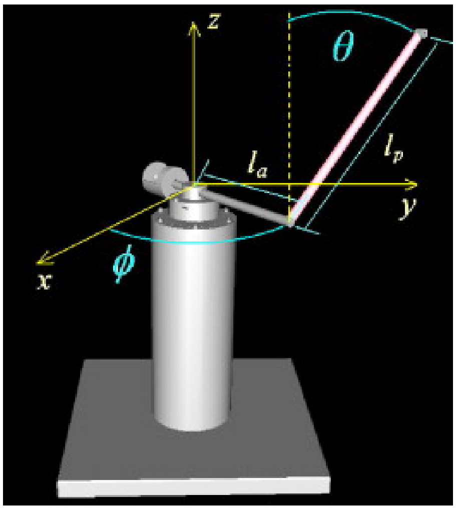

Fig. 1 A typical structure of the Furuta pendulum with the most important variables, figure from La Hera et al. (2009).

Finally, there are two different types of control problems, ”swing-up” and ”stabilization”. The swing-up task consists in bringing the flexible arm from a hanging position to a position which is nearly upright. Stabilization considers the problem when the arm is nearly upright with low enough speed, balancing it to hold this state stable.

1.1 Definitions

The system (see Fig. 1) consists of an arm with length l_a mounted to a DC motor, which is able to apply a torque τ to it. It has a mass of m_a. Another arm with length l_p and mass m_p is attached to the remaining side of the first arm. Both arms have a moment of inertia J_a and J_p respectively. The counter force to the input torque is the viscous damping B_a caused by the bearings of the motor. As the pendulum is not controlled directly, the only force that is directly applied to the pendulum is damping

B_p at the connection to the arm. The angles are given in radian, where an angle θ of zero refers to the upright position.

2 Mathematical Modelling

The kinematics of the pendulum's center of gravity can be described by

$$\mathbf{p_{cg}} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} l_a \cos\phi - l_p \sin\theta \sin\phi \\ l_a \sin\phi + l_p \sin\theta \cos\phi \\ l_p \cos\theta \end{bmatrix}.$$

The relative velocity of the pendulum to the arm can be represented by

$$\mathbf{v_{cg}} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} -l_a \ \sin\phi\dot{\phi} - l_p \ \cos\phi \sin\theta\dot{\phi} - l_p \ \sin\phi\cos\theta\dot{\theta} \\ l_a \ \cos\phi\dot{\phi} - l_p \ \sin\phi\sin\theta\dot{\phi} + l_p \ \cos\theta\cos\phi\dot{\theta} \\ -l_p \ \sin\theta\dot{\theta} \end{bmatrix}.$$

The equations of motion can be derived through the Euler-Lagrange method. The Lagrangian function is given by the difference of the kinetic and potential energies $L = T - V$, with T as the total sum of kinetic energies in the system and V as the total potential energies.
The only potential energy in the system is the one of the pendulum:

$$V_{total} = V_{pendulum} = m_p g z = m_p g l_p \cos\theta.$$

Kinetic energy is available both in the arm and in the pendulum and is composed of the sum of translational energy and rotational energy. The translational energy $T = \frac{1}{2}mv^2$ uses the squared velocity. This is defined by the sum of the squared velocities in all directions:

$$\begin{aligned} v^2 &= \dot{x}^2 + \dot{y}^2 + \dot{z}^2 \\ &= (l_a^2 + l_p^2 \ \sin^2\theta)\dot{\phi}^2 + l_p^2\dot{\theta}^2 + 2l_a l_p\dot{\phi}\dot{\theta}\cos\theta \end{aligned}$$

$$\begin{aligned} T_{arm} &= \frac{1}{2}J_a\dot{\phi}^2 \\ T_{pendulum} &= \frac{1}{2}J_p\dot{\theta}^2 + \frac{1}{2}m_p v^2 \\ &= \frac{1}{2}J_p\dot{\theta}^2 + \frac{1}{2}m_p\left((l_a^2 + l_p^2 \ \sin^2\theta)\dot{\phi}^2 + l_p^2\dot{\theta}^2 + 2l_a l_p\dot{\phi}\dot{\theta}\cos\theta\right) \\ T_{total} &= \frac{1}{2}J_a\dot{\phi}^2 + \frac{1}{2}J_p\dot{\theta}_0{}^2 + \frac{1}{2}m_p\left((l_a^2 + l_p^2 \ \sin^2\theta)\dot{\phi}^2 + l_p^2\dot{\theta}^2 + 2l_a l_p\dot{\phi}\dot{\theta}\cos\theta\right). \end{aligned}$$

From this the Lagrange Function arises:

$$\begin{aligned} L &= T_{total} - V_{total} \\ &= \frac{1}{2}J_a\dot{\phi}^2 + \frac{1}{2}J_p\dot{\theta}^2 + \frac{1}{2}m_p\left((l_a^2 + l_p^2 \ \sin^2\theta)\dot{\phi}^2 + l_p^2\dot{\theta}^2 + 2l_a l_p\dot{\phi}\dot{\theta}\cos\theta\right) \\ &\quad - m_p g l_p \cos\theta. \end{aligned}$$

Lagrange's Equation follows

$$\frac{d}{dt}\left(\frac{\partial L(q,\dot{q},t)}{\partial \dot{q}_i}\right) - \frac{\partial L(q,\dot{q},t)}{\partial q_i} + \frac{\partial F(q,\dot{q},t)}{\partial \dot{q}_i} = 0.$$

Thereby, the variables q_i are called generalized coordinates. F is the friction in the system, which can be described with the damping coefficients (B_a, B_p) in the form of $B_p\dot{\theta}$ and $B_a\dot{\phi}$. For the Furuta pendulum the generalized coordinates are $q(t)^T = [\phi\ \theta]$ with the derivation according to time $\dot{q}(t)^T = \left[\frac{\partial\phi(t)}{\partial t}\ \frac{\partial\theta(t)}{\partial t}\right]$. Together, this leads to

$$\frac{\partial^2 L}{\partial t \partial \dot{\phi}} - \frac{\partial L}{\partial \phi} = \tau - B_a\dot{\phi} \tag{1}$$

$$\frac{\partial^2 L}{\partial t \partial \dot{\theta}} - \frac{\partial L}{\partial \theta} = -B_p\dot{\theta}. \tag{2}$$

The partial derivations of the Lagrangian for the generalized coordinates are:

$$\begin{aligned}
\frac{\partial L}{\partial \phi} &= 0 \\
\frac{\partial L}{\partial \dot{\phi}} &= \dot{\phi}(J_a + m_p(l_a^2 + l_p^2\sin^2\theta)) + \dot{\theta}(m_p l_a l_p \cos\theta) \\
\frac{\partial^2 L}{\partial t \partial \dot{\phi}} &= \ddot{\phi}(J_a + m_p(l_a^2 + l_p^2\sin^2\theta)) + l_p^2 m_p \sin(2\theta)\dot{\phi}\dot{\theta} + m_p l_a l_p(\cos\theta\ddot{\theta} - sin\theta\dot{\theta}^2) \\
\frac{\partial L}{\partial \theta} &= \frac{1}{2}\dot{\phi}^2 m_p l_p^2 \sin(2\theta) - m_p l_a l_p sin\theta\dot{\phi}\dot{\theta} + m_p g l_p sin\theta \\
\frac{\partial L}{\partial \dot{\theta}} &= \dot{\theta}(J_p + m_p l_p^2) + \dot{\phi}(m_p l_a l_p \cos\theta) \\
\frac{\partial^2 L}{\partial t \partial \dot{\theta}} &= \ddot{\theta}(J_p + m_p l_p^2) + \ddot{\phi} m_p l_a l_p \cos\theta - \dot{\phi}\dot{\theta} m_p l_a l_p \sin\theta.
\end{aligned}$$

With these equations it is possible to derive the Euler-Lagrange's Equations:

$$\begin{aligned}
\frac{\partial^2 L}{\partial t \partial \dot{\phi}} - \frac{\partial L}{\partial \dot{\phi}} &= \ddot{\phi}(J_a + m_p(l_a^2 + l_p^2\sin^2\theta)) + l_p^2 m_p \sin(2\theta)\dot{\phi}\dot{\theta} \\
&\quad + m_p l_a l_p(\cos\theta\ddot{\theta} - sin\theta\dot{\theta}^2) \\
\frac{\partial^2 L}{\partial t \partial \dot{\theta}} - \frac{\partial L}{\partial \dot{\theta}} &= \ddot{\theta}(J_p + m_p l_p^2) + \ddot{\phi} m_p l_a l_p \cos\theta - \frac{1}{2}\dot{\phi}^2 m_p l_p^2 \sin(2\theta) - m_p g l_p sin\theta.
\end{aligned}$$

Applying matrix notation and filling in the friction, this results in the nonlinear equations of motion (EOM) for the Furuta pendulum:

$$\begin{bmatrix} J_a + m_p(l_a^2 + l_p^2\sin^2\theta) & m_p l_a l_p \cos\theta \\ m_p l_a l_p \cos\theta & J_p + m_p l_p^2 \end{bmatrix} \begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \end{bmatrix} +$$

$$\begin{bmatrix} m_p l_p^2 \sin(2\theta)\dot{\theta} + B_a & -m_p l_a l_p \sin\theta\dot{\theta} \\ -\frac{1}{2} m_p l_p^2 \sin(2\theta)\dot{\phi} & B_p \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ -m_p l_p g \sin\theta \end{bmatrix} = \begin{bmatrix} \tau \\ 0 \end{bmatrix}.$$

2.1 Linearization of the State-Space Model

As all equations of motion contain a part of a trigonometric function, the equations are non-linear. There are several different methods to transform non-linear to linear equations, for example Taylor expansion with substituting non-linear parts (Hamza et al. 2015), Jacobian linearization (Al-Jodah et al. 2013) or optimal linearization (Zhang and Zhang 2011). For a linearization in the state-space the equation $\dot{x} = Ax + Bu$ with $x = [\phi \;\; \theta \;\; \dot{\phi} \;\; \dot{\theta}]$ is used. The linearization is done around the operating point (Furuta et al. 1992), which is the vertical inverted state and, therefore, uses $x_0 = [0\;0\;0\;0]^T$. This results in the following equations:

$$\dot{x} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \frac{\partial f_1(x,\tau)}{\partial \phi}\Big|_{x=x_0} & \frac{\partial f_1(x,\tau)}{\partial \theta}\Big|_{x=x_0} & \frac{\partial f_1(x,\tau)}{\partial \dot{\phi}}\Big|_{x=x_0} & \frac{\partial f_1(x,\tau)}{\partial \dot{\theta}}\Big|_{x=x_0} \\ \frac{\partial f_2(x,\tau)}{\partial \phi}\Big|_{x=x_0} & \frac{\partial f_2(x,\tau)}{\partial \theta}\Big|_{x=x_0} & \frac{\partial f_2(x,\tau)}{\partial \dot{\phi}}\Big|_{x=x_0} & \frac{\partial f_2(x,\tau)}{\partial \dot{\theta}}\Big|_{x=x_0} \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ \frac{\partial f_1(x,\tau)}{\partial \tau}\Big|_{x=x_0} \\ \frac{\partial f_2(x,\tau)}{\partial \tau}\Big|_{x=x_0} \end{bmatrix} \tau \tag{3}$$

with f_1 as Eq. (1) and f_2 as Eq. (2).
The result is a linear form of the equations of motion which are very close to the system's description by the non-linear model. For an angle of $\theta < 25°$ there is nearly no discrepancy to the actual motion (Kurode et al. 2011).

3 Pendulum Control

The goal of controlling the Furuta pendulum is to bring it from a hanging position into a vertical upright position. Thus, it is necessary to generate enough energy to swing the pendulum up into a nearly upright position, where the linear region begins (Kurode et al. 2011). If this region is reached, the controller should change to the balancing mode to stabilize the upright position.
To check whether the controller needs to be switched from the swing-up task to the stabilization task, the switching criteria can for example be defined by (Hamza et al. 2019):

$$\text{switching criteria} \begin{cases} \text{stabilization} & \begin{cases} |\theta| < \frac{\pi}{9} & \text{and } \dot{\theta} < 2.62 \text{ rad/sec} \\ |E - E_r| < 0.04 \text{ Joule} & \text{and } \dot{\theta} < 2.62 \text{ rad/sec} \end{cases} \\ \text{swing-up} & \text{otherwise} \end{cases} .$$

3.1 Swing-Up

There are many different approaches for solving the swing-up task of the Furuta pendulum: linear ones which use the linearized equations of motion, non-linear ones and model-free approaches. The classical way of solving the swing-up task is an approach with energy control (Seman et al. 2013). The sum of kinetic and potential energy of the pendulum is used to model which amount of energy should be added. The energy of the pendulum can be described by

$$E = \frac{1}{2} J_p \theta^2 + m_p g l_p (\cos\theta - 1),$$

which was already done by Åström and Furuta (2000) for the normal pendulum. As we want to stabilize the pendulum in an upright position, optimally we do not need any further energy in this state. Therefore, this is a logical state to set as state without any energy left (Seman et al. 2013).

There are two different directions in which the energy can be supplied. Either the pendulum seems to be ”pulled” or ”pushed” (Seman et al. 2013). Therefore, we need a sign-function that represents whether to push or pull. The argument of the sign-function is compound by the velocity which specifies whether the pendulum starts to swing in the opposite direction and a cosine function that defines the side of the pendulum. This results in the term, $\mathrm{sign}(\dot\theta \cos\theta)$ (Awtar et al. 2002).

The magnitude of energy which should be provided is selected by an aggressivity factor and the difference of actual and desired system energy $k_a(E - E_0)$, with $E_0 = 0$ as the system should have zero energy at the end. The aggressivity factor handles the amount of input and thereby the number of oscillations which are needed to achieve the swing-up (Awtar et al. 2002). Often there is also a saturation function which limits the signal to the maximum acceleration of the pendulum. This is done with a linear function, in the following described as a wildcard *sat*. The whole swing-up controller is then implemented by

$$u = sat(k_a(E - E_0))\mathrm{sign}(\dot\theta \cos\theta).$$

3.2 Stabilization

There are many different approaches for solving the stabilization problem as well. The most common one is the linear quadratic regulation (LQR) because it guarantees the optimal solution (Hamza et al. 2019). Also the proportional integral derivative (PID) is often used because of its simplicity and robustness and also because of its usage in industry (Hassanzadeh and Mobayen 2011). Nevertheless, there are a lot of other algorithms from sliding modes (Izutsu et al. 2008) to particle swarm optimization and other evolutionary algorithms (Hassanzadeh and Mobayen 2011).

In the following, we will shortly describe how an LQR controller can be modeled for the stabilization problem of the Furuta Pendulum. The LQR is a controller which can be used to make a controllable, but unstable system stable (Park et al. 2011). It uses the linearized equations of motion in the state-space model in form of a state feedback method (Özbek and Efe 2010). The cost function can be defined with $x = [\phi\ \theta\ \dot\phi\ \dot\theta]^T$, R and Q as weighting matrix by (Al-Jodah et al. 2013),

$$J = \int_0^\infty \left(x^T Q x + u^T R u \right) \mathrm{d}t.$$

The constraint for the cost function is the system dynamics in form of a differential equation derived from the linearization $\dot x = Ax + Bu$, see Eq. (3). To solve the cost function, linear state feedback with a constant gain matrix K is used: $u(t) = -Kx(t)$ with $K = -R^{-1}B^TP$. $P(t)$ is the solution matrix which will stabilize ($\dot{P}(t) = 0$) for a convergent system (Chen et al. 2007). It is a symmetric matrix which holds the algebraic Riccati equation (ARE) (Al-Jodah et al. 2013)

$$PA + A^TP - PBR^{-1}B^TP + Q = 0.$$

4 Reinforcement Learning on the Furuta Pendulum

In reinforcement learning, the Furuta pendulum has only a very small influence on research maybe also because of the complexity of the problem. Only a few algorithms were tested on it with diverse results. Reinforcement learning should play a bigger role in solving the Furuta pendulum because it could improve the robustness of the stabilization control (Wang et al. 2004).

Hennig (2011) compares a Gaussian process optimal learner with Kalman filter, TD(λ) and full information as baseline. He used the same baseline function for all algorithms and collated the cumulated loss. Surprisingly, none of the methods could stabilize the swung up pendulum totally, even not the best method, the Gaussian process optimal learner.

An artificial neural network (ANN) was used to model the pendulum (Quyen et al. 2012). They worked with the voltage of the motor as input values and the angles of arm and pendulum as the output. They used a PID controller for a supervised training. Different numbers of hidden neurons were tested with the result that more neurons in the hidden layer led to a smaller mean squared error. The ANN could predict the angle of the pendulum very well.

Another approach to stabilize the pendulum was successfully done with recurrent neural networks (RNN) and a Genetic Algorithm (Shojaei et al. 2011). They used a recurrent neural network identifier and controller with a proportional integral derivative controller, which was responsible for the feedback and the prediction. The RNN identifier checks whether the output of the feedback system is valid or not. The RNN controller handles the power of the system.

The ”Application of Reinforcement Learning Methods” course of the TU Darmstadt implemented different reinforcement learning algorithms on different platforms. Some of the algorithms were also tested on the Qube-v0 environment from Quanser Robots, which implements the Furuta Pendulum. The maximum reward per episode was close to 6.0 with a maximum episode length of 300 steps and a reward per step of 0.02. The results can be found in Fig. 2 with a simulation evaluation of 100 episodes. None of the different algorithms clearly outperform the others. They have very similar results.

Further on, two groups managed to run their algorithm on the physical Furuta Pendulum from Quanser Robots and received received a cumulative reward of 4.88 (PPO, 25 evaluated episodes) and 4.2 (DQN, 10 evaluated episodes).

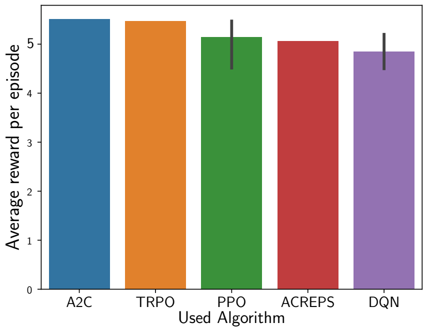

Fig. 2 The figure shows the best results for the different reinforcement learning algorithms of the ”Advanced Reinforcement Learning Methods” course of the TU Darmstadt. Some of the algorithms were solved by different groups, which is shown by a standard deviation of the achieved results.

5 Conclusion

The Furuta pendulum is a very complex control problem, which could be solved by a very wide variety of approaches. The control problem splits into two major problems, the swing-up and the balancing task. For most of the approaches the equations of motion are needed. Depending on the construction of the pendulum, there are different formulations of the problem. The different algorithms show very different approaches from very simple ones to complex or reinforcement learning approaches. Many of them lack stable results, what should be further evaluated with a stronger focus on reinforcement learning (Wang et al. 2004).
It was possible to learn a good policy with many different reinforcement learning algorithms in simulation through a project of the TU Darmstadt. Much more complicated was the usage on the real robot.

References

Akhtaruzzaman M, Shafie AA (2010) Modeling and control of a rotary inverted pendulum using various methods, comparative assessment and result analysis. In: 2010 IEEE International Conference on Mechatronics and Automation, IEEE, pp 1342–1347

Al-Jodah A, Zargarzadeh H, Abbas MK (2013) Experimental verification and comparison of different stabilizing controllers for a rotary inverted pendulum. In: 2013 IEEE International Conference on Control System, Computing and Engineering, IEEE, pp 417–423

Åström KJ, Furuta K (2000) Swinging up a pendulum by energy control. Automatica 36(2):287–295

Awtar S, King N, Allen T, Bang I, Hagan M, Skidmore D, Craig K (2002) Inverted pendulum systems: rotary and arm-driven-a mechatronic system design case study. Mechatronics 12(2):357–370

Chen Y, Atherton DP, et al. (2007) Linear feedback control: analysis and design with MATLAB, vol 14. Siam

Furuta K, Yamakita M, Kobayashi S (1992) Swing-up control of inverted pendulum using pseudo-state feedback. Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering 206(4):263–269

Hamza MF, Yap HJ, Choudhury IA (2015) Genetic algorithm and particle swarm optimization based cascade interval type 2 fuzzy pd controller for rotary inverted pendulum system. Mathematical Problems in Engineering 2015

Hamza MF, Yap HJ, Choudhury IA, Isa AI, Zimit AY, Kumbasar T (2019) Current development on using rotary inverted pendulum as a benchmark for testing linear and nonlinear control algorithms. Mechanical Systems and Signal Processing 116:347–369

Hassanzadeh I, Mobayen S (2011) Controller design for rotary inverted pendulum system using evolutionary algorithms. Mathematical Problems in Engineering 2011

Hennig P (2011) Optimal reinforcement learning for gaussian systems. In: Advances in Neural Information Processing Systems, pp 325–333

Izutsu M, Pan Y, Furuta K (2008) Swing-up of furuta pendulum by nonlinear sliding mode control. SICE Journal of Control, Measurement, and System Integration 1(1):12–17

Kurode S, Chalanga A, Bandyopadhyay B (2011) Swing-up and stabilization of rotary inverted pendulum using sliding modes. IFAC Proceedings Volumes 44(1):10685–10690

La Hera PX, Freidovich LB, Shiriaev AS, Mettin U (2009) New approach for swinging up the furuta pendulum: Theory and experiments. Mechatronics 19(8):1240–1250

Özbek NS, Efe MÖ (2010) Swing up and stabilization control experiments for a rotary inverted pendulum—an educational comparison. In: 2010 IEEE International Conference on Systems, Man and Cybernetics, IEEE, pp 2226–2231

Park M, Kim YJ, Lee JJ (2011) Swing-up and lqr stabilization of a rotary inverted pendulum. Artificial Life and Robotics 16(1):94–97

Quyen ND, Thuyen N, Hoc NQ, Hien ND (2012) Rotary inverted pendulum and control of rotary inverted pendulum by artificial neural network. In: Proc. Natl. Conf. Theor. Phys, vol 37, pp 243–249

Seman P, Juh M, Salaj M, et al. (2013) Swinging up the furuta pendulum and its stabilization via model predictive control. journal of Electrical Engineering 64(3):152–158

Shojaei A, Othman MF, Rahmani R, Rani M (2011) A hybrid control scheme for a rotational inverted pendulum. In: 2011 UKSim 5th European Symposium on Computer Modeling and Simulation, IEEE, pp 83–87

Spong MW (1998) Underactuated mechanical systems. In: Control problems in robotics and automation, Springer, pp 135–150

Tedrake R (2009) Underactuated robotics: Learning, planning, and control for efficient and agile machines: Course notes for mit 6.832. Working draft edition 3

Wang Z, Chen Y, Fang N (2004) Minimum-time swing-up of a rotary inverted pendulum by iterative impulsive control. In: Proceedings of the 2004 American Control Conference, IEEE, vol 2, pp 1335–1340

Zhang J, Zhang Y (2011) Optimal linear modeling and its applications on swing-up and stabilization control for rotary inverted pendulum. In: Proceedings of the 30th Chinese Control Conference, IEEE, pp 493–500