CYBERNETICS AND PHYSICS, VOL. 1, NO. 1, 2012 , 28–35

ADAPTIVE NONLINEAR TRACKING FOR ROBOTIC WALKING

Kamil Dolinský
Dep. of Control Engineering, CTU in Prague
Institute of IT and Automation,
Academy of Sciences of the Czech Republic
kamil.dolinsky@fel.cvut.cz

Sergej Čelikovský
Dep. of Control Engineering, CTU in Prague
Institute of IT and Automation,
Academy of Sciences of the Czech Republic
celikovs@utia.cas.cz

Abstract
This article deals with online adaptation of control strategy for nonlinear tracking of a walking like motion of bipedal robot. Adaptation of the control rule is done according to results of online parameter estimation. Parameter estimation was realized by an extended Kalman filter due to recursive nature of the estimation problem and abundant a priori information. Proposed estimation strategy yields at least three advantages. By utilization of extensive knowledge about the system in consideration a multi-variable estimation problem was reduced to estimation problem involving one parameter only. A heavy computation burden required for recomputation of reference trajectory and feed-forward controller is removed. This approach can also be used to eliminate the modeling mismatch. A practical situation when a robot has to carry a load of an unknown weight is demonstrated.

Key words
Adaptive control, Kalman filter, walking robots.

1 Introduction

The problem of bipedal robotic walking has been extensively studied in last decades [Isidori, 1996], [Spong, 1998] [Grizzle et. al., 2005]. It comprises numerous problems which bring together various fields of science, e.g. nonlinear control design, state estimation and state trajectory planing. From the studies regarding the problem of the nonlinear tracking one can conclude that the performance of the control algorithm depends heavily on its ability to utilize as much information about the controlled system as possible [Anderle et. al., 2010], [Anderle and Čelikovský, 2009], [Anderle et. al., 2009].

However when corrupted information is used the performance degrades considerably. In the light of the research results that were obtained in the field of the state estimation [Simon, 2006], one can conclude that utilization of filtering algorithms is necessary when some measurements are corrupted with noise.

One of the most frequently used filtering algorithm for nonlinear systems is the Extended Kalman Filter (EKF) originating from the work of [Kalman, 1960] or it's continuous counterpart the Extended Kalman-Bucy Filter (EKBF). This approach uses Taylor series to linearize the nonlinear system around the Kalman filter estimate. This idea was originally proposed S. Schmidt for nonlinear spacecraft navigational problems [Bellantoni and Dodge, 1967]. Although many successful applications were reported the general experience is that the performance of the EKBF is not guaranteed and that it is not easy to to tune [Fitzgerald, 1971].

Indeed for the classical parametrization of the system the filter fails to estimate the parameters that are affected by an unknown load that the robot should carry. To overcome mentioned deficiencies a well suited parametrization of the system in consideration is introduced resulting in utilization of as much information about the system as possible. The use of such a parametrization results in problem of standard state estimation and recursive identification of one unknown model parameter. Also the tunning of the filter lies only in finding a suitable value of one parameter, the variance of the noise used to model a random change of this unknown model parameter as a reasonable estimate of variance of the measurement noise is known a priori and the dynamical system is not significantly influenced by any other disturbances that would require any further modeling either in deterministic or stochastic framework.

The walking like motion of the robot is defined by a suitable state trajectory that depends on the model's parameters. The recursive parameter estimation implies the need of online adaptation of the reference trajectory. However the computation of the reference trajectory is a computationally very expensive task. To reduce the heavy computational burden the trajectory can be precomputed off-line and approximated by a table or using regression techniques [Dennis and Schnabel, 1983], [Dennis and Gay, 1981].

The paper is organized as follows, section one describes the mathematical model used for robot modeling. The second section deals with the state estimation and recursive parameter identification. The key ideas of the nonlinear tracking algorithm and the adaptation scheme of the reference trajectory is summarized in the third section. Simulations that support and demonstrate proposed approach are given in the fifth section while the sixth section summarizes the advantages and disadvantages of chosen approach, points out the future works and concludes this article.

2 Model of the Acrobot

In this article we consider following model of Acrobot, depicted on Fig. 1.

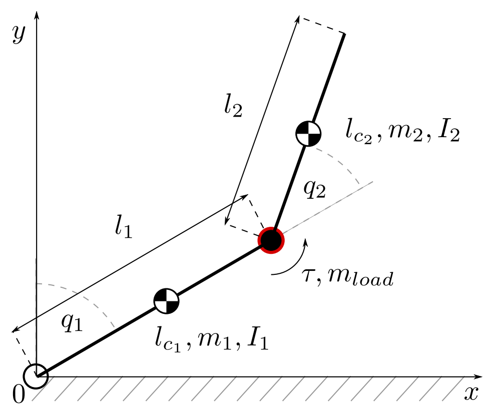

Figure 1. The Acrobot.

Physical quantities that describe model of Acrobot are summed up in Tab. 1.

Table 1. Parameters of the Acrobot

l_1, l_2	length of 1st, 2nd link	$[m]$
l_{c_1}, l_{c_2}	center of gravity of 1st, 2nd link	$[m]$
m_1, m_2	masses of 1st, 2nd link	$[kg]$
m_{load}	mass of applied load	$[kg]$
τ	applied torque	$[N.m]$
g	gravitational acceleration	$[m.s^{-2}]$
I_1, I_2	inertia moments of 1st, 2nd link	$[kg.m^2]$

Two rigid links are joined by a joint. This joint is actuated by a DC motor. Position of the system is uniquely defined by two angles q_1 and q_2, thus the system has two degrees of freedom, yet there is only one control input, torque τ generated by DC motor. Therefore Acrobot represents an underactuated system, with degree of underactuation equal to one and underactuated angle q_1. The state vector $\boldsymbol{x}$ of Acrobot is composed from generalised co-ordinates - angles q_1, q_2 and generalised velocities $\dot{q}_1$, $\dot{q}_2$. Measured state variables of our laboratory model are q_1, $\dot{q}_1$, q_2 measured by a leaser beam sensor, digital gyroscope and an incremental sensor respectively thus

$$\left.\begin{array}{ll} \boldsymbol{x} = (q_1,\ \dot{q}_1,\ q_2,\ \dot{q}_2)^T, & \\ \boldsymbol{q} = (q_1,\ q_2)^T, & \dot{\boldsymbol{q}} = (\dot{q}_1,\ \dot{q}_2)^T, \\ \boldsymbol{u} = (0\ ,\tau)^T, & \boldsymbol{y} = (q_1,\ \dot{q}_1,\ q_2)^T. \end{array}\right\} \quad (1)$$

The unmeasured state $\dot{q}_2$ has to be estimated. Directly measurable model parameters of Acrobot are l_1, l_2, m_1, m_2, g and parameters $l_{c_1},\ l_{c_2}, I_1, I_2$ can be computed using physical laws or identified off-line as well as any other additional parameters, e.g. friction coefficients.

2.1 Equations of motion

To obtain equations of motion (EoM) for Acrobot we use classical Lagrangian approach [Landau and Lifshitz, 1976]. If we introduce following substitution

$$\left.\begin{array}{l} \theta_1 = m_1 l_{c_1}^2 + I_1 + m_2 l_1^2, \\ \theta_2 = m_2 l_{c_2}^2 + I_2, \\ \theta_3 = m_2 l_1 l_{c_2}, \\ \theta_4 = m_1 l_{c_1} + m_2 l_1, \\ \theta_5 = m_2 l_{c_2}, \end{array}\right\} \quad (2)$$

where parameters $m_1, m_2, l_{c_1}, l_{c_2}, I_1, I_2$ depend on the value of m_{load} and some constants $m_1^{'}, m_2^{'}, l_{c_1}^{'}, l_{c_2}^{'}, I_1^{'}, I_2^{'}$ that are known a priori. The resulting Equations of Motion (EoM) of Acrobot in Lagrange formalism are

$$\mathbf{D}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \mathbf{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \mathbf{G}(\boldsymbol{q}) = \boldsymbol{u} \quad (3)$$

where matrices

$$\mathbf{D} = \begin{bmatrix} \theta_1 + \theta_2 + 2\theta_3 \cos(q_2); & \theta_2 + \theta_3 \cos(q_2) \\ \theta_2 + \theta_3 \cos(q_2); & \theta_2 \end{bmatrix},$$

$$\mathbf{C} = \begin{bmatrix} -2\theta_3 \sin(q_2)\dot{q}_2; & -\theta_3 \sin(q_2)\dot{q}_2 \\ \theta_3 \sin(q_2)\dot{q}_1; & 0 \end{bmatrix}, \quad (4)$$

$$\mathbf{G} = \begin{bmatrix} -\theta_4 g \sin(q_1) - \theta_5 g \sin(q_1 + q_2) \\ -\theta_5 g \sin(q_1 + q_2) \end{bmatrix}.$$

Matrix $\mathbf{D}$ contains inertia terms, matrix $\mathbf{C}$ contains centripetal and Coriolis force terms and matrix $\mathbf{G}$ contains gravity terms.

3 Extended Kalman-Bucy filter

The well known EKBF is a estimation tool of choice for many practical applications. Although due to implementation difficulties its discrete or hybrid counterpart is preferred, we will use the continuous version as it easier to present together with continuous control law that will be introduced in next section. With the sufficiently small sampling period other versions provide similar performance.

Acrobot's equations of motion can be represented in following form

$$\left.\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{f}(\boldsymbol{x},\boldsymbol{u}) + \boldsymbol{w} \\ \boldsymbol{y}(t) &= \mathbf{H}\boldsymbol{x} + \boldsymbol{v} \\ \boldsymbol{w}(t) &\sim (0,\mathbf{Q}) \\ \boldsymbol{v}(t) &\sim (0,\mathbf{R}) \end{aligned}\right\} \tag{5}$$

where t stands for time, $\boldsymbol{x} \in \mathbb{R}^n$ is the state and $\boldsymbol{y} \in \mathbb{R}^m$ is the output vector as defined in (1). Vectors $\boldsymbol{w} \in \mathbb{R}^n$, $\boldsymbol{v} \in \mathbb{R}^m$ denote the process and measurement noise respectively and it is assumed that they can be represented as normally distributed random processes with zero mean and covariance matrices $\mathbf{Q} \in \mathbb{R}^{n\times n}$ and $\mathbf{R} \in \mathbb{R}^{m\times m}$ respectively. Dynamics of the system is described by a vector field $\boldsymbol{f} : \mathbb{R}^n \to \mathbb{R}^n$ defined by (3) and matrix $\mathbf{H}$ is given as

$$\mathbf{H} = \begin{bmatrix} 1\,0\,0\,0 \\ 0\,1\,0\,0 \\ 0\,0\,1\,0 \end{bmatrix}. \tag{6}$$

The equations of the EKBF collapses the prediction and the measurement update steps into one resulting in following equations

$$\left.\begin{aligned} \dot{\hat{\boldsymbol{x}}}(t) &= \boldsymbol{f}(\hat{\boldsymbol{x}},\boldsymbol{u}) + \mathbf{K}(\boldsymbol{y} - \mathbf{H}\hat{\boldsymbol{x}}) \\ \mathbf{K}(t) &= \mathbf{P}\mathbf{H}^T\mathbf{R}^{-1} \\ \dot{\mathbf{P}}(t) &= \mathbf{A}\mathbf{P} + \mathbf{P}\mathbf{A}^T + \mathbf{L}\mathbf{Q}\mathbf{L}^T - \mathbf{K}\mathbf{R}\mathbf{K}^T \end{aligned}\right\} \tag{7}$$

where

$$\mathbf{A} = \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\right|_{\hat{\boldsymbol{x}}(t)}, \quad \mathbf{L} = \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{w}}\right|_{\hat{\boldsymbol{x}}(t)}. \tag{8}$$

Initial values for the integration are

$$\left.\begin{aligned} &\hat{\boldsymbol{x}}(0) = \mathbb{E}\{\boldsymbol{x}(0)\}, \\ &\mathbf{P}(0) = \mathbb{E}\{(\boldsymbol{x}(0) - \hat{\boldsymbol{x}}(0))(\boldsymbol{x}(0) - \hat{\boldsymbol{x}}(0))^T\}\,. \end{aligned}\right\} \tag{9}$$

For detailed derivation of the EKBF one can confront [Simon, 2006].

3.1 Parameter estimation

By considering some unknown model's parameter as a state of the system a recursive identification of this parameter is possible. In the particular case of Acrobot dynamics we can consider various options. Regarding the number of unknown parameters the use of the substitution (2) is useful as the number will decrease from six $(m_1, m_2, l_{c_1}, l_{c_2}, I_1, I_2)$ to five. Also use of the substitution will decrease the complexity of computations involved in evaluating the Jacobians (8) greatly. Unfortunately this approach does not work very well. The estimation is slow and often results in divergence of the estimates from the true model's parameters.

A solution to this problem is to use the analytical formulas for the computation of the model's parameters $(m_1, m_2, l_{c_1}, l_{c_2}, I_1, I_2)$, the general rules for the computation of the center of mass and inertia tensors, see [Landau and Lifshitz, 1976], and treating the parameters as a functions of one unknown parameter m_{load} and some a priori known constants $m_1^{'}$, $m_2^{'}$, $l_{c_1}^{'}$, $l_{c_2}^{'}$, $I_1^{'}$, $I_2^{'}$ that can be either measured directly or computed using off-line identification thus

$$\left.\begin{aligned} m_1 &= m_1(m_1^{'}, m_{load}) \\ m_2 &= m_2(m_2^{'}, m_{load}) \\ l_{c_1} &= l_{c_1}(m_1^{'}, l_{c_1}^{'}, m_{load}) \\ l_{c_2} &= l_{c_2}(m_2^{'}, l_{c_2}^{'}, m_{load}) \\ I_1 &= I_1(I_1^{'}, l_1, l_{c_1}^{'}, m_{load}) \\ I_2 &= I_2(I_2^{'}, l_2, l_{c_2}^{'}, m_{load})\,. \end{aligned}\right\} \tag{10}$$

The filter will use additional information and the number of unknown parameters will decrease to one. The use of the substitution (2) is still to filter's advantage as the analytical formulas for Jacobians (8) are much more complicated without the substitution.

The filter's equations can then be written as follows

$$\left.\begin{aligned} \dot{\hat{\overline{\boldsymbol{x}}}}(t) &= \overline{\boldsymbol{f}}(\hat{\overline{\boldsymbol{x}}},\boldsymbol{u}) + \overline{\mathbf{K}}(\boldsymbol{y} - \overline{\mathbf{H}}\hat{\overline{\boldsymbol{x}}}) \\ \overline{\mathbf{K}}(t) &= \overline{\mathbf{P}}\overline{\mathbf{H}}^T\mathbf{R}^{-1} \\ \dot{\overline{\mathbf{P}}}(t) &= \overline{\mathbf{A}}\overline{\mathbf{P}} + \overline{\mathbf{P}}\overline{\mathbf{A}}^T + \overline{\mathbf{L}}\overline{\mathbf{Q}}\overline{\mathbf{L}}^T - \overline{\mathbf{K}}\mathbf{R}\overline{\mathbf{K}}^T \end{aligned}\right\} \tag{11}$$

where

$$\left.\begin{aligned} \hat{\overline{\boldsymbol{x}}} &= [\hat{\boldsymbol{x}}, \hat{m}_{load}]^T & \overline{\boldsymbol{w}} &= [\boldsymbol{w},\ w_5] \\ \overline{\boldsymbol{f}} &= [\boldsymbol{f}, 0]^T & \overline{\mathbf{H}} &= [\mathbf{H}, \mathbf{0}] \end{aligned}\right\} \tag{12}$$

and

$$\overline{\mathbf{A}} = \begin{bmatrix} \mathbf{A} & \boldsymbol{A}^* \\ \mathbf{0} & 0 \end{bmatrix}, \overline{\mathbf{L}} = \begin{bmatrix} \mathbf{L} & \boldsymbol{L}^* \\ \mathbf{0} & 0 \end{bmatrix}, \overline{\mathbf{Q}} = \begin{bmatrix} \mathbf{Q} & \mathbf{0} \\ \mathbf{0} & Q^* \end{bmatrix} \tag{13}$$

with

$$\left.\begin{aligned} A^* &= \sum_{i=1}^{i=5} \frac{\partial \boldsymbol{f}}{\partial \theta_i} \left.\frac{\partial \theta_i}{\partial m_{load}}\right|_{\hat{\overline{\boldsymbol{x}}}(t)}, \\ L^* &= \sum_{i=1}^{i=5} \frac{\partial \boldsymbol{f}}{\partial \theta_i} \left.\frac{\partial \theta_i}{\partial w_5}\right|_{\hat{\overline{\boldsymbol{x}}}(t)}. \end{aligned}\right\} \tag{14}$$

Initial values for the integration are

$$\left.\begin{aligned} &\hat{\overline{\boldsymbol{x}}}(0) = \mathbb{E}\{\overline{\boldsymbol{x}}(0)\}, \\ &\overline{\mathbf{P}}(0) = \mathbb{E}\{(\overline{\boldsymbol{x}}(0) - \hat{\overline{\boldsymbol{x}}}(0))(\overline{\boldsymbol{x}}(0) - \hat{\overline{\boldsymbol{x}}}(0))^T\}\,. \end{aligned}\right\} \tag{15}$$

As the system is not significantly influenced by any other forces that would require stochastic modeling the covariance matrix $\mathbf{Q}$ can be chosen as diagonal matrix with very small values of the diagonal elements. The only remaining parameter that should be specified is the variance Q^* of the random variable w_5. Therefore tunning of the filter is very simple.

4 Adaptation of the control law

The control law used to track the walking like motion consists from a feed-forward and feedback controller. Both are based on feedback linearization of the Acrobot.

4.1 Partial feedback linearization for Acrobot

The partial exact feedback linearization method is based on a system transformation into a new system of co-ordinates that display linear dependence between some auxiliary output and new input [Isidori, 1996].

In [Čelikovský et. al., 2008] was showed that using functions

$$\left.\begin{array}{l}\sigma = \dfrac{\partial \mathcal{L}}{\partial \dot{q}_1} = (\theta_1 + \theta_2 + 2\theta_3 \cos(q_2))\dot{q}_1 + \\ \quad +(\theta_2 + \theta_3 \cos(q_2))\dot{q}_2 \\ p = q_1 + \dfrac{q_2}{2} + \left(\dfrac{\theta_2 - \theta_1}{\sqrt{(\theta_1+\theta_2)^2 - 4\theta_3^2}} \right. \\ \quad \left. \arctan\left(\sqrt{\dfrac{\theta_1+\theta_2-2\theta_3}{\theta_1+\theta_2+2\theta_3}} \tan\frac{q_2}{2} \right) \right). \end{array}\right\} \tag{16}$$

which both have relative degree $r = 3$ one can transform the system (3) into new nearly linear system by following transformation

$$T:\ \xi_1 = p,\ \xi_2 = \sigma,\ \xi_3 = \dot{\sigma},\ \xi_4 = \ddot{\sigma} \tag{17}$$

and the resulting dynamic system will have following form

$$\left.\begin{array}{l}\dot{\xi}_1 = d_{11}^{-1}\xi_2,\ \dot{\xi}_2 = \xi_3,\ \dot{\xi}_3 = \xi_4 \\ \dot{\xi}_4 = \alpha(\boldsymbol{q},\dot{\boldsymbol{q}})\tau + \beta(\boldsymbol{q},\dot{\boldsymbol{q}}) = z\end{array}\right\} \tag{18}$$

where d_{11} is corresponding entry in inertia matrix $\mathbf{D}$ and z the new linearizing input which is well defined whenever $\alpha(\boldsymbol{q},\dot{\boldsymbol{q}})^{-1} \neq 0$. Also

$$\alpha = \frac{\det \mathbf{\Phi}_2}{\det \mathbf{D}(\boldsymbol{q})},$$

$$\begin{array}{l}\beta = \dfrac{\det \mathbf{\Phi}_2}{\det \mathbf{D}(\boldsymbol{q})}(-\mathbf{C}_2(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} - \mathbf{G}_2(\boldsymbol{q})) - \\ \dfrac{(\theta_2\theta_4 g \cos(q_1) - \theta_3\theta_5 g \cos(q_2)\cos(q_1+q_2))}{\det \mathbf{D}(\boldsymbol{q})} \times \\ \times(\mathbf{C}_1(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \mathbf{G}_1(\boldsymbol{q})) - \theta_4 g \dot{q}_1{}^2 \sin(q_1) - \\ +\theta_5 g(\dot{q}_1+\dot{q}_2)^2 \sin(q_1+q_2),\end{array}$$

where

$$\mathbf{\Phi}_2 = \begin{bmatrix} \theta_1+\theta_2+2\theta_3\cos(q_2) & \theta_2+\theta_3\cos(q_2) \\ \begin{array}{c}\theta_4 g\cos(q_1) \\ +\theta_5 g\cos(q_1+q_2)\end{array} & \theta_5 g\cos(q_1+q_2) \end{bmatrix}.$$

Moreover the co-ordinate transformation is locally invertible at each point where

$$\frac{\det \mathbf{\Phi}_2}{d_{11}(\boldsymbol{q})} \neq 0. \tag{19}$$

4.2 Adaptation of the reference and the feed-forward control

The role of the feed-forward controller is particularly important when the tracking of a specified state space trajectory is desired. Its main purpose is to transfer the system along the desired trajectory usually from one equilibrium to another and if there would be no disturbances acting on the controlled system and the system would be perfectly known the feed-forward control would suffice to track any feasible trajectory of the system. Main advantage of the feed-forward control structure is that this controller can be designed to meet such performance standards so that the feedback controller can be tunned more aggressively.

Parameters of some feed-forward controller with well chosen structure together with a suitable reference trajectory can be obtained by a solution of a boundary value problem for the Acrobot's EoM with following boundary conditions

$$\left.\begin{array}{l} q_1(0) = q_{10} \\ q_1(T) = -q_1(0) \\ q_2(0) = \pi + 2|q_1(0)| \\ q_2(T) = \pi - 2|q_1(T)| \\ \dot{q}_1(0) = \dot{q}_1(T^+) \\ \dot{q}_2(0) = \dot{q}_2(T^+). \end{array}\right\} \tag{20}$$

The first four conditions from (20) secure that the configuration of the Acrobot at the beginning and at the end of the step (at time T) will be symmetric with respect to the y-axis, see Fig. 1.

Time T is the time when the swing leg will hit the ground we will call this time the time of the impact. The times T^- and T^+ are the times just before and just after the impact respectively. It is supposed that the impact will not change the angular configuration of the Acrobot, but will affect only its angular velocities $\dot{\boldsymbol{q}}$. The impact can be modeled as follows

$$\begin{bmatrix}\dot{q}_1(T)^+ \\ \dot{q}_2(T)^+\end{bmatrix} = \Phi_{imp}(\boldsymbol{q}(T)) \begin{bmatrix}\dot{q}_1(T)^- \\ \dot{q}_2(T)^-\end{bmatrix} \tag{21}$$

where Φ_{imp} represents the matrix of impact. The details regarding the computation of the impact matrix can be found in [Grizzle et. al., 2001] or more general treatment can be found in [Brogliato, 1996].

The solution of (20) is not unique and can be obtained by numerical techniques for boundary value problems [Shampine et. al., 2003]. Another way of computing the boundary value problem is to use algorithm developed in [Anderle and Čelikovský, 2010b]. In both cases solution provides both the reference trajectories and the feed-forward control. Using the transformation (17) a following reference system is obtained

$$\left.\begin{array}{l}\dot{\xi}_1^{ref} = d_{11}^{-1}\xi_2^{ref},\ \dot{\xi}_2^{ref} = \xi_3^{ref},\ \dot{\xi}_3^{ref} = \xi_4^{ref},\\ \dot{\xi}_4^{ref} = z^{ref}\end{array}\right\} \quad (22)$$

with known initial conditions $\boldsymbol{\xi}^{ref}(0) = \boldsymbol{\xi}_0^{ref}$ and known feed-forward control z^{ref}. In the case when strategy developed in [Anderle et. al., 2009] is chosen the feed-forward control law will be computed as $z^{ref} = a + bt$ with known constants $a,\ b$.

The particular reference trajectory and feed-forward control are however dependent on the parameters of the robot. These are in turn dependent on the load that the robot has to carry. However if this mass is unknown, then both trajectories and the control are unknown as well. Nevertheless thanks to recursive estimation some estimate of the mass of the load is always available. Also off-line computation of a set of reference trajectories for some well chosen range of the values that the mass of the load can attain is possible. Thus a fast selection of a reference trajectory is possible provided some estimate is available. Example of the set of trajectories that were computed for $m_{load} \in [0,\ 0.5]\ [kg]$ is demonstrated on Fig. 2 and Fig. 3. Moreover parameters a, b can be accurately approximated by polynomials

$$\begin{array}{l} a(m_{load}) = a_0 + a_1 m_{load} + \ldots + a_{n_a} m_{load}^{n_a} \\ b(m_{load}) = b_0 + b_1 m_{load} + \ldots + b_{n_b} m_{load}^{n_b} \end{array} \quad (23)$$

where coefficients of polynomials (23) can be estimated off-line using the least-squares method.

4.3 Feedback control design

Denoting the regulation error as $\boldsymbol{e} = \boldsymbol{\xi} - \boldsymbol{\xi}^{ref}$ and subtracting (22) from (18) one obtains

$$\left.\begin{array}{l} \dot{e}_1 = d_{11}^{-1}(\phi_2(\xi_1,\xi_3))\xi_2 - \\ -d_{11}^{-1}(\phi_2(\xi_1^{ref},\xi_3^{ref}))\xi_2^{ref}, \\ \dot{e}_2 = e_3,\ \dot{e}_3 = e_4,\ \dot{e}_4 = z - z^{ref}, \end{array}\right\} \quad (24)$$

where $\phi_2(\xi_1,\xi_3) = q_2$. Using Taylor expansion we can rewrite (24) as follows

$$\left.\begin{array}{l} \dot{e}_1 = \mu_1(t)e_1 + \mu_2(t)e_2 + \mu_3(t)e_3 + o(e) \\ \dot{e}_2 = e_3,\ \dot{e}_3 = e_4,\ \dot{e}_4 = z - z^{ref}, \end{array}\right\} \quad (25)$$

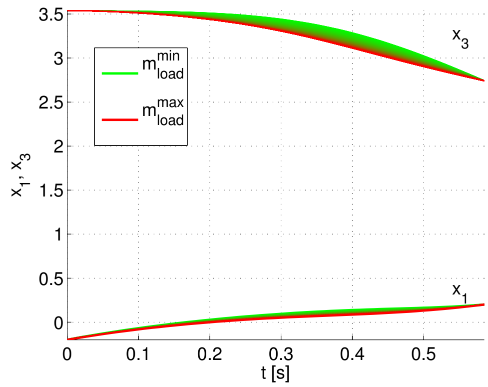

Figure 2. The Acrobot's reference configuration angles.

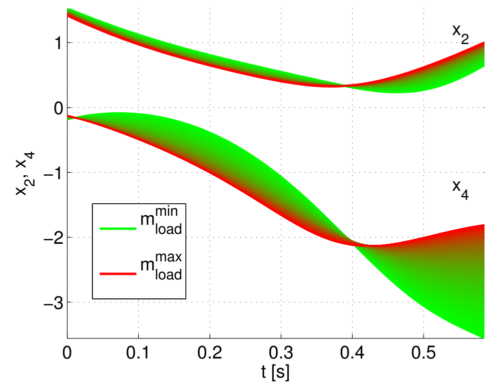

Figure 3. The Acrobot's reference angular velocities.

where $\mu_1(t),\ \mu_2(t),\ \mu_3(t)$ are known smooth time functions [Anderle and Čelikovský, 2009]. Using *Theorem 3.1* from [Anderle and Čelikovský, 2009] the one can stabilize (24) and solve the tracking problem by using following state feedback

$$\begin{array}{l} z = z^{ref} + \Theta^3\tilde{K}_1 e_1 + \Theta^3(\tilde{K}_2 - \\ \quad \tilde{K}_1\mu_3(t))e_2 + \Theta^2\tilde{K}_3 e_3 + \Theta\tilde{K}_4 e_4, \end{array} \quad (26)$$

where constants $\tilde{K}_1,\ \tilde{K}_2,\ \tilde{K}_3,\ \tilde{K}_4, \Theta$ are chosen based on conditions in [Anderle and Čelikovský, 2009].

We can therefore design a feedback control law for the worst case scenario where the load that the robot has to carry attains it's maximal allowed value. Then we can use such feedback to stabilize the system for a set of values of the parameter m_{load}.

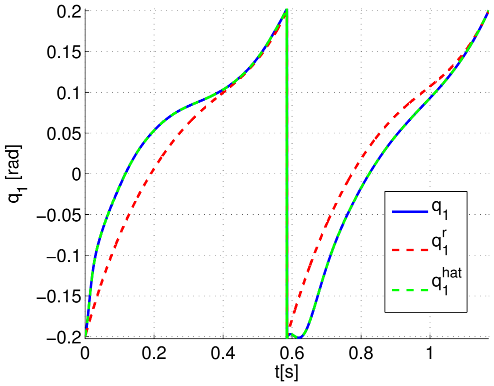

Figure 4. Tracking of the adapting reference trajectory and estimation of the angle q_1.

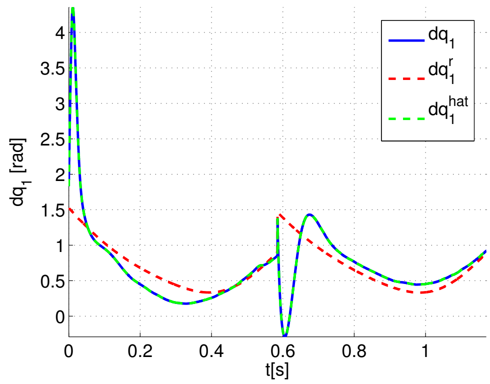

Figure 6. Tracking of the adapting reference trajectory and estimation of the angular velocity $\dot{q}_1$.

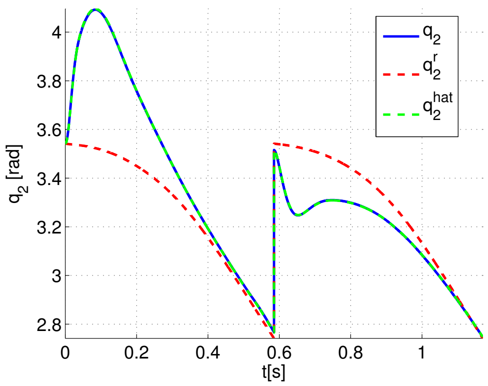

Figure 5. Tracking of the adapting reference trajectory and estimation of the angle q_2.

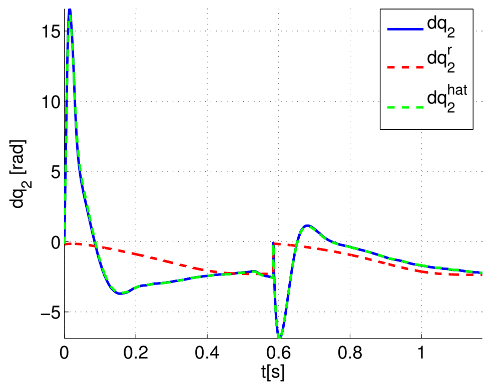

Figure 7. Tracking of the adapting reference trajectory and estimation of the angular velocity $\dot{q}_2$.

5 Simulation results

We have carried out simulations with following parameters of the system

$$\left.\begin{array}{l} m_1 = m_2 = 1[kg] \\ l_1 = l_2 = 1[m] \\ l_{c_1} = l_{c_2} = 0.5455[kg] \\ I_1 = I_2 = 0.0606[kg.m^2] \\ g = 9.81[ms^{-2}] \\ m_{load} = 0.2[kg] \\ \mathbf{R} = 10^{-6}\mathbf{I}, \\ \mathbf{Q} = 10^{-10}\mathbf{I}, \\ Q* = 10^{-2} \end{array}\right\} \quad (27)$$

Tracking of the reference configuration angles is depictured on figures Fig. 4 and Fig. 5. Tracking of the reference angular velocities is demonstrated on figures Fig. 6 and Fig. 7. Estimation of the mass of the load is demonstrated on Fig. 8. After the first two steps wich occur during less than 1.2 $[s]$ the unknown load is estimated with an error less than 8%. The estimate is continually improving in increasing time as well as the tracking performance. Animation of the walking is demonstrated on the Fig. 9.

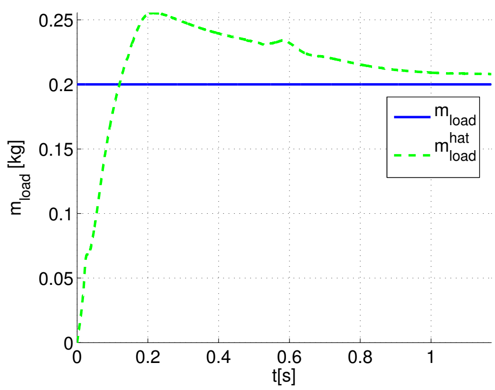

Figure 8. Estimation of the mass of the carried load.

6 Conclusion

An adaptation scheme for nonlinear tracking of a walking like motion for bipedal robot was presented. The task of transportation of an object of unknown but bounded mass was considered. Due to abundant a priori information about the system dynamics and also due to the fact that not all states of the system are measured estimation of the unknown mass of the load was realized using the EKBF. The complicated multi variable estimation problem was reduced to the much simpler

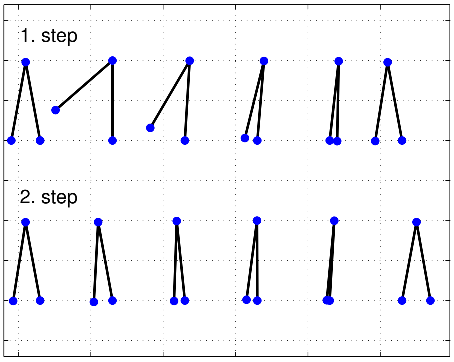

Figure 9. Animation of the walking of Acrobot with sampling period 0.11 [s].

problem consisting of determination of one parameter of the model only. It was also shown that adaptation of the reference trajectory and the feed-forward control law is can be done in a manner that doesn't require much computational power thanks to polynomial approximation of the feed-forward control rule and off-line computation of the reference trajectory. Therefore such adaptation scheme can be applied to real-time control.

Exact linearization was used for the feedback control rule design as it yields a controller that can be designed to stabilize the system for some range of values of the mass of the load that the robot has to carry. The performance of proposed control strategy yields high performance as it benefits from precise knowledge of the states and parameters of the system. A precise estimation of the unknown state of the system is more complicated when the parameters of the system are not known precisely. Therefore utilization of the recursive identification increases both the quality of the state estimate and the performance of the controller.

The responsiveness of the filter or the speed of convergence could be further improved by some suitable regularization of the covariance matrix of the estimation error. Also different filtering schemes can be applied, e.g. Unscented filtering or Particle filtering [Julier, 1995], [Gordon et. al., 1993].

The estimation of the time of impact is another very important problem. Although the bipedal robot was modeled as the Acrobot in the reality the robot can also perform a movement in knees. However this movement is only used to shorten or stretch the legs so that the feet would not hit the ground prematurely. The exact timing of the impact and the stretching of the leg is vital for the tracking performance.

Acknowledgements

This work has been supported by a CTU internal grant SGS 10-802830, Optimization methods for modeling and control and their applications. Both authors were supported by the Czech Science Foundation through the research grant no. P103/12/1794.

References

Anderle, M., Čelikovský, S. (2010b) Sustainable Acrobot walking based on the swing phase exponentially stable tracking. To appear in *The 3rd Annual Dynamic Systems and Control conference*, Boston, USA.

Anderle, M., Čelikovský, S., Henrion, D., Zikmund J. (2010) Advanced LMI based analysis and design for Acrobot walking. In *Internation Journal of Control*, 83:8, pp. 1641–1652.

Anderle, M., and Čelikovský, S. (2009) Analytical Design of the Acrobot Exponential Tracking with Application to its Walking. In *IEEE International Conference on Control Automation (ICCA)*, Christchurch, New Zealand, pp. 163–168.

Anderle, M., Čelikovský, S., Henrion, D., Zikmund J. (2009) LMI Based Design for the Acrobot Walking. In *IFAC Symposium on Robot Control (SYROCO)*, Gifu, Japan, pp. 595–600.

Čelikovský, S., Zikmund, J., Moog, C.H. (2008) Partial exact linearization design for Acrobot walking. *Preprints of the 2008 Americal Control Conference*, Seatle, USA.

Bellantoni, J., Dodge, K. (1967) A square root formulation of the Kalman Schmidt filter. In *AIAA Journal*, **5**, pp. 1309–1314.

Brogliato, B. (1996) *Nonsmooth Mechanics: Models, Dynamics and Control.* Springer-Verlag, London, U. K.

Dennis Jr.J.E., Schnabel, R.B. (1983) *Numerical methods for unconstrained optimization and nonlinear equations*. Prentice-Hall, Upper Saddles River, N.J.

Dennis Jr.J.E., Gay, D.M. (1981) An adaptive nonlinear least-squares algorithm. In *ACM T. Math Software*, **7**, pp. 348–368, 369–383.

Fitzgerald, R. (1971) Divergence of the Kalman filter. In *IEEE Transactions on Automatic Control*, AC-16, pp. 736-747.

Gordon, N.J., Salmond, D.J., and Smith, A.F.M. (1993) Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings-F*, 140(2), pp. 107–113.

Grizzle, J.W., Moog, C.H., and Chevallereau C. (2005) Nonlinear control of mechanical systems with an underactuated cyclic variable. In *IEEE Transactions on Automatic Control*, **50**, pp. 559–576.

Grizzle, J.W., Abba, G., and Plestan, F. (2001) Asymptotically stable walking for biped robots: Analysis via systems with impulse effects. *IEEE Transactions on Automatic Control*, **46**, pp. 51–64.

Isidori, A. (1996) *Nonlinear control systems*. Springer-Verlag. New York.

Julier, S.J., Uhlmann, J.K., and Durrant-Whyte, H.F. (1995) A new approach to filtering nonlinear systems. In *The Proceedings of the American Control Confer-*

ence, Seattle, Washington, pp. 1628–1632.

Kalman, R.E. (1960) A new approach to linear filtering and prediction problems. In *Transactions of the ASME, Journal of basic engineering* , pp. 35–45.

Landau, L.D., and Lifshitz, E.M. (1976) *Mechanics, Third Edition: Volume 1 of Course of Theoretical physics*. Elsevier Science.

Shampine, L.F., Gladwell, I., and Thompson, S. (2003) *Solving ODEs with MATLAB*, Cambridge University Press.

Simon, D. (2006) *Optimal State Estimation*. Wiley-Intersciences. New Jersey.

Spong, M. (1998) *Underactuated Mechanical Systems, Control Problems in Robotics and Automation*. Springer-Verlag. London.