Improvements of the Adaptive Slotine & Li Controller – Comparative Analysis with Solutions Using Local Robust Fixed Point Transformations

JÓZSEF K. TAR, IMRE J. RUDAS
Budapest Tech
John von Neumann Faculty of Informatics
Institute of Intelligent Engineering Systems
Bécsi út 96/B, H–1034 Budapest
HUNGARY
tar.jozsef@nik.bmf.hu, rudas@bmf.hu

JÓZSEF GÁTI
Budapest Tech
"Bánki Donát" Faculty of Mechanical & Safety Engineering
Institute of Materials Sciences & Manufacturing Technologies
Népszínház utca 8, H–1081 Budapest
HUNGARY
gati@bmf.hu

Abstract: One of the most sophisticated classical robot controller, the *"Slotine-Li Adaptive Controller"* is constructed on the basis of exact knowledge on the form of the equations of motion of the system under control, and on the application of *"Lyapunov's* 2^{nd} *Method"*. This generic technique makes it possible to guarantee the stability of the controlled system using only simple estimations without having any detailed knowledge on its motion that is a great advantage. However, in the application of this elegant technique the main problem is the proper construction of the Lyapunov function. Formal elegance usually has the price of limitations in the applicable trajectory tracking policy and in the presence of a huge number of the constant control parameters to be determined in the commencement of the controller's operation. In the here presented approach the controller invented by Slotine and Li is modified in two steps: the first step modifies the original Lyapunov function, the second one –as an addition– modifies the parameter tuning by utilizing the information encoded in the equations of motion for which no any Lyapunov function is needed. The advantage is the limitation in the number of the fixed control parameters, and possibility for faster parameter tuning. The modified controller's operation is compared with that of a simple adaptive technique using locally convergent Cauchy sequences in iterative learning via simulation for a simple Classical Mechanical System as a paradigm.

Key–Words: Adaptive Control, Slotine–Li Adaptive Controller, Parameter Tuning, Iterative Learning, Cauchy Sequences

1 Introduction

In this paper possible improvements of the most sophisticated classical controller, the *"Slotine-Li Adaptive Controller"* [1] based on the use of *formally exact, permanent, and complete* model with *parameter uncertainties* is considered in comparison with a novel adaptive approach applying only *partial*, *incomplete*, *temporal*, and *situation-dependent* model that requires continuous refreshment via observing the behavior of the controlled system in the actual situation. It will be shown how simple geometric considerations can be used for developing iterative learning control. It will also be shown that in the case of the novel approach the conditions of convergence normally can be satisfied by choosing very primitive initial system models and roughly chosen control parameters on the basis of simple geometric considerations.

It will be pointed out that while even the improved Slotine–Li controller is applicable only for learning linearly separable dynamical system parameters and normally it fails for the permanent presence of unknown external disturbances and linearly non-separable (e.g. friction) parameters, the novel approach can well compensate the simultaneous effects of these factors. In contrast to the traditional solutions that normally guarantee global (asymptotic) stability by using Lyapunov functions, the novel approach can assure only local region of stability that in principle can be left by the system's state. To cope with this problem a *real time simulation tool* has been developed to study the robustness of the various controllers against the variation of the adaptive control parameters. The already achieved results will be exemplified by simulation studies using the same paradigm.

At first the original version and the suggested modification of the Slotine–Li controller will be detailed. Following that the idea of the suggested novel method will briefly be outlined. Finally comparative simulation results will be provided for a particular nonlinear system as a paradigm. The paper will be closed by concluding remarks and list of references.

2 The Slotine–Li Controller and Its Suggested Modifications

This control approach utilizes subtle details of the Euler–Lagrange equations of motion that are not observed/utilized in the *Adaptive Inverse Dynamics* approach [1], namely that the terms quadratic in the generalized velocity components are not independent of the inertia matrix: they can be deduced from the inertia matrix, and according to their special position in the equations of motion they can be symmetrized. In this approach the exerted generalized torque/force components are constructed by the use of the actual model as follows:

$$\begin{array}{c} Q = \hat{H}(q)\dot{v} + \hat{C}(q,\dot{q})v + \hat{g} + K_D r \\ e := q^N - q, v := \dot{q}^N + \Lambda e, \\ r := \dot{e} + \Lambda e, \tilde{p} := \hat{p} - p \\ C_{ij} = \frac{1}{2}\sum_z \dot{q}_z \left(-\frac{\partial \hat{H}_{zj}}{\partial q_i} + \frac{\partial \hat{H}_{ij}}{\partial q_z} + \frac{\partial \hat{H}_{iz}}{\partial q_j} \right) \\ Q = Y(q,\dot{q},v,\dot{v})\hat{p} + K_D r \end{array} \quad (1)$$

in which q^N and q denote the generalized co–ordinates of the *nominal* and the *actual* motion, K_D and Λ are symmetric positive definite matrices, matrices $\hat{H}$, $\hat{C}$, and $\hat{g}$ are the actual models of the system's inertia matrix, the Coriolis, and the gravitational terms. The possession of the exact form of the dynamical model makes it possible to linearly separate the system's dynamic parameters p in the expression of the physically interpreted *generalized forces* Q by the use of matrix Y that exclusively consists of known kinematical data. The Lyapunov function of this method is $V = r^T H r + \tilde{p}^T \Gamma \tilde{p}$, with positive definite symmetric matrix Γ. For guaranteeing negative derivative of the Lyapunov function the *skew symmetry* of the C_{ij} matrix and the parameter tuning rule $\dot{\hat{p}} = \Gamma^{-1} Y^T r$ are utilized. Since in this approach no matrix inversion happens, the speed of parameter tuning can be far higher that that of the Adaptive Inverse Dynamics Control [1]. However, this method cannot properly compensate the effects of unknown external disturbances and friction forces. As a kind of deficiency of the method is the fact that it does not contain integrated term in its feedback that usually is very efficient remedy against small, slowly varying but almost permanent tracking error. In the next subsection a modification will be proposed by the use of which integrated feedback term can be added to this control.

2.1 Modified Slotine–Li Controller with Integrated Error-feedback

For making small, slowly varying tracking errors decay the appropriate quantity to be monitored used to be the integral of the error $z := \int_0^t e(\tau) d\tau$. By the use of a symmetric positive definite matrix Λ first order integro–differential equation can be obtained for the error as follows

$$\breve{r} := \left(\frac{d}{dt} + \Lambda \right)^2 z = \ddot{z} + 2\Lambda \dot{z} + \Lambda^2 z = 0 \quad (2)$$

that could guarantee the exponential decay of the integral of the error z, and according to *Barbalat's Lemma*, *for uniformly continuous function* $e(t)$ this means that $e(t) \to 0$ as $t \to \infty$. Equation (2) suggests to modify (1) as follows:

$$\begin{array}{c} Q = \hat{H}(q)\dot{\breve{v}} + \hat{C}(q,\dot{q})\breve{v} + \hat{g} + K_D \breve{r} \\ \breve{v} := \dot{q}^N + 2\Lambda \dot{z} + \Lambda^2 z \\ Q = \breve{Y}(q,\dot{q},\breve{v},\dot{\breve{v}})\hat{p} + K_D \breve{r} \end{array} \quad (3)$$

in which $\breve{Y}$ also is completely known in each instant. As in the case of the *original Slotine–Li Controller*, we can assume, that the so exerted Q is the only agent that influences the state propagation of the system so we can state that

$$\hat{H}(q)\dot{\breve{v}} + \hat{C}(q,\dot{q})\breve{v} + \hat{g} + K_D \breve{r} = H\ddot{q} + C\dot{q} + g \quad (4)$$

in which H, C and g are the "exact" terms. Via subtracting $H\dot{\breve{v}}$, $C\breve{v}$, $K_D\breve{r}$, and g from both sides we obtain that

$$\begin{array}{c} (\hat{H} - H)\dot{\breve{v}} + (\hat{C} - C)\breve{v} + \hat{g} - g = \\ = -K_D \breve{r} - H\dot{\breve{r}} - C\breve{r} \\ \breve{Y}(\hat{p} - p) = -K_D \breve{r} - H\dot{\breve{r}} - C\breve{r}. \end{array} \quad (5)$$

Let the new Lyapunov function be $\breve{V} := \breve{r}^T H \breve{r} + \tilde{p}^T \Gamma \tilde{p}$. Its time–derivative evidently is (due to symmetry reasons) $\dot{\breve{V}} = 2\breve{r}^T H \dot{\breve{r}} + 2\tilde{p}^T \Gamma \dot{\tilde{p}} + \breve{r}^T \dot{H} \breve{r}$. From (5) it is possible to substitute here $H\dot{\breve{r}}$ and the terms quadratic in $\breve{r}$ can be selected, and negative derivative has to be prescribed as:

ISSN: 1790-2769

ISBN: 978-960-474-138-0

$$\dot{\breve{V}} = \breve{r}^T \left[\dot{H} - 2C\right] \breve{r} - 2\breve{r}^T K_D \breve{r} + \\ +2\tilde{p} \left[\Gamma \dot{\tilde{p}} - \breve{Y}^T \breve{r}\right] < 0. \quad (6)$$

As in the original case, due to symmetry reasons $\breve{r}^T \left[\dot{H} - 2C\right] \breve{r} = 0$, the 2^{nd} in the LHS is negative, and for the modified parameter tuning we obtain the rule $\dot{\hat{p}} = \Gamma^{-1} \breve{Y}^T \breve{r}$. Since $\breve{r}$ contains the integral of the tracking error it is expected that the present modification yields better trajectory tracking even in its "learning phase" than the original version. For this purpose simulation investigations will be made in the appropriate section of the paper. In the sequel it will be investigated if it is really significant to apply any Lyapunov function for tuning purposes.

2.2 Dropping the Lyapunov Function: Other Possibility for Tuning Parameters

It can be observed that the above manipulations lead to the direction of using a Lyapunov function. On the other hand we can obtain information on the actual modeling errors by subtracting $\hat{H}\ddot{q}$, $\hat{C}\dot{q}$ instead of $H\dot{\breve{v}}$, $C\breve{v}$, and subtracting $\hat{g}$ from both sides. In this manner $\breve{r}$, and $\dot{\breve{r}}$ will appear in the equations:

$$\hat{H}\dot{\breve{r}} + \hat{C}\breve{r} + K_D \breve{r} = \\ = (H - \hat{H})\ddot{q} + (C - \hat{C})\dot{q} + g - \hat{g} = \\ = \Upsilon(q, \dot{q}, \ddot{q})(p - \hat{p}) \quad (7)$$

where $\Upsilon(q, \dot{q}, \ddot{q})$ is also well known, furthermore the LHS of (7) also is known. ***As a consequence it can be stated that while the exerted generalized forces remain the same as in the case of the modified Lyapunov function in (3), we have satisfactory information for parameter tuning without applying any Lyapunov function.*** Via applying the *Singular Value Decomposition (SVD)* on Υ as $\Upsilon = UDV^T$ in which U and V are orthogonal matrices of appropriate sizes and D is ***diagonal*** with decreasing positive singular values in the main diagonal, with a positive $\gamma > 0$ value fast parameter–tuning can be prescribed as $\dot{\hat{p}} = -\gamma V D^+ U^T \left[\hat{H}\dot{\breve{r}} + \hat{C}\breve{r} + K_D \breve{r}\right]$ in which $D^+_{ij} = \frac{1}{D_{ji}}$ $if\ D_{ji} > \varepsilon$, otherwise $D^+_{ij} = 0$. This corresponds to the application of a kind of pseudo–inverse of Υ. From our application point of view SVD is excellent mathematical method by the use of which the *ingress* and *egress spaces* of a real linear map can be decomposed to *pairwisely orthogonal unit vectors* that are related to each other by nonzero singular values [2], [3]. The bigger the appropriate singular value is the more significant the appropriate directions in the input and the output spaces are in the given map. Zero singular value means that the appropriate directions do not take part in the map at all. This simple geometric interpretability makes SVD an attractive tool in various applications. Due to the associative nature of the matrix product the elements of the column $U^T \left[\hat{H}\dot{\breve{r}} + \hat{C}\breve{r} + K_D \breve{r}\right]$ can be considered at first. They correspond to the components of available information with respect to the orthonormal system of coordinates determined by the columns of U. In the approximation only those columns (unit vectors) are taken into account that have limited but considerable contribution to parameter tuning. The columns of the matrix VD^+ are pairwisely orthogonal and they are "big" if in the appropriate direction too much variation of the parameters would be needed for achieving observable effect in the available information. For keeping the speed of tuning at bay the dangerous or almost singular contributions are excluded from the tuning process.

3 The Excitation – Response Scheme and Fixed Point Transformations

Each control task can be formulated by using the concepts of the appropriate *"excitation"* Q of the controlled system to which it is expected to respond by some prescribed or *"desired response"* r^d. The appropriate excitation can be computed by the use of some *inverse dynamic model* $Q = \varphi(r^d)$. Since normally this inverse model is neither complete nor exact, the actual response determined by the system's dynamics, ψ, results in a *realized response* r^r that differs from the desired one: $r^r \equiv \psi(\varphi(r^d)) \equiv f(r^d) \neq r^d$. It is worth noting that the functions $\varphi()$ and $\psi()$ may contain various hidden parameters that partly correspond to the dynamic model of the system, and partly pertain to unknown external dynamic forces acting on it. Due to phenomenological reasons the controller can manipulate or "deform" the input value from r^d so that $r^r \equiv \psi(r^d_*)$. The main idea is that via the introduction of an iterative process as $r_{n+1} = \Psi(r_n; r^d)$ the solution of the problem can be found as $r_n \to r_*$. If the iteration is convergent and this convergence is fast enough the solution can practically well approximated. In the sequel it will be shown that for *SISO* systems the appropriate deformation can be defined as some *Parametric Fixed Point Transformation*.

3.1 Iteration Using Robust Fixed Point Transformations

Consider the iteration generated by some function as $x_{n+1} = G(x_n; x^d)$. In order to apply iterations let us consider the set of the real numbers $\Re$ as a linear normed space with the common addition and multiplication with real numbers, and with the absolute value $|\bullet|$ as a norm. It is well known that this space is *complete*, i.e. it is a *Banach Space* in which the *Cauchy Sequences* are convergent. Due to that, using the norm–inequality, for a convergent iterative sequence $x_n \to x_*$ it is obtained that

$$|G(x_*) - x_*| \leq |G(x_*) - x_n| + |x_n - x_*| = \\ = |G(x_*) - G(x_{n-1})| + |x_n - x_*|. \tag{8}$$

It is evident from (8) that if G is continuous then the desired fixed point is found by this iteration because in the right hand side of (8) both terms converge to 0 as $x_n \to x_*$. The next question is giving the necessary or at least a *satisfactory condition of this convergence*. It also is evident that for this purpose contractivity of $G(\bullet)$, i.e. the property that $|G(a) - G(b)| \leq K|a-b|$ with $0 \leq K < 1$ is satisfactory since it leads to a *Cauchy Sequence* ($|x_{n+L} - x_n| \to 0 \; \forall L \in \mathsf{N}$):

$$|x_{n+L} - x_n| = |G(x_{n+L-1}) - G(x_{n-1})| \leq ... \\ \leq K^n |x_L - x_0| \to 0 \;\; as \quad n \to \infty \tag{9}$$

For the role of function $G(x;x^d)$ a novel fixed point transformation was introduced in [4] that is rather "robust" as far as the dependence of the resulting function on the behavior of $f(\bullet)$ is concerned (10). This robustness can approximately be investigated by the use of an affine approximation of $f(x)$ in the vicinity of $x_\star$ and it is the consequence of the strong nonlinear saturation of the sigmoid function $\tanh(x)$:

$$\begin{array}{c} G(x|x^d) := (x+K)\times \\ \left[1 + B\tanh(A[f(x) - x^d])\right] - K \\ \texttt{if} f(x_\star) = x^d \texttt{ then} \\ G(x_\star|x^d) = x_\star \\ G(-K|x^d) = -K, \\ G(x_\star|x^d)' = (x_\star + K)ABf'(x_\star) + 1. \end{array} \tag{10}$$

It is evident that the transformation defined in (10) has a proper ($x_\star$) and a false ($-K$) fixed point, but by properly manipulating the control parameters A, B, and K the good fixed point can be located within its basin of attraction, and the requirement of $|G'(x_\star|x^d)| < 1$ can be guaranteed. This means that the iteration can have considerable speed of convergence even nearby $x_\star$, and the strongly saturated $\tanh$ function can make it more robust in its vicinity, that is the properties of $f(x)$ have less influence on the behavior of G. It is not difficult to show that in the case of *Single Input – Single Output (SISO)* systems the $G(x|x^d)$ functions can realize contractive mapping around $x_\star$. Qualitatively it can be stated that a small value of the parameter A opens a wide "window" in the vicinity of the realized response, while parameter K can yield an additional shift to speed up the tuning. Practically these parameters can be set via simulations: by the use of a simple PID-type controller one can observe the order of magnitude of the desired and simulated responses, and A and K can be set accordingly.

A simple possibility for applying the same idea of adaptivity outlined in (10) for *Multiple Input – Multiple Output Systems* is the application of a sigmoid function projected to the direction of the response-error defined in the n^{th} control cycle as $\vec{h} := \vec{f}(\vec{x}_n) - \vec{x}^d$, $\vec{e} := \vec{h}/||\vec{h}||$, $\tilde{B} = \sigma(A||\vec{h}||)$, so that

$$\vec{x}_{n+1} = (1 + \tilde{B})\vec{x}_n + \tilde{B}K\vec{e}. \tag{11}$$

(If $||\vec{h}||$ is very small, instead of normalizing with it the approximation $\vec{x}_{n+1} = \vec{x}_n$ can be applied since then the system already is in the very close vicinity of the fixed point.) It can also be noted that instead of the $\tanh$ function any sigmoidal function with the property of $\sigma(0) = 0$, e.g. $\sigma(x) := x/(1 + |x|)$ can be similarly applied, too.

It has be noted that within each control cycle only one step can be executed in the iteration. If the adaptation is faster than the dynamics of the system to be controlled appropriate result can be expected even in this case, too. This approach is similar to the application of *"Cellular Neural Networks (CNN)"* for image processing. In relation to the operation of CNNs the concept of *"Complete Stability"* can be introduced that means that a static input picture is mapped to a static output picture following a short dynamic transition of the physical state of the CNN. If the input picture is not static but varies "slowly" in comparison with the "speed" of the CNN's internal dynamics varying picture is mapped to varying output [5]. In spite of using a single step during one control cycle from each point of view the improvement may be considerable. In the next subsection simulation examples are give for the use of the realization defined in (11). In the sequel simulation examples will be presented.

4 Simulation Examples

In this part at first the dynamic model of the controlled system as a paradigm is considered then a comparative analysis of simulation results will be given.

4.1 The Dynamic Model of the Cart–Beam–Hamper System

It is assumed that a hamper can be rotated around its mass center point with angle q_2 $[rad]$ at the end of a beam of length $L = 2$ $[m]$ and of negligible mass. The beam can be rotated around a horizontal axis with angle q_1 $[rad]$, and the whole cart can move along the axis q_3 $[m]$. The mass of the cart is $M = 30$ $[kg]$, the hamper's mass is $m = 10$ $[kg]$, its momentum to its rotational axis is $\Theta = 20$ $[kg \times m^2]$. Its Euler–Lagrange Equations of motion are given in (12). The gravitational acceleration is assumed as $g = 10$ $[m/s^2]$. The appropriate *"approximate model parameters"* are $\hat{L} = 2$ $[m]$, $\hat{M} = 60$ $[kg]$, $\hat{m} = 20$ $[kg]$, $\hat{\Theta} = 20$ $[kgm^2]$, and $\hat{g} = 8$ $[m/s^2]$.

$$\begin{bmatrix} (ML^2+\Theta) & \Theta & mLcosq_1 \\ \Theta & \Theta & 0 \\ mLcosq_1 & 0 & (m+M) \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix} + + \begin{bmatrix} -mgLsinq_1 \\ 0 \\ -mLsinq_1\dot{q}_1^2 \end{bmatrix} = \begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \end{bmatrix} \tag{12}$$

The linearly separable parameters to be tuned are as follows: $p = [mL, mL^2+\Theta, \Theta, m+M, mLg]^T$. Furthermore it is assumed that along the axis q_3 nonlinear dynamic friction acts with the following LuGre-type model in which the deformation of the *"bristles"* of some *"brushes"* are applied to describe the "internal deformation" of the surfaces in dynamic contact (a new degree of freedom denoted by z), so friction is described as a dynamic coupling between two systems having their own equations of motion. For z it is given in (13) as

$$\begin{array}{c} \frac{dz}{dt} = v - \frac{\sigma_0|v|}{F_C+F_S\exp(-|v|/v_s)}z \\ F = \sigma_0 z + \sigma_1\frac{dz}{dt} + F_v \times v \end{array} \tag{13}$$

for which the proper direction of F has to be set in the applications, σ_0 describes some "spring constant", σ_1 is a new parameter pertaining to the effect of the bending bristles, and F_v describes the viscous friction coefficient, v is the relative velocity of the surfaces in contact. (In the Slotine–Li picture the parameters of this strongly nonlinear model cannot be separated into the *"parameter vector"*, and friction appears as unknown external disturbance.) The role of the F_C, F_S, and v_s parameters is to describe the reduction of the friction coefficient with increasing velocity in the low velocity domain. This model is physically complete in the sense that no any velocity limit of dubious interpretation must be introduced for its use to declare the velocity to be "zero" in order to describe the phenomenon of "sticking". The behavior of the whole system is described by the dynamic coupling between the hidden internal and the observed degrees of freedom. The appropriate quantities in (13) were as follows: $\sigma_0 = 100$ $[N/m]$, $\sigma_1 = 1500$ $[N \times s/m]$, $F_v = 1$ $[N/m]$, $F_C = 1000$, $F_S = 2000$ $[N]$, and $v_s = 0.1$ $[m/s]$. In spite of certain successful efforts as reported e.g. in [6] the identification of the parameters of the friction models is quite difficult in general. Therefore it was assumed that the friction is not known/not modeled by the controllers. To describe unknown external perturbations 3^{rd} order periodic spline functions were fitted to randomly chosen amplitude values to produce general smooth noise of nonzero mean and wide frequency spectrum as additional force to that driving axis q_3, i.e. an addition to the generalized force component Q_3. ***We note that in each simulation the finite element time–resolution was 1*** ms***.***

4.2 Computational Results

Using the $\Lambda = 10$ s^{-1} $K_D = 100$ the *"ad hoc"* $\Gamma = 0.01 < 1, 1, 1 >$ matrix based tuning, $\gamma = 0.01$ for the modified tuning it can be seen that the modifications of the Slotine-Li method were step–by–step improved in accuracy (Fig. 1). The SVD–based tuning resulted in monotone variation of the tuned parameters while resisted to any speed increasing attempt as the fluctuations in the control parameters reveal it. However, the superior solution seems to be the fixed point transformations based method with $K_{ctrl} = -32000$, $B_{ctrl} = 1$, and $A_{ctrl} = 2 \times 10^{-6}$.

To investigate the effect of friction it has been switched on (Fig. 2). It has to be noted that the "modified Lyapunov function" was calculated with reduced tuning speed ($\Gamma = 0.1 < 1, 1, 1 >$). As in the previous case, the best solution between the approaches is that using modified tuning rule. However, the fixed point transformations based solution is the superior one, it almost completely compensates the effect of friction. In general, the methods based on parameter tuning are disturbed/fobbed/mislead by effects of the linearly non–separable friction terms.

To investigate the simultaneous effects of modeling errors, not modeled friction effect and unknown external disturbance drastic nonlinear external force was added to the body of the cart (Fig. 3). It was created by 3rd order spline functions fitted to randomly selected points within a limited amplitude. Such a noise is periodic, not necessarily has zero mean, and normally could not be treated by Kalman–filter based techniques. However, the fixed point transformations based adaptive controller successfully can compensate its effects. The other methods here numerically

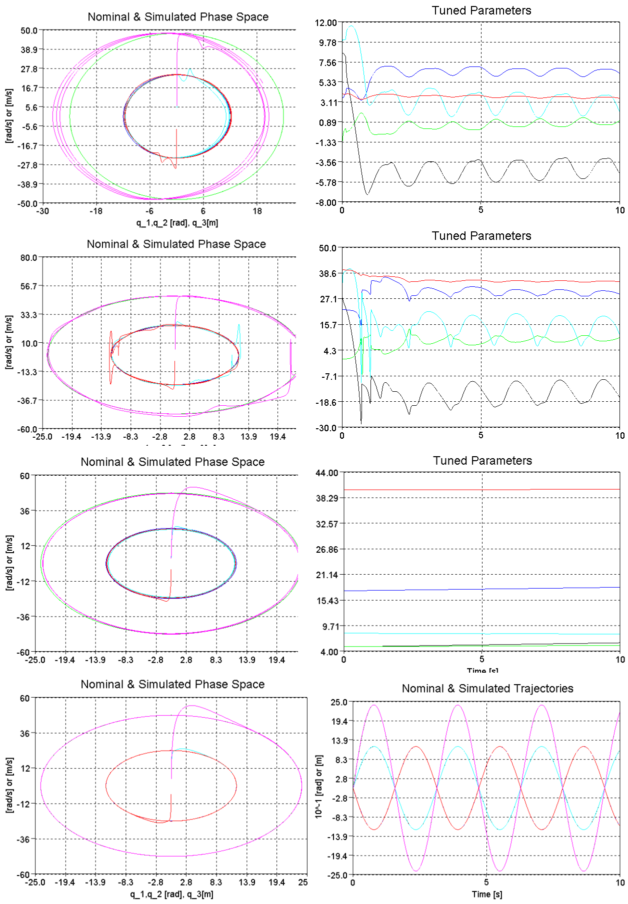

Figure 1: The controlled motion without external disturbances and without friction, according to rows: the original method, the method with modified Lyapunov function, the method with SVD-based tuning, and the Adaptive Fixed Point Transformation (LHS: phase trajectories, RHS: the tuned parameters and the trajectories for the Adaptive Fixed Point Transformation based method)

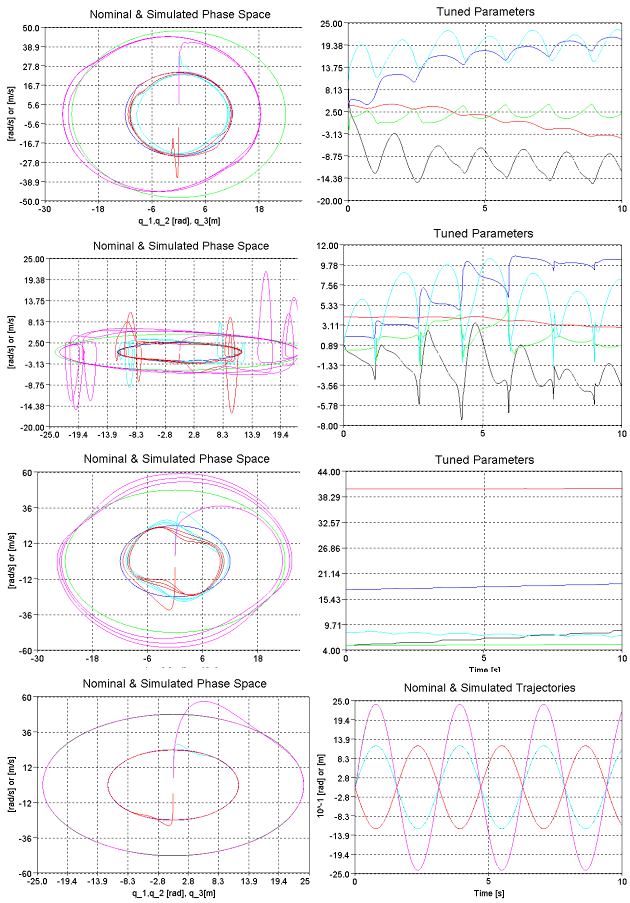

Figure 2: The controlled motion without external disturbances and with friction, according to rows: the original method, the method with modified Lyapunov function, the method with SVD-based tuning, and the Adaptive Fixed Point Transformation (LHS: phase trajectories, RHS: the tuned parameters and the trajectories for the Adaptive Fixed Point Transformation based method)

ISSN: 1790-2769

ISBN: 978-960-474-138-0

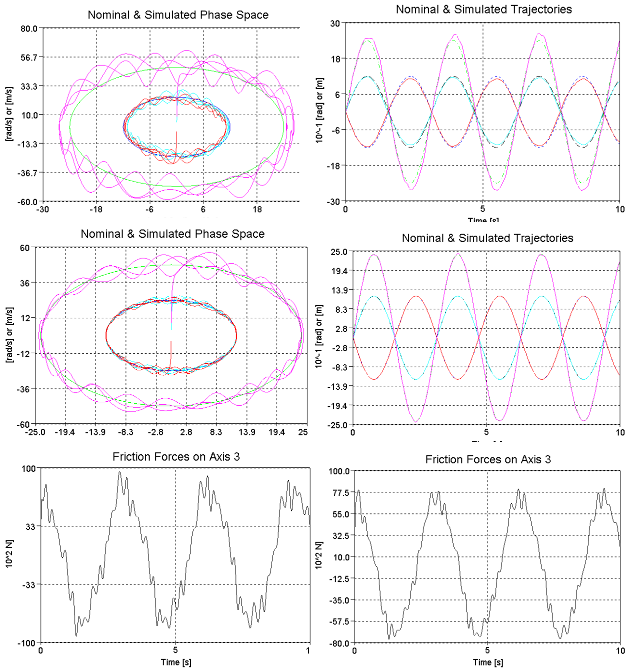

Figure 3: Comparison of the non-adaptive and the Adaptive Fixed Point Transformation–controlled motion with external disturbances and friction: 1st row: non-adaptive case, 2nd row: adaptive case; 3rd row LHS: friction force for the non-adaptive case, RHS: for the adaptive case

investigated are so sensitive to the not modeled and unknown disturbance that it did not make sense to run simulations for them.

5 Conclusion

In this paper two plausible modification was proposed for the traditional adaptive controller developed by Slotine and Li. At first the Lyapunov function was modified with the aim of introducing integrated tracking error feedback to the equations of motion. It was found that this modification improved the precision of trajectory tracking and showed some sensitivity to the speed of parameter tuning. Actually it required slower tuning than the original approach. In the next step, on the basis of formal manipulations, it was shown that the information needed for proper parameter–tuning not necessarily has to originate from a prescribed negative derivative of a Lyapunov function. While exerting the generalized driving forces in a manner quite similar to that of the modified Lyapunov function based case, an SVD–based tuning was proposed that was sensitive to the "direction of tuning". It was found via simulations, that this latter solution resulted in more precise tracking and more quiet parameter tuning than the original and the modified, but still Lyapunov function based technique.

It also was confirmed by the simulations, that according to the theoretical expectations the parameter–tuning based methods are sensitive to the presence of external disturbances and the existence of unknown but dynamically coupled subsystems. For tackling such problems "robust fixed point transformation based" controller was proposed. It was found to be superior in comparison with the other methods. However, this latter approach has the deficiency that it cannot guarantee global stability. Since it operates on the basis of local basins of attractions artificially created for certain iteration, in principle it can quit the region of convergence of the appropriate iteration. Therefore any practical use of this approach needs careful numerical simulations.

Acknowledgements: The authors gratefully acknowledge the support by the *National Office for Research and Technology (NKTH)* using the resources of the *Research and Technology Innovation Fund* within the projects No. *OTKA-K 063405, OTKA-CNK 78168* and *No. RET-10/2006*.

References:

[1] Jean–Jacques E. Slotine, W. Li, Applied Nonlinear Control, *Prentice Hall International, Inc., Englewood Cliffs, New Jersey*, 1991.

[2] G.W. Stewart, On the early history of singular value decomposition, Technical Report TR-92-31, *Institute for Advanced Computer Studies, University of Mariland*, March 1992.

[3] G.H. Golub and W. Kahan, Calculating the singular values and pseudoinverse of a matrix, *SIAM Journal on Numerical Analysis*, Vol. 2, 1965, pp. 205-224.

[4] J.K. Tar, J.F. Bitó, I.J. Rudas, K.R. Kozłowski, and J.A. Tenreiro Machado, Possible adaptive control by tangent hyperbolic fixed point transformations used for controlling the Φ^6-type Van der Pol oscillator, *Proc. of the* 6^{th} *IEEE International Conference on Computational Cybernetics (ICCC 2008)*, November 27–29, 2008, Stará Lesná, Slovakia, pp. 15–20

[5] T. Roska, Development of Kilo Real-time Frame Rate TeraOPS Computational Capacity Topographic Microprocessors, *Plenary Lecture at 10th International Conference on Advanced Robotics (ICAR 2001)*, Budapest, Hungary, August 22–25, 2001.

[6] L. Márton, B. Lantos, Identification and Model-based Compensation of Striebeck Friction, *Acta Polytechnica Hungarica*, Vol. 3, No. 3, 2006.