# Symbiosis of RFPT-Based Adaptivity and the Modified Adaptive Inverse Dynamics Controller*

József K. Tar[1], János F. Bitó[1],
Annamária R. Várkonyi-Kóczy[2], and Adrienn Dineva[3]

[1] Óbuda University, H-1034 Budapest, Bécsi út 96/B, Hungary,
Antal Bejczy Center of Intelligent Robotics
tar.jozsef@nik.uni-obuda.hu, bito@uni-obuda.hu
http://irob.uni-obuda.hu/?q=en

[2] Óbuda University, H-1034 Budapest, Bécsi út 96/B, Hungary,
Donát Bánki Faculty of Mechanical and Safety Engineering
koczy.annamaria@bgk.uni-obuda.hu
http://www.bgk.uni-obuda.hu/indexeng.php

[3] Óbuda University, H-1034 Budapest, Bécsi út 96/B, Hungary,
Doctoral School of Applied Informatics
dineva.adrienn@phd.uni-obuda.hu
http://phd.uni-obuda.hu/

**Abstract.** The use of Lyapunov's "direct" method for designing globally asymptotically stable controllers generates numerous, practically disadvantageous restrictions. The "Adaptive Inverse Dynamic Controller for Robots (AIDCR)" therefore suffers from various difficulties. As alternative design approach the "Robust Fixed Point Transformations (RFPT)" were introduced that instead of parameter tuning adaptively deforms the control signals computed by the use of a fixed approximate system model by observing the behaviour of the controlled system. It cannot guarantee global asymptotic stability but it is robust to the simultaneous presence of the unknown external disturbances and modelling imprecisions. In the paper it is shown that the RFPT-based design can co-operate with a modified version of the AIDCR controller in the control of "Multiple Input-Multiple Output (MIMO)" Systems. On the basis of certain function approximation theorems it is expected that this symbiosis works well in a wider class of physical systems than robots.

**Keywords:** adaptive control, Lyapunov function, adaptive inverse dynamics controller, robust fixed point transformations.

## 1 Introduction: Lyapunov's "Direct" Method and Its Potential Alternatives

In designing the control for strongly non-linear systems whenever the range of the nominal motion cannot be located in the close vicinity of some "working

* The authors thankfully acknowledge the grant provided by the Project *TÁMOP-4.2.2.A-11/1/KONV-2012-0012: Basic research for the development of hybrid and electric vehicles* – The Project is supported by the Hungarian Government and co-financed by the European Social Fund.

J. Fodor and R. Fullér (eds.), *Advances in Soft Computing, Intelligent Robotics and Control,* Topics in Intelligent Engineering and Informatics 8,
DOI: 10.1007/978-3-319-05945-7_6, © Springer International Publishing Switzerland 2014

point" the only efficient method seems to be Lyapunov's "*direct*" one that is based on his doctoral thesis on the stability of motion of non-linear systems [1]. This ingenious problem tackling became well known in the Western World due to translations in the sixties (e.g. [2]) mainly for its exceptional efficiency. It is well known that that the most of the coupled non-linear systems of differential equations cannot be integrated in closed analytical form, therefore the properties of the solutions cannot be concluded by studying the solution itself. This fact means significant difficulty even in our days when efficient numerical tricks and huge, cheap computing power is available since the numerical solutions generally cannot be extrapolated outside the domain of actual computations. *By the application of relatively simple estimations Lyapunov was able to determine various stability properties (e.g. stability, uniform stability, global stability, asymptotic stability, exponential stability) of the solution without knowing or revealing any other significant and interesting details of the motion.*

In the practice of control engineers normally quadratic Lyapunov functions are used. Certain *adaptive techniques* as the AIDCR, or the *"Adaptive Slotine–Li Controller"* [3] assume the existence *formally exactly known* analytical system models in which the parameters are only approximately known. They achieve *globally asymptotically stable* solutions by parameter tuning that corresponds to some machine learning.

Other adaptive techniques as e.g. the *"Model Reference Adaptive Controllers (MRAC)"* (e.g. [4,5,6]) tune *rather control than model parameters* and normally also are designed by the use of Lyapunov functions.

The global stability achieved by these methods seem to be attractive for engineering applications, however, it can be noted that in the great majority of practical applications stability criteria are set only for bounded error regions (e.g. [7]), i.e. it seems to be "too rigorous" for practical use. Since the *primary design intent* in most cases may consists in the precise prescription of the trajectory tracking error relaxation that is not revealed by Lyapunov's technique sometimes the application of *evolutionary methods* is needed for properly setting the control parameters that normally are free ones in the Lyapunov function (e.g. [8,9]). Regarding other disadvantages of the Lyapunov function-based technique the observations as follows can be done: (a) it corresponds to a *satisfactory condition*, i.e. the failure of finding a Lyapunov function for a given problem is not conclusive for the stability of that problem; (b) finding an appropriate Lyapunov function is not an easy task, it cannot be solved by the use of some algorithm.

To evade the difficulties related to the Lyapunov function technique alternative solutions were searched for (well summarized in [11]). The main idea behind them was the use of an *approximate dynamic system model* to calculate the necessary torque or force to realize the second time-derivative that is needed for the *kinematically prescribed trajectory tracking error relaxation.* Instead of tuning the *model parameters* it *adaptively deforms these second derivatives* to compensate the effects of modelling errors and unknown external disturbances. The most efficient deformation was found by a convergent iterative sequence generated by the RFPT transformations (e.g. [10]). The mathematical foundation of this approach was Stefan Bancah's "Fixed Point Theorem" [12] that states that *in a linear, complete, normed metric space the contractive maps generate Cauchy sequences that necessarily are convergent and converge to the fixed point of this map.* Since the necessary conditions of contractivity are valid only in a bounded region this approach generally cannot guarantee global stability. Furthermore, since it is not in the possession of *a priori exact information on the model structure* no asymptotic convergence can be guaranteed: the method permanently utilizes the freshest information on the motion of the controlled system, and according to the principle of causality it utilizes the *past information* in the *future.* In contrast to the abundant number of the free parameters of a quadratic Lyapunov function the RFPT-based method has only three adaptive parameters that can be set easily and can be kept fixed in the case of various control tasks. If needed, i.e. when the basin of convergence may be left various tuning methods were constructed for tuning only one of the adaptive control parameters (e.g. [13,14]). In this manner the RFPT-based method was made competitive with the Lyapunov function based technique regarding global stability. Furthermore, it was found that in this manner a novel design methodology can be built up to create MRAC controllers, too (e.g. [15]).

Till now the coexistence and possible co-operation of the RFPT-based approach and the methods applying adaptive model tuning was not studied. In the present paper it is shown that *a modification of the AIDRC controller* can well co-operate with the RFPT-based adaptivity in the following manner. Whenever no unknown external disturbances are present the two methods complete each other: efficient parameter tuning is going on while the tracking error is kept at low level due to the RFPT-base control; as the tuned model is improved the burdens of the RFPT-based controller step by step decrease and finally no adaptive deformation is needed since *the system finally uses the exact model.* If unknown external perturbations are present the parameter tuning happens on the basis of false information, however, the RFPT-based design keeps the tracking error at low level and efficiently compensates the simultaneous effects of the disturbances and the improper tuning. *The theoretical basis of the convergence always remains Banach's Fixed Point Theorem* [12].

## 2 The AIDCR with Modified Tuning Rule

This method assumes the validity of a special condition: in the equations of motion the dynamic parameters of the system must be separable into an array the is multiplied by a kinematically known matrix. This condition also is valid for the here used 2 Dof system obtained by coupling two generalized 1 DoF van der Pol oscillator the original version of which was developed to describe non-linear oscillations in a triode in 1927 [16]. For the sake of simplicity the properties of the original and the modified version of the AIDCR controller are explained by the use of this paradigm.

### 2.1 The Coupled van der Pol Oscillators

$$\begin{aligned} m_1\ddot{q}_1 + \mu_1\left(q_1^2+q_2^2-c\right)\dot{q}_1 + k_1q_1 + \beta_1 q_1^3 + \lambda_1 q_1^5 = F_1, \\ m_2\ddot{q}_2 + \mu_2\left(q_1^2+q_2^2-c\right)\dot{q}_2 + k_2q_2 + \beta_2 q_2^3 + \lambda_2 q_2^5 = F_2 \end{aligned} \tag{1}$$

in which $m_1 = 10$ kg denotes the *inertia* of oscillator 1, $\mu_1 = 1$ N/(ms) corresponds to some viscous damping if $q_1^2+q_2^2-c>0$ and to some *external excitation* if $q_1^2+q_2^2-c<0$, $c = 3$ m$^2$ describes the coupling of the two subsystems, $k_1 = 100$ N/m is a spring stiffness, $\beta_1 = 1$ N/m$^3$ and $\lambda_1 = 2$ N/m$^5$ are coefficients of non-linear corrections for the distance-dependent spring stiffness, $F_1$ and $F_2$ denote the active control forces, and $q_1$, $q_2$ stand for the *observable* generalized coordinates. The second sub-system has the parameters as $m_2 = 15$ kg, $\mu_2 = 2$ N/(ms), $k_2 = 120$ N/m, $\beta_2 = 2$ N/m$^3$, and $\lambda_2 = 3$ N/m$^5$.

The *initial approximate model* has the following parameters: $\hat{m}_1 = 5$ kg, $\hat{\mu}_1 = 2$ N/(ms), $\hat{c} = 3.5$ m$^2$, $\hat{k}_1 = 110$ N/m, $\hat{\beta}_1 = 0.9$ N/m$^3$, $\hat{\lambda}_1 = 1.5$ N/m$^5$, $\hat{m}_2 = 5$ kg, $\hat{\mu}_2 = 2$ N/(ms), $\hat{k}_2 = 110$ N/m, $\hat{\beta}_2 = 3$ N/m$^3$, and $\hat{\lambda}_2 = 4$ N/m$^5$.

The model parameters can be arranged into an array as

$$\Theta \stackrel{def}{=} \left[m_1, \mu_1, \mu_1 c, k_1, \beta_1, \lambda_1, m_2, \mu_2, \mu_2 c, k_2, \beta_2, \lambda_2\right]^T \tag{2}$$

while the coefficients of this array are the non-zero elements of a matrix of size $2\times 12$ as $Y_{1,1} \stackrel{def}{=} \ddot{q}_1$, $Y_{1,2} \stackrel{def}{=} \dot{q}_1\left(q_1^2+q_2^2\right)$, $Y_{1,3} \stackrel{def}{=} -\dot{q}_1$, $Y_{1,4} \stackrel{def}{=} q_1$, $Y_{1,5} \stackrel{def}{=} q_1^3$, $Y_{1,6} \stackrel{def}{=} q_1^5$, $Y_{2,7} \stackrel{def}{=} \ddot{q}_2$, $Y_{2,8} \stackrel{def}{=} \dot{q}_2\left(q_1^2+q_2^2\right)$, $Y_{2,9} \stackrel{def}{=} -\dot{q}_2$, $Y_{2,10} \stackrel{def}{=} q_2$, $Y_{2,11} \stackrel{def}{=} q_2^3$, and $Y_{2,12} \stackrel{def}{=} q_2^5$ by the use of which (1) takes the form as

$$\begin{bmatrix} F_1 \\ F_2 \end{bmatrix} = Y\left(q, \dot{q}, \ddot{q}\right)\Theta. \tag{3}$$

By the use of this paradigm the AIDCR controller can be built up as follows:

### 2.2 The Modified Tuning Rule

Assume that for the *nominal trajectory* the following kinematic data are *a priori* known: $q^N(t)$, $\dot{q}^N(t)$, and $\ddot{q}^N(t)$. By the use of two positive feedback gains $K_1$ and $K_2$ the *approximate version* of (1) can be used for the calculation of the forces in a manner in which PD-type corrections are applied by the tracking errors $e(t) \stackrel{def}{=} q^N(t) - q(t)$ as

$$\begin{aligned} \hat{m}_1\left(\ddot{q}_1^N + K_1e_1 + K_2\dot{e}_1\right) + \hat{\mu}_1\left(q_1^2+q_2^2-\hat{c}\right)\dot{q}_1 + \hat{k}_1q_1 + \hat{\beta}_1 q_1^3 + \hat{\lambda}_1 q_1^5 = F_1, \\ \hat{m}_2\left(\ddot{q}_2^N + K_1e_2 + K_2\dot{e}_2\right) + \hat{\mu}_2\left(q_1^2+q_2^2-\hat{c}\right)\dot{q}_2 + \hat{k}_2q_2 + \hat{\beta}_2 q_2^3 + \hat{\lambda}_2 q_2^5 = F_2. \end{aligned} \tag{4}$$

In the lack of external disturbances the forces in (4) are the same as in (1). Via eliminating the force terms and subtracting from both sides $\left[\hat{m}_1\ddot{q}_1, \hat{m}_2\ddot{q}_2\right]^T$ the remaining terms can be so rearranged that at one side of the equations the array of the modelling errors multiplied by $Y$ appears, while on the other side the *known quantity* $[\hat{m}_1(\ddot{e}_1 + K_1 e_1 + K_2\dot{e}_1), \hat{m}_2(\ddot{e}_2 + K_1 e_2 + K_2\dot{e}_2)]^T$ remains:

$$\begin{bmatrix} \hat{m}_1(\ddot{e}_1 + K_1 e_1 + K_2\dot{e}_1) \\ \hat{m}_2(\ddot{e}_2 + K_1 e_2 + K_2\dot{e}_2) \end{bmatrix} = Y(q, \dot{q}, \ddot{q}) \left(\Theta - \hat{\Theta}\right). \tag{5}$$

Equation (5) has a *very simple geometric interpretation*: at a given time instant the projections of the 12 dimensional error array $(\Theta - \hat{\Theta})$ are known in the directions of two vectors defined by the two rows of matrix $Y$. The original AIDCR does not directly utilize this information. Instead of that, in order to construct a Lyapunov function, it multiplies both sides of (5) with the *inverse of the approximate inertia matrix* (hence originates the expression "inverse dynamics" in the name of the method), and deduces the parameter tuning rule from the prescription that the Lyapunov function must have negative time-derivative.

The geometrically interpreted information in (5) can directly be utilized as follows. If *exponential decay rate* could be realized for the parameter estimation error the *array equation* $\frac{d}{dt}\left(\Theta - \hat{\Theta}\right) = -\alpha\left(\Theta - \hat{\Theta}\right)$ $(\alpha > 0)$ should be valid. If we multiply both sides of this equation with a *projector* determined by *a few pairwisely orthogonal unit vectors* as $\sum_i e^{(i)} e^{(i)T}$ the equation $\sum_i e^{(i)} \left(\dot{\Theta}_i - \dot{\hat{\Theta}}_i\right) = -\alpha \sum_i e^{(i)} \left(\Theta_i - \hat{\Theta}_i\right)$ is obtained. This situation can well be approximated if we use the Gram-Schmidt algorithm (e.g. [17,18]) for finding the *orthogonal components* of the rows of matrix $Y$ in (5). We can apply the tuning rule *only for the known components* in the form: $\frac{d}{dt}(\Theta - \hat{\Theta}) = -\alpha \sum_i \frac{\tilde{y}^{(i)} \tilde{y}^{(i)T}}{\|\tilde{y}^{(i)}\|^2 + \varepsilon}(\Theta - \hat{\Theta})$ in which $\tilde{y}^{(i)}$ denotes the transpose of the orthogonalized rows of matrix $Y$, and a small $\varepsilon > 0$ evades division by zero whenever the norm of the appropriate row is too small. Since the scalar product is a *linear operation* during the orthogonalization process the appropriate linear combinations of the scalar products in the LHS of (5) can be computed.

## 3 Combination with the RFPT

It is evident that all the above considerations remain valid if in the LHS of (4) instead of $\left(\ddot{q}^N + K_1 e + K_2 \dot{e}\right)$ different feedback terms are used. Therefore this term can be replaced by its iterative variant obtained from the RFPT-base design as follows:

$$\begin{aligned} h(t_n) &\stackrel{def}{=} f\left(r(t_n)\right) - r^d\left(t_{n+1}\right), \quad e(t_n) \stackrel{def}{=} h(t_n)/\|h(t_n)\|, \\ \tilde{B}(t_n) &\stackrel{def}{=} B_c \sigma(A_c \|h(t_n)\|) \\ r(t_{n+1}) &\stackrel{def}{=} \left(1 + \tilde{B}(t_n)\right) r(t_n) + \tilde{B}(t_n) K_c e(t_n) \end{aligned} \tag{6}$$

in which $\sigma(x) \stackrel{def}{=} \frac{x}{1+|x|}$, $r^d_{n+1} \stackrel{def}{=} \ddot{q}^N + K_1 e + K_2 \dot{e}$, $r_n$ denotes the adaptively deformed control signal used in control cycle $n$, and $f\left(r(t_n)\right) \stackrel{def}{=} \ddot{q}(t_n)$, i.e.

the *observed system response* in cycle $n$. It is evident that if $f(r(t_n)) = r^d(t_{n+1})$ then $r(t_{n+1}) = r(t_n)$, that is the solution of the control task (i.e. the appropriate adaptive deformation) is the fixed point of the mapping defined in (6). Therefore the same tuning rule can used in (7) as previously but the known information at the LHS is different. It is evident that if the parameter estimation error is great, the difference between the *adaptively deformed control signal* $r_i(t)$ and the *realized 2nd time-derivative* $\ddot{q}_i(t)$ is significant, therefore at the LHS of (7) considerable quantity is available for parameter tuning.

$$\begin{bmatrix} \hat{m}_1(r_1 - \ddot{q}_1) \\ \hat{m}_2(r_2 - \ddot{q}_2) \end{bmatrix} = Y(q, \dot{q}, \ddot{q}) \left( \Theta - \hat{\Theta} \right). \tag{7}$$

Since the details of the convergence were discussed in ample literature references in the sequel only simulation results will be presented to reveal the co-operation of the RFPT-based adaptivity and model parameter tuning.

## 4 Simulation Results

The simulations were made by a sequential program written in SCILAB with simple Euler integration with fixed time-steps of 0.1 ms. This time-resolution also corresponded to the cycle time of the assumed controller. The *kinematically prescribed trajectory tracking rule* corresponded to the feedback gains $K_1 = \Lambda^2$, $K_2 = 2\Lambda$ with $\Lambda = 10/\mathrm{s}$, the *adaptive control parameters* were $K_c = -10^6$, $B_c = 1$, and $A_c = 10^{-8}$. The learning rate was determined by $\alpha = 5/\mathrm{s}$.

### 4.1 Simulations without Unknown External Disturbances

At first the RFPT-based was studied in the case free of any external disturbances. Figure 1 describes the details of the trajectory tracking and trajectory tracking errors. The precise tracking is evident in spite of the considerable parameter estimation errors. Figure 2 reveals considerable learning activity. *These figures well illustrate the theoretical expectation that the RFPT-based adaptivity and the parameter learning activity can well co-operate.*

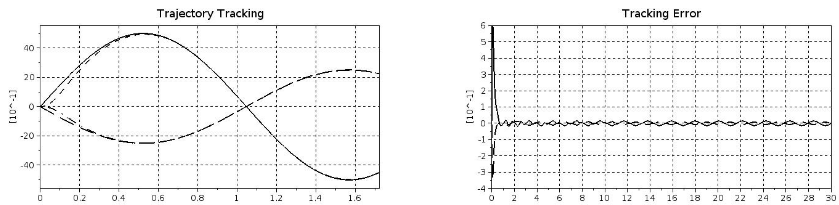


**Fig. 1.** Trajectory tracking [$q_1^N$: solid, $q_2^N$: dashed, $q_1$: dense dash, $q_2$: dash-dot lines LHS], and trajectory tracking error [for $q_1$: solid, for $q_2$: dashed lines RHS] for the RFPT-based design without unknown external disturbances, time is described in the horizontal axes in $s$ units

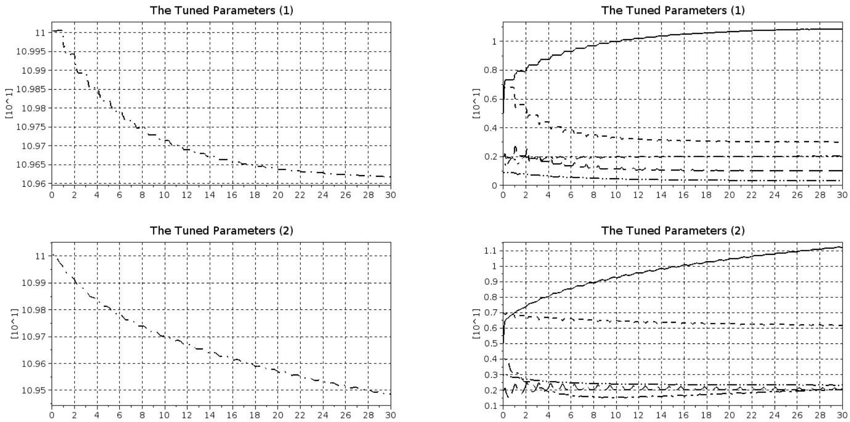


**Fig. 2.** The tuned dynamic system parameters for subsystem 1 (at the top) and for subsystem 2 (at the bottom) for the RFPT-based design without unknown external disturbances, time is described in the horizontal axes in $s$ units [$\hat{\Theta}_1$: solid, $\hat{\Theta}_2$: dashed, $\hat{\Theta}_3$: dense dash, $\hat{\Theta}_4$: dash-dot, $\hat{\Theta}_5$: dash-dot-dot, and $\hat{\Theta}_6$: longdash-dash lines]

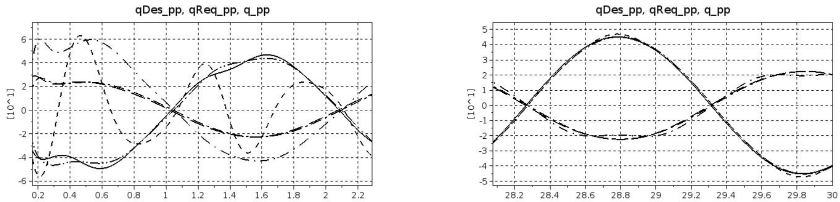


**Fig. 3.** Operation of the RFPT-based adaptivity: [the *kinematically prescribed "Desired" values*: $\ddot{q}_1^{Des}$: solid, $\ddot{q}_2^{Des}$: dashed, the *adaptively deformed "Required" values*: $\ddot{q}_1^{Req}$: dense dash, $\ddot{q}_2^{Req}$: dash-dot, and the *simulated values*: $\ddot{q}_1$: dash-dot-dot, $\ddot{q}_2$: longdash-dash lines] at the early (LHS) and the late (RHS) phases of parameter tuning, time is described in the horizontal axes in $s$ units

The efficiency of parameter learning is well exemplified by Fig. 3 revealing the initial and the late phases of the control session: *at the beginning the dynamic model was very imprecise, therefore considerable adaptive deformation was done* by the RFPT-based design. In the later phase, *when the model already became precise, only minimal extent of adaptive deformation was necessary.*

Figure 4 displays the significance of the RFPT-based design in an alternative manner: it describes the tracking error and tracking of the phase trajectories without external disturbances when the RFPT-based deformation was switched off: the initial tracking error was great and it only slowly decreased as the parameter tuning process proceeded. The details of parameter tuning (Fig. 5) were similar to the case in which the RFPT-based adaptivity was switched on.

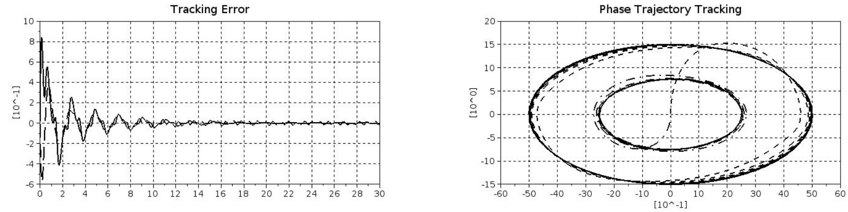


**Fig. 4.** Trajectory tracking error [for $q_1$: solid, for $q_2$: dashed lines, time is described in the horizontal axis in $s$ units (LHS)] and the phase trajectory tracking (i.e. the $\dot{q}_i$ vs. $q_i$ charts) [for $q_1^N$: solid, for $q_2^N$: dashed, for $q_1$: dense dash, for $q_2$: dash-dot lines (RHS)] without unknown external disturbances and without RFPT-based adaptivity

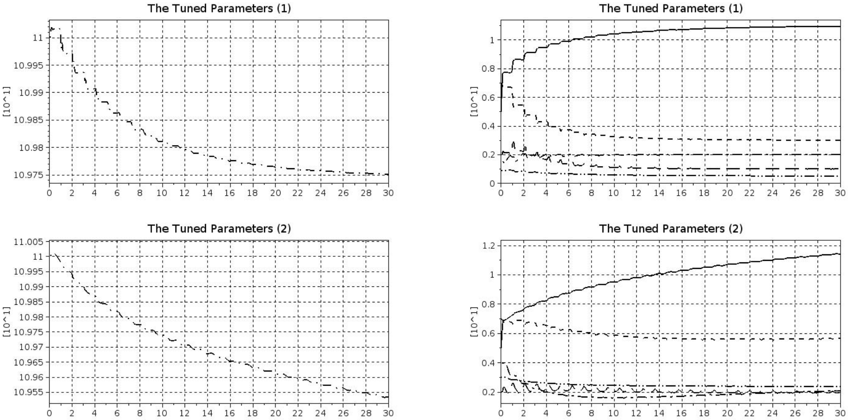


**Fig. 5.** The tuned dynamic system parameters for subsystem 1 (at the top) and for subsystem 2 (at the bottom) without unknown external disturbances and RFPT-based adaptivity, time is described in the horizontal axes in $s$ units [$\hat{\Theta}_1$: solid, $\hat{\Theta}_2$: dashed, $\hat{\Theta}_3$: dense dash, $\hat{\Theta}_4$: dash-dot, $\hat{\Theta}_5$: dash-dot-dot, and $\hat{\Theta}_6$: longdash-dash lines]

## 4.2 Simulations with Temporal Unknown External Disturbances

In this subsection the effects of *temporal unknown external disturbances* are studied via simulations.

At first the the lack of RFPT-based adaptivity is investigated. Figures 6, 7, and 8 reveal that the parameter tuning process was corrupted by the unknown external disturbances and the trajectory tracking errors became quite considerable.

When the RFPT-based adaptivity was switched on the trajectory tracking became precise (Figs. 9, 10), however, since the parameter tuning happened on the basis of false information it was corrupted again (Fig. 8).

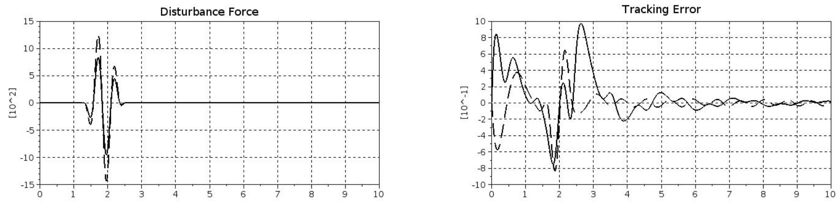


**Fig. 6.** Disturbance forces [for $F_1^{Dist}$: solid, for $F_2^{Dist}$: dashed lines (LHS)], and trajectory tracking error [for $q_1$: solid, for $q_2$: dashed lines (RHS)], and trajectory tracking error [for $q_1$: solid, for $q_2$: dashed lines RHS] with unknown external disturbances and without RFPT-based adaptation, time is described in the horizontal axes in $s$ units

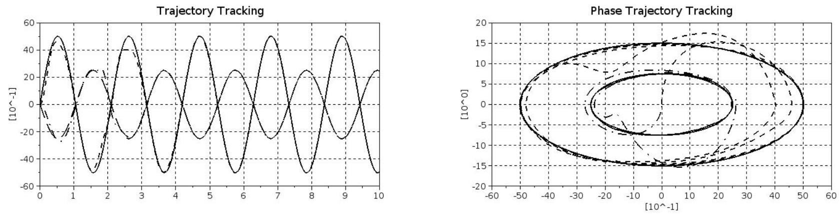


**Fig. 7.** Trajectory (LHS) and phase trajectory (RHS) tracking [for $q_1^N$: solid, for $q_2^N$: dashed, for $q_1$: dense dash, for $q_2$: dash-dot lines] with temporal unknown external disturbances and without RFPT-based adaptation, time is described in the horizontal axes in $s$ units

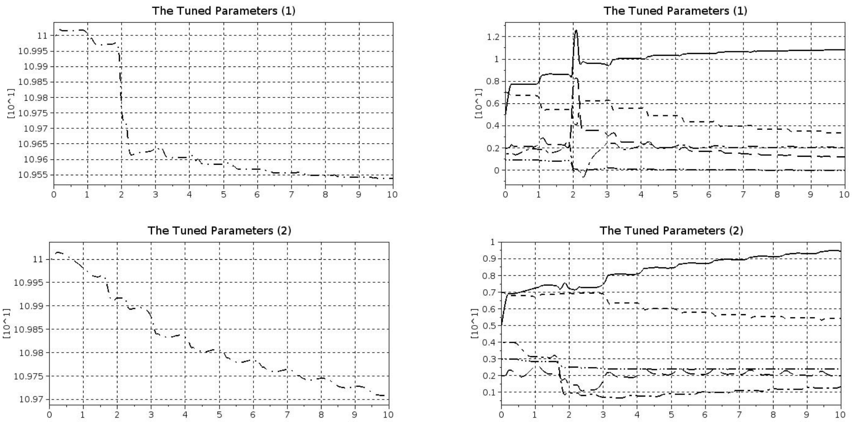


**Fig. 8.** The tuned dynamic system parameters for subsystem 1 (at the top) and for subsystem 2 (at the bottom) with unknown external disturbances and without RFPT-based adaptivity, time is described in the horizontal axes in $s$ units [$\hat{\Theta}_1$: solid, $\hat{\Theta}_2$: dashed, $\hat{\Theta}_3$: dense dash, $\hat{\Theta}_4$: dash-dot, $\hat{\Theta}_5$: dash-dot-dot, and $\hat{\Theta}_6$: longdash-dash lines]

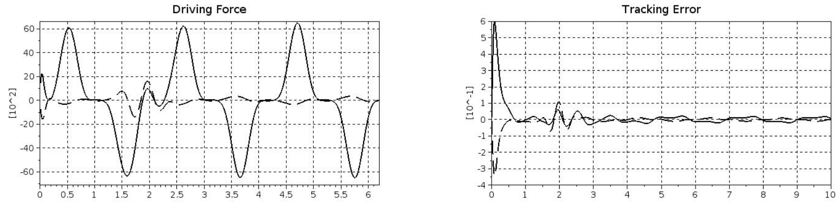


**Fig. 9.** The driving forces [for $F_1$: solid, for $F_2$: dashed lines (LHS)], and trajectory tracking error [for $q_1$: solid, for $q_2$: dashed lines (RHS)], and trajectory tracking error [for $q_1$: solid, for $q_2$: dashed lines RHS] with unknown external disturbances and RFPT-based adaptation, time is described in the horizontal axes in $s$ units

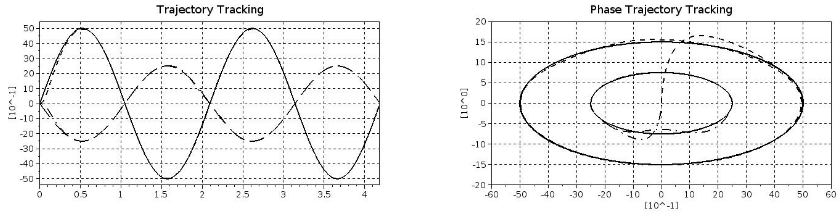


**Fig. 10.** Trajectory (LHS) and phase trajectory (RHS) tracking [for $q_1^N$: solid, for $q_2^N$: dashed, for $q_1$: dense dash, for $q_2$: dash-dot lines] with temporal unknown external disturbances and RFPT-based adaptation, time is described in the horizontal axes in $s$ units

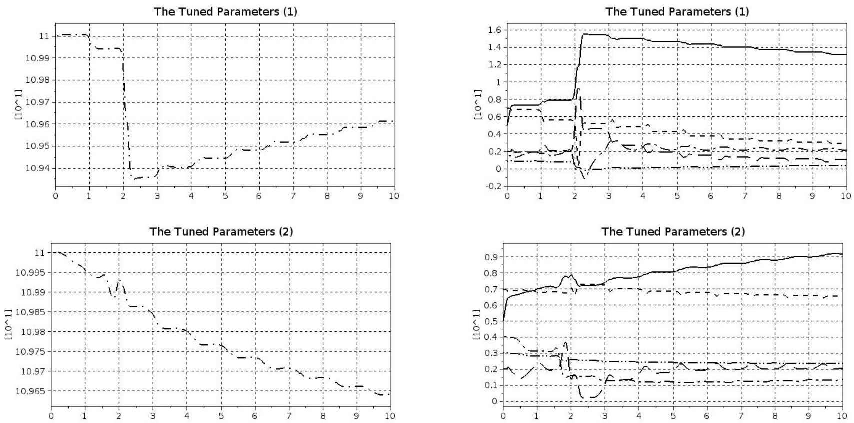


**Fig. 11.** The tuned dynamic system parameters for subsystem 1 (at the top) and for subsystem 2 (at the bottom) with unknown external disturbances and RFPT-based adaptivity, time is described in the horizontal axes in $s$ units [$\hat{\Theta}_1$: solid, $\hat{\Theta}_2$: dashed, $\hat{\Theta}_3$: dense dash, $\hat{\Theta}_4$: dash-dot, $\hat{\Theta}_5$: dash-dot-dot, and $\hat{\Theta}_6$: longdash-dash lines]

## 5 Conclusions

In this paper the symbiosis of RFPT-based adaptivity and the *"Modified Adaptive Inverse Dynamics Controller"* was theoretically proved and illustrated via simulation results for two coupled, strongly non-linear, generalized van der Pol oscillators.

The main idea on the basis of which analytical model learning and the RFPT-based design needing only a rough approximate model can co-exist and co-operate consists in evading the use of the Lyapunov function based stability proof since it demands too rigorous formal limitations. It was shown that tuning of the parameters of the *formally exact analytical model* was possible without the use of any Lyapunov function by direct utilization of the available information on the modelling errors on the basis of simple and lucid geometric interpretation. The stability of the new controllers is guaranteed by *"Banach's Fixed Point Theorem"*.

It is worthy of note that the novel method (as well as its original version) yields correct parameter tuning only in the lack of unknown external disturbances. However, while the trajectory tracking of the original, Lyapunov function based *"Adaptive Inverse Dynamics Controller"* is considerably corrupted by such disturbances, the novel method guarantees precise trajectory and phase trajectory tracking even in this case at least if these disturbances are of temporal nature.

It is also worthy of note that the dynamic model under consideration had the special property that the array of the dynamic model parameters was separable and it was multiplied by a matrix of kinematically known quantities. On the basis of the modern function approximation theorems ([19,20]) it is expected that this approach can be extended for more general cases as Bernard and Slotine already mentioned a possible extension of the Lyapunov function based method for wavelets in [21].

In the future similar investigations are planned for completing the appropriate modification of the *"Slotine-Li Adaptive Robot Controller"* with the RFPT-based adaptive design.

## References

1. Lyapunov, A.M.: A general task about the stability of motion. PhD Thesis, University of Kazan (1892) (in Russian)
2. Lyapunov, A.M.: Stability of motion. Academic Press, New York (1966)
3. Slotine, J.-J.E., Li, W.: Applied Nonlinear Control. Prentice Hall International, Inc., Englewood Cliffs (1991)
4. Nguyen, C.C., Antrazi, S.S., Zhou, Z.-L., Campbell Jr., C.E.: Adaptive control of a Stewart platform-based manipulator. Journal of Robotic Systems 10(5), 657–687 (1993)
5. Somló, J., Lantos, B., Cát, P.T.: Advanced Robot Control. Akadémiai Kiadó, Budapest (2002)

6. Hosseini-Suny, K., Momeni, H., Janabi-Sharifi, F.: Model reference adaptive control design for a teleoperation system with output prediction. J. Intell. Robot. Syst. 1–21 (2010), doi:10.1007/s10846-010-9400-4
7. Kovács, L.: Modern robust control in patophysiology from theory to application. In: IEEE 11th International Symposium on Applied Machine Intelligence and Informatics (SAMI 2013), p. 13 (2013)
8. Sekaj, I., Veselý, V.: Robust output feedback controller design: Genetic algorithm approach. IMA J. Math. Control Info. 22(3), 257–265 (2005)
9. Chen, J.L.: Chang, Wei-Der: Feedback linearization control of a two-link robot using a multi-crossover genetic algorithm. Expert Systems with Applications 2(pt. 2), 4154–4159 (2009)
10. Tar, J.K., Bitó, J.F., Nádai, L., Tenreiro Machado, J.A.: Robust Fixed Point Transformations in adaptive control using local basin of attraction. Acta Polytechnica Hungarica 6(1), 21–37 (2009)
11. Tar, J.K.: Adaptive Control of Smooth Nonlinear Systems Based on Lucid Geometric Interpretation (DSc Dissertation). Hungarian Academy of Sciences, Budapest, Hungary (2012)
12. Banach, S.: Sur les opérations dans les ensembles abstraits et leur application aux équations intégrales (About the Operations in the Abstract Sets and Their Application to Integral Equations). Fund. Math. 3, 133–181 (1922)
13. Tar, J.K., Nádai, L., Rudas, I.J., Várkonyi, T.A.: RFPT-based adaptive control stabilized by fuzzy parameter tuning. In: 9th European Workshop on Advanced Control and Diagnosis (ACD 2011), Budapest, Hungary, pp. 1–8 (2011)
14. Kósi, K., Tar, J.K., Rudas, I.J.: Improvement of the stability of RFPT-based adaptive controllers by observing "precursor oscillations". In: Proc. of the 9th IEEE International Conference on Computational Cybernetics, Tihany, Hungary, July 8-10, pp. 267–272 (2013)
15. Tar, J.K., Bitó, J.F., Rudas, I.J.: Replacement of Lyapunov's direct method in model reference adaptive control with robust fixed point transformations. In: Proc. of the 14th IEEE International Conference on Intelligent Engineering Systems 2010, Las Palmas of Gran Canaria, Spain, pp. 231–235 (2010)
16. Van der Pol, B.: Forced oscillations in a circuit with non-linear resistance (reception with reactive triode). The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science 7(3), 65–80 (1927)
17. Gram, J.P.: Über die Entwickelung reeler Funktionen in Reihen mittelst der Methode der kleinsten Quadrate. Journal für die Reine und Angewandte Mathematik 94, 71–73 (1883)
18. Schmidt, E.: Zur Theorie der linearen und nichtlinearen Integralgleichungen I. Teil: Entwicklung willkürlicher Funktionen nach Systemen vorgeschriebener. Matematische Annalen 63, 442 (1907)
19. Weierstraß, K.: Über die analytische Darstellbarkeit sogenannter willkürlicher Functionen einer reellen Veränderlichen. Sitzungsberichte der Akademie zu Berlin (1885)
20. Stone, M.H.: A generalized Weierstrass approximation theorem. Math. Magazine 21, 167–184, 237–254 (1948)
21. Bernard, C.P., Slotine, J.-J.E.: Adaptive control with multiresolution bases. In: Proceedings of the 36th IEEE Conference on Decision and Control, San Diego, CA, December 10-12, vol. 4, pp. 3884–3889 (1997)