On Using Wegstein's Convergence Accelerator in Iterative Adaptive Control

Bence Varga[1,2*], Richárd Horváth[2] and József K. Tar[1,3]

[1]Doctoral School of Applied Informatics and Applied Mathematics, Obuda University, Bécsi út 96/B, H-1034 Budapest, Hungary, varga.bence@bgk.uni-obuda.hu, tar.jozsef@nik.uni-obuda.hu

[2]Banki Donat Faculty of Mechanical and Safety Engineering, Obuda University, Népszínház utca 8, 1081-Budapest, Hungary horvath.richard@bgk.uni-obuda.hu

[3]John von Neumann Faculty of Informatics, Obuda University, Bécsi út 96/B, H-1034 Budapest, Hungary

*Corresponding: varga.bence@bgk.uni-obuda.hu

Abstract: Robust Fixed-Point Transformation-based (RFPT) Adaptive Control was introduced in 2009, as an alternative to certain Lyapunov function-based design techniques. The basic idea behind this control method is to transform the control task into a problem of finding a fixed-point in a complete metric space by a contraction mapping. The convergence rate of the iteration can be manipulated with some adaptive parameters, and some of them dependent on the dynamics of the controlled system. In this paper the potential of Wegstein's method in enhancing the convergence rate was investigated. As demonstrated in experimental trials on a DC motor, the implementation of Wegstein's method can improve the trajectory tracking performance in certain scenarios. However, the online tuning of the Wegstein parameter is difficult, in the presence of measurement noise. A solution is proposed in this paper, for a moving average estimation of the first order derivative of the iterative function, with exponential smoothing, to ensure stable calculation of the Wegstein parameter.

Keywords: Fixed-Point Iteration; Convergence Acceleration; Wegstein; Adaptive; Motor Control; Robust Fixed-Point Transformation

1 Introduction

Fixed-Point Iteration has been extensively used in numerical analysis, particularly in the case of problems involving non-linear phenomena. In the last few decades, the convergence properties of iterative sequences have been widely investigated, however one of pioneering work was published by Banach in 1922 [1]. According to Banach's fixed point theorem in a complete metric space, which is since then called Banach space, a contractive self-mapping $\Phi: \mathcal{B} \mapsto \mathcal{B}$, that is, any mapping for which there exist $K \in [0,1[$ such that $\forall y, x \in \mathcal{B}$

$$|\Phi(y) - \Phi(x)| \leq K|y - x| \quad (1)$$

converges to a unique fixed point $x_\star$, i.e., for which $\Phi(x_\star) = x_\star$. It was also shown that a simple iteration in the form of $\{x_0, x_1 = \Phi(x_0), \dots, x_{n+1} = \Phi(x_n), \dots\}$ can be used to find the unique fixed point. Nevertheless, many alternative methodologies have been proposed for finding a fixed point. Among these e.g., the Mann [2] and Ishikawa [3] iteration process are particularly prominent.

Speeding up the convergence of certain iterative processes have been key area of interest in the research of fixed-point theory. Some classical notable methods are Aitken's delta-squared process [4], the Steffensen Accelerator [5], Wegstein's method [6] or the Anderson Mixing method [7].

In [8] Aitken's delta-squared process was used to estimate contouring error in case of a 3-axis motion control application. It was demonstrated that convergence acceleration not only ensured sufficient computation time, but iterative divergence was also avoided in some cases. The application of Anderson Mixing method was investigated for an iterative closest point problem [9] which has shown robust behavior, and it was efficiently handling even noisy datasets as well. Anderson Acceleration was also applied in [10], where it was demonstrated that, in the context of an inverse problem in imaging, the Anderson Acceleration reduces the net time of iteration as convergence occurs more rapidly. However, it should be noted that the method requires more time per iterative step due to computational complexity. Wegstein's method is often used in chemical process simulations e.g., [11] [12].

In the field of adaptive control of nonlinear second order systems, a simple fixed-point iteration-based scheme was introduced in 2009 [13]. The authors have suggested the application of a deformation function in order to transform a Computed Torque Control (CTC) scheme into a fixed-point problem. The solution was named Robust Fixed-Point Transformation-based (RFPT) Adaptive Control, that have demonstrated improved trajectory tracking precision, when the dynamic model of the controlled system was neither complete nor precise. In order to further increase the tracking precision of the RFPT method the use of Steffensen Convergence Accelerator was suggested Kósi *et al.* in [14]. Later a similar idea was introduced in case of a novel adaptive sliding mode controller in [15] [16].

The idea of using Steffensen convergence accelerator technique stems from the fact that the speed of convergence can be slow when parameter K in (1) is close to 1. Steffensen realized that it is expedient to break the infinite sequence into finite number excerpts as $\{x_0, x_{n-1} = \Phi(x_0), x_n = \Phi(x_{n-1})\}$ in which x_0 does not originate as a function of a previous point of the sequence. Instead of that, in the vicinity of the fixed point the derivative of $\Phi(x)$, i.e., $\Phi'(x_{n-1})$ can be estimated as:

$$\Phi'(x_{n-1}) \approx \frac{\Phi(x_{n-1}) - \Phi(x_{n-2})}{x_{n-1} - x_{n-2}} = \frac{x_n - x_{n-1}}{x_{n-1} - x_{n-2}} \quad (2)$$

and in first order Taylor estimation, instead of generating $x_{n+1} = \Phi(x_n)$ of the Banach sequence immediately try to find the fixed point as $\Phi(x_{n-1} + \Delta x) = x_\star$, that with the above estimation of the derivative leads to the approximation $\Phi(x_{n-1} + \Delta x) \approx \Phi(x_{n-1}) + \Phi'(x_{n-1})\Delta x$, i.e.:

$$\Delta x = \frac{(x_n - x_{n-1})(x_{n-1} - x_0)}{-x_n + 2x_{n-1} - x_0} \tag{3}$$

The value $x_{n-1} + \Delta x$ will be the initial element of the next excerpt of the sequence that can be written into the next starting variable x_0. To avoid division by zero by the introduction of the small positive constant $0 < \epsilon$ the following approximation can be done

$$x_0^{next} \approx x_{n-1} + \frac{(x_n - x_{n-1})(x_{n-1} - x_0)(-x_n + 2x_{n-1} - x_0)}{\epsilon + (-x_n + 2x_{n-1} - x_0)^2} \tag{4}$$

It must be noted that in his original paper in [5] Steffensen also utilized other special observations on the nature of the Banach sequence that he applied in his formula. If in the adaptive control we restrict ourselves to observations made for the freshest considered excerpt, for the estimation of the derivative obtaining the values $\Phi(x_0) = x_{n-1}$, and $\Phi(x_{n-1}) = x_n$ cannot be avoided. This attitude is reasonable because in our case the function $\Phi(x)$ has a hidden parameter x^{Des} as $\Phi(x; x^{Des})$, and this parameter in the control process is not exactly constant: it can slowly drift according to the kinematic design applied in the adaptive control. Therefore, using the fresh information obtained from the observations seems to be expedient.

In addition to the Steffensen method, the use of other convergence accelerating techniques are not yet explored in the field of Fixed-Point Iteration-based (FPI) Adaptive Control. In this paper Wegstein's method is investigated in order to improve the convergence rate and hence the tracking precision of an FPI controller. This method seems advantageous when it is compared to the Steffensen Accelerator. The latter one requires two iterative steps for each improved estimation according (4), whereas Wegstein’s method, requires only a single iterative step. It appears that this is more appropriate for FPI control, in which a single iteration is executed within each control cycle, resulting in a fixed point that shifts with each cycle. In addition, the experimental results are presented for a simple DC motor control application.

2 Robust Fixed Point Transformation-based Adaptive Control

In case of a trajectory tracking application the trajectory tracking error is defined as $e(t) = q^N(t) - q^R(t)$, where $q^N(t)$ is the nominal trajectory and $q^R(t)$ is the realized trajectory of the generalized coordinates. Defining an error relaxation rule as $\left(\Lambda + \frac{\mathrm{d}}{\mathrm{d}t}\right)^3 e_{int}(t) \equiv 0$, the kinematic prescription for the second order derivative of the generalized coordinate is formed:

$$\ddot{q}^{Des}(t) = \ddot{q}^{N}(t) + \Lambda^3 e_{int}(t) + 3\Lambda^2 e(t) + 3\Lambda\, \dot{e}(t) \quad (5)$$

where $e_{int}(t)$ is integrated trajectory tracking error ($e_{int}(t) = \int_{t_0}^{t}\left(q^N(\xi) - q^R(\xi)\right)\mathrm{d}\xi$), $\ddot{q}^{Des}(t)$ is the desired value of the second order derivative of the generalized coordinates that should be precisely implemented by the controller and finally $\Lambda > 0$ is a single design parameter and $K_p = 3\Lambda^2$, $K_i = \Lambda^3$, $K_d = 3\Lambda$ are the PID gains of the controller.

In Computed Torque Control the inverse dynamic model ($F^{-1}(\cdot)$) is used to linearize and decouple the nonlinear dynamics of the controlled system, so the control force ($Q(t)$) is given as:

$$Q(t) = F^{-1}\left(\ddot{q}^{Des}(t), \dot{q}^R(t), q^R(t)\right) \quad (6)$$

However, in case of an imprecise dynamic model $\tilde{F}^{-1}(\cdot)$, the control force $\tilde{Q}(t) = \tilde{F}^{-1}\left(\ddot{q}^{Des}(t), \dot{q}^R(t), q^R(t)\right)$ applied to the system, results in:

$$\ddot{q}^R(t) = F\left(\tilde{Q}(t), \dot{q}^R(t), q^R(t)\right) \neq \ddot{q}^{Des}(t) \quad (7)$$

To resolve this issue, in [13] it was suggested to introduce a deformed value $\ddot{q}^{Def}(t)$ which in combination with the imprecise model $\tilde{F}^{-1}(\cdot)$ still results in precise implementation of $\ddot{q}^{Des}(t)$ that would be required by (5). In essence, the control task is converted into a problem of finding the correct control force ($Q(t)$) to which the controlled system responds by some desired response ($\ddot{q}^{Des}(t)$). The suggested deformation was formulated as:

$$\ddot{q}^{Def}(t) = \left(K_c + \ddot{q}^{Def}(t-\delta t)\right) \cdot$$
$$\cdot \left[1 + B_c \tanh\left(A_c\left(\ddot{q}^R(t)\,(t-\delta t) - \ddot{q}^{Des}(t)\right)\right)\right] - K_c \quad (8)$$

where A_c, B_c and K_c are parameters that should be set by the user and δt is the sampling time of the controller. Since $(\mathbb{R}, \|\cdot\|)$ forms a complete metric space (8) will converge if (1) holds. In the case of a differentiable function $\varphi(x)\colon \mathbb{R} \mapsto \mathbb{R}$ the integral estimation can be used as

$$\varphi(b) - \varphi(a) = \int_a^b \varphi'(\xi)\mathrm{d}\,\xi \quad (9a)$$

$$|\varphi(b) - \varphi(a)| \leq \int_a^b |\varphi'(\xi)|\mathrm{d}\,\xi \quad (9b)$$

Therefore, if $|\varphi'| < K < 1$ can be guaranteed, since $|\varphi(b) - \varphi(a)| \leq K|b - a|$, i.e., the function will be contractive. If $\ddot{q}^{Des}(t)$ does not vary drastically, a response function can be introduced as:

$$\ddot{q}^R(t) = F\left(\tilde{F}\left(\ddot{q}^{Def}(t), \dot{q}^R(t), q^R(t)\right), \dot{q}^R(t), q^R(t)\right) \approx R(\ddot{q}^{Def}(t))$$

The derivative of (8) is given as:

$$G'(\cdot) = \left(1 + B_c \tanh\left(A_c(F(\cdot) - \ddot{q}^{Des}(t))\right)\right) + \frac{B_c A_c R'(\cdot)\left(\ddot{q}^{Des}(t-\delta t) + K_c\right)}{\cosh^2\left(A_c\left(F(\cdot) - \ddot{q}^{Des}(t)\right)\right)} \quad (10)$$

so (1) holds if $R(\cdot)$ changes slowly and the adaptive parameters are tuned properly, with special emphasis on the A_c that has the most significant effect on the rate of convergence [17].

The operation of the controller, in case of a digital implementation, can be summarized as:

- In each control cycle the $\ddot{q}^{Des}(t)$ is calculated based on (5)
- The $\ddot{q}^{Des}(t)$ value is deformed using (8) hence: $\ddot{q}^{Def}(t) = G(\ddot{q}^{Def}(t-\delta t), \ddot{q}^{Des}(t), \ddot{q}^{R}(t-\delta t))$ is obtained. In the first control cycle a boundary condition is used i.e., $\ddot{q}^{Def}(0) = \ddot{q}^{Des}(0)$
- The deformed value is used to calculate the control force from the available dynamic model $\tilde{Q}(t) = \tilde{F}^{-1}\left(\ddot{q}^{Def}(t), \dot{q}^{R}(t), q^{R}(t)\right)$
- The control force $\tilde{Q}(t)$ is exerted on the system and response is observed

It can be observed that in each control cycle a single step of iteration is made and the fixed point is shifting with each control cycle as given by (8). Moreover, in the case of a second-order system, it is important to note that not only the first-order derivatives of the generalized coordinates are fed back, as indicated by equation (5), but the second-order coordinates are also fed back with a delay of δt.

The authors note that the kinematic prescription (5), can be formulated in several ways. Some suggestions are e.g., a Control Lyapunov Function-based design [18] or back stepping type solution [19]. The same holds for the adaptive deformation (8). Some possible variation was introduced e.g., in [20] in order to extend the solution for MIMO systems, the noise sensitivity was addressed in [21] or the solution in [22] that uses a simple geometric interpretation.

3 On the Use of Wegstein's Convergence Accelerator in Adaptive Control

In his original paper in [6] Wegstein considered finding the fixed point of $\Phi(x_\star) = x_\star \in R$ in a Banach space with the iterative sequence $\{x_0, x_1 = \Phi(x_0), \dots, x_{i+1} = \Phi(x_i), \dots\}$. From this function he introduced a different one by using the parameter $q \in R$ as follows:

$$\Psi(x) \stackrel{\text{def}}{=} qx + (1-q)\Phi(x) \tag{11}$$

Evidently, $\Psi(x_\star) = qx_\star + (1-q)\Phi(x_\star) = qx_\star + (1-q)x_\star = x_\star$ i.e., $\Psi(x)$ has the same fixed point as $\Phi(x)$. With this new function a similar iterative sequence can be generated as $\{x_0, \hat{x}_1 = \Psi(x_0), \dots \hat{x}_{\ell+1} = \Psi(\hat{x}_\ell), \dots\}$. Parameter q can be used for manipulating the convergence properties of the new sequence that can be used for finding $x_\star$ instead of the original iteration. For this purpose, we can consider (9) as the satisfactory and necessary condition of convergence as:

$$-1 < \frac{\mathrm{d}\Psi}{\mathrm{d}x} = q + (1-q)\Phi' < 1 \quad \text{i.e.}$$

$$-1 - \Phi' < q(1-\Phi') < 1 - \Phi' \tag{12}$$

1) Case $\Phi' > 1$: Then $1 - \Phi' < 0$, therefore it yields

$$1 < \frac{-1-\Phi'}{1-\Phi'} = \frac{\Phi'+1}{\Phi'-1} > q > \frac{1-\Phi'}{1-\Phi'} = 1 \tag{13}$$

2) Case $\Phi' = 1$: Then (12) states that $-2 < 0$ that does not have any restriction for q.

3) Case $-1 < \Phi' < 1$: Then $1 - \Phi' > 0$, therefore it is obtained that:

$$0 > \frac{-1-\Phi'}{1-\Phi'} = \frac{1+\Phi'}{\Phi'-1} < q < 1 \tag{14}$$

4) Case $\Phi' = -1$: Then (12) means that $0 < q < 1$

5) Case $\Phi' < -1$: Then $1 - \Phi' > 2 > 0$ and (12) means that:

$$0 < \frac{-1-\Phi'}{1-\Phi'} = \frac{1+\Phi'}{\Phi'-1} < q < 1 \tag{15}$$

The above considerations guarantee that it is always possible to find a parameter q that makes the modified sequence $\{x_0, \hat{x}_1 = \Psi(x_0), \ldots\ \hat{x}_{\ell+1} = \Psi(\hat{x}_\ell), \ldots\}$ convergent.

For the practical use of the formulae it can be noted that the *starting value of the considerations* can be $q = \pm\varepsilon$, in which $\varepsilon > 0$ is a small positive number. If at least the sign of Φ' is fixed in the problem, going over or under the value of $q = 1$ convergence can be achieved.

3.1 Online Parameter Tuning

According to (12) the fastest convergence would be achieved at $q + (1-q)\Phi' = 0$ leading to:

$$q = \frac{\Phi'}{\Phi'-1} \tag{16}$$

therefore the estimation of the derivative e.g., by the trick suggested by Steffensen in [5] or by any other approximation can be expedient. A simple estimation for q can be given by the use of a backward difference formulae:

$$\Phi' \approx \frac{\Phi(\hat{x}_n) - \Phi(\hat{x}_{n-1})}{\hat{x}_n - \hat{x}_{n-1}} = \frac{x_{n+1} - x_n}{\hat{x}_n - \hat{x}_{n-1}} \tag{17}$$

The above solution requires a single step of iteration in order to obtain $x_{n+1} = \Phi(\hat{x}_n)$.

3.2 Application for the FPI-based Adaptive Control

In the case of single variable systems, the deformation function of the Robust Fixed-Point Transformation in digital control framework can be used for constant $i \in N$ for slowly varying $\ddot{q}^{Des}(t)$ as:

$$\ddot{q}^{Def}(i+1) = G\left(\ddot{q}^{Def}(i), \ddot{q}^{R}(i), \ddot{q}^{Des}(i+1)\right) \approx \Phi\left(\ddot{q}^{Def}(i)\right) \quad (18)$$

The original control parameters K_c, B_c, and A_c made it possible to manipulate the speed of convergence within certain limits determined by the *''response function''* $R(\cdot)$. The introduction of Wegstein's relaxation parameter allows direct manipulation of the convergence behavior of fixed-point iterations, potentially accelerating convergence or mitigating divergence. However, some limitations may arise for application in control systems. Particularly, estimation of the relaxation parameter can be compromised by measurement noise. This makes the direct use of analytical expressions such as equation (17) unreliable. A more robust estimation of the derivative of the fixed-point mapping can be achieved using moving window estimation in the following manner:

$$\Phi'_w \approx \frac{\sum_{\ell=0}^{w}(x_{n+1-\ell}-x_{n-\ell})}{\sum_{\ell=0}^{w}(\hat{x}_{n-\ell}-\hat{x}_{n-1-\ell})} = \frac{x_{n+1}-x_{n-w}}{\hat{x}_n-\hat{x}_{n-w}} \quad (19)$$

This telescoping formulation reduces computational complexity and improves numerical stability, making it well-suited for real-time applications where noise and limited precision are concerns.

4 Experimental Results

The use of Wegstein's method in an FPI control application was experimentally tested with an FIT0185 12V DC motor, that was driven by a BTS7960 chip-based dual half bridge drive. The control algorithm was implemented on an ARM Cortex-M7 microprocessor, and the adaptive deformation was applied on the PWM output of the controller. The angular position of the motor's shaft was measured with the inbuilt encoder by the use of quadrature decoding. Higher derivatives were estimated using second order backward difference formulae:

$$\dot{q}^{R}(t) = \frac{3q^{R}(t)-4q^{R}(t-\delta t)+q^{R}(t-2\delta t)}{2\delta t} \quad (20a)$$

$$\ddot{q}^{R}(t) = \frac{2q^{R}(t)-5q^{R}(t-\delta t)+4q^{R}(t-2\delta t)-q^{R}(t-3\delta t)}{\delta t^2} \quad (20b)$$

Due to the quantization error of the position measurement, the noisy higher order derivatives were filtered using second order Butterworth low pass filter with $f_c \approx 30$Hz cut-off frequency. The design parameter for the PID-type feed according (5) was set to $\Lambda = 12$s^{-1} and the adaptive design parameters in (8) were $B_c = -1$

and $K_c = 10^6$. The sampling time of the controller was $\delta t = 0.001$s. The controller was tested in a *"quasi"* model free form as:

$$\tilde{Q}(t) = \ddot{q}^{\mathrm{Def}}(t) \quad (21)$$

The derivative of the fixed-point mapping was estimated using eq. (19) with a window length of $w = 15$. Additionally, exponential smoothing was applied on the estimate in the form of:

$$\Phi'^{S}{}_{w}(t) = \alpha\Phi'_{w}(t) + (1-\alpha)\Phi'^{S}{}_{w}(t-\delta t) \quad (32)$$

where $\alpha = 0.5$ controls the smoothing factor. This approach helps mitigate abrupt changes in the derivative estimate, promoting smoother convergence behavior. Furthermore, in order to ensure stable convergence of the iteration and avoid excessive deformation of $\ddot{q}^{Def}(t)$, the Wegstein relaxation parameter q was constrained in the $[-1,1]$ interval.

In order to enhance the non-linear behavior of the control system a spring with unknown stiffness was attached to the motor's shaft. The experimental setup is shown in Fig. 1.

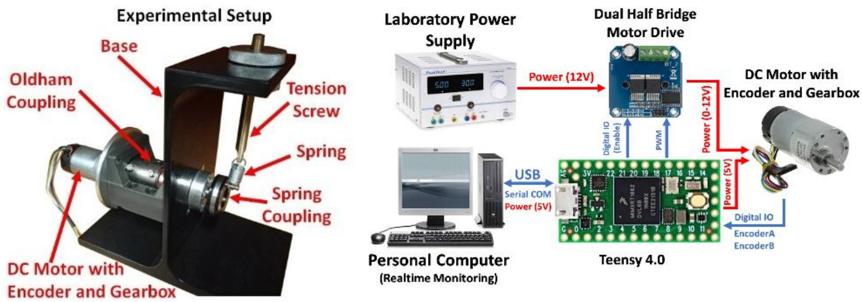

Figure 1

Experimental setup with a DC motor: mechanical design (left), control electronics (right)

The measurements were carried out with different springs and adaptive parameter (A_c) settings. For comparison purposes, the following control performance indices were introduced:

- Maximum absolute tracking error: $e_{max} = max_{i=1,2,..,N}(|e(i)|)$
- Average absolute tracking error: $\mu_e = \frac{1}{N}\sum_{i=1}^{N}|e(i)|$
- Standard deviation of the trajectory tracking error:

$$\sigma_e = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(|e(i)| - \mu_e)^2}$$

Some measurements results are presented in Fig. 2, for a DC motor without convergence acceleration. In this application the motor must track some sinusoid trajectory as precisely as possible without any load. As it is presented by the position measurement and phase trajectory, after short settling time, the tracking is precise. The quantization error in position measurement gives rise to substantial noise components in the higher derivatives however, it is effectively attenuated by the applied filter without significant delay.

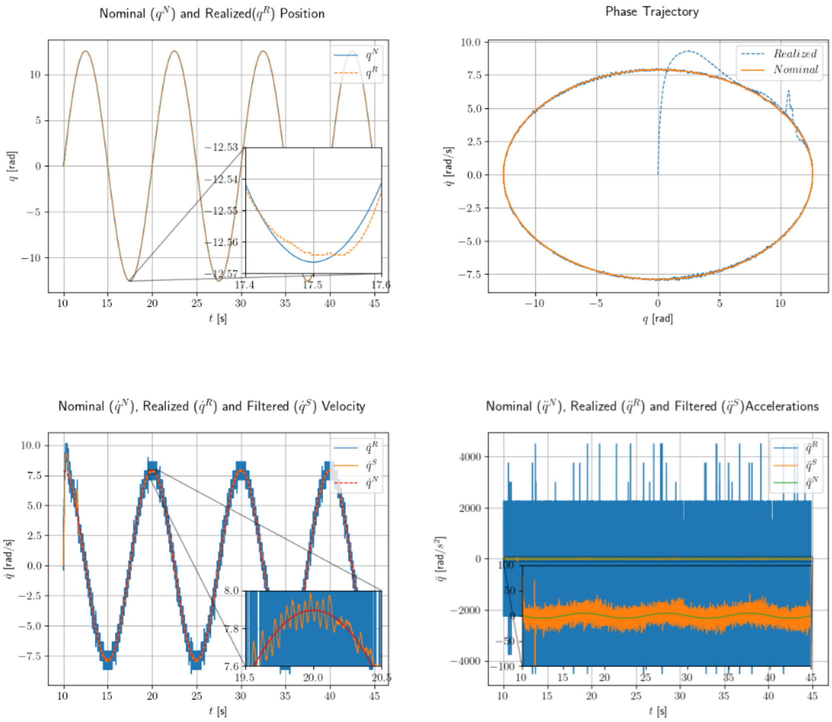

Figure 2

Measurement Results for DC motor without load – without convergence acceleration

The effect of the adaptive deformation is presented in Fig 3. where, significant difference can be observed in the deformed and desired signals. The controller effectively compensates for modelling imprecisions, the $\ddot{q}^{Des}(t)$ trajectory nicely follows the nominal prescription $\ddot{q}^{N}(t)$. For comparison purposes the trajectory tracking error for the RFPT and the Accelerated RFPT (with Wegstein) is presented here as well. The adaptive parameters were tuned with trial-and-error method trying to achieve the best control performance for both control scenarios. After a settling time of approximately $t_s < 3$s, precise trajectory tacking was achieved with maximum absolute trajectory remaining under 0.006rad despite the 0.00075rad encoder resolution. At the beginning of the control process, slightly less overshoot can be observed for the Accelerated RFPT controller. However, the quantitative comparison in Table I indicates that the standard RFPT controller performed marginally better overall.

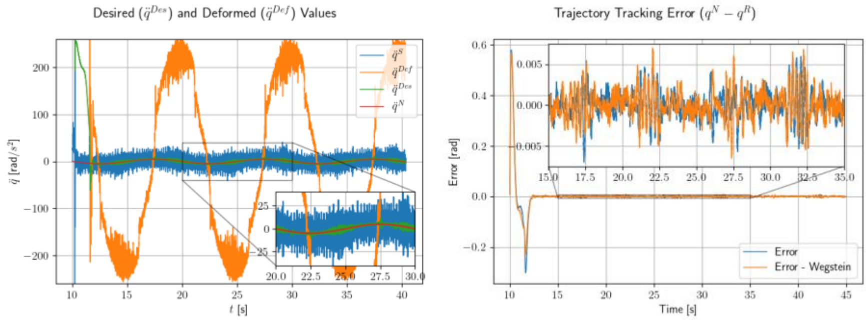

Figure 3

The effect of adaptive deformation (left) and trajectory tracking error comparison (right) – Adaptive Parameter setting for RFPT: $A_c = \frac{0.4}{K_c}$, Accelerated RFPT: $A_c = \frac{0.3}{K_c}$

Table I

Control Performance Indices

	e_{max}	μ_e	σ_e
RFPT (Spring: -)	0.00702	0.00133	0.00111
Accelerated RFPT (Spring: -)	0.00784	0.00164	0.00124
RFPT (Spring: D_1)	0.03794	0.00387	0.00388
Accelerated RFPT (Spring: D_1)	0.03140	0.00364	0.00361
RFPT (Spring: D_2)	0.24170	0.01766	0.03338
Accelerated RFPT (Spring: D_2)	0.16466	0.01168	0.02075

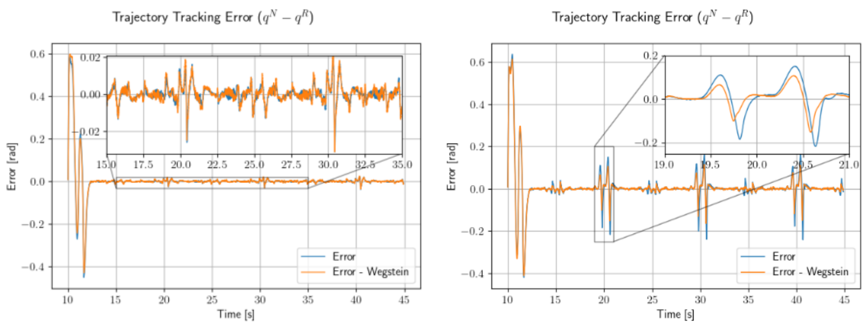

Figure 4

Trajectory tracking error comparison for DC motor with spring load D_1 (left) $<$ D_2(right) stiffness – Adaptive Parameter setting for RFPT: $A_c = \frac{0.4}{K_c}$, Accelerated RFPT: $A_c = \frac{0.3}{K_c}$

Additional experiments were conducted using the same DC motor loaded with springs of varying, but unknown, stiffness, that is $D_1 < D_2$. The results are presented in Fig. 4 which show the trajectory tracking error for the RFPT and Accelerated RFPT controller. In the case of the spring with D_2 stiffness, the motor experienced overloading, which resulted in the divergence of the iteration and poor tracking performance. Nevertheless, the accelerated controller recovered faster with shorter transient phase and less overshoot. This is further supported by the data presented in Table I. Alternatively, the Accelerated RFPT controller improved marginally in case of the softer spring with D_1 stiffness.

Conclusions

In this paper, the application of Wegstein's convergence acceleration method was investigated, in the domain of RFPT-based adaptive control. The proposed solution for online tuning of the relaxation parameter, based on Wegstein’s idea, resulted in increased trajectory performance, in some control scenarios. Specifically, less overshoot and shorter transient behavior was observed, when an overload occurred in the control regime. However, it seems that in most scenarios, by the proper tuning of the adaptive parameters of the RFPT controller, sufficient tracking performance can be achieved and the use of convergence acceleration is unnecessary, especially since, the process of online tuning of the Wegstein parameter through difference estimation is challenging. This is due to the implementation of fixed-point iteration on the second-order derivatives of the generalized coordinates, which are already significantly influenced by estimation noise.

Future investigations can be made for other implementations of FPI control method.

Acknowledgements

This work was supported by the 2024-2.1.1 University Research Scholarship Program of the Ministry for Culture and Innovation from the National Research, Development and Innovation Fund. We would like to acknowledge the support of this work by the Doctoral School of Applied Informatics and Applied Mathematics of Óbuda University.

References

[1] S. Banach, “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) *Fundamenta Mathematicae*, Vol. 3, pp. 133-181, 1922

[2] W. R. Mann, “Mean value methods in iteration,” *Proceedings of the American Mathematical Society*, Vol. 4, No. 3, pp. 506-506, Mar. 1953

[3] S. Ishikawa, “Fixed points by a new iteration method,” *Proceedings of the American Mathematical Society*, Vol. 44, No. 1, pp. 147-150, May 1974

[4] A. C. Aitken, "XXV.—On Bernoulli's Numerical Solution of Algebraic Equations," *Proceedings of the Royal Society of Edinburgh*, Vol. 46, pp. 289-305, Jan. 1927

[5] J. F. Steffensen, "Remarks on iteration.," *Scandinavian Actuarial Journal*, Vol. 1933, No. 1, pp. 64-72, Jan. 1933

[6] J. H. Wegstein, "Accelerating convergence of iterative processes," *Communications of the ACM*, Vol. 1, No. 6, pp. 9-13, Jun. 1958

[7] D. G. Anderson, "Iterative Procedures for Nonlinear Integral Equations," *Journal of the ACM*, Vol. 12, No. 4, pp. 547-560, Oct. 1965, doi: https://doi.org/10.1145/321296.321305

[8] Z. Wang, C. Hu, and Y. Zhu, "Accelerated Iteration Algorithm Based Contouring Error Estimation for Multiaxis Motion Control," *IEEE/ASME Transactions on Mechatronics*, Vol. 27, No. 1, pp. 452-462, Feb. 2022, doi: https://doi.org/10.1109/tmech.2021.3066188

[9] J. Zhang, Y. Yao, and B. Deng, "Fast and Robust Iterative Closest Point," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1-1, 2021, doi: https://doi.org/10.1109/tpami.2021.3054619.

[10] D. Gilton, G. Ongie, and R. Willett, "Deep Equilibrium Architectures for Inverse Problems in Imaging," *IEEE Transactions on Computational Imaging*, Vol. 7, pp. 1123-1133, 2021, doi: https://doi.org/10.1109/tci.2021.3118944

[11] S. Abrol and C. M. Hilton, "Modeling, simulation and advanced control of methanol production from variable synthesis gas feed," *Computers & Chemical Engineering*, Vol. 40, pp. 117-131, May 2012, doi: https://doi.org/10.1016/j.compchemeng.2012.02.005

[12] Jacopo De Tommaso, F. Rossi, N. Moradi, C. Pirola, G. S. Patience, and F. Galli, "Experimental methods in chemical engineering: Process simulation," The Canadian Journal of Chemical Engineering, Vol. 98, No. 11, pp. 2301-2320, Oct. 2020, doi: https://doi.org/10.1002/cjce.23857

[13] J. K. Tar, J. F. Bitó, László Nádai, and A. Tenreiro, "Robust fixed point transformations in adaptive control using local basin of attraction," *Acta Polytechnica Hungarica*, Vol. 6, No. 1, pp. 21-38, Dec. 2009

[14] K. Kosi and A. Awudu, "Increasing The Convergence Speed of Robust Fixed Point Transformaton-based Adaptive Control by Steffensen's Method in SISO Case," *2022 IEEE 20th Jubilee International Symposium on Intelligent Systems and Informatics (SISY 2022)*, pp. 285-290, Sep. 2022, doi: https://doi.org/10.1109/sisy56759.2022.10036260

[15] Awudu Atinga, Krisztián Kósi, and J. K. Tar, "Improving the Tracking Precision of a Robust Sliding Mode Controller Using Steffensen's Convergence Accelerator Strategy," i*n 2024 IEEE 18th International*

Symposium on Applied Computational Intelligence and Informatics (SACI 2024), pp. 000285-000290, May 2024, doi: https://doi.org/10.1109/saci60582.2024.10619830

[16] Awudu Atinga, Krisztián Kósi, and J. K. Tar, "Multivariable steffensen's accelerator in adaptive sliding mode control," *Acta Polytechnica Hungarica*, Vol. 21, No. 10, p. 413-438, 2024

[17] B. Varga, Richárd Horváth, and J. K. Tar, "Improved Adaptation Speed of the Robust Fixed Point Transformation-based Controller Using Virtual Response Forecast," *Acta Polytechnica Hungarica*, Vol. 22, No. 3, pp. 121-143, Jan. 2025, doi: https://doi.org/10.12700/aph.22.3.2025.3.7

[18] M. Stefanoni, Dominik Csík, P. Sarcevic, Ákos Odry, and J. K. Tar, "Adaptive Control Lyapunov Function-Based Controller Using Fixed Point Iteration," in 2024 IEEE 18th International Symposium on Applied Computational Intelligence and Informatics (SACI 2024), pp. 000479-000484, May 2024, doi: https://doi.org/10.1109/saci60582.2024.10619780

[19] M. Merei and J. K. Tar, "Adaptive Backstepping Control Design for Nonlinear System," in 2023 IEEE 17th International Symposium on Applied Computational Intelligence and Informatics (SACI 2023), pp. 000147-000152, May 2023, doi: https://doi.org/10.1109/saci58269.2023.10158606

[20] Adrienn Dineva, J. K. Tar, A. Varkonyi-Koczy, and V. Piuri, "Adaptive control of underactuated mechanical systems using improved 'Sigmoid Generated Fixed Point Transformation' and scheduling strategy," in 2016 IEEE 14th International Symposium on Applied Machine Intelligence and Informatics (SAMI 2016), pp. 193-197, Jan. 2016, doi: https://doi.org/10.1109/sami.2016.7423006

[21] H. Issa and J. K. Tar, "Noise Sensitivity Reduction of the Fixed Point Iteration-based Adaptive Control," *2021 IEEE 19th International Symposium on Intelligent Systems and Informatics (SISY 2021)*, pp. 171-176, Sep. 2021, doi: https://doi.org/10.1109/sisy52375.2021.9582514

[22] B. Csanádi, P. Galambos, J. K. Tar, G. Györök and A. Serester, "A Novel, Abstract Rotation-Based Fixed Point Transformation in Adaptive Control," *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC 2018)*, Miyazaki, Japan, 2018, pp. 2577-2582, doi: 10.1109/SMC.2018.00441