Application of Heavy and Underestimated Dynamic Models in Adaptive Receding Horizon Control Without Constraints

Awudu Atinga
Doctoral School of Applied Informatics and Applied Mathematics
Óbuda University
Budapest, Hungary
UGN9N8.uni-obuda.hu@stud.uni-obuda.hu

József K. Tar
ABC iRob, University Research and Innovation Center
Inst. of Appl. Math., John von Neumann Faculty of Informatics
Óbuda University
Budapest, Hungary
tar.jozsef@nik.uni-obuda.hu

***Abstract*—In the heuristic *“Adaptive Receding Horizon Controller”* (ARHC) the available dynamic model of the controlled system usually is placed in the role of a constraint under which various cost functions can be minimized over a horizon. A possible secure design can be making calculations for a “heavy dynamic model” that may produce high dynamical burden that is efficiently penalized by the cost functions and instead of the original nominal trajectory results a “deformed” one that can be realized by the controlled system of “less heavy dynamics”. In the lack of accurate system model a fixed point iteration-based adaptive approach is suggested for the precise realization of this deformed trajectory. To reduce the computational burden of the control the usual approach in which the dynamic model is considered as constraint and Lagrange-multipliers are introduced as co-state variables is evaded. The heavy dynamic model is directly built in the cost and the computationally greedy Reduced Gradient Algorithm is replaced by a transition between the simple and fast Newton-Raphson and the slower Gradient Descent algorithms (GDA). In the paper simulation examples are presented for two dynamically coupled van der Pol oscillators as a strongly nonlinear system. The comparative use of simple non-differentiable and differentiable cost functions is considered, too.**

***Index Terms*—Newton-Raphson Algorithm, Gradient Descent Algorithm, Reduced Gradient Algorithm, Receding Horizon Control, Fixed Point Iteration-based Adaptive Control**

I. INTRODUCTION

This paper is a further developed, extended version of the conference publication [1] in which preliminary calculations were made for two coupled van der Pol oscillators. The originally electrical system (an externally excited triode in [2]) was “transformed” into a mechanical one that allowed working with more convenient physical concepts and units.

The scientific antecedents can be briefly summarized as follows. The idea of optimal controllers can be considered as a generalization of the variational principles of Classical Mechanics in which functionals are minimized. The subject area can be related to the action functional generalized by Bellman as the Hamilton-Jacobi Bellman equation that in the advent of the appearance of powerful computers resulted the idea of “Dynamic Programming” [3], [4]. However, the computational power of the processors even in the beginning of the nineties of the past century was not satisfactory for this purpose in robot control where fast motion was considered. The optimal control framework was completely evaded in the “Computed Torque Control” [5] in which the dynamic model was directly use for the computation of the necessary control forces. In the seventies, to reduce the computational burden of dynamic programming the idea of the “Receding Horizon Controller” was introduced in [6], in which the cost terms are computed in discrete points of a finite horizon length, they are summarized, and the dynamic model’s output (certain integer order derivative of the generalized coordinate of the controlled system) are estimated as finite element approximations over this discrete grid. Evidently, the resolution of the grid had to be fine enough to mathematically underpin this approach. The constraint terms were considered as relationships between the neighboring grid points and Lagrange’s Reduced Gradient method [7] was applied for the cost minimization.

Certain special cases of this method lead to very popular approaches. In the case of Linear Time-invariant dynamic models and quadratic cost terms the occurring terms can be even “formally treated”: based on Riccati’s observation in 1724 [8] according to which certain second order differential equation’s solution was constructed by solving first order differential equations, and on Schur’s lemma that made it possible to tackle quadratic matrix problems with linear ones [9], the idea of “Linear Quadratic Regulator” was developed [10]. These solutions are built into efficient MATLAB codes and the practice of transforming the real problems to such approximate form and solving them by the use of appropriate MATLAB packages was announced by Boyd et al. in 1994 in [11].

Even in the simplified, time grid-based formalism the Lagrange multipliers used for gradient reduction maintain their formal property, i.e., they are counterparts of the canonical

We acknowledge the support of this work by the Doctoral School of Applied Informatics and Applied Mathematics of Óbuda University.

Cite as: A. Atinga and J. K. Tar, “Application of Heavy and Underestimated Dynamic Models in Adaptive Receding Horizon Control Without Constraints”, *Syst. Theor. Control Comput. J.,* vol. 2, no. 2, pp. 1–8, Dec. 2022.
DOI: 10.52846/stccj.2022.2.2.36

momentum coordinates in the Hamiltonian equations of motion he introduced in 1834 [12], [13] from which strict analogy with the flow of incompressible fluids can be deduced (e.g., [14]) together with its mathematical consequences.

However, by keeping in mind the fact that in the original problem statement the co-state variables do not appear, attempts were made for the elimination of their computation. At first, instead of solving the usual set of linear equations used for gradient reduction in EXCEL's Solver package [15], [16], the simpler Gram-Schmidt method [17], [18] –earlier also invented by Laplace [19]–was suggested in [20]. As further computational reduction possibility, instead using the separate $\{g^{(i)}(x) = 0; i = 1, 2, \dots, K\}$ constraint terms with their associated Lagrange multipliers a single term $G(x) := \sum_\ell {g^{(\ell)}}^2 = 0$ was applied with which only a single Lagrange multiplier was associated. In [21] the dynamic model was not treated as a constraint term. Instead of that it was directly built in the cost function belonging to the whole horizon. Consequently no gradient reduction was necessary, and the way was opened for using the more efficient Newton-Raphson algorithm [22], [23].

In this paper and in its immediate predecessor [1] the same method is applied for the adaptive RHC control of two coupled van der Pol oscillators. In the computations different nominal trajectories were investigated. Besides the original "fragmented linear" cost functions their differentiable approximation was also investigated. Furthermore, for speeding up the calculations a new stopping condition was built in the system because it was experimentally observed that the original construction sometimes was apt to spend too much time in certain points. Simply the number of the numerical steps was limited.

In the sequel at first the dynamic model of the two coupled oscillators is given (it is identical to that used in [1]).

II. The Controlled System

In the simulations three different parameter settings were applied for the same dynamic model, according to Tables I and II.

TABLE I
THE DYNAMIC MODEL PARAMETERS USED IN THE SIMULATIONS.

Parameter	*Unit*	*Value*
m_1, "exact" mass	[kg]	1.0
$\hat{m}_1$, "heavy" mass	[kg]	$2m_1$
$\check{m}_1$, "approx." mass	[kg]	$0.8m_1$
m_2, "exact" mass	[kg]	2.0
$\hat{m}_2$, "heavy mass"	[kg]	$2m_2$
$\check{m}_2$, "approx." mass	[kg]	$0.9m_2$

Based on [2] the dynamic model is given in (1). Its main format is compatible with the dynamic equations of robots having the structure $H(q)\ddot{q} + h(q, \dot{q}) = Q$. The originally electrical system (an externally excited triode) has been transformed into a 2 DoF "mechanical" system with *generalized coordinates* q_1, q_2, *generalized forces* $Q_1 \equiv F_1$, $Q_2 \equiv F_2$, and parameters given in Table I. Parameters a_1 and a_2 separate the zones of damping and excitation. Evidently, for certain coordinate values the system is excited, and for others it is damped, depending on the signs of the terms $\left(a_1^2 - q_1^2\right)$ and $\left(a_1^2 - (q_2 - L_2)^2\right)$.

TABLE II
THE DYNAMIC MODEL PARAMETERS USED IN THE SIMULATIONS.

Parameter	*Unit*	*Value*
k_1, "exact" spring stiffness	$[\mathrm{N \cdot m^{-1}}]$	100.0
$\hat{k}_1$, "heavy" spring stiffness	$[\mathrm{N \cdot m^{-1}}]$	$1.2k_1$
$\check{k}_1$, "approx." spring stiffness	$[\mathrm{N \cdot m^{-1}}]$	$0.8k_1$
k_2, "exact" spring stiffness	$[\mathrm{N \cdot m^{-1}}]$	150.0
$\hat{k}_2$, "heavy" spring stiffness	$[\mathrm{N \cdot m^{-1}}]$	$1.5k_2$
$\check{k}_2$, "approx." spring stiffness	$[\mathrm{N \cdot m^{-1}}]$	$0.7k_2$
b_1, "exact" excitation coeff.	$[\mathrm{N \cdot m^{-3} \cdot s}]$	1.0
$\hat{b}_1$, "heavy" excitation coeff.	$[\mathrm{N \cdot m^{-3} \cdot s}]$	$1.5b_1$
$\check{b}_1$, "approx." excitation coeff.	$[\mathrm{N \cdot m^{-3} \cdot s}]$	$0.8b_1$
b_2, "exact" excitation coeff.	$[\mathrm{N \cdot m^{-3} \cdot s}]$	1.50
$\hat{b}_2$, "heavy" excitation coeff.	$[\mathrm{N \cdot m^{-3} \cdot s}]$	$1.25b_2$
$\check{b}_2$, "approx." excitation coeff.	$[\mathrm{N \cdot m^{-3} \cdot s}]$	$0.7b_2$
c_1, "exact" visc. damping	$[\mathrm{N \cdot m^{-1} \cdot s}]$	1.0
$\hat{c}_1$, "heavy" visc. damping	$[\mathrm{N \cdot m^{-1} \cdot s}]$	$1.25c_1$
$\check{c}_1$, "approx." visc. damping	$[\mathrm{N \cdot m^{-1} \cdot s}]$	$0.8c_1$
c_2, "exact" visc. damping	$[\mathrm{N \cdot m^{-1} \cdot s}]$	1.50
$\hat{c}_2$, "heavy" visc. damping	$[\mathrm{N \cdot m^{-1} \cdot s}]$	$1.5c_2$
$\check{c}_2$, "approx." visc. damping	$[\mathrm{N \cdot m^{-1} \cdot s}]$	$0.9c_2$
d_1, "exact" turb. damping	$[\mathrm{N \cdot m^{-2} \cdot s^2}]$	1.0
$\hat{d}_1$, "heavy" turb. damping	$[\mathrm{N \cdot m^{-2} \cdot s^2}]$	$1.5d_1$
$\check{d}_1$, "approx." turb. damping	$[\mathrm{N \cdot m^{-2} \cdot s^2}]$	$0.8d_1$
d_2, "exact" turb. damping	$[\mathrm{N \cdot m^{-2} \cdot s^2}]$	1.50
$\hat{d}_2$, "heavy" turb. damping	$[\mathrm{N \cdot m^{-2} \cdot s^2}]$	$1.25d_2$
$\check{d}_2$, "approx." turb. damping	$[\mathrm{N \cdot m^{-2} \cdot s^2}]$	$0.7d_2$
a_1, "exact" separator	[m]	3.0
$\hat{a}_1$, "heavy" separator	[m]	$1.5a_1$
$\check{a}_1$, "approx." separator a_2, "exact" separator	[m] [m]	$0.9a_1$ 4.0
$\hat{a}_2$, "heavy" separator	[m]	$1.25a_2$
$\check{a}_2$, "approx." separator	[m]	$0.8a_2$
L_2, "exact" shift	[m]	3.0
$\hat{L}_2$, "heavy" shift	[m]	$0.8L_2$
$\check{L}_2$, "approx." shift	[m]	$1.2L_2$
k, "exact" coupling stiffness	$[\mathrm{N \cdot m^{-\sigma}}]$	200.0
$\hat{k}$, "heavy" coupling stiffness	$[\mathrm{N \cdot m^{-\sigma}}]$	$1.5k$
$\check{k}$, "approx." coupling stiffness	$[\mathrm{N \cdot m^{-\sigma}}]$	$0.8k$
L, "exact" coupling length	[m]	0.5
$\hat{L}$, "heavy" coupling length	[m]	$0.9L$
$\check{L}$, "approx." coupling length	[m]	$1.3L$
σ "exact" nonlinearity	[nondimensional]	1.50
$\hat{\sigma}$ "heavy" nonlinearity	[nondimensional]	1.2σ
$\check{\sigma}$ "approx." nonlinearity	[nondimensional]	0.9σ

$$
\begin{aligned}
&m_1\ddot{q}_1 + k_1 q_1 - b_1\left(a_1^2 - q_1^2\right)\dot{q}_1 + \\
&c_1\dot{q}_1 + d_1 \mathrm{sign}(\dot{q}_1)\dot{q}_1^2 \\
&- k\mathrm{sign}(q_2 - q_1 - L)|q_2 - q_1 - L|^\sigma = F_1 \\
&m_2\ddot{q}_2 + k_2(q_2 - L_2) - b_2\left(a_1^2 - (q_2 - L_2)^2\right)\dot{q}_2 + \\
&c_2\dot{q}_2 + d_2 \mathrm{sign}(\dot{q}_2)\dot{q}_2^2 + \\
&k\mathrm{sign}(q_2 - q_1 - L)|q_2 - q_1 - L|^\sigma = F_2
\end{aligned}
\tag{1}
$$

Parameters c_1 and c_2 correspond to the usual *viscous damping* that is typical for low velocities, while d_1 and d_2

belong to the drag force that normally is generated during fast motion in turbulent gases or liquids. The oscillators are coupled by a spring of zero force length L, and $\sigma > 1$ "nonlinearity parameter" according to which the differential stiffness of the coupling spring varies with its compression or dilatation. This system is evidently burdened with strong nonlinearities therefore it can serve as a good paradigm for our investigations.

III. RHC Without Gradient Reduction

The dynamic model in (1) is considered in a *function format* as

$$\begin{aligned}\ddot{q}_1(t_i) &= \mathfrak{f}_1(q_1(t_i), q_2(t_i), \dot{q}_1(t_i), \dot{q}_2(t_i), F_1(t_i), F_2(t_i)),\\ \ddot{q}_2(t_i) &= \mathfrak{f}_2(q_1(t_i), q_2(t_i), \dot{q}_1(t_i), \dot{q}_2(t_i), F_1(t_i), F_2(t_i))\end{aligned} \quad (2)$$

for each point of the horizon. The *initial conditions*, furthermore $F_1(t_1)$, and $F_2(t_1)$ determine $\ddot{q}_1(t_1)$ and $\ddot{q}_2(t_1)$. Based on the possible interpretation of the forward differences, this determines $\dot{q}_1(t_2) = \dot{q}_1(t_1) + \Delta t \ddot{q}_1(t_1)$, and $\dot{q}_2(t_2) = \dot{q}_2(t_1) + \Delta t \ddot{q}_2(t_1)$. Again using the forward differences it is obtained that $q_1(t_3) = q_1(t_2) + \Delta t \dot{q}_1(t_2)$, and $q_2(t_3) = q_2(t_2) + \Delta t \dot{q}_2(t_2)$. Therefore, the initial conditions and the forces in the first grid point determine $q_1(t_3)$ and $q_2(t_3)$. Via continuing this calculation with $F_1(t_2)$ and $F_2(t_2)$ the values $q_1(t_4)$ and $q_2(t_4)$ can be computed, etc. Therefore, the force components $\{F_1(t_1), \dots, F_1(t_{N-2})\}$, and $\{F_2(t_1), \dots, F_2(t_{N-2})\}$ determine the new coordinates $\{q_1(t_3), \dots, q_1(t_N)\}$, and $\{q_2(t_3), \dots, q_2(t_N)\}$. The cost function of optimization may have the form of

$$\begin{aligned}&\Psi\left(F_1(t_1), \dots, F_1(t_{N-2}), F_2(t_1), \dots, F_2(t_{N-2})\right) =\\ &= \sum_{\ell=1}^{N-2} \left(\psi_{q1}\left(q_1^N(t_{\ell+2}), q_1^o(t_{\ell+2})\right) + \right.\\ &\left. + \psi_{q2}\left(q_2^N(t_{\ell+2}), q_2^o(t_{\ell+2})\right)\right) +\\ &+ \sum_{\ell=1}^{N-2} \left(\psi_{F1}\left(F_1(t_\ell)\right) + \psi_{F2}\left(F_2(t_\ell)\right)\right) \quad ,\end{aligned} \quad (3)$$

in which the contributions $\{\psi_{q1}\left(q_1^N(t_{\ell+2}), q_1^o(t_{\ell+2})\right)\}$ denote the cost contributions for the tracking error of variable q_1, $\{\psi_{q2}\left(q_1^N(t_{\ell+2}), q_1^o(t_{\ell+2})\right)\}$ mean similar terms for tracking the coordinate q_2, $\{\psi_{F1}\left(F_1(t_\ell)\right)\}$ and $\{\psi_{F2}\left(F_2(t_\ell)\right)\}$ are the penalty contributions for the applied control forces. The optimized trajectories $\{q_1^o(t_\ell)\}$, $\{q_2^o(t_\ell)\}$, their time-derivatives $\{\dot{q}_1^o(t_\ell)\}$, $\{\dot{q}_2^o(t_\ell)\}$, and second time-derivatives $\{\ddot{q}_1^o(t_\ell)\}$, $\{\ddot{q}_2^o(t_\ell)\}$ are built up of the initial conditions and the control forces. The ψ_{q1}, ψ_{q2}, ψ_{F1}, and ψ_{F2} functions may have various forms, they can differ from each other even in the different grid points, too. On this reason the heuristic RHC method obtains a high degree of flexibility.

For the minimization of (3) in principle the GDA algorithm can be used via calculating $\nabla\Psi$. Its success evidently may depend on the structure of the cost functions.

Instead of the complicated cost functions of [21], the simple ones of common shape were applied for both trajectory tracking, and force limitation. The functions had the *"width parameter"* ($w_q > 0$ for trajectory tracking, and $w_F > 0$ for the force limitation, respectively), and linear increase or *"steepness parameter"* ($s_q > 0$ for trajectory tracking, and $s_F > 0$ for force limitation, respectively), as follows:

$$\psi(x) = \begin{cases} \psi = -s(x+w) \text{ if } x \le -w \\ \psi = 0 \text{ if } -w < x \le w \\ \psi = s(x-w) \text{ if } w < x \end{cases} \quad . \quad (4)$$

This function can be so interpreted that small values, i.e., that for which $|x| \le w$ are tolerated without causing any cost contribution, but the terms with greater absolute values generate finite, nonzero contribution. The linearity in cost generation is expected to evade the occurrence of numerical overflow problems. In the range of the occurring numerical values *smoothed version* of the above cost function in the form

$$\Psi = \mathfrak{a}|x|^{\mathfrak{n}} \quad (5)$$

was also used for the sake of comparison. The new parameters were so fitted that at $x = 2w$ the new function value had to be equal to the original resulting

$$\begin{aligned}&\mathfrak{a}(2w)^{\mathfrak{n}} = s(2w - w) \text{ resulting}\\ &\mathfrak{a} = \frac{sw}{(2w)^{\mathfrak{n}}} \quad .\end{aligned} \quad (6)$$

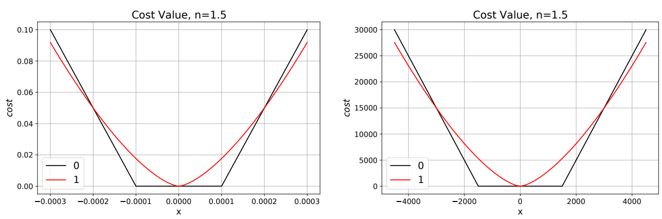

Fig. 1. The original and smoothed cost functions for trajectory tracking, $\mathfrak{n} = 1.5$ (LHS: $s_q = 500\text{m}^{-1}$, $w_q = 10^{-4}\text{m}$) and for force limitation (RHS: $s_F = 10.0\text{N}^{-1}$, and $w_F = 1500.0\text{N}$)

The next question is how to speed up the classic *"Gradient Descent Algorithm"* for finding the local minima. Normally the procedure is stopped when the reduced gradient becomes "zero". However, in a numerical solution, during finite time, only some "approximation of 0" can be achieved, on which the time-need of the method can drastically depend. Here the method suggested in [21] was applied.

In the first step it was assumed, that –as in the case of the Newton-Raphson Algorithm (e.g., [23])– that in a singe step the value of zero as absolute minimum of the error can be achieved. This corresponds to a step $-\alpha\beta\nabla\psi(x)$ in which $\beta\|\nabla\psi(x)\|^2 = \psi(x)$ with a starting value of $\alpha = 1$. This step-length was maintained while the condition $\psi(x(n+1)) < \psi(x(n))$ was met. If $\psi(x(n+1) \ge \psi(x(n)))$ happened, the positive parameter α was halved. This procedure was repeated until the last value of α achieved one tenth of its initial value. Then the procedure was stopped and the so found value x was accepted. Since later it was experimentally observed that this solution sometimes can "stick in" in certain point, maximum number of allowed steps was limited to 100.

Since the so optimized trajectory $q^o(t)$ can be quite noisy, according to an idea borrowed from [24], it was smoothed/filtered by a simple low pass filter by tracking it according to the equation

$$\left(\Lambda_f + \frac{\mathrm{d}}{\mathrm{d}t}\right)^3 q^{of}(t) = \Lambda_f^3 q^o(t) \tag{7}$$

with the initial conditions $q^{of}(t_0) = 0$, $\dot{q}^{of}(t_0) = 0$, and $\ddot{q}^{of}(t_0) = 0$. Following that a kinematically designed fixed point iteration-based tracking was planned for the actual trajectory $q(t)$ as

$$\left(\Lambda + \frac{\mathrm{d}}{\mathrm{d}t}\right)^3 \int_{t_0}^{t} \left(q^{of}(\xi) - q(\xi)\right) \mathrm{d}\xi \equiv 0\,. \tag{8}$$

The adaptive controller tried to realize (8) according to the principles published in [25]. This approach is based on the idea of the *"response function"*, that in the case of a second order system takes the fact into consideration that in the given physical state of the system $\{q(t), \dot{q}(t)\}$, the actual control force $F(t)$, **immediately determines the realized 2nd time-derivative** as $\ddot{q}(t) = \mathfrak{F}\left(q(t), \dot{q}(t), F(t)\right)$, in which the force is computed by using the *available approximate inverse dynamic model* as $F(t) = \hat{\mathfrak{F}}^{-1}\left(q(t), \dot{q}(t), \ddot{q}^{Des}(t)\right)$. Altogether this results a function in the form $\ddot{q}(t) = \mathfrak{R}\left(q(t), \dot{q}(t), \ddot{q}^{Des}(t)\right)$ in which $\ddot{q}^{Des}(t)$ can be very quickly (even abruptly) varied, while its other arguments, i.e., $q(t)$ and $\dot{q}(t)$ vary only slowly, therefore approximately it can be stated that $\ddot{q}(t) \approxeq \mathfrak{R}\left(\ddot{q}^{Des}(t)\right)$, i.e., the slowly varying parts are approximated as parameters. According to this approach, in the case of a digital controller, during one control cycle only one step of modification is possible in the input argument $q^{Des}(t)$. The basic idea is the construction of a deformation $q^{Des}(t) \mapsto q^{Def}(t)$ so that $\ddot{q}^{Des}(t) = \mathfrak{R}\left(\ddot{q}^{Def}(t)\right)$. Evidently, in the case of a strongly nonlinear system, especially in the lack of knowledge on the *exact model parameters*, this deformation cannot be computed in a single step because even the exact form of the function $\mathfrak{R}\left(\ddot{q}(t)\right)$ is unknown, too. However, by observing the appropriate input – output pairs, the behavior of this function can be experimentally observed in similar manner as a car driver observes and learns the behavior of a different car. Let Δx be a small variation in the input argument for which the variation of the output will be

$$\Delta\mathfrak{R} := \mathfrak{R}(x + \Delta x) - \mathfrak{R}(x) \approxeq \frac{\partial \mathfrak{R}(x)}{\partial x}\Delta x\ . \tag{9}$$

This function can be called *"approximately direction keeping"* if $\Delta\mathfrak{R}^T\Delta x > 0$, i.e., the angle between the two vectors is *acute*. Since for an arbitrary quadratic real matrix M can be decomposed as a sum of its symmetric and skew-symmetric parts

$$\begin{aligned} &\Delta x^T \frac{1}{2}\left[\left(M + M^T\right) + \left(M - M^T\right)\right]\Delta x = \\ &= \frac{1}{2}\Delta x^T\left(M + M^T\right)\Delta x\ , \end{aligned} \tag{10}$$

due to symmetry reasons only its symmetric part plays role in the direction keeping property. Let $x_\star$ be so chosen that the *desired goal*, i.e., g is the output of this function: $g = \mathfrak{R}(x_\star)$. Let $\alpha > 0$ a small positive number, and consider the sequence of points generated as $\{x_{n+1} = x_n + \alpha(g - x_n)\}$. For this sequence the following estimation can be done:

$$\mathfrak{R}(x_{n+1}) \approxeq \mathfrak{R}(x_n) - \alpha\frac{\partial\mathfrak{R}}{\partial x}\left(\mathfrak{R}(x_n) - g\right) \tag{11a}$$

$$\mathfrak{R}(x_{n+1}) - g \approxeq \left[I - \alpha\frac{\partial\mathfrak{R}}{\partial x}\right]\left(\mathfrak{R}(x_n) - g\right)\ . \tag{11b}$$

Evidently, if the matrix $M := \frac{\partial\mathfrak{R}}{\partial x}$ in (11b) is approximately direction keeping, for an arbitrary vector w it can be written that

$$\|(I - \alpha M)w\|^2 = w^T(I - \alpha M^T)(I - \alpha M)w = \tag{12a}$$

$$= \|w\|^2 - \alpha w^T\left(M^T + M\right)w + \alpha^2 w^T M^T M w\ , \tag{12b}$$

where in (12b) the first term is negative for an approximately direction keeping function, while the last one is always positive. Since the negative term is proportional to the small α, while the positive one is proportional to α^2, with a cautiously chosen α it can be achieved that $x_n \to x_\star$, and $\mathfrak{R}(x_n) \to \mathfrak{R}(x_\star) = g$. A mathematically more formal proof can be based on Banach's fixed point theorem [26]. Of course, the input values can be slightly adjusted to approach towards the goal in various manners than using a small parameter $\alpha > 0$. In this sequence generation method one cannot easily determine whether an *ad hoc* choice for α will be good enough, i.e., it will result at least a convergent sequence and that the convergence will be fast enough for the purposes of the controller. To reduce this burden in the design, in [25] a more stable design method was suggested. Each vector under consideration was so *augmented* by a physically not interpreted new dimension, that they obtained identical Frobenius norms. Consequently it became possible to rotate these vectors into each other with a full angle, or it was possible to move toward each other with an interpolated rotational angle. The interpolation happened by modifying the full angle of rotation. Evidently, the physically interpreted projections of the augmented vectors also moved toward each other. By choosing a large common norm the occurrence of only small angles of rotation can be guaranteed, and the interpolation factor can be easily set. The necessary rotations can be expressed in closed analytical form by the generalization of the Rodrigues formula [27].

IV. The Simulation Results

In the simulations acceptable discrete step length that is appropriate to the dynamics of the nominal trajectory to be tracked as well as to the additional dynamic burden of the controller's PID-type trajectory corrections must be determined. Instead complicate theoretical considerations simulations can be done for a stable system at time resolution Δt with digital horizon length H that must be compatible with that obtained for a resolution of $\Delta t/2$, $2H$. As a result of such simulations the control parameters given in Table III were obtained.

TABLE III
THE CONTROL PARAMETERS USED IN THE SIMULATIONS.

Parameter	***Unit***	***Value***
Δt, time-resolution	[s]	10^{-3}
H, horizon length in Δt units	[nondimensional integer]	12
w_q, tracking error tolerance	[m]	10^{-4}
s_q, tracking error steepness	m^{-1}	500.0
w_F, force tolerance	[N],	1500
s_F, force penalty steepness,	$[\mathrm{N}^{-1}]$	10^1
Λ_f, noise filtering parameter	$[\mathrm{s}^{-1}]$	100.0
Λ, adaptive tracking parameter	$[\mathrm{s}^{-1}]$	30.0
R_a, common augmented $\|\ddot{q}\|$	$[\mathrm{m}\cdot\mathrm{s}^{-2}]$	10^6
λ_a, adaptive interpolation factor	[nondimensional]	0.8

The impart of force limitation is examined using simulation results for both the original cost and the smoothed cost function. According to the simulation pair $w_F = 1500\,\mathrm{N}$, $s_F = 10\,\mathrm{N}^{-1}$ the Trajectory Tracking of the original cost function in Fig. 2 reveals that the controller follows the optimal trajectory with high amount of error. Therefore, it is well revealed that the force limiting cost contributions can corrupt the trajectory tracking precision for the overestimated heavy dynamic model that needs higher forces than the less heavy realistic one. However, the distorted optimized trajectory is well tracked by the adaptive controller, too. Fig. 6 reveals that the adaptive control forces remained in the reasonable order of magnitude and do not show hectic variation, due to the efficient noise filtering strategy applied for tracking the optimized trajectory. The tracking error that can be observed in the free of force limitation case mainly is generated by this simple low pass-type filter. Figs. 7, 8 and 9 clearly testify the efficiency of the adaptation mechanism: due to the considerable extent of the adaptive deformation the realized and the desired 2nd time-derivatives are in each other's close vicinity, due to which the kinematically designed tracking policy is quite precisely realized. Fig. 10 testifies that the drastic force limitation sometimes results in quite small computational time, but this effect is not even therefore its advantages can be realized mainly in offline applications.

The counterparts of Figs. 2, 3 and 4 that belong to the trajectory tracking of the smoothed cost function are described in Figs. 11, 12 and 13. It is clear that quite similar effects caused by the force limitation can observed in the case of a smoothed cost function as in the case of the original cost function one. Fig. 16 reveals that the fluctuation of the optimized control forces is considerably has been reduced due to the smooth nature of the cost function. Fig. 17 shows smoothly varying adaptive control forces and according to Fig 18 it can be stated that mechanism of adaptation worked well. too. In Fig. 19 it can observed that the computational time-need is more even in the case of the smooth cost function, so it does not allow to spare too much time in the offline applications. The computational need of the optimization was estimated for Julia language version `1.8.1 (2022-09-06)` running under `Linux 5.10.84-1-MANJARO x86_64 21.2.0 Qonos` on a `DELL inspiron 15R` laptop.

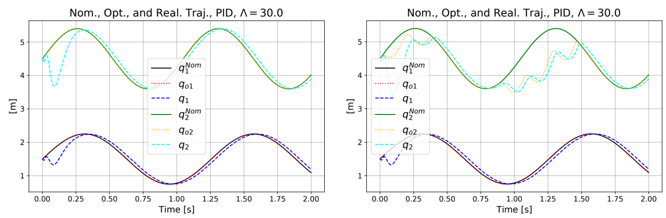

Fig. 2. Trajectory tracking of the original cost functions without (LHS) and with force limitation (RHS) with $w_F = 1500\,\mathrm{N}$ and $s_F = 10\,\mathrm{N}^{-1}$ (RHS)

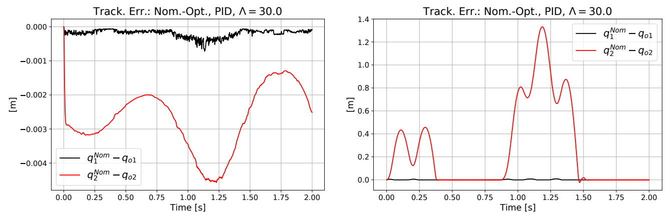

Fig. 3. The "nominal-optimal" trajectory tracking error of the original cost functions without (LHS) and with force limitation (RHS) with $w_F = 1500\,\mathrm{N}$ and $s_F = 10\,\mathrm{N}^{-1}$ (RHS)

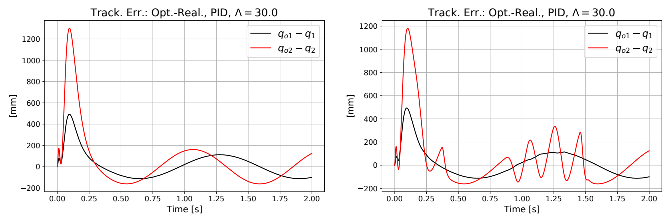

Fig. 4. The "optimal-realized" trajectory tracking error belonging to the original cost functions without (LHS) and with force limitation (RHS) with $w_F = 1500\,\mathrm{N}$ and $s_F = 10\,\mathrm{N}^{-1}$ (RHS)

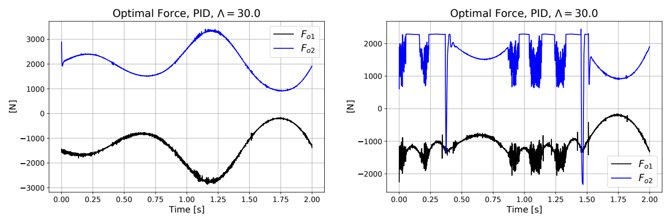

Fig. 5. The optimized control forces belonging to the original cost functions without (LHS) and with force limitation (RHS) with $w_F = 1500\,\mathrm{N}$ and $s_F = 10\,\mathrm{N}^{-1}$ (RHS)

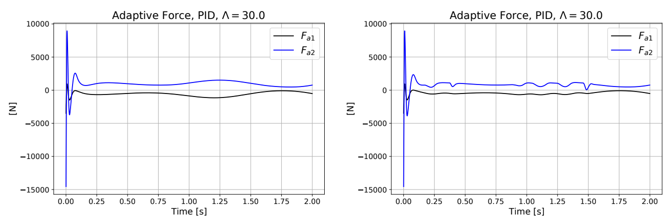

Fig. 6. The adaptive control forces belonging to the original cost functions without (LHS) and with force limitation (RHS) with $w_F = 1500\,\mathrm{N}$ and $s_F = 10\,\mathrm{N}^{-1}$ (RHS)

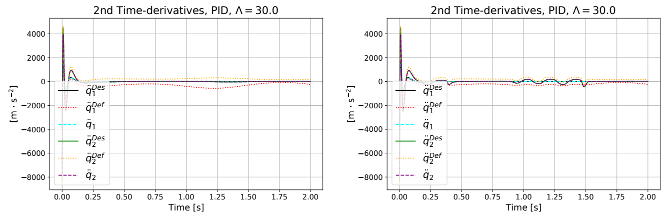

Fig. 7. The 2nd time derivatives belonging to the original cost functions without (LHS) and with force limitation (RHS) with $w_F = 1500\,\mathrm{N}$ and $s_F = 10\,\mathrm{N}^{-1}$ (RHS)

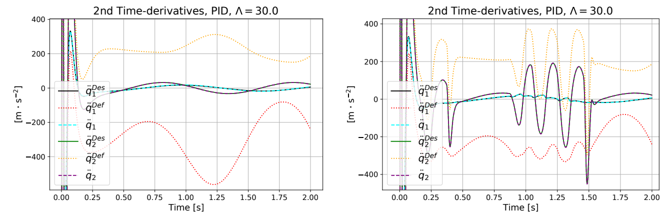

Fig. 8. The 2nd time derivatives belonging to the original cost functions without (LHS) and with force limitation (RHS) with $w_F = 1500\,\mathrm{N}$ and $s_F = 10\,\mathrm{N}^{-1}$ (RHS) – zoomed in excerpts

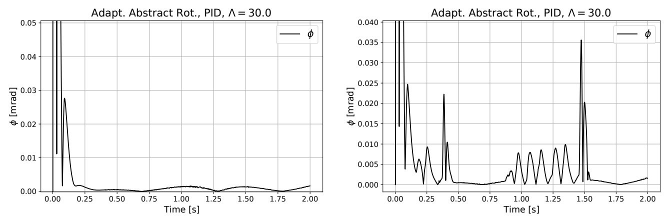

Fig. 9. The angle of the adaptive abstract rotations belonging to the original cost functions without (LHS) and with force limitation (RHS) with $w_F = 1500\,\mathrm{N}$ and $s_F = 10\,\mathrm{N}^{-1}$ (RHS) – zoomed in excerpts

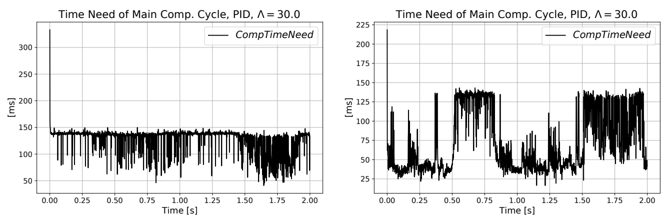

Fig. 10. The computational time of the main cycle belonging to the original cost functions without (LHS) and with force limitation (RHS) with $w_F = 1500\,\mathrm{N}$ and $s_F = 10\,\mathrm{N}^{-1}$ (RHS)

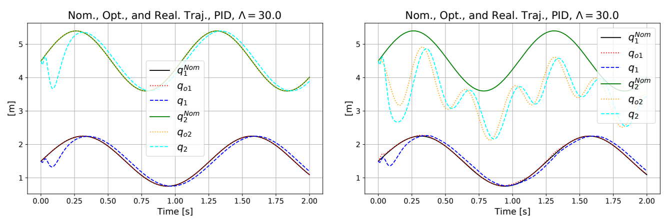

Fig. 11. Trajectory tracking of the smoothed cost functions without (LHS) and with force limitation (RHS) with $w_F = 1500\,\mathrm{N}$ and $s_F = 10\,\mathrm{N}^{-1}$ (RHS)

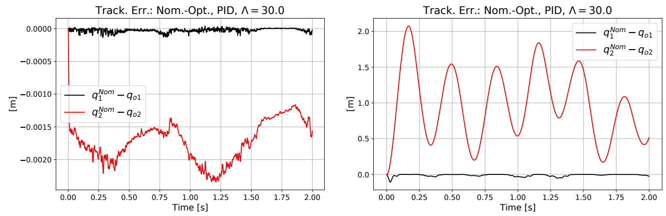

Fig. 12. The "nominal-optimal" trajectory tracking error of the smoothed cost functions without (LHS) and with force limitation (RHS) with $w_F = 1500\,\mathrm{N}$ and $s_F = 10\,\mathrm{N}^{-1}$ (RHS)

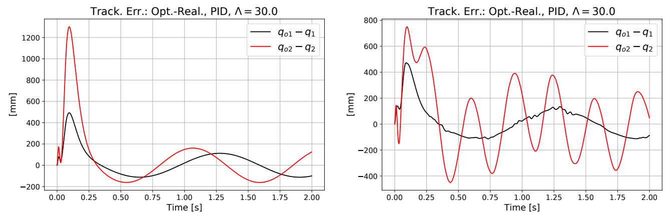

Fig. 13. The "optimal-realized" trajectory tracking error belonging to the smoothed cost functions without (LHS) and with force limitation (RHS) with $w_F = 1500\,\mathrm{N}$ and $s_F = 10\,\mathrm{N}^{-1}$ (RHS)

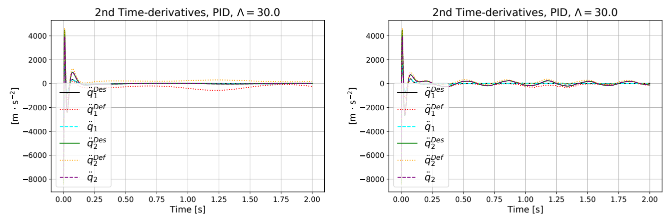

Fig. 14. The 2nd time derivatives belonging to the smoothed cost functions without (LHS) and with force limitation (RHS) with $w_F = 1500\,\mathrm{N}$ and $s_F = 10\,\mathrm{N}^{-1}$ (RHS)

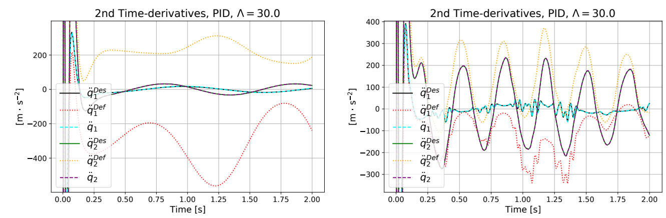

Fig. 15. The 2nd time derivatives belonging to the smoothed cost functions without (LHS) and with force limitation (RHS) with $w_F = 1500\,\mathrm{N}$ and $s_F = 10\,\mathrm{N}^{-1}$ (RHS) – zoomed in excerpts

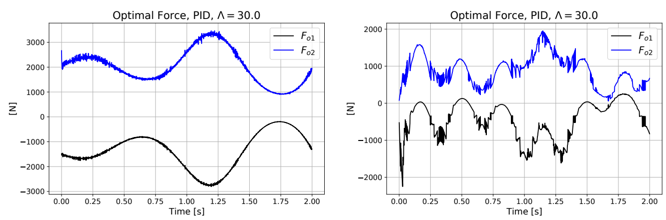

Fig. 16. The optimized control forces belonging to the smoothed cost functions without (LHS) and with force limitation (RHS) with $w_F = 1500\,\mathrm{N}$ and $s_F = 10\,\mathrm{N}^{-1}$ (RHS)

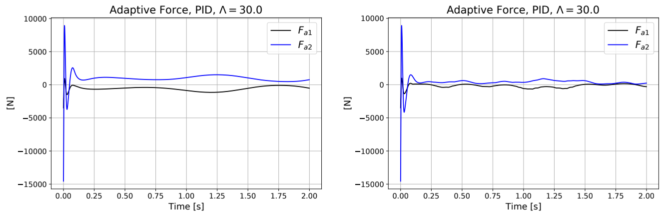

Fig. 17. The adaptive control forces belonging to the smoothed cost functions without (LHS) and with force limitation (RHS) with $w_F = 1500\,\mathrm{N}$ and $s_F = 10\,\mathrm{N}^{-1}$ (RHS)

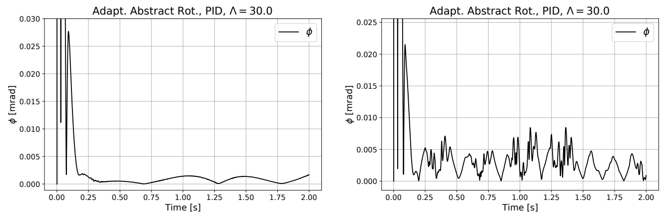

Fig. 18. The angle of the adaptive abstract rotations belonging to the smoothed cost functions without (LHS) and with force limitation (RHS) with $w_F = 1500\,\mathrm{N}$ and $s_F = 10\,\mathrm{N}^{-1}$ (RHS) – zoomed in excerpts

V. Conclusions

In this paper a heuristic *"Adaptive Receding Horizon Controller" (ARHC)* was studied in which tracking the "nominal trajectory" was formulated as minimizing a cost term by the use of a "heavy dynamic model", and the so obtained "optimized path" was adaptively tracked by the use of a "less heavy engine" for which only an approximate dynamic model was available. The optimization phase is based on numerical computations only, no any measurement is necessary in it, since the "heavy dynamic model" as well as the nominal trajectory to be tracked are *a priori* known. For reducing the computational burden of optimization the heavy dynamic model was directly used in building up the horizon with forward differences, therefore the number of the free variables of optimization was drastically decreased, neither the computation of constraint terms, nor gradient reduction with Lagrange multipliers were necessary. The usual quadratic cost functions were substituted by much simpler ones consisting of constant zero regions for tolerated errors with linearly increasing "edges" outside of these regions to evade numerical overflow problems. For comparison, smooth cost functions were also fitted to these functions within a region of practical interest. The optimization was realized by a combination of the gradient descent and the Newton–Raphson methods. This approach drastically reduced the computational burden of optimization. For tracking this optimized trajectory a Fixed Point Iteration-based adaptive controller was applied. In general this approach needs only the measurement of the actually controlled components of the system's state variable, therefore it is much simpler than the Lyapunov function-based approaches in which either measurement or at least estimation of the full state variable is needed.

The operation of the suggested method was exemplified by two nonlinearly coupled van der Pol oscillators as a paradigm of nonlinear dynamical system. According to the simulation results the method seems to be promising for breaking out of the realm of the traditional quadratic cost functions and linear time-invariant dynamic models.

The application of the method can be so interpreted that if in the offline simulations the force limitations do not considerably distort the nominal trajectory, the optimized trajectory obtained for the heavy model surely can be tracked by the less "heavy" engine.

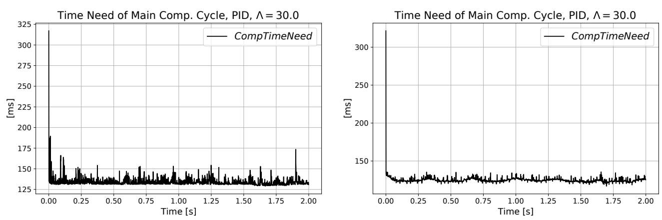

Fig. 19. The computational time of the main cycle belonging to the smoothed cost functions without (LHS) and with force limitation (RHS) with $w_F = 1500\,\mathrm{N}$ and $s_F = 10\,\mathrm{N}^{-1}$ (RHS)

References

[1] A. Atinga, A. Wirtu, and J. K. Tar, "Adaptive receding horizon control for nonlinear systems exemplified by two coupled van der Pol oscillators," in Proceedings of the IEEE 16th International Symposium on Applied Computational Intelligence and Informatics SACI 2022, Timisoara, Romanis. IEEE, 2022, pp. 317–322

[2] B. van der Pol, "Forced oscillations in a circuit with non-linear resistance (reception with reactive triode)," The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science, vol. 7, no. 3, pp. 65–80, January, 1927.

[3] R. Bellman, "Dynamic programming and a new formalism in the calculus of variations," Proc. Natl. Acad. Sci., vol. 40, no. 4, pp. 231–235, April, 1954.

[4] ——, Dynamic Programming. Princeton Univ. Press, Princeton, N. J., 1957.

[5] B. Armstrong, O. Khatib, and J. Burdick, "The explicit dynamic model and internal parameters of the PUMA 560 arm," Proc. IEEE Conf. On Robotics and Automation 1986, pp. 510–518, 1986.

[6] J. Richalet, A. Rault, J. Testud, and J. Papon, "Model predictive heuristic control: Applications to industrial processes," Automatica, vol. 14, no. 5, pp. 413–428, September, 1978.

[7] J. Lagrange, J. Binet, and J. Garnier, Mécanique analytique (Analytical Mechanics) (Eds. J.P.M. Binet and J.G. Garnier). Ve Courcier, Paris, 1811.

[8] J. Riccati, "Animadversiones in aequationes differentiales secundi gradus (observations regarding differential equations of the second order)," Actorum Eruditorum, quae Lipsiae publicantur, Supplementa,, vol. 8, pp. 66–73, 1724.

[9] A. Laub, A Schur Method for Solving Algebraic Riccati Equations (LIDS-P 859 Research Report). MIT Libraries, Document Services, After 1979.

[10] R. Kalman, "Contribution to the theory of optimal control," Boletin Sociedad Matematica Mexicana, vol. 5, no. 1, pp. 102–119, April, 1960.

[11] S. Boyd, L. Ghaoui, E. Feron, and V. Balakrishnan, Linear Matrix Inequalities in Systems and Control Theory. SIAM books, Philadelphia, 1994.

[12] W. Hamilton, "On a general method in dynamics," Philosophical Transactions of the Royal Society, part II for 1834, pp. 247–308, 1834.

[13] ——, "Second essay on a general method in dynamics," Philosophical Transactions of the Royal Society, part I for 1835, pp. 95–144, 1835.

[14] V. Arnold, Mathematical Methods of Classical Mechanics. Springer - Verlag, 1989.

[15] L. S. Lasdon, A. D. Waren, A. Jain, and M. Ratner, “Design and testing of a generalized reduced gradient code for Nonlinear Programming,” ACM Transactions on Mathematical Software (TOMS), vol. 4, no. 1, pp. 34–50, March, 1978.

[16] D. Fylstra, L. Lasdon, J. Watson, and A. Waren, “Design and use of the Microsoft EXCEL Solver,” Interfaces, vol. 28, no. 5, pp. 29–55, September-October, 1998.

[17] J. Gram, “Über die Entwickelung reeler Funktionen in Reihen mittelst der Methode der kleinsten Quadrate,” Journal für die reine und angewandte Mathematik, vol. 94, pp. 71–73, 1883.

[18] E. Schmidt, “Zur Theorie der linearen und nichtlinearen Integralgleichungen. I. Teil: Entwicklung willkürlicher Funktionen nach Systemen vorgeschriebener,” Matematische Annalen, vol. 63, p. 442, December, 1907.

[19] P.-S. Laplace, Théorie Analytique des Probabilités, in: Oeuvres completes de Laplace (Analytical Theory of Probabilities, in: Laplace’s Colleceted Works)/ publiés sous les auspices de l’Académie des sciences, par MM. les secrétaires perpétuel. Gauthier-Villars, Paris, 1886.

[20] H. Issa and J. Tar, “Speeding up the Reduced Gradient Method for constrained optimization,” Proc. of the IEEE 19th World Symposium on Applied Machine Intelligence and Informatics (SAMI 2021), January 21-23, Herl’any, Slovakia, pp. 485–490, 2021.

[21] H. Issa, H. Khan, and J. K. Tar, “Suboptimal adaptive Receding Horizon Control using simplified nonlinear programming,” in 2021 IEEE 25th International Conference on Intelligent Engineering Systems (INES), July 2021, pp. 000 221–000 228.

[22] J. Raphson, Analysis aequationum universalis (Analysis of Universal Equations). typis TB prostant venales apud (printed for sale by) A. & I. Churchill, 1702.

[23] T. J. Ypma, “Historical development of the Newton-Raphson method,” SIAM Review, vol. 37, no. 4, pp. 531–551, December, 1995.

[24] B. Lantos and Z. Bodó, “High level kinematic and low level nonlinear dynamic control of unmanned ground vehicles,” Acta Polytechnica Hungarica, vol. 16, no. 1, pp. 97–117, April, 2019

[25] B. Csanádi, P. Galambos, J. Tar, G. Györök, and A. Serester, “A novel, abstract rotation-based fixed point transformation in adaptive control,” In the Proc. of the 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC2018), October 7-10, 2018, Miyazaki, Japan, pp. 2577–2582, 2018.

[26] 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),” Fund. Math., vol. 3, pp. 133–181, 1922.

[27] O. Rodrigues, “Des lois géometriques qui regissent les déplacements d’ un systéme solide dans l’ espace, et de la variation des coordonnées provenant de ces déplacement considérées indépendent des causes qui peuvent les produire (Geometric laws which govern the displacements of a solid system in space: and the variation of the coordinates coming from these displacements considered independently of the causes which can produce them),” J. Math. Pures Appl., vol. 5, pp. 380–440, 1840