FACTA UNIVERSITATIS
Series: **Mechanical Engineering**
DOI: 10.22190/FUME190919003V

Original scientific paper

COMPUTED TORQUE CONTROL FOR A SPATIAL DISORIENTATION TRAINER

Jelena Vidaković[1], Vladimir Kvrgić[2], Mihailo Lazarević[3], Pavle Stepanić[1]

[1]Lola Institute Belgrade, Serbia
[2]Institute Mihajlo Pupin Belgrade, Serbia
[3]Faculty of Mechanical Engineering, University of Belgrade, Serbia

Abstract. *A development of a robot control system is a highly complex task due to nonlinear dynamic coupling between the robot links. Advanced robot control strategies often entail difficulties in implementation, and prospective benefits of their application need to be analyzed using simulation techniques. Computed torque control (CTC) is a feedforward control method used for tracking of robot's time-varying trajectories in the presence of varying loads. For the implementation of CTC, the inverse dynamics model of the robot manipulator has to be developed. In this paper, the addition of CTC compensator to the feedback controller is considered for a Spatial disorientation trainer (SDT). This pilot training system is modeled as a 4DoF robot manipulator with revolute joints. For the designed mechanical structure, chosen actuators and considered motion of the SDT, CTC-based control system performance is compared with the traditional speed PI controller using the realistic simulation model. The simulation results, which showed significant improvement in the trajectory tracking for the designed SDT, can be used for the control system design purpose as well as within mechanical design verification.*

Key Words: *Computed Torque Control, Robot, Motion Control, Spatial Disorientation*

1. INTRODUCTION

The challenge of robot control stems from nonlinear and time-variable coupling effects in the dynamic model. Different advanced control strategies based on adaptive

Received September 19, 2019 / Accepted December 18, 2019
Corresponding author: Jelena Vidaković
Lola Institute, Kneza Višeslava 70a, Belgrade, Serbia
E-mail: jelena.vidakovic@li.rs

control [1], intelligent control [2], soft computing schemes [3, 4], optimization techniques [5], etc. have been used to overcome nonlinearities and uncertainties in robot dynamics. To select a proper control method, different factors have to be taken into account. Application for which the robot is designed defines motion (range of velocities, accelerations) and performance requirements. Characteristics of the mechanical design, applied actuators and implementation requirements [6] have a great practical value for making a choice of the potential control strategy.

Robot modeling and control methods can be applied to various multibody systems that are not necessarily flexible in their application. Herein, a control strategy for the Spatial disorientation trainer (SDT), Fig. 1, a flight simulation training device designed to train pilots to avoid and cope with in-flight illusions, is considered. The SDT is modeled as a 4DoF robot manipulator with revolute joints [7].

Modern combat aircraft are capable of unconventional flight with unusual orientations. Spatial disorientation (SD) is one of the major threats to the pilots of combat aircraft [8-10]. According to the most widely used definition, SD refers to: "a failure to sense correctly the position, motion or attitude of the aircraft or of him/her within the fixed coordinate system provided by the surface of the earth and the gravitational vertical" [11]. Training within SD simulators of different levels of complexity is considered the most effective countermeasure to spatial disorientation [10]. The SDT considered herein is a robot manipulator specifically designed to examine the pilot's ability to recognize unusual flight orientations, to train the pilot to adapt to them and to persuade the pilot to believe in the aircraft instruments for orientation, and not into his senses [7].

The simplest approach in robot control design is to adopt an LTI-model of the process and to consider variable nonlinear robot dynamics as a disturbance. The traditional control method is PID control. This approach can be justified for highly geared manipulators, as the influence of a nonlinear variable dynamics decreases significantly with high gear ratio [12], and also for stiff manipulators realizing slow trajectories, as the stiffer mechanical design enables the adoption of the larger controller gains. In the case of the SDT, direct drive motors are used for three axes, but the device typically does not achieve high values of speed [7].

Within a choice of a control strategy for the SDT, the influence of nonlinear coupling effects in the dynamic model on the tracking capability of the considered controllers has to be investigated. The feedforward computed torque control (CTC) method [13] implies the cancelation of nonlinear coupled terms in a robot dynamic model. The use of feedforward control is considered as a solution capable of suppressing the speed error in cases with known disturbances [14]. However, besides the complexity of dynamic modeling for multiple DoFs robots [15], the CTC method suffers from drawbacks related to 1) errors due to structured and unstructured uncertainties in a dynamic model; 2) possible difficulties in implementation.

The purpose of this study is to compare, using appropriate simulation techniques, the performances of the traditional PI controller and the controller with CTC compensation added to feedback for the case of the SDT. Motivation is not only to determine a proper control strategy but also to verify the mechanical structure design of the device.

2. TRAJECTORY PLANNER DEVELOPMENT

In the previous work [7], kinematic and dynamic models of the SDT are derived and implemented into the trajectory planner. For the SDT, the discrete control system is developed with a trajectory planner that calculates joint trajectories in the offline regime. At the path update rate defined by interpolation period Δt, reference joint trajectories calculated in trajectory planner are sent to motor controllers.

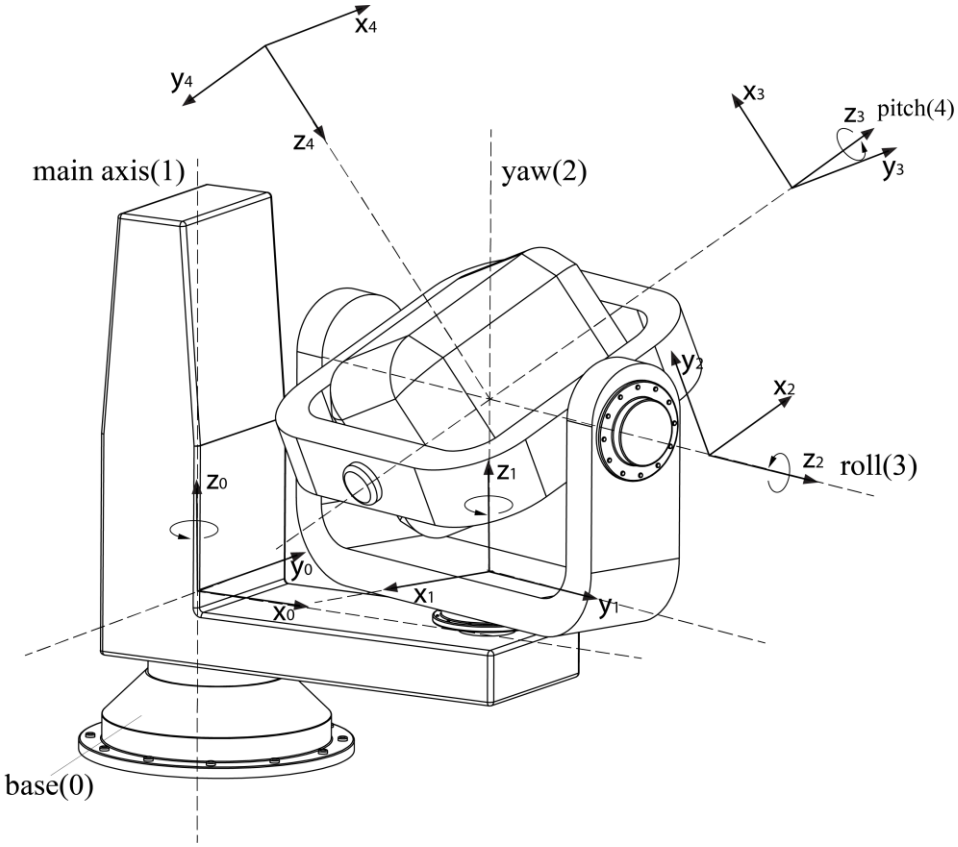

Fig. 1 SDT with 4 DoF [7, 16]

Joint trajectories q_k, $\dot{q}_k$, $\ddot{q}_k$, k=1, 2,..4, Fig. 2, that are used as reference values in this study are obtained by the trajectory planner presented in [7]. Joint accelerations of the trajectories previously studied in [7], obtained after applying limitations of angular accelerations according to maximum torques that chosen actuators can achieve [7] are used herein, and numerical integration is performed to obtain reference speeds and positions of joints.

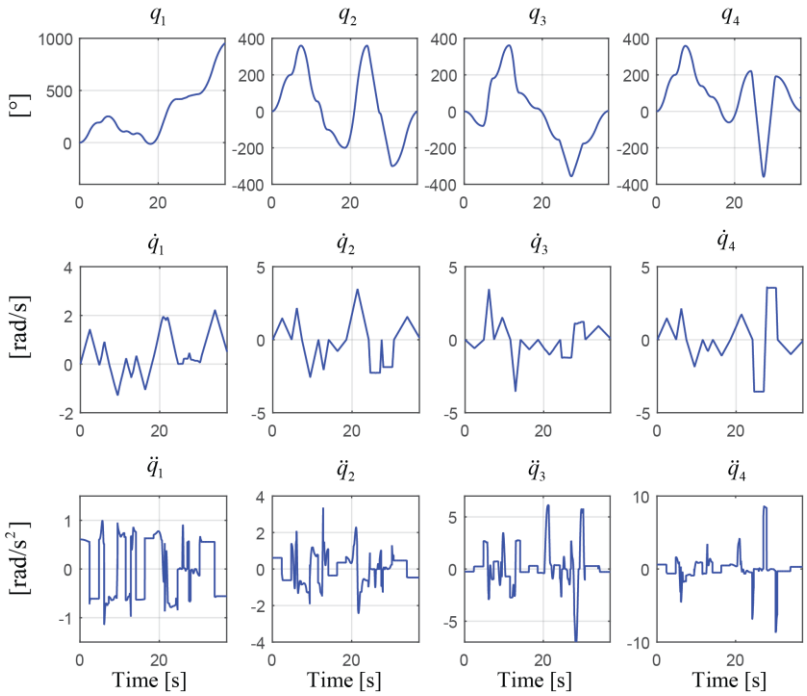

Fig. 2 Reference trajectories of the SDT joints

3. CONTROL SYSTEM DESIGN FOR THE SDT

In this section, the design of the PI speed controller and the CTC-based controller are presented. The model of the motor's mechanical subsystem is based on inertia reflected on the rotor's shaft (effective inertia), obtained from the inverse dynamic (ID) model of the SDT. The load torque calculation from the ID model is presented.

3.1 Model of the Motor's Mechanical Subsystem

From the equation of motion of rigid body rotation about an axis, the nonlinear time-variant model of the motor's mechanical subsystem in robot's joint k can be given in the form:

$$I_{\text{eff}k}\ddot{q}_{\text{m}k} = \tau_{\text{M}k} - \tau_{\text{L}k}. \tag{1}$$

where $q_{\text{m}k}= q_{\text{m}k}(t)$ is the angular position of the rotor; $I_{\text{eff}k}= I_{\text{eff}k}(\mathbf{q})$ is effective inertia (resulting from the coupling of motor with inertial load, as seen from the side of the rotor shaft) which is a function of the instantaneous manipulator configuration $\mathbf{q}=\mathbf{q}(t)=(q_1(t), q_2(t),.., q_n(t))$; n is the number of degrees of freedom; $\tau_{\text{M}k}= \tau_{\text{M}k}(t)$ is the driving torque generated by the motor; $\tau_{\text{L}k}=\tau_{\text{L}k}(t)$ is the load torque, t is time. When the motor in joint k is coupled with the inertial load using gear train with gear ratio r_k, the relation with the angular position of joint k is $q_{\text{m}k}= r_k\, q_k$. In Eq. (1), the bounded nonlinear friction terms

are neglected and treated as disturbances [17]. The deterministic part of load torque τ_{Lk}, τ_{LDk}, and effective inertia I_{effk} are calculated from the ID model.

Herein, the robot ID model is given in the form of a set of n coupled nonlinear differential equations:

$$\sum_{j=1}^{n} d_{kj}(\mathbf{q})\ddot{q}_j + \sum_{j=1}^{n}\sum_{i=1}^{n} h_{kji}(\mathbf{q})\dot{q}_j\dot{q}_i + g_k(\mathbf{q}) = \tau_k. \tag{2}$$

Each equation (for every joint k) in the presented set of n differential equations contains the torque or force terms classified into four groups: 1) inertial- $d_{kk}(\mathbf{q})\ddot{q}_k$, 2) reaction terms generated by accelerations of other joints- $d_{kj}(\mathbf{q})\ddot{q}_j$, $j\neq k$, 3) reaction-velocity generated (centrifugal and Coriolis) terms-$h_{kji}(\mathbf{q})\dot{q}_j\dot{q}_i$, 4) force or torque generated at the joint by gravity in the current manipulator configuration-$g_k(\mathbf{q})$. In Eq. (2), τ_k is the actuating torque for joint k. The ID model of the SDT obtained by the recursive Newton–Euler method was presented in [7].

The effective inertia for the motor's mechanical subsystem in joint k, Eq. (1) can be calculated for every interpolation period in the form of Eq. (3), [13]:

$$I_{effk}(\mathbf{q}) = \left(I_{mk} + d_{kk}(\mathbf{q})/r_k^2\right). \tag{3}$$

where I_{mk} is the inertia of motor and gearbox, and $d_{kk}(\mathbf{q})$ is calculated from the ID model, Eq. (2).

In this study, the model parameters for motors' mechanical subsystems are chosen based on motors selected in [7]. It should be noted that the algorithm that calculates the achievable joint trajectories based on maximum torques that motors can achieve presented in [7] is implemented into the trajectory planner.

3.2 Decoupling of Robot Dynamics and its Implementation in the Simulation Models

With the decoupling of a robot dynamics, a single joint control that takes into account a dynamic model through the deterministic part of the motor load torque τ_{Lk}, Eq. (1), denoted herein as τ_{LDk}, can be considered. Herein, the method for decoupling of robot dynamics given in [13] is extended to include the case when the motion of other links (actuated by their motors) alleviates the load of the motor in the observed joint. Within this method, load torque τ_{LDk} is calculated for every interpolation period from the ID model, Eq. (2), rewritten in the following form:

$$\begin{aligned} &d_{kk}(\mathbf{q})\ddot{q}_k + \tau_{coupledk} = \tau_k, \\ &\tau_{coupledk} = \sum_{j=1,\, j\neq k}^{n} d_{kj}(\mathbf{q})\ddot{q}_j + \sum_{j=1}^{n}\sum_{i=1}^{n} h_{kji}(\mathbf{q})\dot{q}_j\dot{q}_i + g_k(\mathbf{q}). \end{aligned} \tag{4}$$

Torque $\tau_{coupledk} = \tau_{coupledk}(t)$ in Eq. (4) consists of load torque $\tau_{LDk} = \tau_{LDk}(t)$, and torque $\tau_{alleviatek} = \tau_{alleviatek}(t)$ produced by the motion of other links actuated by other motors, which contributes to the motion of link k (it acts in the direction of desired angular acceleration $\ddot{q}_k$) and reduces the driving torque that the motor in joint k has to generate to

achieve the desired link motion. $\tau_{\mathrm{LD}k}$ and $\tau_{\mathrm{alleviate}k}$ are calculated for every interpolation period from the ID model, Eq. 4, as given in Algorithm 1:

Algorithm 1:

If sign ($\ddot{q}_k$)=sign($\tau_{\mathrm{coupled}k}$), then $\tau_{\mathrm{LD}k}=\tau_{\mathrm{coupled}k}$ / r_k, $\tau_{\mathrm{alleviate}k}=0$;

Else if abs($d_{kk}(\mathbf{q})\ddot{q}_k$)> abs($\tau_{\mathrm{coupled}k}$), then $\tau_{\mathrm{LD}k}=0$, $\tau_{\mathrm{alleviate}k}= -\tau_{\mathrm{coupled}k}$ / r_k;

Else $\tau_{\mathrm{LD}k}= -\tau_k$ / r_k, $\tau_{\mathrm{alleviate}k}= (\tau_k -\tau_{\mathrm{coupled}k})$ / r_k;

In Algorithm 1, abs stands for absolute value. The application of the proposed algorithm in the simulation models used in this study is given below in Fig. 3.

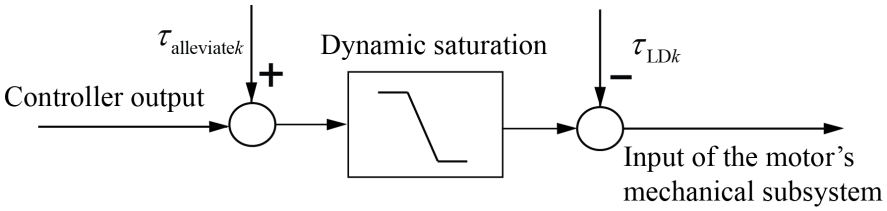

Fig. 3 Decoupling of the inverse dynamic model

The dynamic saturation is implemented in the following way: if sign ($\ddot{q}_k(t)$)>0 then the upper motor saturation limit is increased by value of $\tau_{\mathrm{alleviate}k}(t)$, and if sign ($\ddot{q}_k(t)$)<0 then $\tau_{\mathrm{alleviate}k}(t)$ is added to the lower motor saturation limit ($\tau_{\mathrm{alleviate}k}(t)$ and $\ddot{q}_k$ are always the same in sign). In this way, the motor saturation in the simulation model does not influence $\tau_{\mathrm{alleviate}k}$, nor does it with the real device.

3.3 Design of the PI Speed Controller for Simulation Models

PI controller is the most commonly used control algorithm in the process control industry [18]. Herein, PI controller is selected to obtain a second-order closed-loop system with a characteristic polynomial in a form $s^2+2\zeta_k\omega_{\mathrm{n}k}s+\omega_{\mathrm{n}k}^2$, ($\zeta_k$ is damping factor), for comparison of the natural frequency of closed-loop system $\omega_{\mathrm{n}k}$ with lowest natural frequency ω_{r} of the mechanical structure. The characteristic polynomial of the closed-loop system is equal to the denominator den(s) of the closed-loop transfer function and for the PI speed controller, it is given in Eq. (5):

$$\mathrm{den}(s) = s^2 + \mathrm{K}_{\mathrm{PS}k}\, s / I_{\mathrm{eff}k} + \mathrm{K}_{\mathrm{IS}k} / I_{\mathrm{eff}k}\,, \tag{5}$$

where $\mathrm{K}_{\mathrm{PS}k}$ and $\mathrm{K}_{\mathrm{IS}k}$ are proportional and integral gains, respectively. For joint k, k=1, 2..4, the load torque due to the motion of the chain of other interconnected links, $\tau_{\mathrm{LD}k}$, is treated as a disturbance.

As said before, $I_{\mathrm{eff}k}$ in Eq. (5) depends on robot configuration. Herein, tuning is performed for the LTI-model with the highest load [19], i.e. for the maximum value of effective inertia. Given that the structural flexibilities of the system are not modeled, special attention must be paid not to excite structural resonances. A rule of thumb is that the maximal natural frequency of closed-loop system $\omega_{\mathrm{n}k}$ is at least two times smaller than the lowest natural frequency of mechanical structure ω_{r} ($\omega_{\mathrm{n}k\mathrm{max}}=0.5\omega_{\mathrm{r}}$) [20]. However, if we consider a request that the motion of the robot link is never underdamped [12], the

value of damping factor ζ_k=1 for the maximum value of effective inertia $I_{effkmax}$, Eq. (6), achieves the fastest response without oscillations for all values of I_{effk} [6].

$$\zeta_k = \mathrm{K}_{PSk} / \left(2\sqrt{I_{effk}(\mathbf{q})\mathrm{K}_{ISk}}\right) \tag{6}$$

Considering that the SDT is not flexible in application, $I_{effkmax}$ can be determined beforehand, in the offline regime. Here, $I_{effkmax}$, for which it applies ζ_k=1, is obtained from ID model simulation using Eq. (3), for the motion given in Fig. 2. For realistic simulation purposes, to take into account possibilities for a higher value of $I_{effkmax}$ for different SDT trajectories, choice of PI speed controller gains takes into account the lowest structural natural frequency, with integral gain K_{ISk} chosen to be $\mathrm{K}_{ISk} = 0.25\omega_r^2 I_{effkmax}$.

If a basic structure of the PI speed controller is used, overshoots are present for the values ζ>1 [14]. In Fig.4 step response for the SDT first axis' closed-loop system with the LTI model process in which I_{eff1}= $I_{eff1max}$, obtained using the basic structure of PI speed controller with K_{ISk} set as 0.25 ω_r^2 $I_{eff1max}$ and for the damping factor ζ_1=1, is given in red line. The rise time is 0.022, the settling time is 0.16, and the overshoot is 13.53%. In an attempt to achieve smaller overshoot, (in many practical servo control applications, the overshoot for the speed step response is usually limited below 10% of the step level [21]), a different structure of PI controller is used here, Fig. 5. Proportional gain relocated in the feedback path avoids the overshoot for the values of $\zeta\geq$1 due to the closed-loop zero removal, while at the same time keeps the denominator unchanged [14]. The response is now slower and, to obtain a faster response, values of damping factor ζ_k are chosen to be slightly lesser than 1 for $I_{effkmax}$. In Fig. 4, step response for the SDT first axis' closed-loop system, with the applied PI speed controller structure shown in Fig. 5 [14] is given in blue line. For the LTI model process with $I_{eff1max}$, with K_{ISk} set as 0.25 ω_r^2 $I_{eff1max}$, and for damping factor ζ_1=0.95, the rise time is 0.094, the settling time is 0.159, and the overshoot is zero. This type of PI speed controller is adopted for all four axes of the SDT in simulation models.

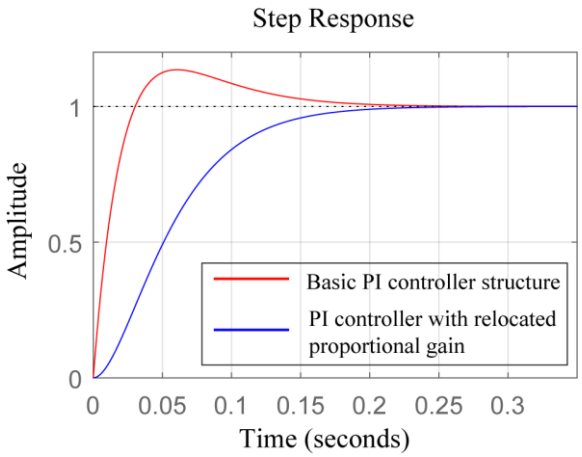

Fig. 4 Step responses for the SDT's first axis with PI speed controllers

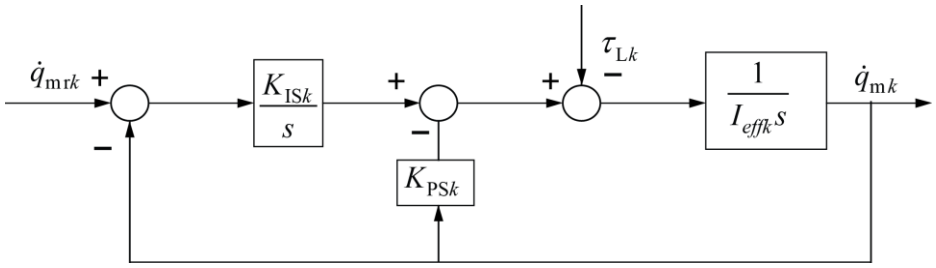

Fig. 5 The PI speed controller with proportional gain in the feedback path [14], $\dot{q}_{\mathrm{m}rk}$ is the reference speed for joint k

It should be noted that in simulation models that use only PI speed control, load torque $\tau_{Lk}=\tau_{LDk}$ and torque contribution of other links' motion $\tau_{\mathrm{alleviate}k}$, Eq. (4) and Algorithm 1, are simulated, and the dynamic saturation is included as presented in Fig. 3.

3.4 Feedforward Computed Torque Method

In the single joint computed torque control method, the load torque due to the motion of the chain of robot's interconnected links, $\tau_{\mathrm{LD}k}= \tau_{\mathrm{LD}k}(\mathbf{q})$, calculated from the ID model for every interpolation period, Eq. (4) and Algorithm 1, is canceled with a feedforward signal. The feedback controller is added to improve the reference-tracking capability (to suppress the errors in dynamic modeling, as well as the effects of stochastic disturbances). For achieving of realistic comparison in a simulation model, to account for modeling errors and stochastic disturbances, the load torque is simulated as $\tau_{Lk}= \tau_{\mathrm{LD}k}(\mathbf{q})(1+\mathrm{A}\sin\omega_{\mathrm{D}k}t)$ where A and $\omega_{\mathrm{D}k}$ are the amplitude and frequency of the simulated disturbances [6], Fig. 6. τ_{LDk} and $\tau_{\mathrm{alleviate}k}$, Eq. (4) and Algorithm 1, are simulated, and the dynamic saturation is included as presented in Fig. 3.

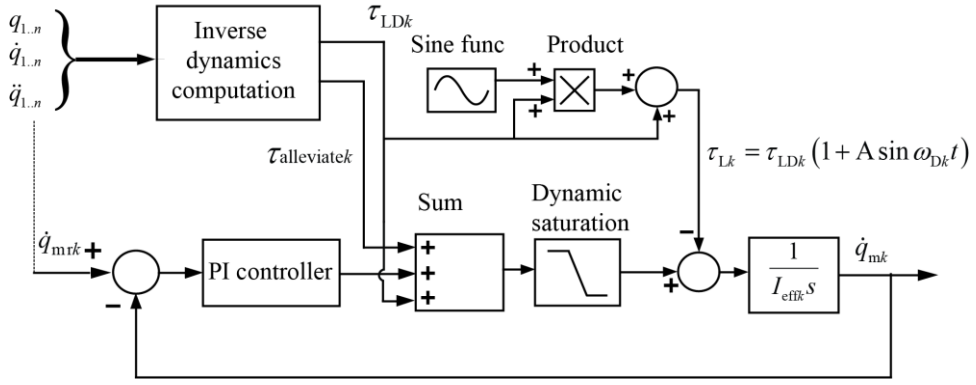

Fig. 6 Simulation model for single joint CTC method

4. SIMULATION RESULTS

In this section, simulation results for the two single-joint control methods presented in Section 3 are given. The performance of the PI speed controller is compared to the same feedback controller with added CTC compensation.

From Catia software, the lowest structural natural frequency of the SDT is obtained to be ω_r =10.5028 Hz. For axis 1, an AC motor is chosen with a maximum torque of 203.2 Nm and a gearbox ratio 67.2, moment inertia of the motor is $I_{m1}=1291\cdot10^{-4}$ kgm^2 [7, 22]. Axes 2, 3 and 4 are actuated by torque motors. The motor for axis 2 achieves a maximum torque of 3950 Nm and has a moment of inertia $I_{m2}=173\cdot10^{-2}$ kgm^2. The motors for axes 3 and 4 achieve maximum torques of 2150 Nm and have moments of inertia $I_{m3,4}=53.1\cdot10^{-2}$ kgm^2 [7, 23]. The simulated gondola payload is 180 kg [7].

Simulink models are designed for all 4 axes of the SDT for processes with maximum loads (maximum effective inertias). The reference speeds are simulated as a series of discrete values obtained from the trajectory planner, Fig. 2. In the models with PI speed feedback only, Fig. 5, load torque τ_{LDk} and torque contribution of other links' motion $\tau_{alleviatek}$, Eq. (4) and Algorithm 1, Fig. 3, are simulated for every interpolation period Δt=5 ms. In models with PI speed feedback plus CTC compensators, load torque τ_{LDk} is compensated in every interpolation period, while load torque τ_{Lk} is simulated as $\tau_{Lk}=\tau_{LDk}(\mathbf{q})(1+A\sin\omega_{Dk}t)$, A is chosen to be 0.05 (meaning that the load torque estimation error is about 5 %); τ_{LDk} and $\tau_{alleviatek}$ are calculated from Eq. (4) and Algorithm 1, Fig. 6. Dynamic saturation presented in Section 3.2 is applied at the outputs of controllers for all simulation models.

Process and controller parameters for PI speed control are given in Table 1. Variation of the effective inertia in percent is given, and the variation of damping factor ζ_k for the motion given in Fig 2. is presented for the minimum, the median and the maximum value of effective inertia. It should be noted that the oscillation frequency of the closed-loop system time response is smaller than the natural frequency of the closed-loop system $\omega_{Ok}=\omega_{nk}\sqrt{1-\zeta_k^2}$. For ζ_k=0.95 for $I_{effkmax}$, the oscillation frequency of the closed-loop system time response with the adopted gains is $\omega_{Ok}=0.32\omega_{nk}=0.16\omega_r$.

Table 1 Process and controller parameters, variation of effective inertia and variation of damping factor ζk for the motion given in Fig. 2

Joint	$I_{effkmax}$ [kgm^2]	Gain		Variation of eff. inertia [%]	ζ_k		
		K_{PSk}	K_{ISk}		$I_{effkmax}$	$I_{effkmed}$	$I_{effkmin}$
1	4.26	267.1	$4.64\cdot10^3$	39.95	0.95	1.06	1.23
2	98.01	$6.34\cdot10^3$	$1.07\cdot10^5$	31.82	0.98	1.07	1.18
3	796.3	$5.2\cdot10^4$	$8.67\cdot10^5$	7.29	0.99	1.009	1.03
4	250.28	$1.57\cdot10^4$	$2.73\cdot10^5$	25.4	0.95	1.017	1.099

In Fig. 7, trajectory tracking for axes k=1, 2.. 4 with two considered types of controllers are presented. The reference value (a series of discrete values obtained from the trajectory planner, Fig. 2) is given in blue color, the outputs obtained by the PI speed

controller are given in red, while the outputs obtained by the PI speed controller with added CTC are given in green color. The errors $e_k = \dot{q}_k - \dot{q}_{rk}$, k=1, 2.. 4 in obtained speeds are given in Fig. 8.

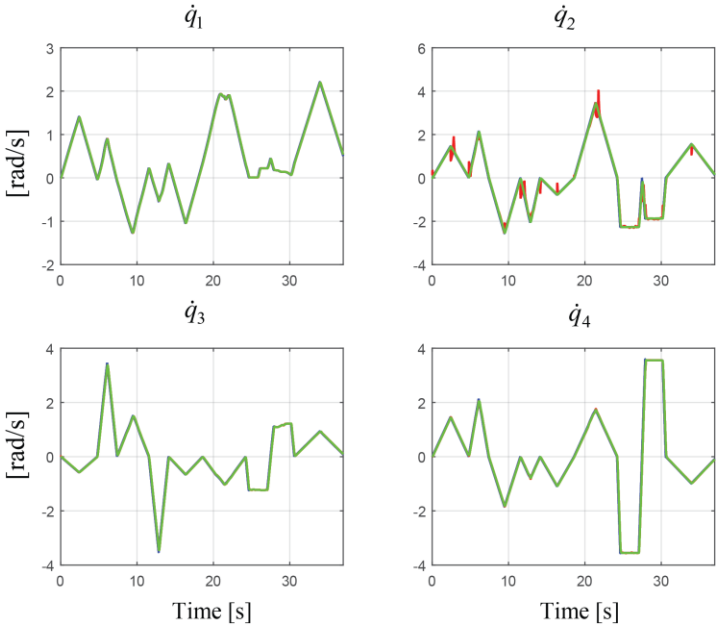

Fig. 7 The obtained trajectory tracking results: reference value is in the blue line, tracking obtained by the PI speed controller is in the red line, tracking obtained by CTC compensation added to the PI speed controller is in the green line

As can be seen, there is an improvement in trajectory tracking when CTC compensation is added to PI speed feedback for the desired SDT motion. The constant reference speed is simulated in segments for the joints 2 and 4 (denoted in Fig. 8), and it can be seen that the small steady-state error is present with the PI speed control as a result of varying disturbance and limited controller gains. The CTC addition achieves zero error in steady-state. The short-lasting high values of error in PI speed control for joint 2 at certain time instants are caused by sudden and large changes in load torque τ_{LD2} in those instants.

The gains for PI speed controllers are selected for the maximum values of effective inertia I_{effk}, and control system performance is expected to deteriorate to a certain extent for smaller values of I_{effk}. To examine the performance decay, the simulation models with the same PI controllers (designed for $I_{effkmax}$) and processes with the minimum value of I_{effk} are developed. The performance deterioration is not significant except for joint 2. For example, the maximum error for joint 2 with the PI speed controller for the process with $I_{eff2max}$ (circled in Fig 8.) is 1.08 rad/s, and for the process with $I_{eff2min}$ this error is 1.204 rad/s. When CTC is added, the difference in these errors (for $I_{eff2max}$ and $I_{eff2min}$) is insignificant.

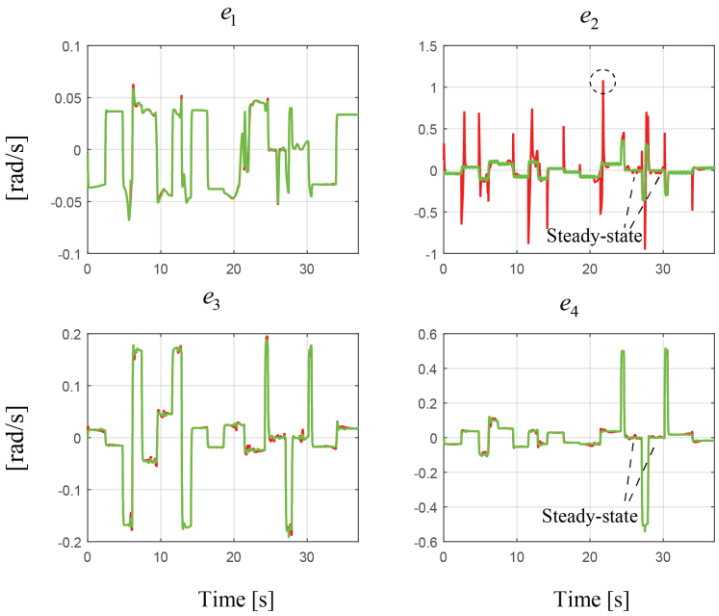

Fig. 8 Errors in speed: obtained by the PI speed controller in the red line, obtained by CTC compensation added to the PI speed controller in the green line

Simulation results showed that for the designed SDT the influence of the nonlinear dynamic model on the control system performance is not negligible for relatively small values of joint speeds. With the reduction of inertia/mass of the mechanical structure, the significance of improvements in trajectory tracking achieved by the addition of CTC compensation may be higher (due to gains limitation of general PID controller to avoid unwanted resonant effects). Considering that weight reduction is performed to reduce the motors' size (power) (to achieve the design and usage cost reduction), this possibility should be tested by control system performance simulation for the chosen smaller motors and selected PID controller structure with the adopted tuning method. The benefits of the achieved improvements should be weighted with the complexity of practical implementation.

5. CONCLUSION

In this paper, the application of the computed torque method for the motion control of the Spatial disorientation trainer is investigated using realistic simulation. The SDT device is modeled as a 4DoF robot manipulator with revolute joints. Models for the motors' mechanical subsystems used in simulation examples account for robot dynamic model through inertia reflected on the rotor shaft. Gains of the applied PI speed controllers are limited taking into account the lowest natural frequency of the mechanical structure obtained from CAE software. The dynamic saturation based on the maximum torques for the selected actuators is applied at the outputs of controllers. Within CTC compensation, the reasonable error in load torque calculation from the dynamic model is assumed. The addition of CTC

compensator to the PI speed feedback controller achieved considerable improvement in trajectory tracking in simulation example. The simulation results are significant regarding the choice of the control method for the SDT, and also in reference to the design of the mechanical structure of the manipulator and the appropriate choice of motors.

Acknowledgements: *This research has been supported by a research grant of the Serbian Ministry of Education, Science and Technological Development under numbers TR 35023 and TR 35006.*

REFERENCES

1. Slotine, J-JE., Weiping, L., 1988, *Adaptive manipulator control: A case study*, IEEE transactions on automatic control, 33(11), pp. 995-1003.
2. Chen, C-S., 2008, *Dynamic structure neural-fuzzy networks for robust adaptive control of robot manipulators*, IEEE Transactions on Industrial Electronics, 55(9), pp. 3402-3414.
3. Li, X., Chien, C.C., 2013, *Adaptive neural network control of robot based on a unified objective bound*, IEEE Transactions on Control Systems Technology, 22(3), pp. 1032-1043.
4. Peng, J., Yan, L., Jie, W., 2014, *Fuzzy adaptive output feedback control for robotic systems based on fuzzy adaptive observer*, Nonlinear Dynamics, 78(2), pp. 789-801.
5. Daş, M.T., Dülger, L.C., Daş, G.S., 2013, *Robotic applications with particle swarm optimization (pso)*, Proc. International Conference on Control, Decision and Information Technologies (CoDIT) IEEE, pp. 160-165.
6. Vidakovic, J., Kvrgic, V., Lazarevic, M., 2018, *Control system design for a centrifuge motion simulator based on dynamic model*, Strojniski vestnik/Journal of Mechanical Engineering, 64 (7-8), pp. 465-474.
7. Kvrgic, V.M., Visnjic, Z.M., Cvijanovic, V.B., Divnic, D.S., Mitrovic, S.M., 2015, *Dynamics and control of a spatial disorientation trainer*, Robotics and Computer-Integrated Manufacturing, 35, pp. 104-125.
8. Previc, F.H., Ercoline, W.R., 2004, *Chapter 1. Spatial Disorientation in Aviation: Historical Background, Concepts, and Terminology*, Progress in astronautics and aeronautics, 203, pp. 1-36.
9. Lawson, B.D., Curry, I.P., Muth, E.R., Hayes, A.M., Milam, L.S., Brill, J.C., 2017, *Training as a countermeasure for spatial disorientation (SD) mishaps: Have opportunities for improvement been missed*, Educational Notes Paper NATO-STO-EN-HFM 265.
10. Lewkowicz, R., Kowaleczko, G., 2019, *Kinematic issues of a spatial disorientation simulator*, Mechanism and Machine Theory, 138, pp. 169-181.
11. Gradwell, D., Rainford, D., 2006, *Ernsting's Aviation and Space Medicine 4E*, CRC Press, 433 p.
12. Craig, J.J., 2005, *Introduction to Robotics: Mechanics and Control (3rd ed.)*, Pearson Prentice Hall, Upper Saddle River, 264 p., 281 p.
13. Spong, M.W., Vidyasagar, M., 2008, *Robot dynamics and control*, John Wiley & Sons, 244 p., 247 p., 238 p.
14. Vukosavic, S., 2007, *Digital Control of Electrical Drives*, Springer-Verlag US, New York, pp. 26- 27, 40 p.
15. Kvrgic, V., Vidaković, J., 2020, *Efficient method for robot forward dynamics computation*, Mechanism and Machine Theory, 145, 103680.
16. Dančuo, Z., Kvrgić, V., Rašuo, B., Vidaković, J., 2013, *On Dynamics of a Spatial Disorientation Trainer for Pilot Training*, Proc. Fourth Serbian Congress on Theoretical and Applied Mechanics, Vrnjačka Banja, pp. 681-686.
17. Lee, H.S., Tomizuka, M., 1996, *Robust motion controller design for high-accuracy positioning systems*, IEEE Transaction on Industrial Electronnics, 43(1), pp. 48-55.
18. Demirtas, M., 2011, *Off-line tuning of a PI speed controller for a permanent magnet brushless DC motor using DSP*, Energy Conversion and Management, 52(1), pp. 264-273.
19. Aström, K., Hägglund, T., 1995, *PID Controllers: Theory, Design, and Tuning, (2nd ed.)*, Isa, Research Triangle Park NC, 52 p.
20. Paul, R.P., 1981, *Robot manipulators: mathematics, programming, and control: the computer control of robot manipulators*, MIT Press, Cambridge MA, 200 p.
21. Dhaouadi, R., Kubo, K., Tobise, M., 1993, *Two-degree-of-freedom robust speed controller for high-performance rolling mill drives*, IEEE transactions on industry applications, 29(5), pp. 919-926.
22. Siemens Configuration Manual, (PFT6), Edition 12, 2004, 6SN1197-0AD12-0BP0.
23. Siemens Configuration Manual, 2009, 05/2009, 6SN1197-0AE00-0BP3.