BULLETIN OF THE POLISH ACADEMY OF SCIENCES
TECHNICAL SCIENCES, Vol. 71(3), 2023, Article number: e144620
DOI: 10.24425/bpasts.2023.144620

SPECIAL SECTION

Motion tracking of a rigid-flexible link robotic system in an underactuated control mode

Elżbieta JARZĘBOWSKA[1]*, Krzysztof AUGUSTYNEK[2], and Andrzej URBAŚ[2]

[1] Warsaw University of Technology, Nowowiejska 24, 00-665 Warsaw, Poland
[2] University of Bielsko-Biala, Willowa 2, 43-309 Bielsko-Biala, Poland

Abstract. The paper presents its contribution to tracking control design of mechanical systems in underactuated mode conditions, i.e. when the number of actuators is less than the number of possible control inputs. Fully actuated mechanical systems are quite well-researched and controller designs are well-developed for them as well. However, due to costs, weight, design, and performance regimes or due to an actuator failure, the underactuated control mode is required in applications. With the aid of the computational procedure for constrained dynamics (CoPCoD), the constrained dynamics, i.e. the reference motion dynamics, and tracking control in an underactuated mode are designed for an example of a three-link planar manipulator model with rigid and flexible links. A dynamic optimization problem is formulated in the paper to obtain optimal time courses of manipulator joint coordinates in underactuated mode conditions in order to apply them to a manipulator driving links controller.

Key words: manipulator; actuator failure; CoPCoD method; optimization.

1. INTRODUCTION

Mechanical control systems like robots, aircraft, satellites and space platforms, underwater vehicles, and human servicing systems are designed to perform work and services and participate in exploration, rescue, military and other missions. These tasks and missions require high-performance levels, durability, and reliability of these systems. There are domains where failure is not an option, like in space missions or health-related operations, and control actions have to be as good as possible. Also, some system functionalities or mission demands require cost, weight, or size reductions. These two factors, failure avoidance, and specific operation demands, lead to active research in dynamics, nonlinear and optimal control of underactuated control systems. A fully actuated system, i.e. the one enjoying the number of control inputs equal to the number of degrees of freedom, performs its mission as long as the actuators work properly. Questions arise about how to minimize risks and danger of damages to the workplace in case of failure and continue a predefined task up to bringing the system to some safe rest position. In the case of an intentionally underactuated design, a controller has to be dedicated to the system and its mission. Controls for underactuated systems are usually driven by two main design approaches, i.e. by attempts to control underactuated system models using current control techniques or by designing new ones that can support reliability problems in practice. These approaches motivate a lot of current research in underactuated control systems. The literature on underactuated systems like aerospace, sea vessels, mechanical, and robotics is quite vast, see e.g. [1–5] and references there. Most of these papers tackle work conditions where there are no other motion limitations and demands, i.e. constraints. Underactuation comes mostly from system designs. Also, some literature addresses optimal control designs for these systems. In [6] multibody systems are designed underactuated and as such they might be non-minimum phase. Optimization is based on structural and control design methodology there to convert systems to minimum phase ones. The proposed integrated design approach is based on an optimization procedure for either the system output or the structural design, or both. Also, many works study specific system examples, e.g. Furuta's pendulum, pendubot, or planar robot manipulators. For example in [7] a specific optimal control problem for planar underactuated manipulators with two revolute joints and brakes at unactuated joints is studied. The brake at an unactuated joint gives rise to two operating modes for that joint. These systems can change their dynamics and the optimal control problem is formulated to find the sequences of modes, the corresponding trajectory, and available control inputs that satisfy the dynamics of the manipulator and steer it between the initial and the final states optimizing a cost function during the motion. Some research considers underactuated systems in geometric settings; see e.g. [8] and references there. Techniques for optimal control designs rely upon adaptations of the classical Skinner and Rusk approaches for cases of Lagrangian dynamics.

The constrained underactuated control systems dynamics are presented in [9, 10] for rigid models of underactuated systems. In [11] a dynamics approach to modeling, taking advantage of the automated computational procedure for constrained dynamics (CoPCoD) is presented. This procedure is based upon constrained dynamics generation, in which the constraints that come from a system work regime can be incorporated. It works

*e-mail: elajarz@meil.pw.edu.pl

Manuscript submitted 2022-11-09, revised 2022-11-09, initially accepted for publication 2022-11-27, published in June 2023.

for rigid and flexible multibody systems which are fully actuated. Tracking control of underactuated system models is still a challenging topic, see e.g. [12–15].

The novelty of the paper is the proposition of an optimal controller design for constrained underactuated systems equipped with flexible links. The underactuation in our scenario comes from the failure of one of the system actuators. The paper and its contribution aim to propose an algorithm for controlling a manipulator system when it becomes underactuated due to failure or its intentional design. Based on the authors' previous experience [9–12], underactuated systems require a specific control approach. In the presented research, the dynamic optimization problem is formulated, in which optimal drive displacements in underactuated mode conditions are calculated. This can be used to design either an intentionally underactuated system or an emergency option for a controller when its failure occurs.

The paper is organized as follows. After the introduction, Section 2 presents the problem formulation including the CoPCoD method short reporting, the constraints on a system model formulation, and the optimization problem formulation together with a control objective function. In Section 3 numerical study results are presented and discussed. The paper closed with the conclusion section and the list of references.

2. PROBLEM FORMULATION

For an illustration of the theoretical approach to the optimization control problem formulation for constrained underactuated system models, a model of a three-link planar manipulator is adopted. It is presented in Fig. 1. It is assumed that link 2 can be treated as rigid or flexible. The flexible link is discretized using the rigid finite element method (RFEM). The motion of the manipulator is forced by three independent servo drives with driving torques $\mathbf{t}_{\mathrm{dr}}^{(p)}$, $p = 1,2,3$.

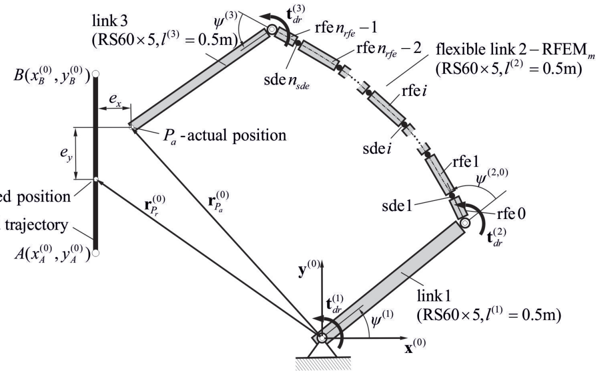

Fig. 1. Model of a rigid-flexible link manipulator: sde$i$ – i-th spring-damping element, rfe$i$ – i-th rigid finite element

To describe the manipulator motion, the generalized coordinates vector is defined as follows

$$\mathbf{q} = \begin{cases} \left[\psi^{(1)} \ \vdots \ \psi^{(2)} \ \vdots \ \psi^{(3)}\right] & \text{if rigid link,} \\ \left[\psi^{(1)} \ \vdots \ \psi^{(2,0)} \quad \tilde{\mathbf{q}}_f^{(2)^T} \ \vdots \ \psi^{(3)}\right]^T & \text{if flexible link,} \end{cases} \tag{1}$$

where: $\tilde{\mathbf{q}}_f^{(2)} = \left[\psi^{(2,1)} \quad \psi^{(2,i)} \quad \psi^{(2,n_{\mathrm{rfe}}-1)}\right]^T$ is a vector containing generalized coordinates of n_{rfe} rigid finite elements.

A controller action aims to move the manipulator end-effector from point A to B along the straight line connecting these points within some prescribed time T. The reference time courses of the joint coordinates can be calculated using the CoPCoD method, see [4, 5] for details. This method takes advantage of the generalized programmed motion equations (GPME) as in [3] and reformulates them for the automation of constrained dynamics derivation and to facilitate numerical calculations. They take the following form

$$\frac{\partial R_1}{\partial \dot{q}_v} + \sum_{w\in i_{d_c}} \frac{\partial R_1}{\partial \dot{q}_w}\frac{\partial \dot{q}_w}{\partial \dot{q}_v} = 0, \quad v \in i_{i_c}\,, \tag{2}$$

where: i_{d_c}, i_{i_c} are vectors of the indexes of dependent and independent coordinates, respectively,

$$R_1 = E_k - 2\sum_{v=1}^{n_{\mathrm{dof}}} \frac{\partial E_k}{\partial q_v}\dot{q}_v + \sum_{v=1}^{n_{\mathrm{dof}}} \frac{\partial E_p}{\partial q_v}\dot{q}_v + \sum_{v=1}^{n_{\mathrm{dof}}} \frac{\partial R_{f_l}}{\partial \dot{q}_v}\dot{q}_v + \sum_{v=1}^{n_{\mathrm{dof}}} Q_v\dot{q}_v,$$

E_k is the kinetic energy, $E_p = E_{p,g} + E_{p,f_l}$ is the potential energy which can be calculated as a sum of the potential energy of the gravity forces and energy of spring deformations of the flexible link, R_{f_l} is the Rayleigh dissipation function of the flexible link, Q are the nonpotential forces, n_{dof} is the number of generalized coordinates.

The system has to satisfy the programmed constraints equations $\mathbf{\Phi}_p(t,\mathbf{q})$ which can be written at the position level in the following way

$$\mathbf{\Phi}_p(t,\mathbf{q}) \equiv \mathbf{r}_{P_a}^{(0)}(\mathbf{q}) - \mathbf{r}_{P_r}^{(0)}(t) = \mathbf{0}, \tag{3}$$

where $\mathbf{r}_{P_a}^{(0)}$, $\mathbf{r}_{P_r}^{(0)}$ are vectors describing the actual and required positions of the end-effector in the reference frame $\{0\}$.

The CoPCoD method uses equations (2) and (3) and generates the constrained dynamics model, which we refer to as reference dynamics. Its solutions satisfy the constraints and can be used to plan motion according to (3) and design a controller to track this motion. The same CoPCoD method is used to generate control dynamics and design a controller. Also, it should be noticed, as described in [3] and references there, that the GPME method and the CoPCoD which is based on it, are both free of the constraint reaction forces. The constraint reaction forces are eliminated during the constrained dynamics generation and this distinguishes these methods from the traditional Lagrange approach. The elimination of the reaction forces is the advantage in the subsequent controller design. These make the CoPCoD a powerful tool for effective constrained dynamics generation and controller designs.

The control of manipulator systems is well studied when a system is fully-actuated, i.e. the number of actuators is equal to the number of degrees of freedom of a system. A major problem is in controlling the manipulator when it becomes underactuated during the manoeuvre execution. In our study, it is assumed that the analyzed manipulator is underactuated due to a drive failure in link 2 at time t_f. The same approach can be applied when link 3 become underactuated.

In the paper, we propose an algorithm based on the solution of the nonlinear optimization problem in which time courses of displacements of each link are calculated. For this purpose, the time courses of joint coordinates of links 1 and 3 are divided into n intervals, starting from the time t_f, in which the displacements are interpolated by means of cubic splines. The time courses of the joint coordinates $\psi_1^{(p)}$, $p = 1,3$, before the time t_f are the same as the ones obtained from the CoPCoD.

The optimization aims to minimize to following functional

$$\Omega(\mathbf{X}) = \frac{w_1}{T - t_f}\sqrt{\int_{t_f}^{T}\left(e_x^2 + e_y^2\right)\,\mathrm{d}t + w_2 E_k(t_k)}, \tag{4}$$

where design variables vector $\mathbf{X}$ contains values of displacements of links 1 and 3 in the discrete time steps after the failure of drive 2; see Fig. 2, e_x, e_y are distances between the actual P_a and required P_r position of the end-effector; see Fig. 1, w_1, w_2 are the weights and $w_1 > 0$, $w_2 > 0$.

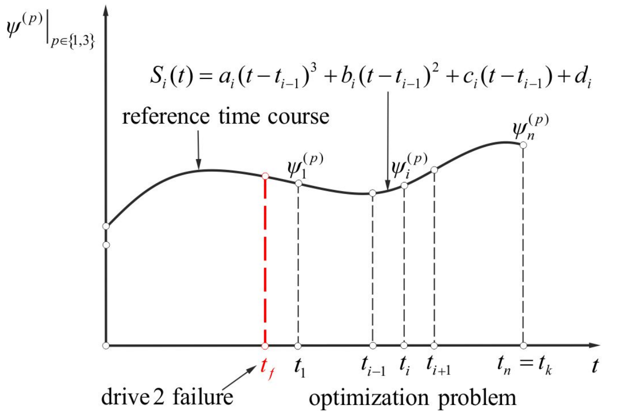

Fig. 2. Time course of the link displacement applied in optimization

This design variables vector can be written as follows

$$\mathbf{X} = \begin{bmatrix}\psi_1^{(1)} & \cdots & \psi_n^{(1)} & \psi_1^{(3)} & \cdots & \psi_n^{(3)}\end{bmatrix}^T. \tag{5}$$

Design variables have to satisfy the following constraints

$$\psi_{j,\min}^{(p)} \le \psi_j^{(p)} \le \psi_{j,\max}^{(p)}\big|_{p\in\{1,3\}}, \tag{6}$$

where $\psi_{j,\min}^{(p)}$ and $\psi_{j,\max}^{(p)}$ are the lower and upper bounds specified a priori.

The structure of the proposed optimization algorithm is presented in Fig. 3.

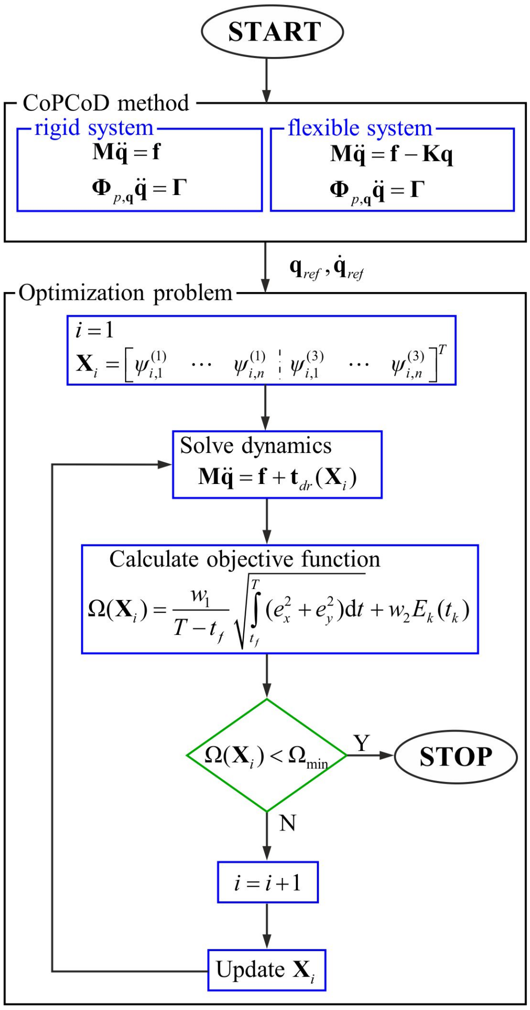

Fig. 3. The proposed optimization algorithm

It is implemented in the following steps:

1. Use the CoPCoD method to calculate the reference time courses of joint coordinates $\mathbf{q}_{\mathrm{ref}}$, $\dot{\mathbf{q}}_{\mathrm{ref}}$ satisfying the assumed programmed constraints, which specify the work regime for the system. The following notation is applied in Fig. 3: $\mathbf{M}$ is a mass matrix, $\mathbf{f}$ is a vector of external potential and nonpotential forces, $\mathbf{K}$ is a stiffness matrix of the flexible link, $\mathbf{\Phi}_{p,\mathbf{q}} = \dfrac{\partial \mathbf{\Phi}_p}{\partial \mathbf{q}}$ is the constraint matrix, $\mathbf{\Gamma}$ is a vector of constraint accelerations.
2. Apply the obtained reference courses to set a starting point for the optimization procedure
$$\mathbf{X}_1 = \begin{bmatrix}\psi_{\mathrm{ref},1}^{(1)} & \cdots & \psi_{\mathrm{ref},n}^{(1)} & \vdots & \psi_{\mathrm{ref},1}^{(3)} & \cdots & \psi_{\mathrm{ref},n}^{(3)}\end{bmatrix}^T.$$
3. Solve the constrained nonlinear optimization problem for the rigid/flexible link manipulator with a failure of drive 2 using the Nelder–Mead procedure together with the penalty method.
4. Verify the obtained optimal solutions
$$\mathbf{X}_{\mathrm{opt}} = \begin{bmatrix}\psi_{\mathrm{opt},1}^{(1)} & \cdots & \psi_{\mathrm{opt},n}^{(1)} & \vdots & \psi_{\mathrm{opt},1}^{(3)} & \cdots & \psi_{\mathrm{opt},n}^{(3)}\end{bmatrix}^T.$$

The critical point of the presented algorithm can be the long computation time of the objective function. This is due to the need of solving the dynamics task each time when the value of the objective function is estimated. This problem can be partially mitigated by using parallel processing methods or methods enabling the approximation of the objective function value [16, 17].

However, this study can give quite a lot of insight into the behavior of underactuated systems when their links are rigid and/or flexible. Specifically, some calculations can be performed offline to determine the potential tracking controller in emergencies. The tracking controller could be used to finish a desired task or bring a system to rest conditions safely. Also, some analysis can be conducted to evaluate the properties of underactuated system behaviors. Lessons learned from this and similar studies can support the potential design or implementation of underactuated systems in cases when the reduction of costs and weight may be desired.

3. NUMERICAL SIMULATION STUDY RESULTS

The proposed optimal control algorithm was implemented in C++, which enables increasing the numerical efficiency of the optimization procedure. The geometry of the manipulator model is given in Fig. 1. It is assumed that all links are made of steel ($E = 2.1 \times 10^{11}$ Pa, $\nu = 0.3$). The flexible link is divided into 4 rigid finite elements. The dynamics equations are integrated using the 4$^{\text{th}}$-order Runge-Kutta scheme with a constant steps size $h = 1 \times 10^{-3}$ s, if link 2 is rigid, or $h = 1 \times 10^{-4}$ s, if link 2 is flexible. Two cases are analyzed: case I – link 2 is treated as rigid, and case II – link 2 is flexible. Both cases are analyzed for the failure times 2 s, 3 s, and 4 s. The coordinates of the starting and end points of the linear trajectory are as follows: $A(x_A^{(0)} = 0.5$ m, $y_A^{(0)} = 0$ m, $B(x_B^{(0)} = 0.5$ m, $y_B^{(0)} = 1.0$ m. The effector moves between these points within 5 s. The following weights of the objective function are assumed: $w_1 = 1000$, $w_2 = 10000$.

Tables 1 and 2 present the values of the objective function and maximum positioning error $e_x^{\max}$ before and after optimization. It can be noted that the values of these parameters after optimization are significantly smaller compared to their values before optimization for all analyzed cases. The optimization is also more efficient when all links are rigid. In case II, when link 2 is treated as flexible, $\Omega(\mathbf{X})$ and $e_x^{\max}$ take greater values.

Table 1
Optimization results obtained for case I

t_f, s	Before optimization		After optimization	
	$\Omega(\mathbf{X})$	$e_x^{\max}$, m	$\Omega(\mathbf{X})$	$e_x^{\max}$, m
2	201.296	0.1806	4.159	0.0123
3	123.706	0.1177	2.529	0.0049
4	35.450	0.0269	3.065	0.0096

Table 2
Optimization results obtained for case II

t_f, s	Before optimization		After optimization	
	$\Omega(\mathbf{X})$	$e_x^{\max}$, m	$\Omega(\mathbf{X})$	$e_x^{\max}$, m
2	207.935	0.1816	7.278	0.0148
3	132.037	0.1230	10.191	0.0161
4	46.466	0.0341	15.594	0.0269

Figures 4–6 show numerical time courses of joint coordinates for all analyzed cases. Figures 7–8 show the trajectories of the end-effector before and after optimization in the reference plane $\mathbf{x}^{(0)}\mathbf{y}^{(0)}$. The time courses of the driving torques are presented in Figs. 9–11.

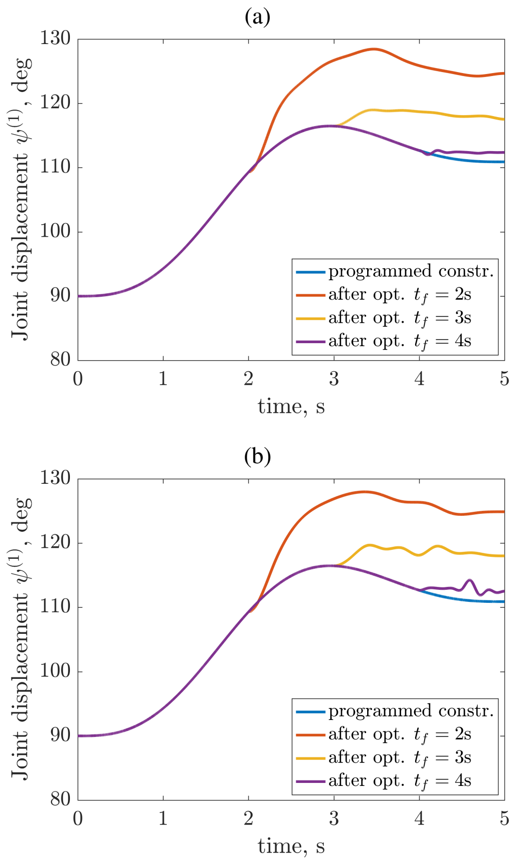

Fig. 4. Time course of link 1 displacement if link 2 is (a) rigid, (b) flexible

When analyzing the results of the calculations, the following observations and conclusions can be made:

- The time courses of joint displacements in their constrained motion obtained for the model with and without the flexible link 2 are similar.
- The flexibility of link 2 has a significant influence on the end-effector trajectory.
- In the case of the drive 2 failure, when the manipulator is equipped with rigid links, a tracking controller enables completing the required task with some small errors.
- In the case of the drive 2 failure, when the manipulator is equipped with rigid and flexible links, it is much more difficult and sometimes even impossible to design a controller to continue tracking a predefined trajectory without the application of some optimization procedure.

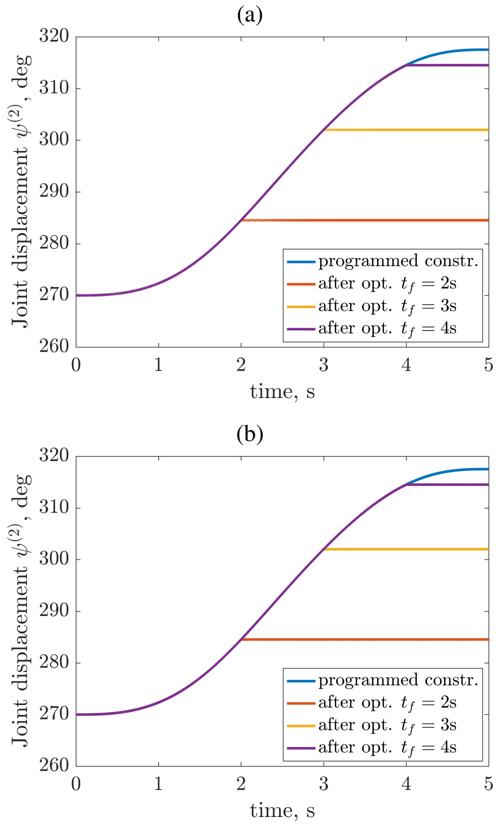

Fig. 5. Time course of link 2 displacement if link 2 is (a) rigid, (b) flexible

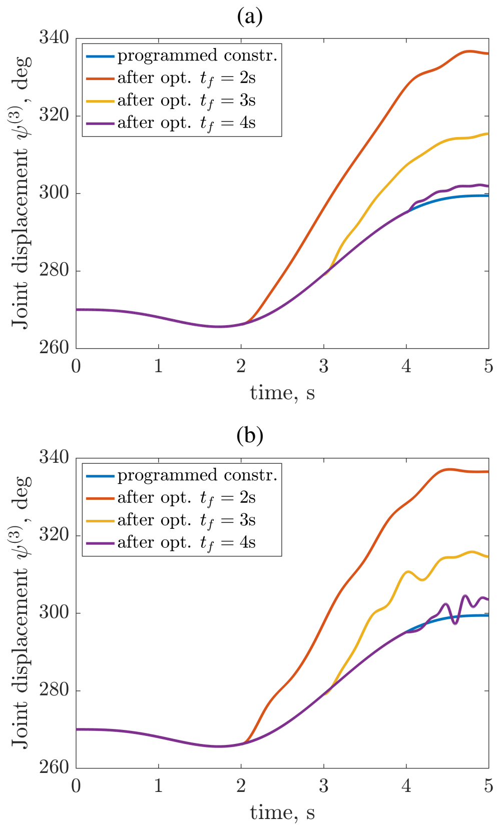

Fig. 6. Time course of link 3 displacement if link 2 is (a) rigid, (b) flexible

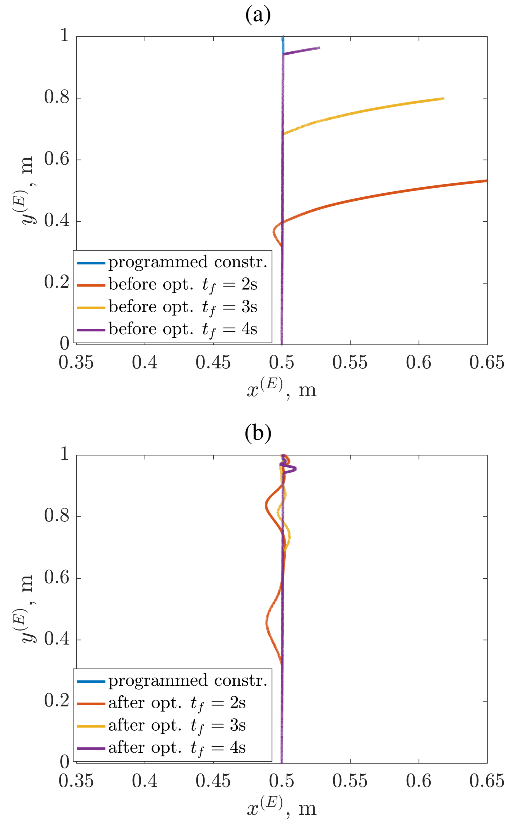

Fig. 7. Trajectory of the end-effector in plane $\mathbf{x}^{(0)}\mathbf{y}^{(0)}$ obtained for the model with rigid link 2 (a) before and (b) after optimization

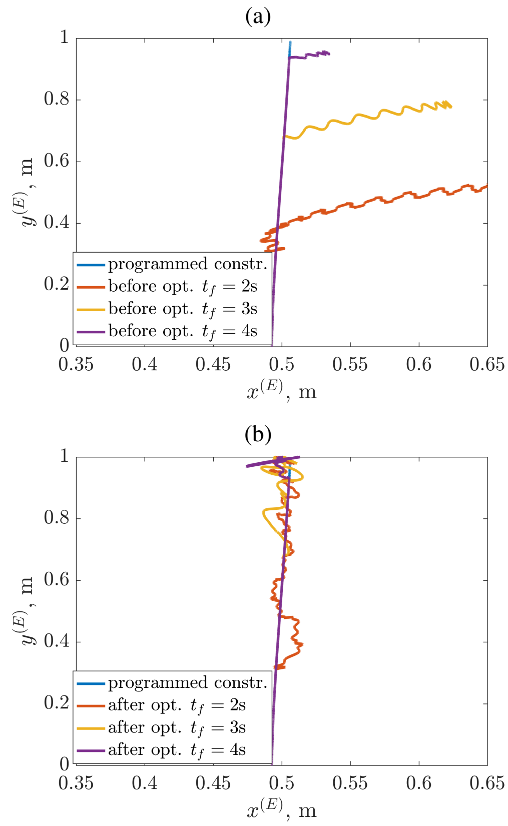

Fig. 8. The trajectory of the end-effector in $\mathbf{x}^{(0)}\mathbf{y}^{(0)}$ plane obtained for the model with flexible link 2 (a) before and (b) after optimization

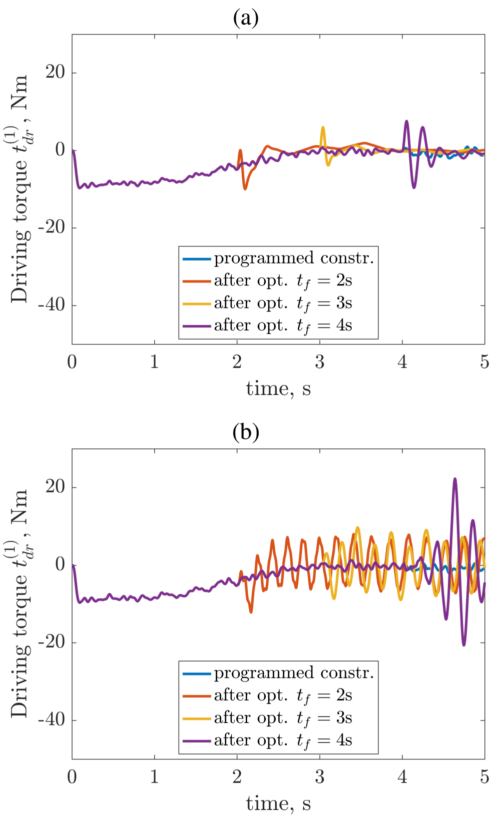

Fig. 9. Time course of driving torque $t_{\mathrm{dr}}^{(1)}$ if link 2 is (a) rigid, (b) flexible

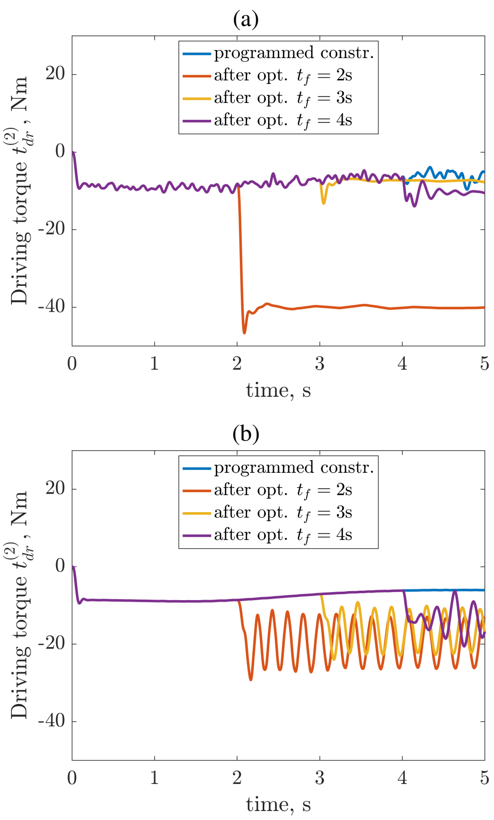

Fig. 10. Time course of driving torque $t_{\mathrm{dr}}^{(2)}$ if link 2 is (a) rigid, (b) flexible

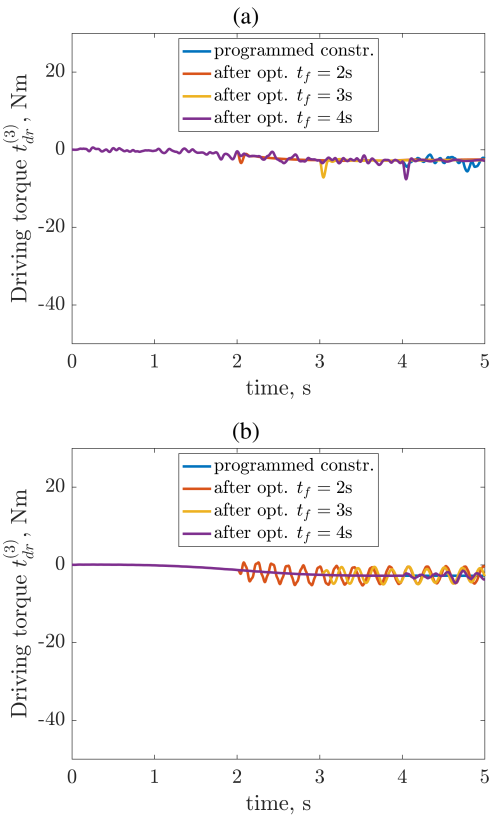

Fig. 11. Time course of driving torque $t_{\mathrm{dr}}^{(3)}$ if link 2 is (a) rigid, (b) flexible

- The application of a popular PID controller in the failure mode without any optimization procedure for the presented task, which is in fact a simple one, is ineffective, see e.g. studies in [5]. Employing the optimization algorithm allows us to eliminate the effects of failures in drive 2 to some extent and, at the same time, facilitates the implementation of the assumed task tracking.

4. CONCLUSIONS

In the presented research, a design of a tracking controller for constrained underactuated system models is proposed. It is accomplished based on the dynamic optimization problem solution, which facilitates calculating optimal drive displacements in underactuated mode conditions.

The mathematical model of the three-link manipulator equipped with rigid and flexible links illustrates the theoretical approach. The novelty of this study is in the optimal controller design when a constrained system with a flexible link becomes underactuated.

The optimization algorithm for the manipulator in the drive failure conditions is proposed and it enables effective desired motion tracking despite the failure of one of the drives. Simulation results illustrate the optimized controller actions.

This study can give some insight into the behavior of underactuated systems when their links are rigid and/or flexible. Analysis can be conducted to evaluate the properties of underactuated system behaviors. Lessons learned from this and similar studies can support the potential design or implementation of underactuated systems in cases when reduction of costs and weight may be desired or to design controllers for failure mode cases.

REFERENCES

[1] Y. Liu, and H. Yu, "A survey of underactuated mechanical systems," *IET Contr. Theory Appl.*, vol. 7, no. 7, pp. 921–935, 2013.

[2] S. Krafes, Z. Chalh, and A. Saka, "A review on the control of second order underactuated mechanical systems," *Complexity*, vol. 2018, p. 9573514, 2018, doi: 10.1155/2018/9573514.

[3] P. Masarati, M. Morandini, and A. Fumagalli, "Control constraint of underactuated aerospace systems," *J. Comput. Nonlinear Dyn.*, vol. 9, no. 2, p. 021014, 2014.

[4] K.D. Do and J. Pan, "Control of Ships and Underwater Vehicles: Design for Underactuated and Nonlinear Marine Systems," in *Advances in Industrial Control*, Springer-Verlag, London, UK, 2009.

[5] J. Moreno-Valenzuela and C. Aguilar-Avelar, "Motion control of underactuated mechanical systems," in *Intelligent Systems, Control and Automation: Science and Engineering*, vol. 88, Springer, Cham, Switzerland, 2018.

[6] R. Seifried, "Dynamics of Underactuated Multibody Systems: Modeling, Control and Optimal Design," in *Solid Mechanics and its Applications*, vol. 205, Springer International Publishing, 2014.

[7] O. Staffetti and E. Staffetti, "Embedded Optimal Control of Robot Manipulators with Passive Joints," *Math. Probl. Eng.*, vol. 2015, p. 348178, 2015.

[8] L. Colombo, D.M. de Diego, and M. Zuccalli, "Optimal control of underactuated mechanical systems: A geometric approach," *J. Math. Phys.*, vol. 51, p. 083519, 2010, doi: 10.1063/1.3456158.

[9] E. Jarzębowska and A. Szewczyk, "An Emergency tracking controller design for a manipulator after its actuator failure," *Proc. ASME, IDETC/CIE*, USA, 2015, doi: 10.1115/DETC2015-47304.

[10] E. Jarzębowska, "Tracking control design for underactuated constrained systems," *Robotica*, 24(5):591-593, 2006.

[11] E. Jarzębowska, A. Urbaś, and K. Augustynek, "Analysis of influence of a crane flexible supports, link flexibility, and joint friction on vibration associated with programmed motion execution," *J. Vibr. Eng. Technol.*, vol. 8, pp. 337–350, 2020.

[12] E. Jarzębowska, K. Augustynek, and A. Urbaś, "Motion tracking of a rigid-flexible link manipulator in a controller failure condition," *Perspectives in dynamical systems II: mathematical and numerical approaches: Springer Proceedings in Mathematics and Statistics*, 2022. (in review)

[13] S.G. Nersesov, H. Ashrafiuon and P. Ghorbanian, "On the stability of sliding mode control for a class of underactuated nonlinear systems," *Proc. of the 2010 American Control Conference*, USA, 2010, pp. 3446–3451.

[14] J. Zhong-Ping, "Controlling underactuated mechanical systems: A review and open problems," in *Lecture Notes in Control and Information Sciences*, 2010, vol. 407, pp. 77–88.

[15] B. He, S.A. Wang and Y. Liu, "Underactuated robotics: A review," *Int. J. Adv. Robot. Syst.*, vol. 16, no. 4, pp. 1–29, 2019, doi: 10.1177/1729881419862164.

[16] K. Augustynek and K. Warwas, "Optimization of deployment process of a satellite by applying genetic algorithms and distributed computing," *Informative Safety, Volodymyr Dahl East Ukrainian National University*, vol. 1, no. 7, pp. 117–126, 2012.

[17] K. Warwas, K. Augustynek, V. Martsenyuk, O. Korchenko, and V. Karpinskyi, "Application of EVCA algorithm in dynamic optimisation of the articulated vehicle motion for control of stability," *ICCAS 2016: 16th International Conference on Control, Automation and Systems*, Korea, 2016, doi: 10.1109/ICCAS.2016.7832408.