Inverse Dynamics Control of a Crane-Like Underactuated Multibody System with Penalty Formulation

Roland Reginald Zana*[1] and Ambrus Zelei[2]

[1]Deptartment of Applied Mechanics, Budapest University of Technology and Economics, Budapest, Hungary

[2]Department of Whole Vehicle Engineering, Audi Hungaria Faculty of Automotive Engineering, Széchenyi István University, Győr, Hungary

*Corresponding author: E-mail: zana_r@mm.bme.hu

Abstract: Inverse dynamics method is an effective approach if the mathematical model of the controlled system is available. Underactuated mechanical systems are special in the sense that the number of independent inputs is less than the degrees of freedom. The Augmented Lagrangian formalism is a well-known and useful approach in forward dynamics simulation and in the control of multibody systems if the mathematical description of the system does not include redundant geometric constraints. In case of redundant constraints, the Penalty formulation serves as a plausible solution. We are not aware of any application of the Penalty formulation for control purposes of underactuated crane-like multibody systems. The Penalty formulation not only allows the inclusion of redundant constraints in the controlled physical system description, but allows redundancy in the task formulation too, i.e. in the servo constraints. Furthermore, the Penalty formulation reduces the computational demand compared to the Augmented Lagrangian formalism in most of the cases. The feasibility and the performance of the Penalty formulation is demonstrated on a crane-like underactuated robot.

1. Introduction

Determining the motion of multibody systems that results from the application of external forces is often mentioned as *forward dynamics*. In a forward dynamic simulation, a system of second order differential equations – generally called the *equations of motion* – describes the mechanical system. In many cases, these equations are extended by some *geometric constraints*, transforming the problem into a set of *differential algebraic equations* (DAE) [1, 2] (see Section 2.1).

If the goal is to control the motion related to some or all degrees of freedom (DoF), the control task can be described by *servo-constraint* equations [3, 4], which are mathematically identical with explicitly time dependent geometric constraints. At this point, the solution possibilities have a variety depending on the controllability properties of the system. If the number of independent control inputs equals the DoF of the system, the *inverse dynamic problem* can be solved directly by solving the inverse kinematics and then inverting the control input matrix. In the inverse dynamics methods in general, the control forces are determined, with which the (partially) prescribed motion can be realized. However, in the case of *underactuated* systems [5], where the number of independent control inputs is smaller than the DoF of the system, the real-time inverse dynamics is more challenging to solve, if even possible. Several inverse dynamics methods were developed for the calculation of control input(s), one of them for instance is based on the *Method of Lagrange Multipliers* (MLM), which was originally developed for forward dynamic simulations (see Section 2.2). MLM requires that the control system is collocated [1, 2], where the DoFs of that the motion is prescribed by the control task are directly driven by a control input.

In [1] derived the so-called *Penalty formulation* from MLM, by eliminating the unknown Lagrangian multipliers from the equations, thus transforming the DAE problem into a set of ordinary differential equations (see Section 2.3). The Penalty formulation directly incorporates the constraints as a dynamical system, penalized by large penalty factor(s). The improper choice of the penalty number can lead to numerical problems. Therefore, as a solution, an iterative version of this method was also developed (see Section 2.4). These Penalty methods were developed for efficient numerical simulation of over-constrained multibody systems. Until now, we are not aware of any applications, where these methods were used to calculate the desired control inputs for a given control task. Hence, our goal is to demonstrate the applicability of the two Penalty methods and compare their results to the MLM on a crane-like underactuated robot shown in Fig. 1.

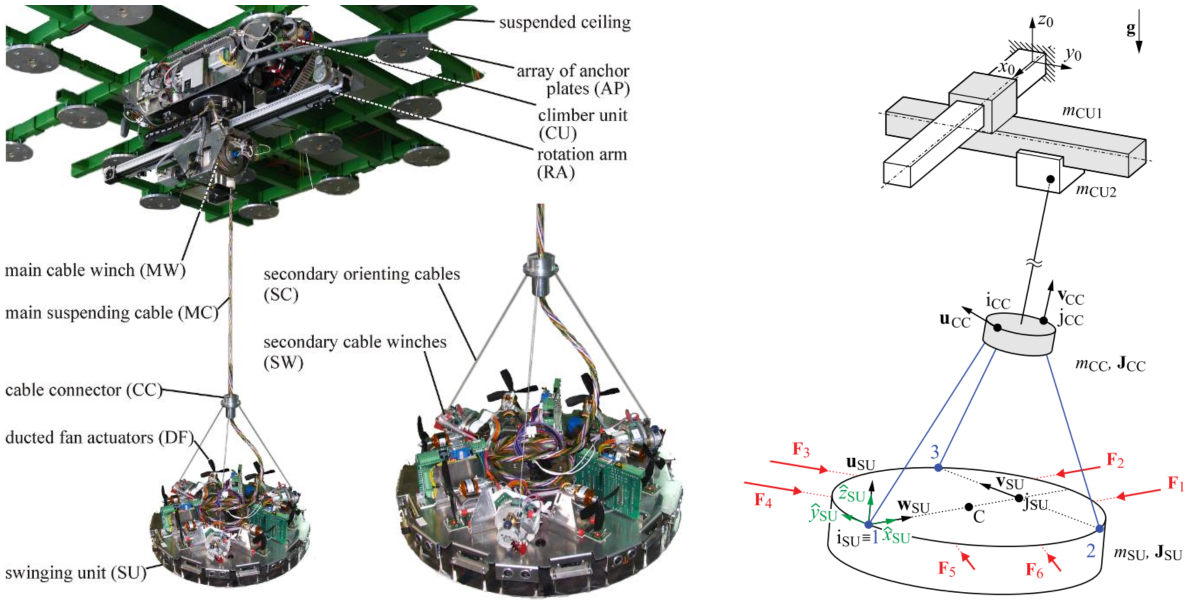

Fig. 1. The crane-like robot prototype Acroboter (left panel) and its mechanical model (right panel)

2. Mathematical modeling and control methods

2.1. A mathematical description of the controlled multibody model

The dynamic Equation 1, and the geometric constraint Equation 2 describes the dynamic behaviour of the controlled system. While the control task is defined by the servo-constraint in Equation 3, which is an algebraic equation similarly to the geometric constraints. Together, these equations form a system of index-3 differential algebraic equations (DAE):

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{c} + \boldsymbol{\varphi}_{\mathbf{q}}^{\mathrm{T}}\boldsymbol{\lambda} = \mathbf{H}\mathbf{u}, \tag{1}$$

$$\boldsymbol{\varphi} = \mathbf{0}, \tag{2}$$

$$\boldsymbol{\sigma} = \mathbf{0}, \tag{3}$$

where $\mathbf{q} \in \mathbb{R}^n$ are the general coordinates, the mass matrix is $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{n\times n}$, the forces which are not related to the acceleration are in $\mathbf{c}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^n$, the vector of the independent control inputs is $\mathbf{u} \in \mathbb{R}^l$., the control distribution matrix is $\mathbf{H}(\mathbf{q}) \in \mathbb{R}^{n\times l}$. The geometric constraints of the $n - m$ DoF mechanical system are defined by $\boldsymbol{\varphi}(\mathbf{q}, t) \in \mathbb{R}^m$, and the servo-constraints of the same number of independent control inputs are defined in $\boldsymbol{\sigma}(\mathbf{q}, t) \in \mathbb{R}^l$.

2.2. A control based on the Augmented Lagrangian formalism

In the Augmented Lagrangian formalism (alternatively called as MLM), DAE index reduction is carried out: the constraints are differentiated twice w.r.t. time and are organized in the following matrix form, from which the control input $\mathbf{u}$ is calculated along with the vectors of the acceleration $\ddot{\mathbf{q}}$ and the Lagrange multipliers $\boldsymbol{\lambda}$:

$$\begin{bmatrix} \mathbf{M} & \boldsymbol{\varphi}_{\mathbf{q}}^{\mathrm{T}} & -\mathbf{H} \\ \boldsymbol{\varphi}_{\mathbf{q}} & \mathbf{0} & \mathbf{0} \\ \boldsymbol{\sigma}_{\mathbf{q}} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \boldsymbol{\lambda} \\ \mathbf{u} \end{bmatrix} = \begin{bmatrix} -\mathbf{c} \\ -\dot{\boldsymbol{\varphi}}_{\mathbf{q}}\dot{\mathbf{q}} - \dot{\boldsymbol{\varphi}}_t \\ -\dot{\boldsymbol{\sigma}}_{\mathbf{q}}\dot{\mathbf{q}} - \dot{\boldsymbol{\sigma}}_t - 2\mathbf{k}_D(\boldsymbol{\sigma}_{\mathbf{q}}\dot{\mathbf{q}} - \boldsymbol{\sigma}_t) - \mathbf{k}_P^2\boldsymbol{\sigma} \end{bmatrix}, \tag{4}$$

with the usually diagonal gain matrices $\mathbf{k}_P$ and $\mathbf{k}_D$, which guarantee the convergence of the servo-constraints to zero. The framework requires linearly independent constraint equations.

2.3. The proposed inverse dynamics control approach based on the Penalty formulation

The penalty formulation is adopted from the work in [1, 2], where it is developed for the efficient numerical simulation of overconstrained multibody systems. We apply the approach for the calculation of the control inputs. The Penalty formulation is based on the introduction of fictitious kinetic energy T^*, dissipative energy D^*, and potential function U^*. At this point, we modify the original formalisms by including not only the geometric constraints but the servo-constraints too in these energy formulae:

$$T^* = \frac{1}{2}(\dot{\boldsymbol{\varphi}}^{\mathrm{T}}\tilde{\boldsymbol{\alpha}}\,\dot{\boldsymbol{\varphi}} + \dot{\boldsymbol{\sigma}}^{\mathrm{T}}\hat{\boldsymbol{\alpha}}\,\dot{\boldsymbol{\sigma}}), \tag{5}$$

$$D^* = \dot{\boldsymbol{\varphi}}^{\mathrm{T}}\tilde{\boldsymbol{\alpha}}\,\boldsymbol{\Omega}\,\boldsymbol{\mu}\,\dot{\boldsymbol{\varphi}} + \dot{\boldsymbol{\sigma}}^{\mathrm{T}}\hat{\boldsymbol{\alpha}}\,\hat{\boldsymbol{\Omega}}\,\hat{\boldsymbol{\mu}}\,\dot{\boldsymbol{\sigma}}, \tag{6}$$

$$U^* = \frac{1}{2}\left(\boldsymbol{\varphi}^{\mathrm{T}}\tilde{\boldsymbol{\alpha}}\,\boldsymbol{\Omega}^2\boldsymbol{\varphi} + \boldsymbol{\sigma}^{\mathrm{T}}\hat{\boldsymbol{\alpha}}\,\hat{\boldsymbol{\Omega}}^2\boldsymbol{\sigma}\right). \tag{7}$$

The newly introduced gain matrices $\hat{\boldsymbol{\alpha}}$, $\hat{\boldsymbol{\Omega}}$, and $\hat{\boldsymbol{\mu}}$ are positive definite and usually diagonal too similarly to $\tilde{\boldsymbol{\alpha}}$, $\boldsymbol{\Omega}$, and $\boldsymbol{\mu}$. These gains represent the penalty numbers, the natural frequencies and the damping ratios of the penalty system. By applying the Lagrangian equation of motion of the second kind on $T + T^*$, $D + D^*$, and $U + U^*$, we obtain the expression

$$\begin{aligned}&\left(\mathbf{M} + \boldsymbol{\varphi}_{\mathbf{q}}^{\mathrm{T}}\tilde{\boldsymbol{\alpha}}\boldsymbol{\varphi}_{\mathbf{q}} + \boldsymbol{\sigma}_{\mathbf{q}}^{\mathrm{T}}\hat{\boldsymbol{\alpha}}\boldsymbol{\sigma}_{\mathbf{q}}\right)\ddot{\mathbf{q}} + \mathbf{c} + \\ &\quad + \boldsymbol{\varphi}_{\mathbf{q}}^{\mathrm{T}}\tilde{\boldsymbol{\alpha}}\left(\dot{\boldsymbol{\varphi}}_{\mathbf{q}}\dot{\mathbf{q}} + \dot{\boldsymbol{\varphi}}_t + 2\boldsymbol{\Omega}\boldsymbol{\mu}\dot{\boldsymbol{\varphi}} + \boldsymbol{\Omega}^2\boldsymbol{\varphi}\right) + \boldsymbol{\sigma}_{\mathbf{q}}^{\mathrm{T}}\hat{\boldsymbol{\alpha}}\left(\dot{\boldsymbol{\sigma}}_{\mathbf{q}}\dot{\mathbf{q}} + \dot{\boldsymbol{\sigma}}_t + 2\hat{\boldsymbol{\Omega}}\hat{\boldsymbol{\mu}}\dot{\boldsymbol{\sigma}} + \hat{\boldsymbol{\Omega}}^2\boldsymbol{\sigma}\right) = \mathbf{0},\end{aligned} \tag{8}$$

from which the acceleration $\ddot{\mathbf{q}}$ can be directly expressed. By comparing Equations 1 and 8, and by acknowledging that $\ddot{\boldsymbol{\varphi}} = \boldsymbol{\varphi}_{\mathbf{q}}\ddot{\mathbf{q}} + \dot{\boldsymbol{\varphi}}_{\mathbf{q}}\dot{\mathbf{q}} + \dot{\boldsymbol{\varphi}}_t$ the identity for the geometric-constraint-related Lagrange multipliers

$$\boldsymbol{\lambda} = \tilde{\boldsymbol{\alpha}}(\ddot{\boldsymbol{\varphi}} + 2\boldsymbol{\Omega}\boldsymbol{\mu}\dot{\boldsymbol{\varphi}} + \boldsymbol{\Omega}^2\boldsymbol{\varphi}) \tag{9}$$

can be recognized [1, 2]. By generalizing this idea, we write that $\ddot{\boldsymbol{\sigma}} = \boldsymbol{\sigma}_{\mathbf{q}}\ddot{\mathbf{q}} + \dot{\boldsymbol{\sigma}}_{\mathbf{q}}\dot{\mathbf{q}} + \dot{\boldsymbol{\sigma}}_t$ and the Lagrange multipliers $\hat{\boldsymbol{\lambda}}$ related to the servo-constraints are

$$\hat{\boldsymbol{\lambda}} = \hat{\boldsymbol{\alpha}}\left(\ddot{\boldsymbol{\sigma}} + 2\hat{\boldsymbol{\Omega}}\hat{\boldsymbol{\mu}}\dot{\boldsymbol{\sigma}} + \hat{\boldsymbol{\Omega}}^2\boldsymbol{\sigma}\right). \tag{10}$$

Furthermore, the $\mathbf{Hu}$ term in Equation 1 must be identical with the fictitious constraint forces that force the system to move according to the servo-constraints. Thus, we can set up the identity

$$\mathbf{Hu} = -\boldsymbol{\sigma}_{\mathbf{q}}^{\mathrm{T}}\hat{\boldsymbol{\lambda}}\ . \tag{11}$$

After the acceleration $\ddot{\mathbf{q}}$ and the Lagrange multiplier $\hat{\boldsymbol{\lambda}}$ are determined from Equations 8 and 10 respectively, Equation 11 provides the possibility to calculate the control input in the form

$$\mathbf{u} = -(\mathbf{H}^{\mathrm{T}}\mathbf{H})^{-1}\mathbf{H}^{\mathrm{T}}\boldsymbol{\sigma}_{\mathbf{q}}^{\mathrm{T}}\hat{\boldsymbol{\lambda}}\ . \tag{12}$$

Alternatively, the Equation of motion 1 might be used for the control input calculation:

$$\mathbf{u} = (\mathbf{H}^{\mathrm{T}}\mathbf{H})^{-1}\mathbf{H}^{\mathrm{T}}\left(\mathbf{M}\ddot{\mathbf{q}} + \mathbf{c} + \boldsymbol{\varphi}_{\mathbf{q}}^{\mathrm{T}}\boldsymbol{\lambda}\right)\ . \tag{13}$$

Both in Equations 12 and 13, $\mathbf{H}^{\dagger} = (\mathbf{H}^{\mathrm{T}}\mathbf{H})^{-1}\mathbf{H}^{\mathrm{T}}$ are regarded as the Moore-Penrose generalised inverse [6] (pseudo-inverse) of $\mathbf{H}$. Alternatively, we can compute the Lagrange multipliers from Equation 9 and then we use Equation 13 for obtaining $\mathbf{u}$. As an alternative, $\mathbf{H}^{\dagger} = \kappa(\mathbf{H}^{\mathrm{T}}\mathbf{H})^{-1}\mathbf{H}^{\mathrm{T}} + (\mathbf{1} - \kappa)\boldsymbol{\sigma}_{\mathbf{q}}$ can be calculated for non-collocated cases similarly to [4].

A special case is when $\mathbf{H}$ is in the following relation with servo-constraint Jacobian:

$$\mathbf{H} = -\boldsymbol{\sigma}_{\mathbf{q}}^{\mathrm{T}} \ . \tag{14}$$

In such case, Equation 12 automatically simplifies as follows in Equation 15. Furthermore, it is guaranteed that there is solution for the inverse dynamics problem (Equation 4 can be solved too).

$$\mathbf{u} = \hat{\boldsymbol{\lambda}} \ . \tag{15}$$

The interesting situation is that when the matrix-inversion is not possible in Equation 4 because of the redundant constraints, but the necessary control input (to satisfy the servo-constraints) can be determined by using the Penalty formulation and Equation 12, 13 (or 15).

2.4. Iterative extension of the Penalty formulation and application in inverse dynamics control

As it is explained in [1,2], the practical application of the penalty formulation requires the fine tuning of the penalty numbers $\tilde{\boldsymbol{\alpha}}$ and $\hat{\boldsymbol{\alpha}}$ and still, numerical problems might occur. As a solution, the Equation of motion 1 is augmented by the penalty terms. This approach is developed for the geometric constraints in [1,2]. After Equation 11 is applied and the $\mathbf{Hu}$ term is substituted by the constraint force $\boldsymbol{\sigma}_{\mathbf{q}}^{\mathrm{T}}\hat{\boldsymbol{\lambda}}$, the same idea is applied for the servo-constraints:

$$\begin{aligned}\mathbf{M}\ddot{\mathbf{q}} + \mathbf{c} + \boldsymbol{\varphi}_{\mathbf{q}}^{\mathrm{T}}\left(\tilde{\boldsymbol{\alpha}}\left(\dot{\boldsymbol{\varphi}}_{\mathbf{q}}\dot{\mathbf{q}} + \dot{\boldsymbol{\varphi}}_t + 2\boldsymbol{\Omega}\boldsymbol{\mu}\dot{\boldsymbol{\varphi}} + \boldsymbol{\Omega}^2\boldsymbol{\varphi}\right) + \boldsymbol{\lambda}^*\right) + \\ + \boldsymbol{\sigma}_{\mathbf{q}}^{\mathrm{T}}\left(\hat{\boldsymbol{\alpha}}\left(\dot{\boldsymbol{\sigma}}_{\mathbf{q}}\dot{\mathbf{q}} + \dot{\boldsymbol{\sigma}}_t + 2\hat{\boldsymbol{\Omega}}\hat{\boldsymbol{\mu}}\dot{\boldsymbol{\sigma}} + \hat{\boldsymbol{\Omega}}^2\boldsymbol{\sigma}\right) + \hat{\boldsymbol{\lambda}}^*\right) = \mathbf{0},\end{aligned} \tag{16}$$

Note that the penalty terms in Equation 16 are zero when the constraints (both geometric and servo-constraints) are fully satisfied. Therefore, in such situation the corrective term $\boldsymbol{\lambda}^*$ is identical with $\boldsymbol{\lambda}$ and indeed $\hat{\boldsymbol{\lambda}}^* = \hat{\boldsymbol{\lambda}}$. The iteration Explained in [1,2] is applied:

$$\boldsymbol{\lambda}_{i+1} = \boldsymbol{\lambda}_i + \tilde{\boldsymbol{\alpha}}(\ddot{\boldsymbol{\varphi}} + 2\boldsymbol{\Omega}\boldsymbol{\mu}\dot{\boldsymbol{\varphi}} + \boldsymbol{\Omega}^2\boldsymbol{\varphi})_{i+1}; \quad i = 0,1,2\ldots\ , \tag{17}$$

$$\hat{\boldsymbol{\lambda}}_{i+1} = \hat{\boldsymbol{\lambda}}_i + \hat{\boldsymbol{\alpha}}\left(\ddot{\boldsymbol{\sigma}} + 2\hat{\boldsymbol{\Omega}}\hat{\boldsymbol{\mu}}\dot{\boldsymbol{\sigma}} + \hat{\boldsymbol{\Omega}}^2\boldsymbol{\sigma}\right)_{i+1}; \quad i = 0,1,2\ldots\ . \tag{18}$$

The iteration formulae defined in Equations 17 and 18 are plied in Equation 16. The resulting formula is then used for the iterative enhancement of the acceleration.

$$\begin{aligned}&\left(\mathbf{M} + \boldsymbol{\varphi}_{\mathbf{q}}^{\mathrm{T}}\tilde{\boldsymbol{\alpha}}\boldsymbol{\varphi}_{\mathbf{q}} + \boldsymbol{\sigma}_{\mathbf{q}}^{\mathrm{T}}\hat{\boldsymbol{\alpha}}\boldsymbol{\sigma}_{\mathbf{q}}\right)\ddot{\mathbf{q}}_{i+1} = \\ &= \mathbf{M}\ddot{\mathbf{q}}_i - \boldsymbol{\varphi}_{\mathbf{q}}^{\mathrm{T}}\tilde{\boldsymbol{\alpha}}\left(\dot{\boldsymbol{\varphi}}_{\mathbf{q}}\dot{\mathbf{q}} + \dot{\boldsymbol{\varphi}}_t + 2\boldsymbol{\Omega}\boldsymbol{\mu}\dot{\boldsymbol{\varphi}} + \boldsymbol{\Omega}^2\boldsymbol{\varphi}\right) - \boldsymbol{\sigma}_{\mathbf{q}}^{\mathrm{T}}\hat{\boldsymbol{\alpha}}\left(\dot{\boldsymbol{\sigma}}_{\mathbf{q}}\dot{\mathbf{q}} + \dot{\boldsymbol{\sigma}}_t + 2\hat{\boldsymbol{\Omega}}\hat{\boldsymbol{\mu}}\dot{\boldsymbol{\sigma}} + \hat{\boldsymbol{\Omega}}^2\boldsymbol{\sigma}\right)\end{aligned} \tag{19}$$

with the initial step $\mathbf{M}\ddot{\mathbf{q}}_0 = -\mathbf{c}$. This formulation allows us to determine the acceleration $\ddot{\mathbf{q}}$ coherent with the geometric and servo constraints. Redundant constraints are handled due to the penalty formalism and the iterative process helps to avoid numerical problems. Again, we have two options to obtain the control inputs. One possibility is to obtain the Lagrange multipliers $\hat{\boldsymbol{\lambda}}$ related to the servo-constraints by using Equation 10. Then Equation 12 or 15 gives $\mathbf{u}$. As a second option, we compute the Lagrange multipliers with Equations 9 and 10. Then we use Equation 13 to obtain $\mathbf{u}$.

3. Mechanical system and simulation methods

The feasibility study of the control methods was carried out on the numerical model of a crane-like underactuated robot (see: Fig. 1 and [7]). The desired XYZ trajectory and the yaw angle were prescribed for the swinging unit. This desired trajectory defined the edges of a 0.25 x 0.25 meters wide square based 0.1 meters high column with an average trajectory speed 0.04m/s, where the robot had to rotate around its vertical axis with 90°, see Fig. 2. The sampling frequency was 50Hz.

The model of the robot consists of two rigid bodies: the swinging unit and the cable connector (both possessing 6 degrees of freedom). The system was described by redundant set of 27 coordinates (12 for each rigid body and 3 for the secondary cables). Since the number of degrees of freedom is 11, 16 geometric constraints were introduced (6 for each rigid body, and 4 for the 4 cables).

The control input calculation and the forward dynamic simulation were executed on the same desktop computer with Intel(R) Core(TM) i7-8700 CPU at 3.20GHz in MATLAB R2019b script.

4. Numerical results

The trajectory tracking performance of the different algorithms was judged based on the deviation from the desired path shown in Fig. 2. and on the violation of the servo-constraints shown in Fig. 3. The results of the MLM and Penalty method are almost totally overlapping, the iterative Penalty method resulted in somewhat larger deviations from the desired trajectory. However, for all the three methods the positioning errors remained under ±5.3mm for displacement, and under ±1.1° for rotation.

An important factor in the evaluation of a control algorithm is the computational demand. We measured the runtime of the control input calculation in each time-step and we calculated the average value for all three control algorithms. The average time consumption was 0.30906ms for the MLM, 0.30478ms for the Penalty and 0.38428ms, 0.38541ms and 0.41062ms for 3, 4 and 5 iterations in the Iterative Penalty method, respectively.

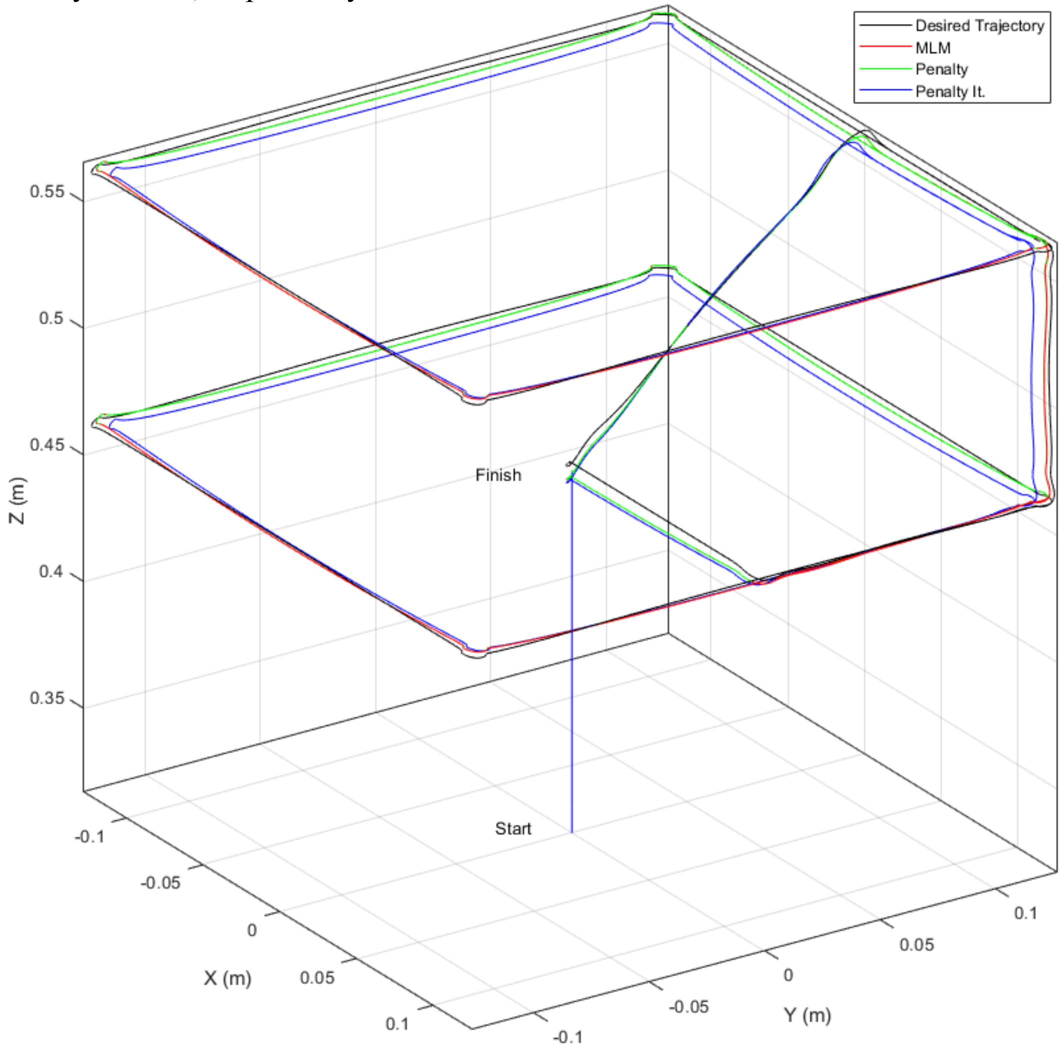

Fig. 2. The desired path and the realized paths of the robot with different controls

5. Conclusion

Similarly to the MLM, the original Penalty formulation requires that the control system is collocated. Otherwise, singularity problems occur. However, in the Penalty formulation the alternative calculation of the pseudo-inverse of the control distribution matrix can be applied to avoid singularity problems. While the MLM requires independent constraint equations, redundant servo-constraints can be handled efficiently by the Penalty method. The Penalty method and the Iterative Penalty method was successfully applied in the inverse dynamics control algorithm.

According to the literature, the Iterative Penalty method helps in intricate situations, when the system is very sensitive to the parameter choices. Since, our case example system is not sensitive to the Penalty numbers, the advantages of the Iterative method were not discovered. In future work, we will answer the question whether the Iterative Penalty method can be advantageous over the non-iterative Penalty formulation in case of control input calculation. Furthermore, we are going to conduct experimental tests to check the practical applicability of these control methods.

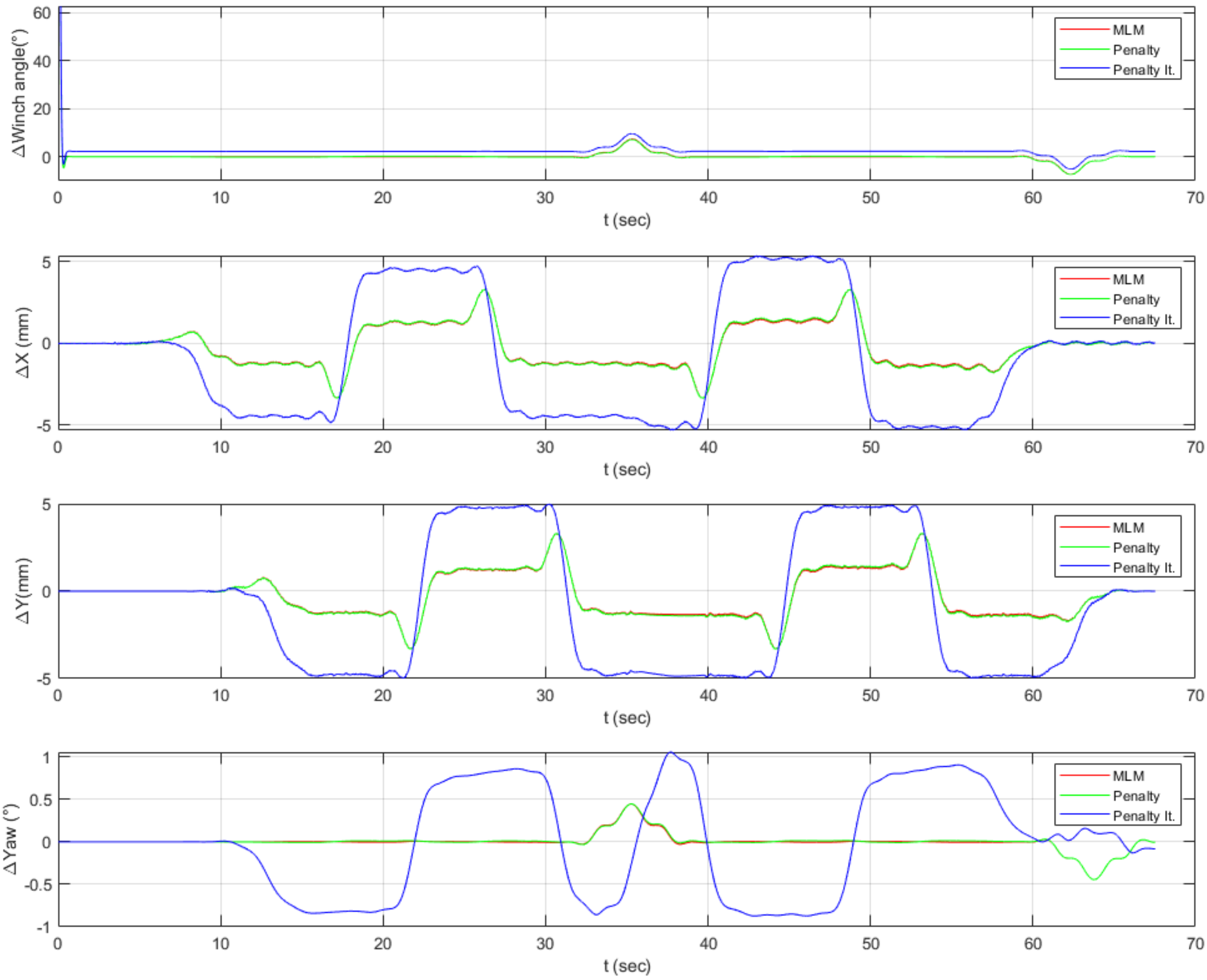

Fig. 3. The error of the servo-constraints

6. References

[1] Bayo, E., de Jalón, J. G., & Serna, M. A. (1988). A modified Lagrangian formulation for the dynamic analysis of constrained mechanical systems. Computer Methods in Applied Mechanics and Engineering, 71 (2), 183-195.

[2] de Jalón, J. G. & Bayo, E. (1994). Kinematic and dynamic simulation of multibody systems: the real-time challenge. Springer-Verlag.

[3] Kirgetov, V. I. (1967). The motion of controlled mechanical systems with prescribed constraints (servoconstraints). PMM, 31(3), 433-446.

[4] Blajer, W., Seifried, R. & Kolodziejczyk, K. (2015). Servo-constraint realization for underactuated mechanical systems. Archive of Applied Mechanics, 85(9-10), 1191-1207.

[5] Seifried, R. (2014). Dynamics of Underactuated Multibody Systems: Modeling, Control and Optimal Design. Springer-Verlag.

[6] Ben-Israel, A. & Greville, T. N. E. (2003). Generalized Inverses: Theory and Applications, 2nd ed.. Springer-Verlag.

[7] Zana, R. R. & Zelei, A. (2021). Feedback motion control of a spatial double pendulum manipulator relying on swept laser based pose estimation. International Journal of Optomechatronics, 15(1), 32-60.