Kinematics and Dynamics Model via Explicit Direct and Trigonometric Elimination of Kinematic Constraints

Moritz Schappler[1], Torsten Lilge[2], and Sami Haddadin[3]

[1] Institute for Mechatronic Systems, Leibniz University Hannover, Germany
moritz.schappler@imes.uni-hannover.de
[2] Institute for Automatic Control, Leibniz University Hannover, Germany
lilge@irt.uni-hannover.de
[3] Chair of Robotics Science and Systems Intelligence, Technical University of Munich, Germany, haddadin@tum.de

Abstract. The efficient implementation of kinematics and dynamics models is a key to model based control of mechatronic systems such as robots and wearable assistive devices. This paper presents an approach for the derivation of these models in symbolic form for constrained systems based on the explicit elimination of the kinematic constraints using substitution variables with trigonometric expressions and the Lagrange equations of the second kind. This represents an alternative solution to using the implicit form of the constraints or using the explicit elimination at comparable computational effort. The method is applied to a novel exoskeleton designed for craftsmen force assistance, which consists of multiple planar closed kinematic loops and gear mechanisms.

Keywords: dynamics, closed-loop, Lagrangian equations, substitution variables, explicit form, trigonometric expressions

1 Introduction and State of the Art

Active mechatronic systems such as industrial robots, exoskeletons or prostheses can only be used to full capacity with model-based control relying on a computationally efficient model of the system, which has to run in real-time. Systems with closed kinematic loops require a greater modeling effort due to an increased number of passive joints. Unlike in classical robotics, exoskeleton systems can contain passive degrees of freedom (DoF) that are equalized by the user at the end-effector. Therefore exoskeleton mechanisms usually have more DoF than classical robots with the same number of end-effector DoF.

The equations of motion of mechanical systems with kinematic constraints can in general be found with the implicit formulation of these constraints. After a partitioning of the coordinates [1], the dynamics equations can be derived via the inverse of the constraints Jacobian in combination with the unconstrained dynamics [2], with Lagrange multipliers [1, 3] or the Udwadia-Kalaba-equation

[4]. The former is implemented in symbolic multi body algorithms such as Robotran [5] or openSymoro [6]. The constraints Jacobian can be either derived from the explicit form of the constraints [2], or from the time derivative of the implicit form [7]. For the computationally efficient analysis of mechanisms it is advantageous to derive the equations of motion in symbolic form [5]. This also facilitates or improves further steps such as grouping dynamics parameters to a minimal set in the regressor form of the dynamics equations [8] or aspects such as linearisation, control and optimization [7].

An explicit representation of the constraints often requires manual generation of the expressions and is only possible for a small set of mechanisms, e. g. some types of planar closed loops. The implicit representation is available for all systems and is therefore used in the methods and tools mentioned above. The direct calculation of all active and passive joint coordinates of a mechanism is only possible with the explicit form of the kinematic constraints. If only the implicit form is available, dependent coordinates have to be determined with numeric iterative procedures [7].

In this paper, the derivation of symbolic equations for the kinematics and dynamics is demonstrated for the example of an upper limb exoskeleton. The structure of the dynamics equations is used for symbolic simplifications leading to improved computational efficiency. The approach can be regarded as an alternative to existing methods with comparable performance. It allows to verify the correctness of dynamics models derived by standard solutions by offering a different way of approaching the problem. This may also open up new paths for researchers working in the area of kinematics and dynamics modeling. The contributions of the paper are

- a formalism for the derivation of dynamics models in symbolic form based on the explicit elimination of the kinematic constraints using substitution variables with trigonometric expressions and the Lagrange equations of the second kind,
- the presentation of a novel exoskeleton mechanism designed for supporting craftsmen when working with power tools, and
- a comparison of the presented approach with other known approaches for modeling mechanical structures with constraints.

The exoskeleton mechanism can be regarded as an example to demonstrate the proposed approach of explicit trigonometric elimination of kinematic constraints. Unlike industrial robots with closed loops which contain only lower kinematic pairs (i. e., single-DoF joints), the exoskeleton also comes with higher kinematic pars (i. e., contacts along a curve generated by gears).

Therefore, the paper is organized as follows: In Sec. 2 the components of the considered exoskeleton are briefly presented. In Sec. 3 the kinematics and dynamics modelling is shown in detail, focussing on the formulation of the explicit constraint equations with substitution variables based on trigonometric expressions. Section 4 provides a comparison of the efficiency of the different methods to calculate the inverse dynamics. Sec. 5 concludes the paper.

2 Exoskeleton Case Study

The upper-body exoskeleton presented in this paper is targeted as an assistive device for craftsmen working with power tools like a hand-held power drill. It fulfills the requirements of attachment at the user's upper body and of neither blocking the users workspace nor sight. The main structure of the exoskeleton shows similar DoF as the human arm. The design was developed by the orthopedics company OTW as part of a BMBF project consortium [9]. It consists of the following parts, which are depicted in Fig. 1 in a right-hand position.

Three-Axis Elbow Joint: The elbow joint of the mechanism is aligned to the axis of rotation of the user's elbow. To reduce associated constraint forces, the joint consists of three parallel axes, which couple the upper arm and the forearm via a central gear wheel. The mechanism is shown in detail in the right part of Fig. 1. The lever mechanism connecting the linear spring-damper-system with the elbow is fixed to the central elbow gear.

Three-Axes Shoulder Joint: The shoulder joint imitates the degrees of freedom of a simplified model of the human shoulder with three consecutive rotations with perpendicular axes. The last axis of the shoulder joint corresponding to the shoulder flexion/extension is parallel to the axes of the elbow and the coupling mechanism, leaving the greatest part of the joints in one plane to simplify manufacturing and kinematics modeling.

Coupling Mechanism Shoulder-Elbow: The shoulder flexion/extension joint and the elbow joint are coupled via a mechanism consisting of two crank-lever mechanisms. This removes one DoF from the system and forces the shoulder and first elbow rotation to move on a common trajectory. With this mechanical coupling it is possible to transfer forces from the shoulder to the elbow and therefore relocate a motor actuating the elbow to the shoulder. This concept is known from industrial robots with a parallelogram structure to relocate a motor closer to the base, reducing the inertia and gravitational load of the system [3].

Spring-Damper System: The elbow joint is connected via the central gear and a linear spring damper system with the forearm. The spring is transferring the moment created by the weight of the powertool towards the central gear and reduces oscillations originating from largely disturbing processes like drilling.

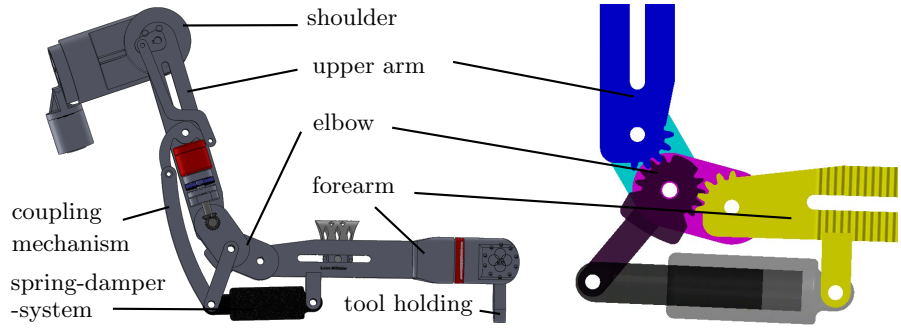

Fig. 1. CAD model of the complete exoskeleton mechanism (left) and detail of the elbow joint (right). © by Michael Winkler and BMBF consortium "Third Arm" [9].

3 Kinematics and Dynamics Modelling

In the following, the single steps to gain dynamics equations required for control and simulation purposes are laid out at the example of the exoskeleton system introduced in the previous Sec. 2. The required steps include the creation of the general kinematic model and coordinate definitions in Sec. 3.1 and the elimination of supernumerous coordinates in Sec. 3.2. These steps allow to derive the dynamics equations as shown in Sec. 3.3. The standard coordinate partitioning method is presented in Sec. 3.4 for comparison.

3.1 Kinematic Model and Coordinate Definition

All rotational axes of the mechanism are described as single-DoF joints using the modified Denavit-Hartenberg notation [8] to describe the position and orientation of the frames $\mathscr{F}_i$ attached to all rigid bodies $\mathscr{B}_i$ of the mechanism. The parameters describing the transformations between these frames are given in the left part of Fig. 2 next to a kinematic sketch of the mechanism in the right part of Fig. 2 with annotations for frames, bodies and coordinates to improve the confirmability of the table.

Due to the structure of the mechanism, the extended notation with anteceding link index $a(i)$, joint type marker σ_i (0 for rotational joint, 1 for prismatic joint, 2 for fixed connection) and actuation marker μ_i (0 for passive, 1 for active joint) is used. The marker μ_i is set to 1 for the joints representing the generalized coordinates, regardless whether they are actuated or not. The order of rotations $\boldsymbol{R}$ and translations $\boldsymbol{T}$ is given in the head row for γ_i to r_i. Only rows 1 to 15 of the table mark frames associated with rigid bodies. Rows 16-18 denote the first frame and rows 19-21 the second frame of the cut joints belonging to the three closed kinematic loops. The coordinates of the cut joint frames 16-18 are marked with a "$*$" and are not regarded further, since they are not needed for the dynamics. Additionally, the origins O_i of the coordinate systems are given in Fig. 2 for a better overview.

The mechanism is considered without the kinematic constraints by virtually cutting the loop-closing joints resulting in a tree structure [8] and removing the other constraints that are referred to as "user constraints" in [5]. According to [2], the coordinates $\boldsymbol{q} = \left(\boldsymbol{q}_1^{\mathrm{T}}\ \boldsymbol{q}_2^{\mathrm{T}}\right)^{\mathrm{T}}$ of the single-DoF joints of this unconstrained system can be separated into the generalized coordinates $\boldsymbol{q}_1$ and the dependent coordinates $\boldsymbol{q}_2$. The constant kinematics parameters can be grouped into a separate vector $\boldsymbol{p}_{\mathrm{kin}}$. Hereby the joint coordinates of the main structure are denoted with ρ, the coordinates and constant angles of the parallel coupling and support mechanism with η and the variable elongation of the spring with L_{s}.

Without loss of generality, the coordinates ρ and η can be grouped according to the joint type into rotational coordinates $\boldsymbol{q}_{\mathrm{R}}$ corresponding to $\sigma_i = 0$ and translational coordinates $\boldsymbol{q}_{\mathrm{T}}$ corresponding to $\sigma_i = 1$ with

$$\boldsymbol{q}_1 = \left(\boldsymbol{q}_{1\mathrm{R}}^{\mathrm{T}}\ \boldsymbol{q}_{1\mathrm{T}}^{\mathrm{T}}\right)^{\mathrm{T}}, \boldsymbol{q}_2 = \left(\boldsymbol{q}_{2\mathrm{R}}^{\mathrm{T}}\ \boldsymbol{q}_{2\mathrm{T}}^{\mathrm{T}}\right)^{\mathrm{T}}. \tag{1}$$

i	a	μ_i	σ_i	γ_i	ϵ_i	α_i	d_i	θ_i	r_i	O_i
				$\boldsymbol{R}_z$	$\boldsymbol{T}_z$	$\boldsymbol{R}_x$	$\boldsymbol{T}_x$	$\boldsymbol{R}_z$	$\boldsymbol{T}_z$	
1	0	1	0	0	0	0	0	$\rho_1 - \frac{\pi}{2}$	L_1	O_1
2	1	1	0	0	0	$-\frac{\pi}{2}$	0	$\rho_2 + \frac{\pi}{2}$	L_2	O_2
3	2	0	0	0	0	$\frac{\pi}{2}$	0	ρ_3	L_3	O_2
4	3	1	0	0	0	0	L_4	ρ_4	0	O_4
5	4	1	0	0	0	0	$2L_5$	ρ_5	0	O_5
6	5	0	0	0	0	0	$2L_6$	$\rho_6 + \frac{\pi}{2}$	0	O_6
7	6	1	0	0	0	$\frac{\pi}{2}$	0	ρ_7	L_7	O_7
8	2	0	2	0	0	$\frac{\pi}{2}$	0	$\eta_1 + \frac{\pi}{2}$	L_3	O_2
9	8	0	0	0	0	0	$-L_8$	$-\eta_2$	0	F
10	3	0	0	0	0	0	L_9	$-\eta_3$	0	D
11	10	0	0	0	0	0	L_{10}	$\pi - \eta_4$	0	C
12	4	0	0	0	0	0	$2L_5$	$\eta_5 - \frac{\pi}{2}$	0	O_5
13	12	0	2	0	0	0	0	$\eta_6 - \frac{\pi}{2}$	0	O_5
14	13	0	0	0	0	0	L_{11}	$3\frac{\pi}{2} - \eta_7$	0	B
15	14	0	1	0	0	$\frac{\pi}{2}$	0	0	L_s	A
16	10	0	0	η_{12}	0	0	L_{12}	$*$	0	E
17	13	0	0	0	0	0	L_{11}	$*$	0	B
18	6	0	0	η_{13}	0	0	L_{13}	$*$	0	A
19	9	0	2	0	0	0	L_{14}	0	0	E
20	11	0	2	0	0	0	L_{15}	0	0	B
21	15	0	2	$-\frac{\pi}{2}$	0	0	0	0	0	A

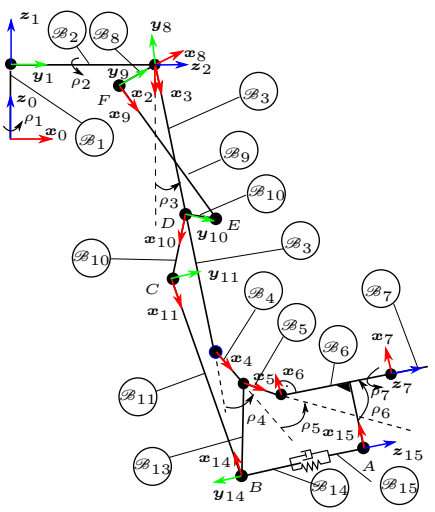

Fig. 2. Left: Table with the kinematic parameters of the considered structure. Right: Kinematic sketch of the mechanism with frames according to the table. Body numbers are indicated with circles. Bottom: Definitions of geometric parameters and coordinates.

In the following, the dependent coordinates are expressed in the explicit form

$$\boldsymbol{q}_2 = \boldsymbol{f}(\boldsymbol{q}_1), \quad \boldsymbol{q}_{2\text{R}} = \boldsymbol{f}_{\text{R}}(\boldsymbol{q}_1), \quad \boldsymbol{q}_{2\text{T}} = \boldsymbol{f}_{\text{T}}(\boldsymbol{q}_1) \tag{2}$$

as a function of the generalized coordinates $\boldsymbol{q}_1$. This will be used to determine the configuration $\boldsymbol{q}$ of the complete system depending on the pose of the main structure and to generate dynamics equations for simulation and model based controllers.

3.2 Elimination of Kinematic Constraints

The kinematic constraints of the system originate from the rolling condition of the elbow gears and the closed loop constraints in the coupling mechanism. The expressions for the different dependent coordinates in $\boldsymbol{q}_2$ can be calculated by intersecting circles "$\mathscr{C}$" resulting from the planar motion of the single parts of the four-bar-linkages. Calculating $\{C, C'\} = \mathscr{C}_1 \cap \mathscr{C}_2$ for the elbow-shoulder-coupling from Fig. 3 (a) gives equations for the dependent coordinates η_3 and

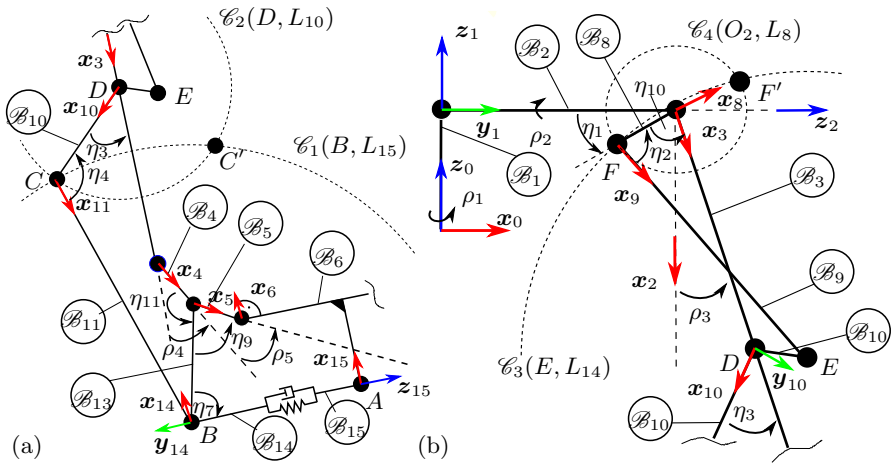

Fig. 3. Kinematic sketch of (a) the lower part and (b) the upper part of the mechanism with annotations for circles, angles and frames. Missing z-axes point out of the paper.

η_4, corresponding to the loop of transformation matrices in eq. (10). The intersection $\{F, F'\} = \mathscr{C}_3 \cap \mathscr{C}_4$ of the second set of circles in Fig. 3 (b) give ρ_3 and η_2, which also shows in the loop equation eq. (11). The variables η_5 and ρ_6 can be solved by equating the velocities of the pitch point of the two gear contacts from both sides. Solving the slider-crank mechanism of the forearm finally gives η_7 and L_s, also occurring in eq. (12). This provides the explicit definitions of the kinematic constraints defined in eq. (2), which are needed for the complete kinematic description of the frames and rigid bodies defined in Fig. 2.

It shall be emphasized that the explicit form of the rotatory coordinates $\boldsymbol{q}_{2\mathrm{R}}$ can be calculated using eq. (2). However also a specific implicit form can be defined with the angles as an argument of the sine and cosine functions with

$$\sin(\boldsymbol{q}_{2\mathrm{R}}) = \boldsymbol{f}_{\mathrm{Rsin}}(\boldsymbol{q}_1), \quad \cos(\boldsymbol{q}_{2\mathrm{R}}) = \boldsymbol{f}_{\mathrm{Rcos}}(\boldsymbol{q}_1). \tag{3}$$

The translatory coordinates $\boldsymbol{q}_{2\mathrm{T}}$ and velocities can still be used in the explicit form

$$\boldsymbol{q}_{2\mathrm{T}} = \boldsymbol{f}_{\mathrm{T}}(\boldsymbol{q}_1), \quad \dot{\boldsymbol{q}}_2 = \boldsymbol{f}_{\mathrm{diff}}(\boldsymbol{q}_1, \dot{\boldsymbol{q}}_1). \tag{4}$$

The specific implicit form of eq. (3) allows to produce simpler symbolic expressions when solving the set of equations for the unknown joint variables $\boldsymbol{q}_2$. This is facilitated by the use of the angle sum identities. The form in eq. (3) is called *trigonometric explicit form* in this paper since trigonometric functions are used as substitution variables and the form is explicit in these variables. The expressions of this form have a tendency to additions and products of sine and cosine terms. The explicit form in eq. (2) on the contrary contains nested expressions with arctangent functions, since every successive substitution of the angles in the calculation of the closed loops results in a new nested term of the arctangent

function in the expression. The latter has shown to be inefficient without further optimizations for processing with Maple, a computer algebra system that was used to create the symbolic expressions and to generate code for dynamics functions.

3.3 Dynamics Model using Explicit Constraints

The dynamics equations are derived in this work using Lagrange equations of the second kind. This method needs the Lagrangian

$$L(\boldsymbol{q}, \dot{\boldsymbol{q}}) = T(\boldsymbol{q}, \dot{\boldsymbol{q}}) - (U_\mathrm{g}(\boldsymbol{q}) + U_\mathrm{s}(\boldsymbol{q})) \tag{5}$$

expressed in minimal coordinates $\boldsymbol{q}_1$. Rotation matrices from all the frame transformations from Fig. 2 are used to calculate the kinetic energy T as well as the potential energies U_g from gravity and U_s from the spring. Separating the rotational and translational coordinates as introduced in eq. (1) yields the Lagrangian from eq. (5) in the structure

$$\begin{aligned} L(\boldsymbol{q}, \dot{\boldsymbol{q}}) &= L(\sin(\boldsymbol{q}_\mathrm{R}), \cos(\boldsymbol{q}_\mathrm{R}), \boldsymbol{q}_\mathrm{T}, \dot{\boldsymbol{q}}) \\ &= L(\sin(\boldsymbol{q}_{1\mathrm{R}}), \cos(\boldsymbol{q}_{1\mathrm{R}}), \boldsymbol{q}_{1\mathrm{T}}, \dot{\boldsymbol{q}}_1, \sin(\boldsymbol{q}_{2\mathrm{R}}), \cos(\boldsymbol{q}_{2\mathrm{R}}), \boldsymbol{q}_{2\mathrm{T}}, \dot{\boldsymbol{q}}_2). \end{aligned} \tag{6}$$

After substitution of eq. (3) and eq. (4) into (7), the Lagrangian

$$\begin{aligned} L(\boldsymbol{q}_1, \dot{\boldsymbol{q}}_1) = L(&\sin(\boldsymbol{q}_{1\mathrm{R}}), \cos(\boldsymbol{q}_{1\mathrm{R}}), \boldsymbol{q}_{1\mathrm{T}}, \dot{\boldsymbol{q}}_1, \\ &\boldsymbol{f}_{\mathrm{R}\sin}(\boldsymbol{q}_1), \boldsymbol{f}_{\mathrm{R}\cos}(\boldsymbol{q}_1), \boldsymbol{f}_\mathrm{T}(\boldsymbol{q}_1), \boldsymbol{f}_\mathrm{diff}(\boldsymbol{q}_1, \dot{\boldsymbol{q}}_1))) \end{aligned} \tag{7}$$

is represented in the minimal coordinates $\boldsymbol{q}_1$. This allows using the Lagrange equations

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L(\boldsymbol{q}_1, \dot{\boldsymbol{q}}_1)}{\partial \dot{\boldsymbol{q}}_1} - \frac{\partial L(\boldsymbol{q}_1, \dot{\boldsymbol{q}}_1)}{\partial \boldsymbol{q}_1} = \boldsymbol{\tau}_1^c, \tag{8}$$

to get the inverse dynamics joint torques[4] $\boldsymbol{\tau}_1^c$ of the constrained system in coordinates $\boldsymbol{q}_1$. This allows to obtain the dynamics equations in explicit form

$$\boldsymbol{M}_1\ddot{\boldsymbol{q}}_1 + \boldsymbol{c}_1(\boldsymbol{q}_1, \dot{\boldsymbol{q}}_1) + \boldsymbol{g}_1(\boldsymbol{q}_1) + \boldsymbol{\tau}_{1\mathrm{s}}(\boldsymbol{q}_1) = \boldsymbol{\tau}_1^c, \tag{9}$$

where $\boldsymbol{M}_1$, $\boldsymbol{c}_1$, $\boldsymbol{g}_1$, $\boldsymbol{\tau}_{1\mathrm{s}}$ denote the inertia matrix, and the vectors of centrifugal/Coriolis torques, gravity torques and spring torques, respectively.

3.4 Dynamics Model using Implicit Constraints

For comparison of the computational efficiency of the approach proposed in the previous section, the standard method [2, 7, 8, 5] of using kinematic constraints in implicit form is briefly summarized.

[4] joint forces in the case of prismatic joints are also referred to as torques for the sake of readability

The presented exoskeleton contains three kinematic loops which are cut in the joints corresponding to the rows 16, 17 and 18 of Fig. 2 (points E, B and A). These joints do not articulate any bodies and are associated with a second virtual frame in the entries 19, 20 and 21 of Fig. 2 that are fixed to the other link of the respective cut joint [8, 5]. For the given mechanism, three vector loops can be defined with the homogeneous transformation matrices

$$ {}^{3}\boldsymbol{T}_{4}(\rho_4){}^{4}\boldsymbol{T}_{12}(\eta_5){}^{12}\boldsymbol{T}_{13}{}^{13}\boldsymbol{T}_{17}(*) = {}^{3}\boldsymbol{T}_{10}(\eta_3){}^{10}\boldsymbol{T}_{11}(\eta_4){}^{11}\boldsymbol{T}_{20} \tag{10} $$

$$ {}^{2}\boldsymbol{T}_{3}(\rho_3){}^{3}\boldsymbol{T}_{10}(\eta_3){}^{10}\boldsymbol{T}_{16}(*) = {}^{2}\boldsymbol{T}_{8}{}^{8}\boldsymbol{T}_{9}(\eta_2){}^{9}\boldsymbol{T}_{19} \tag{11} $$

$$ {}^{4}\boldsymbol{T}_{5}(\rho_5){}^{5}\boldsymbol{T}_{6}(\rho_6){}^{6}\boldsymbol{T}_{18}(*) = {}^{4}\boldsymbol{T}_{12}(\eta_5){}^{12}\boldsymbol{T}_{14}(\eta_7){}^{14}\boldsymbol{T}_{15}(L_\mathrm{s}){}^{15}\boldsymbol{T}_{21}. \tag{12} $$

Each of these equations provides two linearly independent entries in the translational part of the transformation matrix. Therefore, neglecting the rotational part, it is not required to take the angles "$*$" of the cut joints into consideration. Together with the two gear constraints the eight implicit constraints can be written as $\boldsymbol{h}(\boldsymbol{q}_1, \boldsymbol{q}_2) = \boldsymbol{0}$ with the partial derivatives

$$ \boldsymbol{J}_1(\boldsymbol{q}_1, \boldsymbol{q}_2) = \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{q}_1}, \quad \boldsymbol{J}_2(\boldsymbol{q}_1, \boldsymbol{q}_2) = \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{q}_2}. \tag{13} $$

The projection matrix $\boldsymbol{W}$ from [2, 7] is used in the form

$$ \boldsymbol{W} = \frac{\partial \boldsymbol{q}}{\partial \boldsymbol{q}_1} = \begin{pmatrix} \boldsymbol{E} \\ \frac{\partial \boldsymbol{q}_2}{\partial \boldsymbol{q}_1} \end{pmatrix} = \begin{pmatrix} \boldsymbol{E} \\ -\boldsymbol{J}_2^{-1}\boldsymbol{J}_1 \end{pmatrix} \tag{14} $$

in order to obtain the joint torques $\boldsymbol{\tau}_1^c$ of the constrained system in minimal coordinates $\boldsymbol{q}_1$ as

$$ \boldsymbol{\tau}_1^c = \boldsymbol{W}^{\mathrm{T}}\boldsymbol{\tau}. \tag{15} $$

The matrix from eq. (14) still contains the dependent joint coordinates $\boldsymbol{q}_2$ which have to be calculated numerically with iterative methods using the Jacobians from eq. (13) or with the explicit form in eq. (2). The advantage of this method is the availability of efficient tools to symbolically compute the inverse dynamics $\boldsymbol{\tau}$ of the unconstrained system. Further, for most systems the passive joint constraint Jacobian $\boldsymbol{J}_2$ is sparse and can be efficiently inverted symbolically [7].

4 Computational Costs of the proposed Methods

The inverse dynamics of the constrained system in eq. (9) was derived using the different approaches presented in this paper. The computational efforts to compute the terms $\boldsymbol{g}_1$ and $\boldsymbol{M}_1$ are compared in Tab. 1. The used methods are

1. *Trigonometric elimination* ("TE") of the constraints by substituting with eq. (3) as described in Sec. 3.3,
2. *Direct elimination* ("DE") of the constraints by directly using eq. (2) to substitute the dependent joint coordinates in eq. (5),
3. using *implicit constraint equations* ("IC") as described in Sec. 3.4.

The expressions are generated with Maple using the procedure `optimize` with `tryhard` and the number of operations is counted with the procedure `cost`.

Table 1. Comparison of the computational effort for different implementations of the gravitational load $\boldsymbol{g}_1(\boldsymbol{q}_1)$ and the mass matrix $\boldsymbol{M}_1(\boldsymbol{q}_1)$.

	$\pm$	$*$	$/$	$\sqrt{}$	sin	atan	$:=$	Sum	t_{Gen}
Factor	1	1	4	5	10	15	1	(weighted)	
Method	gravitational load $\boldsymbol{g}_1(\boldsymbol{q}_1)$								
1: TE	710	985	12	3	22	0	365	**2343**	1.4 h
2: DE	491	646	24	3	38	5	358	**1975**	0.4 h
3: IC	462	554	6	0	50	0	264	**1804**	2 s
$\boldsymbol{W}$	40	66	6	0	22	0	36	386	1 s
$\boldsymbol{g}$	422	488	0	0	28	0	228	1418	1 s
Method	mass matrix $\boldsymbol{M}_1(\boldsymbol{q}_1)$								
1: TE	1138	1671	28	3	22	0	598	**3754**	2.0 h
2: DE	1142	1656	25	3	32	5	591	**3899**	2.0 h
3: IC	1230	1730	6	0	48	0	537	**4001**	17 s
$\boldsymbol{W}$	40	66	6	0	22	0	36	386	1 s
$\boldsymbol{M}$	1190	1664	0	0	26	0	501	3615	16 s

To give an estimate of the computational cost, the different operations are counted in a weighted sum. An estimation of the lower bound for the cost factor of the floating point operation types relative to simple additions is adapted from [10], [11] and given in the second row in Tab. 1. The exact time for computation depends on the concrete hard- and software and further optimizations of the implementation.

Since the inverse dynamics of the open loop tree structure can be calculated very efficiently and the sparsity of the projection matrix $\boldsymbol{W}$ is high, benefiting the symbolic inversion, the standard method 3 has an appr. 10 % lower computational effort compared to method 2 for the gravitational load as shown in the upper part of Tab. 1. Method 2 outperforms method 1 when using an optimized order of symbolic substitutions. The time t_{Gen} needed to optimize and generate the code is several orders of magnitude slower than the time for method 3.

As summarized in the lower part of Tab. 1, the elimination methods 1 and 2 perform slightly better than the standard method 3 for the inertia matrix. This can be explained by the high number of DoF for the 15×15 matrix $\boldsymbol{M}$ from method 3 compared to a smaller 5×5 matrix $\boldsymbol{M}_1$ for methods 1 and 2.

5 Conclusions and future work

This paper presented a novel concept for a force assistance exoskeleton to support workers in carrying and using powertools and its dynamics modeling. The details of the solution of the kinematic constraints of the multi-loop mechanism focus on the elimination of trigonometric expressions of dependent variables. This reduces the computational load when creating symbolic code for the terms of the dynamics equation with a similar efficiency of the output. However, the presented approach does not completely reach the computational performance of standard solutions for the given mechanism. On the upside it can be used to

test the correctness of dynamics models with the standard solution by offering a different way of approaching the problem. This may also give new ideas to researchers working in the area of kinematics and dynamics modeling.

In future works, the featured approach of implementing kinematic constraints will be applied to different serial-chain industrial robots with parallel mechanisms invoking kinematic constraints, i. e. hybrid robots. The results will be compared to the classical approach using the projection of constraint forces and with non-trigonometric elimination of the constraints.

Acknowledgements

The presented work was funded by the Federal Ministry of Education and Research of Germany (BMBF) under grant number 16SV6175 and has also received funding from European Union's Horizon 2020 research and innovation programme under grant agreement No. 688857 ("SoftPro").

References

1. Wehage, R., Haug, E.: Generalized coordinate partitioning for dimension reduction in analysis of constrained dynamic systems. Journal of mechanical design **104**(1), 247–255 (1982). DOI 10.1115/1.3256318
2. Nakamura, Y., Ghodoussi, M.: Dynamics computation of closed-link robot mechanisms with nonredundant and redundant actuators. IEEE Transactions on Robotics and Automation (1989). DOI 10.1109/70.34765
3. Luh, J., Zheng, Y.F.: Computation of input generalized forces for robots with closed kinematic chain mechanisms. IEEE Journal on Robotics and Automation **1**(2), 95–103 (1985). DOI 10.1109/JRA.1985.1087008
4. Udwadia, F.E., Kalaba, R.E.: A new perspective on constrained motion. Proceedings: Mathematical and Physical Sciences pp. 407–410 (1992). DOI 10.1098/rspa.1992.0158
5. Samin, J.C., Fisette, P.: Symbolic modeling of multibody systems, vol. 112. Springer Science & Business Media (2013). DOI 10.1007/978-94-017-0287-4
6. Khalil, W., Vijayalingam, A., et al.: OpenSYMORO: An open-source software package for Symbolic Modelling of Robots. In: IEEE/ASME International Conference on Advanced Intelligent Mechatronics, pp. 1206–1211. Besançon, France (2014). DOI 10.1109/AIM.2014.6878246
7. Park, F., Choi, J., Ploen, S.: Symbolic formulation of closed chain dynamics in independent coordinates. Mechanism and machine theory **34**(5), 731–751 (1999). DOI 10.1016/S0094-114X(98)00052-4
8. Khalil, W., Bennis, F.: Symbolic calculation of the base inertial parameters of closed-loop robots. The International journal of robotics research **14**(2), 112–128 (1995). DOI 10.1177/027836499501400202
9. Nülle, K., Schappler, M., et al.: Projektabschlussbericht "3. Arm". Tech. rep., Mechatronik Zentrum Hannover (2017). DOI 10.2314/GBV:1014030161
10. Atkinson, L.: A simple benchmark of various math operations (2014). Online: http://www.latkin.org/blog/2014/11/09/, accessed 20.08.2018
11. Hindriksen, V.: How expensive is an operation on a CPU? (2012). Online: https://streamhpc.com/blog/2012-07-16/, accessed 20.08.2018