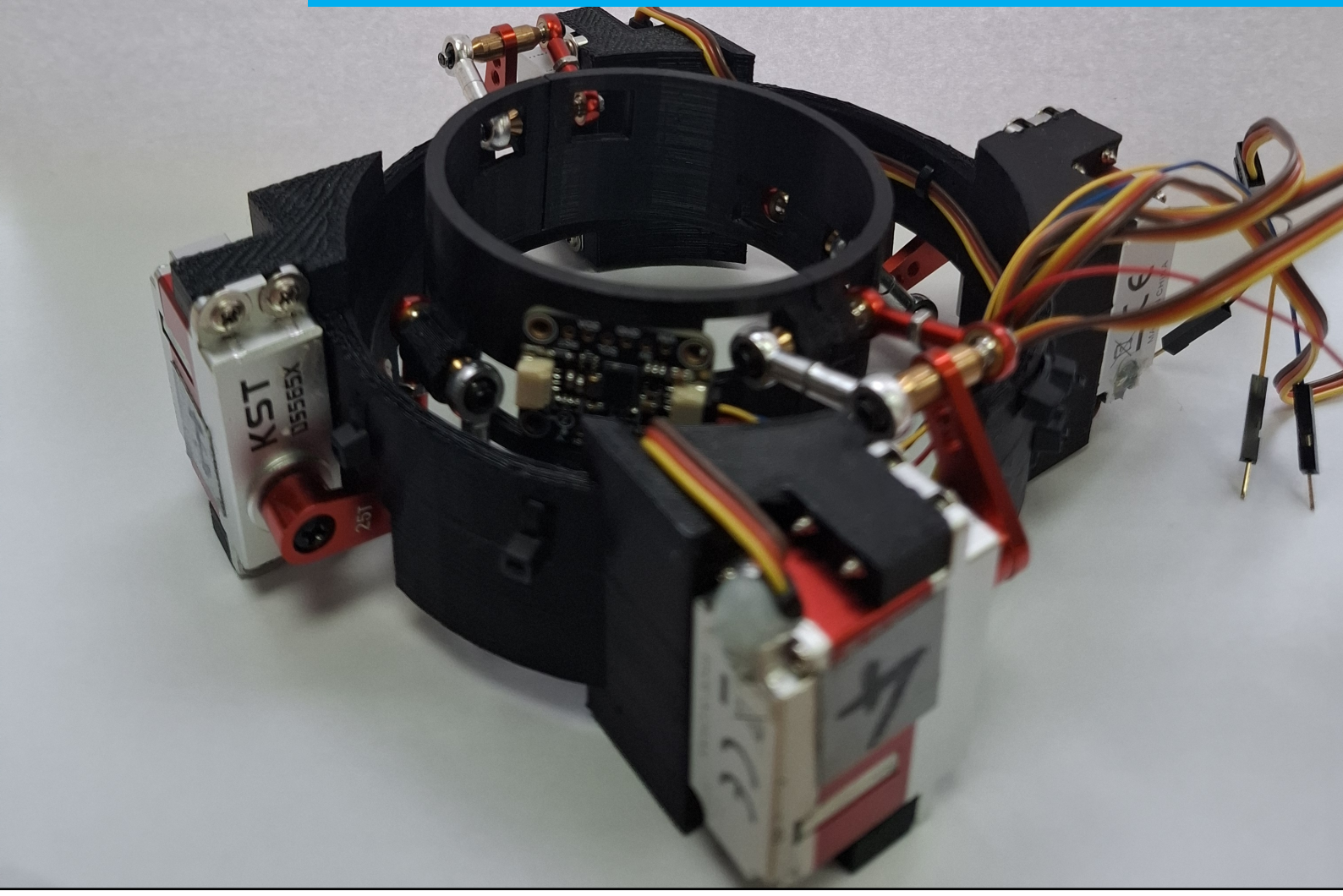
Department of Precision and Microsystems Engineering

DESIGN AND VALIDATION OF A LIGHTWEIGHT PARALLEL MANIPULATOR FOR UNGROUNDED 3-DOF TRANSLATIONAL WRIST PERTURBATIONS WITH ORIENTATION SENSING

MATTHIAS WEERHEIM

Report no : 2025.075
Coaches : Dr.ir. J.P. Meijaard; Dr.ir. W. Mugge
Professor : Prof.dr.ir. J.L. Herder
Specialisation : Mechatronic System Design
Type of report : Master Thesis
Date : December 2, 2025



Design and Validation of a Lightweight Parallel Manipulator for Ungrounded 3-DoF Translational Wrist Perturbation with Orientation Sensing

by

Matthias Weerheim

Daily supervisor:	Dr.ir. J.P. Meijaard
Supervisors:	Dr.ir. W. Mugge
	Prof.dr.ir. J.L. Herder
Project Duration:	September, 2024 - December, 2025
Faculty:	Mechanical Engineering, Delft
Deparment:	Precision and Microsystems Engineering
Specialisation:	Mechatronic System Design

TU Delft

Preface

This thesis contributes to the fields of mechanisms and neuromuscular control for system identification. *"Design and Validation of a Lightweight Parallel Manipulator for Ungrounded 3-DoF Translational Wrist Perturbation with Orientation Sensing"* investigates a wearable measurement device that can generate ungrounded forces in prescribed directions to perturb the human arm while a subject performs everyday tasks. The device is intended for medical and haptic applications and includes orientation sensing to characterize the applied perturbations.

The work presented here was carried out to fulfill the requirements of the High-Tech Engineering track of the Master's program in Mechanical Engineering at Delft University of Technology. The project was organized as a joint effort between the High-Tech Engineering and Biomechanical Design departments. Following an initial discussion with Jaap Meijaard, I became interested in a project that combined mechanism analysis with a specific real-world application. Winfred Mugge provided the haptic and medical perspective that guided the problem formulation. Just Herder supervised the mechatronic system design aspect, and Jaap Meijaard acted as the daily supervisor, also with a lot of experience in mechanism design. His regular, often inspiring discussions were essential when exploring mechanism configurations and experimental setups.

Research related to this topic was initiated in 2022 by Robbert Koene, who developed a proof-of-concept prototype based on the problem statement provided by Winfred Mugge. Koene's prototype and results provided both motivation and a starting point for this thesis. Building on that foundation, this project focused on improving several aspects of the concept: deriving analytical inverse kinematics, using simulations to optimize main dimensions, and integrating an orientation sensor for enhanced estimation of the device's state.

I would like to thank my supervisors for their stimulating and occasionally divergent perspectives, which ultimately helped me decide independently which directions to pursue. Special thanks to Volkert van der Wijk for sharing insights and documentation from Robbert Koene's earlier work, which proved highly valuable for identifying gaps and opportunities. I am also grateful to Jacques Brenkman for generously lending measurement equipment and for the constructive discussions on the prototype and experimental setup.

Finally, I thank my family and friends for their continued interest and support throughout this project. Although they were not always familiar with the technical details, their respect for the time and effort invested, and their questions that helped me return to the fundamentals, were greatly appreciated.

Matthias Weerheim
Delft, December 2025

Summary

Measurements of human limbs with system identification can improve understanding of human movement, enable robotic replication of human behavior, and quantify rehabilitation progress. Current system identification setups that apply force or position perturbations are typically bulky and confined to laboratory environments. A wearable device capable of perturbing and measuring during daily activities can therefore provide more valid data.

A previous master's thesis presented a proof-of-concept mechatronic system designed to generate controlled force perturbations in three degrees of freedom for the wrist. To support this application, a literature review was conducted on wearable sensor devices, filtering techniques, calibration procedures, and validation methods for wrist orientation measurement. Based on this review, a compact and lightweight 4-RUU parallel manipulator was proposed; it provides three translational DoF around the wrist and can generate ungrounded forces. This 4-RUU architecture offers improved singularity avoidance, a more favorable torque distribution, and enhanced controllability. The inverse kinematics of the mechanism were derived analytically.

A Simulink model (MATLAB) of the mechanism was developed and used to optimize the main dimensions for a suitable servomotor. An improved prototype was subsequently fabricated with a total mass of 0.30 kg, of which 0.25 kg is the moving platform. The inner ring has a maximum displacement capacity of 20 mm. Using the analytical inverse kinematics, an open-loop feedforward controller with separate calibration was implemented to produce controlled perturbations from four servomotors.

Force measurements demonstrated repeatable force-generation profiles along each principal axis and for combined-axis excitations, with a minimum force peak of 5.0 N. The lack of end-effector acceleration, due to limiting servo performance, resulted in a loss of forces in the mechanism ranging from 14 % to 50 %. Orientation estimates obtained from a BNO-055 inertial and magnetic measurement unit (IMMU) processed with a Madgwick filter were fast and accurate in real time, yielding a maximum total angle error of 5°.

Contents

1

Literature review: Non-obstructive wrist orientation measurements: A review of wearable sensor devices, filtering, calibration, and validation methods

Non-obstructive wrist orientation measurements: A review of wearable sensor devices, filtering, calibration, and validation methods

Matthias Weerheim, Student

***Abstract*— Wearable devices capable of perturbing and measuring wrist orientation during daily activities are crucial, as traditional measurement systems are often constrained to controlled environments, limiting natural movement. While a proof-of-concept prototype can apply ungrounded forces to the wrist, an integrated system for measuring wrist orientation is lacking. This review examines state-of-the-art sensors, filtering, calibration, and validation techniques and identifies which are suitable for the ungrounded wrist perturbator. Techniques are identified as suitable when capable of handling nonlinear motions in real-time. Studies on non-obstructive wearable devices for tracking upper limb orientation were analyzed. First, three wearable sensing devices were reviewed. Sensor fusion algorithms combining data from these sensors enhance accuracy and mitigate errors. Among the evaluated algorithms, the extended Kalman filter and the Madgwick filter are both suitable for the application. However, the Madgwick filter shows more promising results, offering lower RMSE and requiring fewer computational steps for implementation. For calibration, both static and dynamic calibration are recommended for mitigating static and dynamic errors, respectively. Static calibration is conducted with a steady pose and dynamic calibration entails swinging the arm multiple times. To ensure data reliability, recalibration is necessary every 15 minutes. Validation, required only once, can be performed using an optical motion capture system. Both passive and active markers are suitable for this application, with active markers offering the advantage of avoiding misidentification while requiring extra wiring. These techniques integrated into the wrist perturbator device can support real-time measurements during daily activities.**

I. Introduction

In the medical field, there is a need for compact, ambulatory sensor systems capable of measuring the kinematics of human limbs. For example, for assessment of the rehabilitation progress of stroke patients [1]. But also relevant to the rise of virtual reality experiences for applications in entertainment, gaming, and training environments [2]. Traditional methods for measuring and characterizing the mechanical properties of human limbs typically require bulky and expensive setups confined to controlled laboratory environments [3][4]. A significant limitation of these methods is that they restrict participants' natural movement, making it impossible to measure mechanical properties during normal daily activities.

A proof-of-concept prototype developed by Koene et al. [5] aims to generate ungrounded forces through a wearable wrist device. However, this ungrounded wrist perturbator prototype does not measure the orientation of the wrist and perturbation motion, data that would be valuable for determining limb impedance and orientation. This leads to the following research question: What are the state-of-the-art sensors, filtering, calibration, and validation techniques for non-obstructively determining wrist orientation, and which of these are suitable for the ungrounded wrist perturbator? The techniques are identified as suitable when they can handle nonlinear motions. Also should be feasible to compute orientations in real-time, with a latency in the order of $\mathcal{O}(10^1$ ms$)$ or less, while the device is worn on the wrist. Such that the sensor system can provide information on how to perturb the wrist.

To answer this question and review the literature, a strategical search method for relevant research papers is developed as described in Section II. The results of this search, along with comparison tables are presented in Section III and discussed in Section IV. This discussion provides insights into the feasible filtering, calibration, and validation methods tailored for the application of the ungrounded wearable wrist device. Finally, Section V concludes the literature review. An appendix is also included, offering an overview table of the reviewed papers, detailing the filtering, validation, and calibration methods employed. In combination with the resulting root-mean-square errors from the experiments.

II. Method

The databases Scopus and Google Scholar were utilized to conduct the search, employing specific terms and Boolean operators as outlined in Table I. This review exclusively considers studies related to upper limb measurements. Because certain solutions tailored to lower limb orientation tracking, such as contact recognition with the ground [6] or gait analysis [7], are not applicable to upper limb tracking. Regarding criteria relative to the wearability of the device, all sensors must facilitate non-obstructive measurements to ensure that the device does not hinder natural motion or comfort during use. The search was limited to English-language publications, resulting in an initial set of 307 articles.

Abstracts and figures from the retrieved papers were reviewed to assess their relevance to the aim of this study. Only papers with a mechanical focus on human upper limb motion estimation were included in the final selection. Although the search terms targeted upper-limb studies, some lower-limb-focused articles were retrieved and manually excluded. Additionally, articles focused on machine learning, deep learning, and gesture recognition were excluded, as well as papers that did not meet the non-obstructive measurement criteria, which included motor control/actuation and wrong sensor usage.

Following these exclusions, 30 relevant articles remained. Additionally, four 'snowball' papers were identified through references within the selected articles. Figure 1 illustrates the exclusion process based on the outlined criteria.

TABLE I: Overview of search terms used. The columns are combined with OR Boolean operators, while the rows are combined with AND Boolean operators. An asterisk (*) replaces 0 or more characters.

	AND		
OR	wearable sens*	upper limb*	human motion
	inertial sens*	shoulder*	human movement
	wearable system*	arm*	orientation
	Wearable device	elbow*	
		forearm*	
		hand*	
		wrist*	

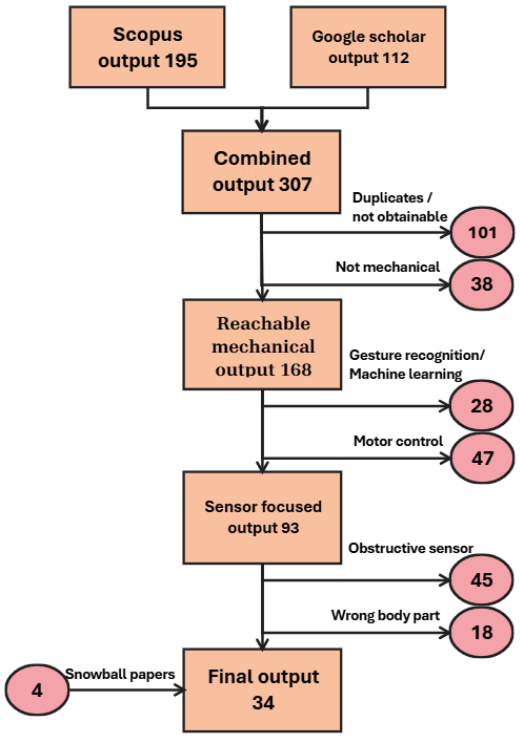


Fig. 1: 307 papers were obtained from Scopus and Google Scholar with the search terms outlined in Table I. Initially, duplicate entries were removed, along with 6 papers that were no longer available in the databases, a reduction of 101 papers. Additionally, 38 articles unrelated to mechanical systems, such as those focusing on neuroscience or computer science, were excluded. Papers addressing machine learning or gesture recognition were also eliminated, as were studies involving motor control or actuation systems that render the system obstructive. Similarly, papers utilizing obstructive sensors, such as goniometers or flexible sensors, were excluded. Studies measuring outcomes unrelated to the upper limb were also deemed irrelevant. Finally, incorporating four snowball papers identified during the research, the final selection comprised 34 articles.

The identified papers are analyzed in the following section, focusing on four key aspects of wearable orientation measurements: wearable sensing devices, filtering methods, calibration, and validation. The findings of these last three key aspects are summarized in tables that evaluate various relevant parameters. First, the tables indicate which sensing devices are compatible with the respective methods. The analysis then evaluates the capability of these methods to address nonlinear as well as linear problems, and it identifies the feasible orientation representation techniques (Euler angles and/or quaternions).

The accuracy of filtering methods is assessed using the root-mean-square errors (RMSE) in degrees reported in the papers, relative to the used filter. Eventually, the mean for the values found in every filtering method is computed. Note that not every paper reports

the RMSE and sometimes it was computed with the given results from the specific paper. When there is a comparison between real-time and offline estimation, real-time results are always reported, because we consider a real-time application. Calibration accuracy is evaluated by identifying the specific errors mitigated by each calibration method. The accuracies of the validation techniques are given in an order of accuracy based on multiple datasheets from the identified validation techniques/devices.

Additionally, the number of steps required to integrate each filter, calibration, and validation method is identified. These steps may include the mathematical computations (for one state update cycle) or the setup procedures for specific devices. The real-time feasibility of filtering methods is analyzed using Big $\mathcal{O}$-notation, which classifies the computational complexity of algorithms based on how their runtime scales with state vector size (n). The real-time feasibility of calibration methods is determined by assessing whether they can be applied while wearing the sensors on the body. For validation methods, an order of computation time is given based on data from multiple datasheets.

III. Results

A. Wearable sensing devices

This subsection explains three fundamental wearable sensing devices—accelerometers, gyroscopes, and magnetometers—along with their respective advantages and limitations.

1) Accelerometers: An accelerometer measures the proper linear acceleration of an object along the x, y, and z axes relative to an observer in free fall. It provides acceleration data in terms of m/s^2 or g-force. To interpret accelerometer data accurately, two coordinate frames must be established: the global navigation coordinate frame (n) and the sensor body frame (b). The spatial orientation of the rigid body can be determined by defining the axis orientation of the body frame relative to the navigation frame.

The measurements of accelerometers in the body frame (b) can be represented as

$$\boldsymbol{a}_m = \boldsymbol{C}_n^b \boldsymbol{M}^a (\boldsymbol{a} + \boldsymbol{g}) + \boldsymbol{b}_a + \boldsymbol{n}_a, \tag{1}$$

where $\boldsymbol{a}_m$ is the measured acceleration vector, $\boldsymbol{C}_n^b$ denotes the orientation rotation matrix defining the transformation from the navigation frame to the body frame, $\boldsymbol{g} \approx \begin{bmatrix} 0, & 0, & -9.81 \end{bmatrix}^T$ the gravity vector and $\boldsymbol{a} \in R^3$ is the inertial acceleration of the body, respectively, expressed in the navigation frame. $\boldsymbol{M}^a$ is a 3 x 3 matrix that scales the accelerometers outputs; $\boldsymbol{b}_a$ is the vector of accelerometer bias; $\boldsymbol{n}_a \in R^3$ is the Gaussian noise vector with zero-means [8].

Despite calibration efforts, accelerometers are still prone to errors related to noise and temperature fluctuations. Additionally, they can only detect linear motion, necessitating the use of a gyroscope for rotational measurements.

2) Gyroscopes: Gyroscopes measure angular velocity, capturing the rate of rotation around the x, y, and z axes. They typically output data in radians per second or degrees per second. A key limitation is integration drift, a cumulative error resulting from small noise offsets and biases in angular velocity measurements. El-Gohary et al. [9] demonstrated that integrating gyroscope data can result in a drift of 10–25° after just one minute, highlighting the importance of calibration and correction to maintain accuracy over time. The general form of gyroscope measurements is expressed as:

$$\boldsymbol{\omega}_m = \boldsymbol{\omega} + \boldsymbol{b}_g + \boldsymbol{n}_g, \tag{2}$$

where $\boldsymbol{\omega}_m$ is the measured angular velocity, $\boldsymbol{\omega}$ is the true angular velocity vector, $\boldsymbol{b}_g$ is gyroscope bias, and $\boldsymbol{n}_g$ is the noise vector that supposed to be Gaussian with zero-means. Temperature variations significantly affect gyroscope readings by introducing offsets [10]. However, during stable environmental conditions, a calibration step can effectively compensate for these offsets.

3) Magnetometers: Magnetometers measure the Earth's magnetic field strength and direction, providing a heading relative to the true north and remaining unaffected by gravitational forces. The magnetic field vector in the navigation frame, represented as $\boldsymbol{H}_h$ is used to derive directional information, with measurements in the body frame b expressed as:

$$\boldsymbol{h}_m = \boldsymbol{C}_n^b \boldsymbol{M}^h \boldsymbol{H}_h + \boldsymbol{b}_h + \boldsymbol{n}_h, \tag{3}$$

Where $\boldsymbol{M}^h$ is a 3 x 3 matrix scaling the magnetometer outputs, $\boldsymbol{b}_h$ denotes the disturbance vector including magnetometer bias and magnetic effects, and $\boldsymbol{n}_h \in R^3$ is Gaussian noise with zero mean.

Magnetometers are vulnerable to interference from nearby ferromagnetic metals and electronic devices generating magnetic fields. Such disturbances can compromise the stability of the magnetic field and the accuracy of orientation measurements. Therefore, calibration during commissioning is essential to ensure reliable readings.

B. Filtering methods

Linear acceleration, angular velocity, and magnetic field vectors are used in various sensor fusion algorithms to estimate the three-dimensional orientation of inertial and magnetic measurement units (IMMUs) relative to a global coordinate system. Not all filtering methods incorporate all sensor data; magnetometer data is excluded in certain filters.

1) Strapdown integration: Strapdown integration is the simplest method for orientation determination. This process updates the orientation, velocity, and position of a device, based on accelerometer and gyroscope data within a dynamic reference frame at a defined sampling frequency [11].

The first step in this process involves determining the device's orientation, which evolves over time as the device rotates. Orientation is represented by a rotation matrix derived from the gyroscope output at time step k:

$$\dot{\boldsymbol{C}}_{n,k}^{b} = \boldsymbol{C}_{n,k}^{b} \cdot \boldsymbol{\Omega}_{k}^{b}, \tag{4}$$

where $\boldsymbol{\Omega}_{k}^{b}$ is the skew-symmetric matrix form of the gyroscope vector.

$$\boldsymbol{\Omega}_{k}^{b} = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \tag{5}$$

$$\boldsymbol{C}_{n,k+1}^{b} = \boldsymbol{C}_{n,k}^{b} + \dot{\boldsymbol{C}}_{n,k}^{b} \Delta t \tag{6}$$

The accelerometer measures acceleration in the device's local body frame. To express this acceleration in the global frame, the data must be rotated using the orientation matrix obtained from gyroscope integration.

$$\boldsymbol{a}_{n,k} = \boldsymbol{C}_{n,k}^{b} \cdot \boldsymbol{a}_{b,k} - \boldsymbol{g} \tag{7}$$

Once the acceleration is expressed in the global frame, the updates for velocity and position are calculated as follows:

$$\boldsymbol{v}_{k+1} = \boldsymbol{v}_{k} + \boldsymbol{a}_{n,k} \Delta t \tag{8}$$

$$\boldsymbol{p}_{k+1} = \boldsymbol{p}_{k} + \boldsymbol{v}_{k+1} \Delta t \tag{9}$$

This integration process, consisting of three steps, repeats continuously, with each incremental change in angle updating the orientation. The mean RMSE accuracy found across the used literature is 5.8°, based on the results of two papers. The computational complexity of this algorithm is $\mathcal{O}(n)$ where n is the size of the state vector, making it a fairly simple and real-time feasible filtering method. However, since the gyroscope is prone to drift, the integrated angle can accumulate errors over time without correction. To mitigate these errors, more advanced filters combine data from accelerometers, gyroscopes, and sometimes magnetometers, correcting measurements for better accuracy.

The output of the calculations is presented in 3D space using either Euler angles or quaternions, each with distinct properties and use cases. Euler angles describe rotation around the principal axes (x, y, and z) in a defined sequence, commonly referred to as roll, pitch, and yaw. Euler angles are prone to gimbal lock, a condition in which two rotational axes align, leading to a loss of one degree of freedom.

Quaternions provide an alternative, robust representation of 3D orientation, free from gimbal lock. A quaternion consists of four components: one scalar (ω) and three vector components (x, y, z). It is expressed as:

$$\boldsymbol{q} = \omega + xi + yj + zk, \tag{10}$$

where $\boldsymbol{i}$, $\boldsymbol{j}$, and $\boldsymbol{k}$ are the imaginary unit vectors. Quaternions encode both the rotation axis and angle. For a rotation by angle θ around a unit axis $\boldsymbol{v} = (v_x, v_y, v_z)$, the quaternion is:

$$\boldsymbol{q} = (\cos\frac{\theta}{2}, v_x \sin\frac{\theta}{2}, v_y \sin\frac{\theta}{2}, v_z \sin\frac{\theta}{2}). \tag{11}$$

Quaternions avoid the successive rotations required by Euler angles, eliminating gimbal lock. They also allow smooth and continuous transitions between orientations using Spherical Linear Interpolation (SLERP), which is particularly advantageous for orientation tracking [12]. Additionally, quaternion multiplication simplifies the combination of rotations, whereas combining Euler angles involves trigonometric calculations.

2) Complementary filtering: The complementary filter applies low-pass and high-pass filters to different sensor outputs. The fusion of the filtered data will effectively estimate the orientation [13]. The principle of the complementary filter is illustrated in Figure 2.

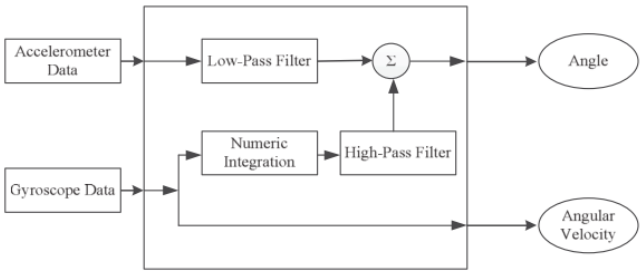


Fig. 2: Complementary filter algorithm block diagram. The gyroscope data gets high-pass filtered after numerical integration. Accelerometer data is low-pass filtered, by combining both outputs the orientation of the sensor is obtained.

Accelerometers are immune to external accelerations not originating from gravity, while gyroscopes are prone to significant drift over time [14]. The mathematical model used to derive the angles of a sensor with the complementary filter can be represented as:

$$\theta_{t+1} = \alpha \cdot (\theta_t + \omega_g \cdot \Delta t) + (1 - \alpha) \cdot \theta_{acc}, \tag{12}$$

where θ_t is the tilting angle (pitch or roll) at a particular time instance and θ_{t+1} is the estimated tilting angle at the next time instance, α is the filter gain, ω_g is the angular velocity from the gyroscope, Δt is the time differential (sampling period), and θ_{acc} is the angle computed from the accelerometer data, given the following equation for 3D space.

$$\theta_{acc} = \arctan\left(\frac{\sqrt{a_x^2 + a_y^2}}{a_z}\right) \tag{13}$$

This computes the angle between the gravity vector and the horizontal plane.

The complementary filter relies on the gyroscope data for short-term accuracy but incorporates accelerometer data, after it is numerically integrated, to correct for long-term drift. However, IMUs need to be calibrated before each use to avoid incorrect measurements. The parameter α, typically between 0 and 1, determines the weighting between the gyroscope and the accelerometer. Incorrectly weighted data can lead to erroneous results. The value of α is typically computed using the following equation:

$$\alpha = \frac{\tau}{\tau + \Delta t}, \tag{14}$$

where τ is the time constant of the filter.

For a low-pass filter, signals with periods much longer than the time constant pass through the filter unchanged, while signals with shorter periods are filtered out. The reverse is true for a high-pass filter. During each time interval, the gyroscope data is first integrated with the current angle and then combined with the low-pass accelerometer data. This approach effectively merges low- and high-frequency data to mitigate errors in a simple way. But this filter is limited for linear estimations around two rotation axes, pitch and roll.

The RMSE is based on three papers, resulting in a mean RMSE of 4.5°. We identified five steps in this algorithm: measure, numerical integration, low-pass filtering, high-pass filtering, and summation. And this filter can be considered as real-time feasible with a computational complexity of $\mathcal{O}(n)$.

3) Kalman filtering: The Kalman filter is a recursive algorithm designed to estimate the state of a dynamic system from a sequence of noisy measurements. It operates on a prediction-correction model, which includes both a prediction phase and an update phase.

In the prediction phase, the Kalman filter generates a state-space estimate to predict the orientation at a given time. This estimate is used to predict the error covariance, which provides an estimation of potential errors. The Kalman gain is then calculated for the subsequent state update. Finally, using the sensor output, the state update is performed. The schematic structure of the Kalman filter is shown in Figure 3. We identified six steps in this algorithm: predict state, predict covariances, measure, compute Kalman gain, update state, and update covariances. The computational complexity of this algorithm scales with $\mathcal{O}(n^2)$, making it less suitable for real-time computations than the complementary filter.

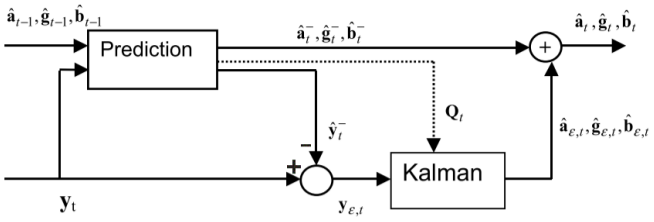


Fig. 3: Structure of the Kalman filter. Previous estimates of the acceleration, gravity and offset ($\hat{\mathbf{a}}_t^-, \hat{\mathbf{g}}_t^-, \hat{\mathbf{b}}_t^-$) are used to predict the sensor output vector. The difference ($\mathbf{y}_{\varepsilon,t}$) between the predicted sensor output ($\hat{\mathbf{y}}_t^-$) and actual sensor output ($\mathbf{y}_t$) is a function of the prediction error of the acceleration, gravity, and offset. A Kalman filter attributes this difference to the prediction errors ($\hat{\mathbf{a}}_{\varepsilon,t}, \hat{\mathbf{g}}_{\varepsilon,t}, \hat{\mathbf{b}}_{\varepsilon,t}$) using the variances of the predicted components ($\mathbf{Q}_t$). A hat on top of a symbol is used to indicate an estimate, the minus sign an *a priori* estimate based on the signal model and a plus sign an *a posteriori* estimate after correction by the Kalman filter [15].

The difference in estimation between gyroscope and accelerometer outputs is primarily due to prediction errors. These errors, along with the filter states, are presented in matrix form to assess and correct inaccuracies. By estimating noise and errors in covariance matrices, the Kalman filter helps mitigate the propagation of errors. As illustrated in Figure 4, the Kalman filter significantly reduces drift in orientation estimation when compared to simple strapdown integration.

However, it is important to note that this experiment involves a simple one-axis orientation change. When multiple orientations are altered simultaneously, the motion becomes nonlinear. A simple Kalman filter

struggles with nonlinear problems, causing the estimated orientation to drift away from the true orientation over time. The mean RMSE obtained from four papers is 4.5°.

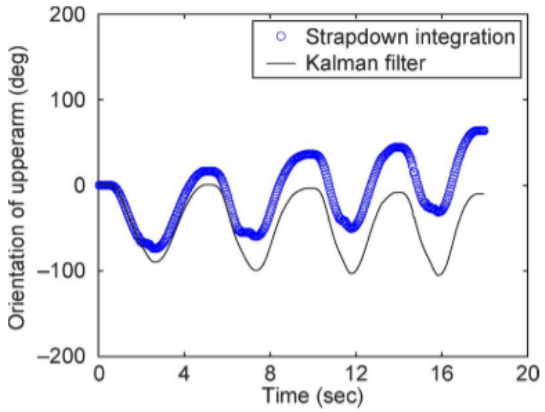


Fig. 4: Example trials of the orientation of the upper arm around one particular axis parallel to the ground surface during arm elevation. Orientations are computed with a strapdown integration algorithm and a Kalman filter over time [16].

4) Extended Kalman filtering: By combining the measured angular velocity, acceleration, and magnetic field values from a single Inertial Magnetic Measurement Unit (IMMU), the Extended Kalman Filter (EKF) can stably determine the orientation relative to a global coordinate system. The EKF constructs a measurement model by stacking the accelerometer, gyroscope, and magnetometer measurement vectors into a nonlinear measurement function $\boldsymbol{h}(\boldsymbol{x}_k)$ that maps the state $\boldsymbol{x}_k$ to the measurement space.

Due to the nonlinear nature of the measurement function $\boldsymbol{h}(\boldsymbol{x}_k)$, the EKF requires a first-order Taylor expansion. This expansion is performed around the current state estimation and observation model by computing the Jacobian matrix and linearizing the covariances of different noise sources:

$$\mathbf{H}_k = \left.\frac{\partial \boldsymbol{h}}{\partial \mathbf{x}}\right|_{\mathbf{x}_k=\mathbf{x}_k^-} \tag{15}$$

These linearizations, combined with an initial estimate, yield the Kalman gain, which updates the state vector using the sensor output. This update produces a new estimate for the sensor's orientation.

However, linearization can lead to reduced accuracy if the system dynamics are highly nonlinear. The linearized model may not accurately approximate the actual system, especially during fast movements, where lag and overshoot are observable [17]. The mean RMSE found is 4.3°, combining experiments from eight different papers. Figure 5 provides a block diagram of the EKF algorithm.

The identified steps for this algorithm are: predict state, linearize model, predict covariances, measure, linearize measurement, compute Kalman gain, update state, and update covariances. These steps come down to a total of eight. The EKF models state variables using first- and second-order moments, a method that is most effective when the distribution is Gaussian, which is not always the case. Furthermore, EKF computations are resource-intensive, with a computational complexity that scales with $\mathcal{O}(n^3)$. But this filter is still considered feasible for real-time applications, although offline, the accuracy can be optimized [18][19].

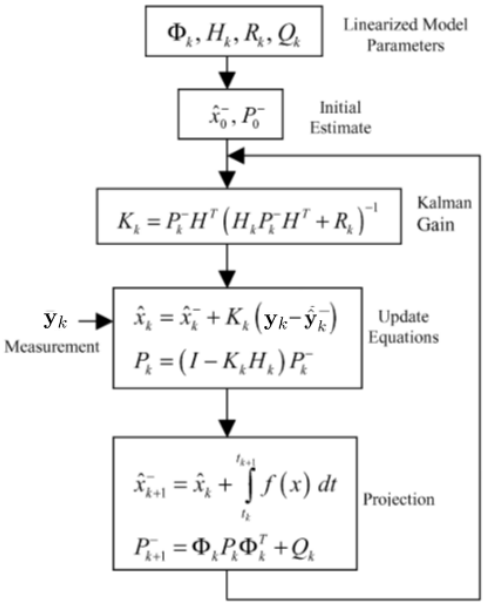


Fig. 5: Block diagram of the EKF. The process begins with the linearization of the state equations and an initial guess, which are combined into a predicted error covariance matrix at time step k ($\boldsymbol{P}_k^-$). This matrix, together with the measurement matrix ($\boldsymbol{H}$) and the measurement noise covariance matrix ($\boldsymbol{R}_k$), is used to compute the Kalman gain (K_k), The Kalman gain is applied to the measurement to estimate and update the measurement uncertainty. The updated estimation is then converted into a new orientation, which serves as the starting point for the next iteration of the cycle.

5) Unscented Kalman filtering: The Unscented Kalman Filter (UKF) shares nearly the same computational requirements as the Extended Kalman Filter but offers improved accuracy in handling nonlinear effects. Unlike the EKF, which relies on linearization, the UKF uses a deterministic sampling technique called the scaled unscented transformation. This technique selects a small set of sample points, known as sigma points, around the mean. These sigma points are then

propagated through the nonlinear functions, from which the mean and covariance of the estimate are accurately recovered, thereby approximating the probability distribution. This makes the UKF particularly well-suited for non-Gaussian estimation problems.

The quaternion-based UKF consists of the following nine steps:

1) Predict state: An initial estimate of the state vector is established.
2) Predict covariances: Together with the state the covariances are estimated.
3) Generate sigma points: Using these estimates, a Gaussian distribution is created through a set of sigma points and associated weights.
4) Propagate sigma points through nonlinear function: the set of sigma points and associated weights are propagated through the nonlinear function.
5) Generate prediction measurement: The outcomes from the nonlinear function combined generate a prediction measurement.
6) Measure: Then all the data from the gyroscope, accelerometer, and magnetometer are obtained by a measurement.
7) Compute Kalman gain: With the prediction measurement, state prediction, and covariance prediction the Kalman gain is computed.
8) Update state: The state is updated by integrating the Kalman gain with the measured data.
9) Update covariances: This step updates the covariances based on the Kalman gain and the prediction measurement.

Through its deterministic sampling technique that generates a probability distribution, the UKF offers robust orientation tracking, making it a valuable advancement over other filtering methods, particularly in dynamic and complex environments. But the computational complexity of this algorithm scales with $\mathcal{O}(k \cdot n^3)$, where k is the number of sigma points. Concluding that this filter is not feasible for real-time applications [9][20]. Furthermore, compared to the EKF, the UKF is more complex to implement and highly sensitive to parameter tuning. Incorrect tuning can lead to unstable filter behavior, poor convergence, and reduced accuracy. The accuracies reported in three papers have a mean of 3.2° as RMSE.

6) Madgwick filtering: The Madgwick filter utilizes data from the three primary sensors—accelerometer, gyroscope, and magnetometer—to provide drift-reduced quaternion-based orientation estimates [21]. This algorithm calculates the error between the predicted and measured quaternion values, using this error to update the quaternion estimate. Fine-tuning the algorithm parameters and sensor fusion weights can optimize the filter for specific application requirements, making it a robust and efficient tool for orientation estimation.

The algorithm starts with a state and error function estimation. Then the sensor data is read and normalized. Next, the Jacobian matrix of the cost function with respect to the quaternion is computed to evaluate the gradient of the cost function ($\nabla \boldsymbol{f}_k$) at time step k. The cost function depends on the quaternion estimate, normalized accelerometer data, and the normalized Earth's magnetic field vector.

$$\nabla \boldsymbol{f}_k = \boldsymbol{J}_k^T \boldsymbol{f}_k \left(\hat{\boldsymbol{q}}_{n,k}^b, \hat{\boldsymbol{a}}_k^b, \hat{\boldsymbol{m}}_k^b \right) \tag{16}$$

While gravity and angular velocity are reliable for short-term orientation estimation, they may become unreliable over extended periods, particularly for heading orientation. To counter this, the Earth's magnetic field, normalized as a vector ($\hat{\boldsymbol{m}}_k^b$), is incorporated as a reference.

Using the gradient descent algorithm, the quaternion is updated. The quaternion update process integrates the gradient of the cost function and the orientation increment derived from gyroscope measurements. The update steps are as follows:

$$\boldsymbol{q}_{\nabla,k+1} = -\beta \frac{\nabla \boldsymbol{f}_k}{||\boldsymbol{f}_k||} \tag{17}$$

$$\dot{\boldsymbol{q}}_{\omega,k+1} = \frac{1}{2} \hat{\boldsymbol{q}}_k \otimes \begin{bmatrix} 0, & \boldsymbol{\omega}_{k+1} \end{bmatrix}^T \tag{18}$$

$$\boldsymbol{q}_{k+1} = \hat{\boldsymbol{q}}_k + \Delta t \left(\hat{\boldsymbol{q}}_{\omega,k+1} + \boldsymbol{q}_{\nabla,k+1} \right) \tag{19}$$

Five steps can be identified from this algorithm: estimate state, estimate error function, measure, update gradient descent, and update state.

Figure 6 compares the performance of the Madgwick filter against a Kalman filter in an experiment validated by an optical tracking system. The results indicate that the Madgwick filter (blue line) provides smoother and more accurate orientation estimates than the Kalman filter (red line). The mean RMSE is 2.9°, based on the results of two papers.

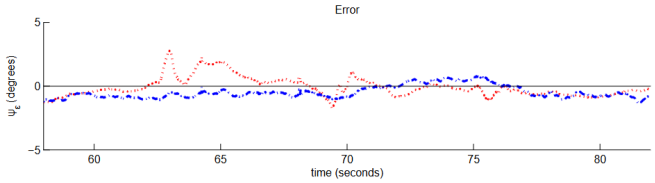


Fig. 6: Angle error in degrees during experiment over time, validated by optical tracking, comparing the Kalman filter (red) and the Madgwick filter (blue) [22].

In conclusion, the Madgwick filter is computationally efficient, with a computational complexity of $\mathcal{O}(n)$, and includes an effective drift correction mechanism. Unlike the Kalman filter, it focuses exclusively on orientation estimation, without estimating additional system states such as linear acceleration or position. This specialization makes it an excellent choice for real-time applications requiring accurate orientation tracking [21].

C. Calibration methods

Calibration is a critical step in addressing data processing issues, particularly biases affecting sensor signals in orientation estimation. Biases include a deterministic component, known as bias offset, and a random component referred to as drift [23]. Calibration mitigates these effects by modeling the bias accurately under controlled conditions.

1) Static calibration: Static calibration focuses on correcting constant parameters, such as sensor gain. In this method, subjects maintain a predefined, stable pose for a set duration, allowing sensor readings to be compared against known reference values, such as gravity or zero-velocity points. Averaging these readings minimizes noise, enabling precise bias estimation [24]. Static calibration is straightforward to compose with only one measuring instance, during the steady pose. However, static calibration alone overlooks dynamic errors, potentially leading to improperly scaled sensor outputs or inaccurate noise estimation over time.

2) Dynamic calibration: Dynamic calibration addresses time-dependent sensor errors and is typically performed immediately before recording limb orientations. It involves controlled motions, such as arm swings, to gather data on sensor behavior during movements. This method compensates for changes in dynamic characteristics, magnetic field distortions, and temperature variations [25]. Dynamic calibration is always performed in multiple measuring periods. The more periods are measured, the better the noise can be canceled from the sensors.

Although more complex than static calibration, dynamic calibration offers better correction for time-varying errors. Regular recalibration in response to environmental changes ensures ongoing accuracy and synchronizes results with the validation technique [26].

3) Multi-position calibration: Ferraris et al. [27] introduced a practical method for calibrating three-axis gyroscopes and accelerometers in-field using multiple sensor orientations. The sensors are placed in a quasi-cubic aluminum case positioned on a horizontal support base with a reference block, see Figure 7.

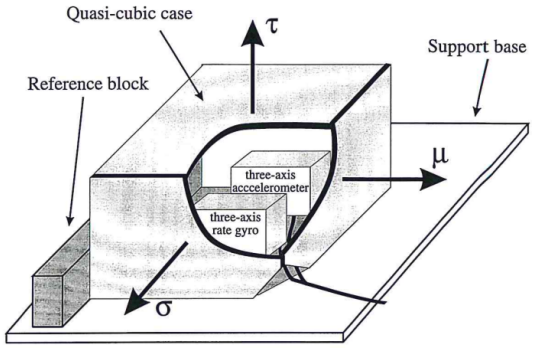


Fig. 7: Sensors enclosed in an aluminum quasi-cubic case used to establish the Cartesian reference system. The case is placed on a horizontal support base, equipped with a reference block.

The calibration procedure involves a simple manipulation of the case, mainly consisting of two steps as:

1) sequentially placing each side of the case onto the support base.
2) completing a full rotation of the case, returning to the initial reference block position.

The procedure involves six measuring instances and calibrates the static and dynamic errors. But the magnetometer cannot be calibrated in this procedure due to the involvement of ferromagnetic materials. This calibration method remains valid for about one hour under standard conditions without requiring additional instrumentation. The estimated alignment error is minimal, with a maximum misalignment angle of 0.45°.

D. Validation methods

Accurate validation of motion-tracking systems depends on the precise alignment of coordinate frames between the validation and sensor systems. Four validation methods were extracted from the literature and discussed in this section. Validation typically requires additional equipment with greater accuracy than the used IMMUs, ensuring reliable accuracy validation measurements.

1) Passive optical motion capturing system: Passive optical motion capture systems are widely used for human motion tracking. These systems utilize retro-reflective markers attached to specific body points and captured by multiple cameras positioned at various angles. Software processes these inputs using trigonometric relations to generate a detailed three-dimensional representation of motion. They are known for their millimeter-scale precision and ease of use due to the simplicity of markers, which require no further wiring. The integration of the system is separated into seven steps. Set up cameras, place markers, set up software, calibrate, measure, process data, and compare. The accuracy of the optical motion tracking system is in the order of 0.01° to 0.1°. With a processing time in the order of 10 ms. However, these systems have limitations including sensitivity to lighting changes, occlusion, and marker confusion in dense setups [28]. Afterward, the motion video has to be manually checked for correct marker identification, making real-time identification more difficult. Despite these challenges, passive optical motion capturing remains an effective validation method for measuring wearable sensor system accuracy.

2) Active optical motion capturing system: In contrast to passive optical motion-capturing systems, a system with active markers requires a power source. The markers emit light, mostly as LEDs, that are tracked by the cameras. This offers a precise identification as markers emit unique signals, for example in different frequencies or flashing patterns.

The systems allow real-time validation with precise marker identification, eliminating confusion between markers. However, they are more complex with an extra integration step due to the need for wiring or battery management and are also therefore heavier [28]. The accuracy and processing time of active systems lies in the same order as the passive systems.

3) Robotic arm: High-precision industrial robotic arms can simulate accurately controlled motions for an upper limb, making them useful for evaluating sensor accuracy. By comparing sensor readings with the known positions of the robotic arm, performance can be assessed across different movement speeds [29].

Robotic arms eliminate human-related variability, enabling highly controlled validation. However, they have limitations, including their inability to fully replicate human arm dynamics and the potential for magnetic interference affecting magnetometer performance [18]. These factors restrict their use in broader human motion studies, though they are effective for testing specific motion patterns. The six steps involved in setting up this validation method are: set up software, place sensors, calibrate, measure, process data, and compare. The accuracy of a high-performance robotic arm lies in the order of 0.001° to 0.01° and the processing time in the order of 1 to 10 ms.

4) Goniometer: Goniometers are straightforward devices that measure joint angles. They consist of two metal arms and a potentiometer; as the angle between the arms changes, the resistance in the potentiometer varies. Digital goniometers also exist, which measure angles by detecting the amount of light passing through optic fibers. Setting up this validation technique is simple and involves only four steps: place goniometer, measure, process data, and compare.

The accuracy of a goniometer is approximately ±1°, factoring in device, mounting, and environmental errors [30]. And the processing time lies in the order of 1–10 ms. The primary limitation is that goniometers only measure angles along one axis. Sometimes the placement can result in restriction of the upper limbs movement. This restricts their utility in human upper limb validation tasks, though they remain valuable for basic angle measurement.

E. Overview

This section presents a comparative analysis of filtering methods, calibration principles, and validation techniques in the form of three tables. The data given in the table follows the used method explained in Section II. The caption of the tables also give insight in where the data is coming from. Table II focuses on the six filtering methods. Table III outlines the identified calibration methods. Table IV provides the comparison of the validation techniques.

TABLE II: Comparison table of the six filters extracted from the used literature. Covering which sensing devices are compatible. Further identifying whether the method can handle non-linear problems and which orientation representation can be used. The accuracies are assessed using the root-mean-square errors (RMSE) in degrees. The mean RMSE was taken from the analyzed papers for every method. The number of steps required to compute one state update cycle is given in the 'integration steps' row, the steps were identified in Section III. The real-time feasibility shows how the computation complexity scales with the size of the state vector (n). For UKF, the scaling also considers the number of sigma points (k).

	Strapdown integration	Complementary filter	Kalman filter	Extended Kalman filter	Unscented Kalman filter	Madgwick filter
Sensors	Gyr+Acc	Gyr+Acc	Gyr+Acc	Gyr+Acc+Mag	Gyr+Acc+Mag	Gyr+Acc+Mag
(Non)linear	Linear	Linear	Linear	Nonlinear	Nonlinear	Nonlinear
Euler/ Quaternion	Euler/ Quaternion	Euler/ Quaternion	Quaternion	Quaternion	Quaternion	Quaternion
Accuracy	5.8°	4.5°	4.5°	4.3°	3.2°	2.9°
Integration steps	3	5	6	8	9	5
Real-time feasibility	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\mathcal{O}(n^2)$	$\mathcal{O}(n^3)$	$\mathcal{O}(k \cdot n^3)$	$\mathcal{O}(n)$

TABLE III: The calibration comparison table summarizes the three calibration methods identified from the analyzed literature. It first specifies the sensing devices compatible with each method and evaluates the capability to calibrate non-linear errors. All calibration methods support orientation representation using both Euler angles and quaternions. The accuracy assessment highlights the types of errors each method can effectively mitigate. The integration steps indicate the number of measurement periods required to complete the calibration process according to the described procedure. Finally, the table considers whether the calibration method can be applied in real-time while the sensors are worn on the body.

	Static calibration	Dynamic calibration	Multi-position calibration
Sensors	Gyr+Acc+Mag	Gyr+Acc+Mag	Gyro+Acc
(Non)linear	Linear	Nonlinear	Nonlinear
Euler/ Quaternion	Euler/ Quaternion	Euler/ Quaternion	Euler/ Quaternion
Accuracy	Static errors	Dynamic errors	Static and dynamic errors
Integration steps	1	2 or more	6
Real-time feasibility	Feasible	Feasible	Not feasible

TABLE IV: This comparison table presents the four validation techniques identified from the analyzed literature. Here, the identified validation techniques already consider the sensors used and evaluate their compatibility for validating nonlinear problems. All validation methods support orientation representation using both Euler angles and quaternions. The accuracy of each validation device is given in order of magnitude based on data from multiple datasheets. The integration steps indicate the number of steps required to set up the validation method. Real-time feasibility is assessed by analyzing the order of magnitude of computation time, as reported in various datasheets.

Sensors	Passive optical	Active optical	Robotic arm	Goniometer
(Non)linear	Nonlinear	Nonlinear	Nonlinear	Linear
Euler/ Quaternion	Euler/ Quaternion	Euler/ Quaternion	Euler/ Quaternion	Euler/ Quaternion
Accuracy (°)	$\mathcal{O}(10^{-2} - 10^{-1})$	$\mathcal{O}(10^{-2} - 10^{-1})$	$\mathcal{O}(10^{-3} - 10^{-2})$	$\mathcal{O}(10^{0})$
Integration steps	7	8	6	4
Real-time feasibility (ms)	$\mathcal{O}(10^{1})$	$\mathcal{O}(10^{1})$	$\mathcal{O}(10^{0} - 10^{1})$	$\mathcal{O}(10^{0} - 10^{1})$

IV. Discussion

This section discusses the interpretations and implications of the results from the preceding analysis, offering insights into the selection of suitable filtering, calibration, and validation techniques for the ungrounded wearable wrist perturbator.

Each wearable sensing device exhibits unique advantages and limitations, as discussed in Subsection III-A. To enhance measurement accuracy, various sensor fusion filtering methods have been developed to integrate these strengths. However, not every filtering method leverages all available sensors for orientation estimation. For instance, strapdown integration, complementary filtering, and the standard Kalman filter rely solely on gyroscope and accelerometer data. In contrast, other Kalman filter variants and the Madgwick filter incorporate magnetometer data into their fusion algorithms. With advancements in sensor affordability, integrating multiple sensor types, including magnetometers, has become increasingly feasible. Consequently, most contemporary studies utilize a 9-axis inertial and magnetic measurement unit (30/34), although some of those studies omit magnetometer data (5/30).

Tracking wrist orientation is particularly challenging due to its complex rotations, influenced by upper limb dynamics and joint constraints. The nonlinear relationship between wrist angles and limb orientation renders linear filtering methods, such as strapdown integration, complementary filtering, and the standard Kalman filter, inadequate. These methods often result in inaccuracies and drift when applied to such scenarios.

Real-time orientation is another critical requirement for this application, as it enables timely feedback to actuate the wearable device accurately. Among the filtering algorithms evaluated, the unscented Kalman filter was deemed unsuitable due to its high computational complexity. Other algorithms demonstrated feasibility for real-time applications, with expected computation times in the order of $\mathcal{O}(10^0 - 10^1$ ms). Although offline estimation can improve accuracy by approximately 1–2°.

The extended Kalman filter and the Madgwick filter were identified as the most suitable options for real-time applications. Both provide nonlinear estimations with manageable computational demands. Literature data suggests the Madgwick filter achieves slightly better accuracy than the extended Kalman filter, with a mean difference of 1.4°. However, this conclusion is based on only two studies for the Madgwick filter versus eight studies for the extended Kalman filter, rendering the result statistically insignificant. Variations in test protocols across studies also hinder comparability, underscoring the need for a standardized test protocol for orientation estimation systems. Factors affecting experimental results include movement speed, movement complexity, real-time versus offline testing, and the number of sensors used.

Calibration methods in the reviewed literature are categorized into static, dynamic, and multi-position approaches. Each method addresses specific sensor errors: static calibration targets constant errors, dynamic calibration handles time-varying errors, and multi-position calibration tackles both. However, multi-position calibration is typically conducted off-body, therefore impractical for real-time applications. Moreover, it is unsuitable for calibrating magnetometer sensors due to interference from ferromagnetic materials. Both static and dynamic calibration methods are appropriate for the wrist perturbator application. Static calibration requires only one measurement period, while dynamic calibration involves two or more measurement instances. Increasing the number of instances enhances noise cancellation and improves sensor accuracy.

Despite these advances, the lack of a robust, long-lasting in-use calibration protocol remains a significant research gap. Currently, after calibration, sensor systems have an error increase rate in the order of $\mathcal{O}(10^{-1}$ °/min), and the longest reported testing period without recalibration is 15 minutes [20]. Extending this period while maintaining accuracy presents a challenge for future research.

Validation techniques for the wrist perturbator must ensure accuracy below the threshold of the selected filters while accommodating nonlinear motions with sensors worn on the body. Optical motion capture systems with passive or active markers meet these requirements. Active marker systems, although requiring wiring on the body, avoid marker misidentification.

In contrast, goniometers are unsuitable due to their restriction to linear angular motion around a single axis and their inability to provide robust validation. Similarly, robotic arms, while effective for simple validation tasks, cannot validate performance while sensors are worn on the body. Additionally, the ferromagnetic materials in the robotic arms will distort the magnetometer data.

V. Conclusion

This literature review explores state-of-the-art sensors, as well as filtering, calibration, and validation techniques for determining wrist orientation. Overviews of these techniques are provided. Based on these overviews, suitable techniques are identified for the application of an ungrounded wrist perturbator. The perturbator requires real-time wrist orientation estimation using non-obstructive measurements and must be capable of tracking nonlinear orientations.

Three non-obstructive wearable sensing devices—the accelerometer, gyroscope, and magnetometer—were analyzed. These sensors are frequently used in wearable non-obstructive motion tracking, as indicated in the reviewed studies. Each sensor offers distinct advantages and limitations, underscoring the importance of sensor fusion for achieving accurate orientation measurements.

The review analyzed six filter algorithms designed to integrate data from multiple sensing devices to enhance orientation accuracy. Among these filters, the extended Kalman filter (EKF) and the Madgwick filter are identified as suitable to apply for the wrist perturbator. Based on the mean reported RMSE values in the literature, the Madgwick filter demonstrated more accurate tracking (2.9°) compared to the EKF (4.3°). Additionally, the Madgwick filter is considered easier to implement, with only five computational steps, compared to the EKF, which contains eight computational steps. However, considering the criteria of handling nonlinearity and real-time feasibility, both filters are deemed suitable.

The review highlighted the critical importance of both static and dynamic calibration in achieving accurate error correction. Static calibration, by enhancing a steady pose, will mitigate the static errors. Dynamic calibration, exemplified by swinging an arm multiple times, mitigates the time-varying dynamic errors. While these methods are effective, the absence of a standardized procedure for in-use calibration remains a significant challenge. Additionally, extending calibration intervals beyond the current reported maximum of 15 minutes is also necessary for longer experimental setups.

An optical motion capture system was identified as a suitable technique for validation. These systems provide better accuracy compared to orientation filters, ensuring valuable validation results. Additionally, their minimal intrusion makes them ideal for in-use experiments. Optical motion capture systems can utilize either passive or active markers. Active marker systems require wiring but eliminate the risk of marker misidentification. In contrast, passive marker systems are wireless but may misidentify markers, necessitating post-processing.

In summary, a wrist-worn device for real-time orientation tracking can be developed using an extended Kalman filter or a Madgwick filter sensor fusion algorithm, complemented with static and dynamic calibration methods. For instance, static calibration can be conducted in a steady pose, while dynamic calibration involves multiple arm swings. We expect that both calibration techniques have to be redone every 15 minutes to maintain data reliability. Validation using an optical motion capture system with active or passive markers ensures precise and reliable wrist orientation estimation, this validation only needs to be performed once. Real-time wrist orientation estimation will enable the wrist perturbator to generate ungrounded forces in the direction of application, advancing the measurement of the upper limb's mechanical properties during daily activities.

References

[1] A. Schwarz, M. M. C. Bhagubai, G. Wolterink, J. P. O. Held, A. R. Luft, and P. H. Veltink, "Assessment of Upper Limb Movement Impairments after Stroke Using Wearable Inertial Sensing," *Sensors*, vol. 20, no. 17, p. 4770, 8 2020. [Online]. Available: https://doi.org/10.3390/s20174770

[2] S. Kloiber, V. Settgast, C. Schinko, M. Weinzerl, J. Fritz, T. Schreck, and R. Preiner, "Immersive analysis of user motion in VR applications," *The Visual Computer*, vol. 36, no. 10-12, pp. 1937–1949, 8 2020. [Online]. Available: https://doi.org/10.1007/s00371-020-01942-1

[3] W. Mugge, J. Schuurmans, A. C. Schouten, and F. C. T. Van Der Helm, "Sensory Weighting of Force and Position Feedback in Human Motor Control Tasks," *Journal of Neuroscience*, vol. 29, no. 17, pp. 5476–5482, 4 2009. [Online]. Available: https://doi.org/10.1523/jneurosci.0116-09.2009

[4] P. K. Artemiadis, P. T. Katsiaris, M. V. Liarokapis, and K. J. Kyriakopoulos, "Human arm impedance: Characterization and modeling in 3D space," *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 10 2010. [Online]. Available: https://doi.org/10.1109/iros.2010.5652025

[5] R. Koene, J. Meijaard, M. van de Ruit, W. Mugge, and V. van der Wijk, "Design and validation of a 3-dof wrist perturbator based on an inverted spatial redundant 4-ruu parallel manipulator," in *Advances in Mechanism and Machine Science*, M. Okada, Ed. Cham: Springer Nature Switzerland, 2023, pp. 786–796.

[6] N. T. Liu, N. Y. Inoue, N. K. Shibata, and N. R. Zheng, "Measurement of human lower limb orientations and ground reaction forces using wearable sensor systems," *2007 IEEE/ASME international conference on advanced intelligent mechatronics*, vol. 12, pp. 1–6, 1 2007. [Online]. Available: https://doi.org/10.1109/aim.2007.4412493

[7] T. Liu, Y. Inoue, and K. Shibata, "Development of a wearable sensor system for quantitative gait analysis," *Measurement*, vol. 42, no. 7, pp. 978–988, 2 2009. [Online]. Available: https://doi.org/10.1016/j.measurement.2009.02.002

[8] H. Zhou, H. Hu, and Y. Tao, "Inertial measurements of upper limb motion," *Medical & Biological Engineering & Computing*, vol. 44, no. 6, pp. 479–487, 5 2006. [Online]. Available: https://doi.org/10.1007/s11517-006-0063-z

[9] M. El-Gohary, L. Holmstrom, J. Huisinga, E. King, J. McNames, and F. Horak, "Upper limb joint angle tracking with inertial sensors," *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 8 2011. [Online]. Available: https://doi.org/10.1109/iembs.2011.6091362

[10] H. J. Luinge and P. H. Veltink, "Measuring orientation of human body segments using miniature gyroscopes and accelerometers," *Medical & Biological Engineering & Computing*, vol. 43, no. 2, pp. 273–282, 4 2005. [Online]. Available: https://doi.org/10.1007/bf02345966

[11] H. Zhou, T. Stone, H. Hu, and N. Harris, "Use of multiple wearable inertial sensors in upper limb motion tracking," *Medical Engineering & Physics*, vol. 30, no. 1, pp. 123–133, 1 2007. [Online]. Available: https://doi.org/10.1016/j.medengphy.2006.11.010

[12] R. Pérez, Costa, M. Torrent, J. Solana, E. Opisso, C. Cáceres, J. M. Tormos, J. Medina, and E. J. Gómez, "Upper Limb Portable Motion Analysis System Based on Inertial Technology for Neurorehabilitation Purposes," *Sensors*, vol. 10, no. 12, pp. 10 733–10 751, 12 2010. [Online]. Available: https://www.mdpi.com/1424-8220/10/12/10733

[13] P. Gui, L. Tang, and S. Mukhopadhyay, "MEMS based IMU for tilting measurement: Comparison of complementary and kalman filter based data fusion," *2015 IEEE 10th Conference on Industrial Electronics and Applications (ICIEA)*, 6 2015. [Online]. Available: https://doi.org/10.1109/iciea.2015.7334442

[14] G. J. Garcia, A. Alepuz, G. Balastegui, L. Bernat, J. Mortes, S. Sanchez, E. Vera, C. A. Jara, V. Morell, J. Pomares, J. L. Ramon, and A. Ubeda, "ARMIA: A Sensorized Arm Wearable for Motor Rehabilitation," *Biosensors*, vol. 12, no. 7, p. 469, 6 2022. [Online]. Available: https://doi.org/10.3390/bios12070469

[15] H. Luinge and P. Veltink, "Inclination measurement of human movement using a 3-D accelerometer with autocalibration," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 12, no. 1, pp. 112–121, 3 2004. [Online]. Available: https://doi.org/10.1109/tnsre.2003.822759

[16] N. H. Zhou and N. H. Hu, "Reducing Drifts in the Inertial Measurements of Wrist and Elbow Positions," *IEEE Transactions on Instrumentation and Measurement*, vol. 59, no. 3, pp. 575–585, 9 2009. [Online]. Available: https://doi.org/10.1109/tim.2009.2025065

[17] X. Yun and E. R. Bachmann, "Design, Implementation, and Experimental Results of a Quaternion-Based Kalman Filter for Human Body Motion Tracking," *IEEE Transactions on Robotics*, vol. 22, no. 6, pp. 1216–1227, 12 2006. [Online]. Available: https://doi.org/10.1109/tro.2006.886270

[18] B. Fang, F. Sun, H. Liu, and D. Guo, "Development of a Wearable Device for Motion Capturing Based on Magnetic and Inertial Measurement Units," *Scientific Programming*, vol. 2017, pp. 1–11, 1 2017. [Online]. Available: https://doi.org/10.1155/2017/7594763

[19] G. Škulj, R. Vrabič, and P. Podržaj, "A Wearable IMU System for Flexible Teleoperation of a Collaborative Industrial Robot," *Sensors*, vol. 21, no. 17, p. 5871, 8 2021. [Online]. Available: https://doi.org/10.3390/s21175871

[20] B. Kirking, M. El-Gohary, and Y. Kwon, "The feasibility of shoulder motion tracking during activities of daily living using inertial measurement units," *Gait & Posture*, vol. 49, pp. 47–53, 6 2016. [Online]. Available: https://doi.org/10.1016/j.gaitpost.2016.06.008

[21] V. Vashista and Y. Singh, "Single-Joint Angle Computation using Inertial Sensors with Feedback on Smartphone," *AIR '21: Proceedings of the 2021 5th International Conference on Advances in Robotics*, 6 2021. [Online]. Available: https://doi.org/10.1145/3478586.3480647

[22] S. O. H. Madgwick, A. J. L. Harrison, and R. Vaidyanathan, "Estimation of IMU and MARG orientation using a gradient descent algorithm," *2011 IEEE International Conference on Rehabilitation Robotics*, 6 2011. [Online]. Available: https://doi.org/10.1109/icorr.2011.5975346

[23] S. A. Rawashdeh, D. A. Rafeldt, and T. L. Uhl, "Wearable IMU for Shoulder Injury Prevention in Overhead Sports," *Sensors*, vol. 16, no. 11, p. 1847, 11 2016. [Online]. Available: https://doi.org/10.3390/s16111847

[24] R. Stanzani, P. Dondero, A. Mantero, and M. Testa, "Measurement Accuracy of an Upper Limb Tracking System Based on Two Hillcrest Labs BNO080 IMU Sensors: An Environmental Assessment," *IEEE Sensors Journal*, vol. 20, no. 17, pp. 10 267–10 274, 5 2020. [Online]. Available: https://doi.org/10.1109/jsen.2020.2992733

[25] L. Y. T. Chan, C. S. Chua, S. M. Chou, R. Y. B. Seah, Y. Huang, Y. Luo, L. Dacy, and H. R. B. A. Razak, "Assessment of shoulder range of motion using a commercially available wearable sensor—a validation study," *mHealth*, vol. 8, p. 30, 8 2022. [Online]. Available: https://doi.org/10.21037/mhealth-22-7

[26] M. M. Morrow, B. Lowndes, E. Fortune, K. R. Kaufman, and M. S. Hallbeck, "Validation of Inertial Measurement Units for Upper Body Kinematics," *Journal of Applied Biomechanics*, vol. 33, no. 3, pp. 227–232, 12 2016. [Online]. Available: https://doi.org/10.1123/jab.2016-0120

[27] F. Ferraris, I. Gorini, U. Grimaldi, and M. Parvis, "Calibration of three-axial rate gyros without angular velocity standards," *Sensors and Actuators A Physical*, vol. 42, no. 1-3, pp. 446–449, 4 1994. [Online]. Available: https://doi.org/10.1016/0924-4247(94)80031-6

[28] J. M. Lambrecht and R. F. Kirsch, "Miniature Low-Power Inertial Sensors: Promising Technology for Implantable Motion Capture Systems," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 22, no. 6, pp. 1138–1147, 5 2014. [Online]. Available: https://doi.org/10.1109/tnsre.2014.2324825

[29] M. El-Gohary and J. McNames, "Human Joint Angle Estimation with Inertial Sensors and Validation with A Robot Arm," *IEEE Transactions on Biomedical Engineering*, vol. 62, no. 7, pp. 1759–1767, 2 2015. [Online]. Available: https://doi.org/10.1109/tbme.2015.2403368

[30] G. X. Lee, K. S. Low, and T. Taher, "Unrestrained Measurement of Arm Motion Based on a Wearable Wireless Sensor Network," *IEEE Transactions on Instrumentation and Measurement*, vol. 59, no. 5, pp. 1309–1317, 3 2010. [Online]. Available: https://doi.org/10.1109/tim.2010.2043974

[31] H. Luinge, P. Veltink, and C. Baten, "Ambulatory measurement of arm orientation," *Journal of Biomechanics*,

vol. 40, no. 1, pp. 78–85, 2 2006. [Online]. Available: https://doi.org/10.1016/j.jbiomech.2005.11.011

[32] H. Harms, O. Amft, R. Winkler, J. Schumm, M. Kusserow, and G. Troester, "ETHOS: Miniature orientation sensor for wearable human motion analysis," *SENSORS, 2010 IEEE*, pp. 1037–1042, 11 2010. [Online]. Available: https://doi.org/10.1109/icsens.2010.5690738

[33] Z. Zhang, L. W. Wong, and J.-K. Wu, "3D Upper Limb Motion Modeling and Estimation Using Wearable Micro-sensors," *2010 International Conference on Body Sensor Networks*, vol. 26, pp. 117–123, 6 2010. [Online]. Available: https://doi.org/10.1109/bsn.2010.14

[34] Z.-Q. Zhang, L.-Y. Ji, Z.-P. Huang, and J.-K. Wu, "Adaptive Information Fusion for Human Upper Limb Movement Estimation," *IEEE Transactions on Systems Man and Cybernetics - Part A Systems and Humans*, vol. 42, no. 5, pp. 1100–1108, 4 2012. [Online]. Available: https://doi.org/10.1109/tsmca.2012.2189876

[35] D. Alvarez, J. C. Alvarez, R. C. Gonzalez, and A. M. Lopez, "Ambulatory human upper limb joint motion monitoring," *2012 IEEE International Instrumentation and Measurement Technology Conference Proceedings*, 5 2012. [Online]. Available: https://doi.org/10.1109/i2mtc.2012.6229695

[36] Y.-L. Hsu, J.-S. Wang, Y.-C. Lin, S.-M. Chen, Y.-J. Tsai, C.-L. Chu, and C.-W. Chang, "A wearable inertial-sensing-based body sensor network for shoulder range of motion assessment," *2013 1st International Conference on Orange Technologies (ICOT)*, 3 2013. [Online]. Available: https://doi.org/10.1109/icot.2013.6521225

[37] B. Bouvier, S. Duprey, L. Claudon, R. Dumas, and A. Savescu, "Upper Limb Kinematics Using Inertial and Magnetic Sensors: Comparison of Sensor-to-Segment Calibrations," *Sensors*, vol. 15, no. 8, pp. 18 813–18 833, 7 2015. [Online]. Available: https://doi.org/10.3390/s150818813

[38] R. Ranganathan, R. Wang, B. Dong, and S. Biswas, "Identifying compensatory movement patterns in the upper extremity using a wearable sensor system," *Physiological Measurement*, vol. 38, no. 12, pp. 2222–2234, 11 2017. [Online]. Available: https://doi.org/10.1088/1361-6579/aa9835

[39] J. Chae, Y. Jin, Y. Sung, and K. Cho, "Genetic Algorithm-Based Motion Estimation Method using Orientations and EMGs for Robot Controls," *Sensors*, vol. 18, no. 1, p. 183, 1 2018. [Online]. Available: https://doi.org/10.3390/s18010183

[40] P. Picerno, P. Caliandro, C. Iacovelli, C. Simbolotti, M. Crabolu, D. Pani, G. Vannozzi, G. Reale, P. M. Rossini, L. Padua, and A. Cereatti, "Upper limb joint kinematics using wearable magnetic and inertial measurement units: an anatomical calibration procedure based on bony landmark identification," *Scientific Reports*, vol. 9, no. 1, 10 2019. [Online]. Available: https://doi.org/10.1038/s41598-019-50759-z

[41] B. Oubre, J.-F. Daneault, H.-T. Jung, K. Whritenour, J. G. V. Miranda, J. Park, T. Ryu, Y. Kim, and S. I. Lee, "Estimating Upper-Limb Impairment Level in Stroke Survivors Using Wearable Inertial Sensors and a Minimally-Burdensome Motor Task," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 28, no. 3, pp. 601–611, 1 2020. [Online]. Available: https://doi.org/10.1109/tnsre.2020.2966950

[42] J. A. Goreham, K. F. MacLean, and M. Ladouceur, "The validation of a low-cost inertial measurement unit system to quantify simple and complex upper-limb joint angles," *Journal of Biomechanics*, vol. 134, p. 111000, 2 2022. [Online]. Available: https://doi.org/10.1016/j.jbiomech.2022.111000

APPENDIX

The papers identified through the methodological search are summarized chronologically in Table V. For each paper, the filter method, calibration and validation method, as discussed in the previous sections, have been extracted. Also, the root-mean-square errors were given when they were reported in the paper, sometimes the mean is computed based on multiple RMSE values given or a range was reported. It is important to note that not all papers provide complete information for every cell in the table.

TABLE V: Paper overview showing the research papers used during this literature review that come out of the methodological search given the author(s), year and reference number, and extracted information about the used, filtering, calibration, and validation method. Together with the reported root-mean-square errors in degrees.

Author	**Ref.**	**Filter**	**Calibration**	**Validation**	**RMSE** (°)
Luinge and Veltink (2004)	[15]	KF	Multi	Passive	2
Luinge and Veltink (2005)	[10]	KF	Multi	Passive	3
Zhou et al. (2006)	[8]	CF	Dynamic	Passive	–
Yun and Bachmann (2006)	[17]	EKF	–	Robot	9
Luinge et al. (2006)	[31]	KF	Static	Passive	8
Zhou et al. (2007)	[11]	CF	Multi	Passive	–
Zhou and Hu (2009)	[16]	KF	Dynamic	Passive	4.8
Lee et al. (2010)*	[30]	Int	Dynamic	Gonio	3.2–8.1
Harms et al. (2010)	[32]	CF	–	IMMU	8.9
Zhang et al. (2010)	[33]	UKF	Static	Passive	–
Madgwick et al. (2011)*	[22]	Mad	Dynamic	Passive	1.1
El-Gohary et al. (2011)	[9]	UKF	–	Passive	–
Zhang et al. (2012)	[34]	UKF	Static	Passive	5.7
Alvarez et al. (2012)	[35]	Int	Dynamic	Robot	6
Hsu et al. (2013)	[36]	CF	Dynamic	IMMU	2.34–3.36
Lambrecht and Kirsch (2014)	[28]	EKF	Static Dynamic	Active	3.6
Bouvier et al. (2015)	[37]	EKF	Static	Passive	–
El-Gohary and McNames (2015)	[29]	UKF	Static	Robot	1.1–2.8
Gui et al. (2015)	[13]	CF KF	Static Dynamic	–	–
Kirking et al. (2016)	[20]	UKF	Static	Robot	2
Rawashdeh et al. (2016)	[23]	EKF	Static Dynamic	–	–
Morrow et al. (2017)*	[26]	EKF	Static Dynamic	Passive	6.8–8.2
Ranganathan et al. (2017)	[38]	KF	Static	Passive	–
Fang et al. (2017)	[18]	EKF	Static	Robot	0.02–2.43
Chae et al. (2018)	[39]	EKF	Path	–	–
Picerno et al. (2019)	[40]	EKF	Static	Passive	1.8–4.4
Oubre et al. (2020)	[41]	EKF	–	–	–
Schwarz et al. (2020)	[1]	Mad	Static	–	–
Stanzani et al. (2020)	[24]	EKF	Static Dynamic	Passive	1.3–5.4
Vashista and Singh (2021)*	[21]	Mad	Dynamic	Passive	4.6
Škulj et al. (2021)	[19]	EKF	Steady	Active	2.5
Garcia et al. (2022)	[14]	CF	Path	Robot	1.86
Chan et al. (2022)	[25]	–	Static Dynamic	Passive	5.04–10.08
Goreham et al. (2022)	[42]	EKF	Static	Passive	1.58–4.83

Abbreviations: Int: Strapdown integration, CF: Complementary filter, KF: Kalman filter, EKF: Extended Kalman filter, UKF: Unscented Kalman filter, Mad: Madgwick filter, Gonio: Goniometer, IMMU: Inertial Magnetic Measurement Unit.
–: Information was not reported and/or unclear in the study and/or unable to be obtained from the manufacturer or other referenced paper.
*: This is one of the snowball papers, useful literature found outside of the used search method.

2

Scientific paper: Design and Validation of a Lightweight Parallel Manipulator for Ungrounded 3-DoF Translational Wrist Perturbation with Orientation Sensing

Design and Validation of a Lightweight Parallel Manipulator for Ungrounded 3-DoF Translational Wrist Perturbations with Orientation Sensing

Matthias Weerheim, Student

Abstract— **Wearable devices capable of perturbing and measuring wrist orientation during daily activities are essential, as traditional measurement systems are often restricted to controlled environments and limit natural movement. A compact and lightweight 4-RUU parallel manipulator mounted on the wrist can generate ungrounded forces in three translational degrees of freedom, while keeping the hand and arm unobstructed. The inverse kinematics of this mechanism is analytically derived, enabling control with an open-loop feedforward system for the four servo motors for precise perturbations. A simulation of the concept parallel manipulator is used to optimize the main dimensions of the device for the intended application. A prototype is then constructed and evaluated, both for its ability to generate forces using a force sensor and for its ability to measure real-time orientation using an inertial and magnetic measurement unit with an implemented Madgwick filter. The force measurements showed that along every axis and combination of axes, a certain repeatable force generation profile can be distinguished, where a minimal peak of 5.0 N is generated. The lack of end-effector acceleration, due to limiting servo performance, resulted in a loss of forces in the mechanism lie between 14–50 %. The orientation estimation is fast and accurate in real time with a maximum total angle error of 5°.**

I. Introduction

System identification has been used for decades to determine the mechanical properties of human limbs [1][2]. Such identification can support the quantification of stroke patients' rehabilitation progress [3], assist in diagnosing neuromuscular diseases [4], and enable the realistic simulation of human limb movement for virtual reality applications [5]. Traditional test setups, however, rely on bulky equipment operated in controlled laboratory environments [6][7][8]. A major limitation of these setups is that they restrict participants' natural movement. Moreover, the mechanical properties of human limbs are task-dependent and vary with orientation [9], making experiments conducted during daily activities with more mobility possibilities particularly valuable.

Various haptic actuation principles have been explored to apply forces to human subjects, for example airjet systems [10][11][12], flywheel-based inertia mechanisms [13][14], and crank-cam mechanisms [15][16]. However, these systems often require large and heavy equipment, such as pressure tanks, or generate substantial parasitic forces that complicate the interpretation of measurements.

A more promising working principle is based on Newton's laws of motion: the acceleration of a mass generates an equal and opposite reaction force. Embedding this principle in a compact three-degrees-of-freedom (3-DoF) parallel mechanism yields a lightweight, high-bandwidth device capable of producing measurable ungrounded forces [17]. This proof-of-concept can be further enhanced by optimizing its dimensions to meet specific perturbation requirements and by integrating improved actuators and control strategies. Analytically derived inverse kinematics will also contribute to this optimization. In addition, integrating a suitable orientation-sensing configuration enables perturbation to be applied at the intended wrist orientation and provides richer system identification for studying orientation- and task-dependent limb mechanics.

This paper presents a mechatronic device that applies ungrounded forces to the wrist in 3-DoF according to a prescribed force profile, achieving a maximum short-duration peak of 7.0 N. The device is also capable of estimating its own orientation in real time, enabling perturbations to be applied while maintaining a total angular error within 5°.

Additional requirements are discussed in Section II, together with further explanation and justification. Based on these requirements, Section III analyzes the kinematics of the parallel manipulator. Section IV presents simulations based on the derived kinematics to validate the main dimensions for a new prototype, which is further discussed in Section V. The prototype is then experimentally validated in Section VI and

Section VII, with results discussed in Section VIII. Finally, Section IX concludes the study.

II. Requirements

In this section, the requirements for selecting a suitable concept of a parallel mechanism are described with additional requirements.

A parallel manipulator with three translational DoF along the principal axes is required to apply forces to the wrist in all directions without introducing significant reaction torques. High torques on the wrist could corrupt the measurements and lead to incorrect conclusions. Research showed that torques of approximately 0.06–0.4 N m are the thresholds for the human arm to be perceptible for [18][19]. The threshold can differ depending on the movement and orientation of the human arm.

Trumbower et al. [20] conducted experiments in which perturbations were applied to the hand as random displacements with a flat power spectrum between 0–5 Hz and a standard deviation of 3 mm. The resulting forces ranged between 5 and 10 N. Krutky et al. [21] used a ramp-and-hold perturbation, where a velocity of 400 mm/s was maintained for a duration of 100 ms. Based on these studies, a short-duration impulse of 7.0 N within the first 10 ms is selected. Subsequently, a deceleration equal to one-seventh of the initial acceleration over a period of 70 ms, see Figure 1 for the force profile over time. This produces a counteracting force of 1.0 N, which is considered negligible for the influence on the human response, since it is masked by the weight of the parallel manipulator. Other force profiles can be applied as long as they can be realized with the available range of motion of the same device.

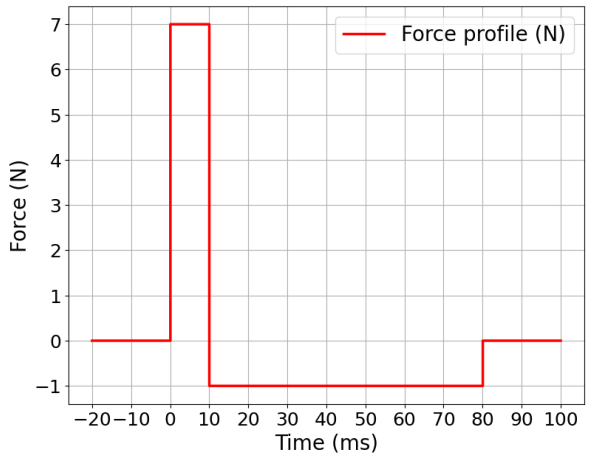


Fig. 1: Force profile over time, derived from the requirements described above. The profile exhibits a peak force of 7.0 N sustained for 10 ms, followed by a deceleration of −1.0 N lasting 70 ms.

To generate sufficient force without requiring an excessively large range of motion, the moving platform is assigned a mass of 0.25 kg. The overall weight of the prototype should not exceed 0.30 kg. With these mass and force specifications, a maximum displacement of 14.0 mm is required. However, for additional flexibility, a displacement capacity of 20.0 mm is chosen. With this additional workspace, singularities and controllability loss is avoided. The definition of the starting position and subsequent displacement paths depends on user input, specifically by selecting the axes along which the wrist should be perturbed. Since the prototype must provide three translational degrees of freedom, perturbations along combinations of axes can also be applied.

To minimize both dimensions and weight, the moving body is initially positioned eccentrically with respect to the wrist. Then the perturbation force profile is applied such that the moving body moves across the center position and comes to rest in an eccentric position on the opposite side.

Real-time estimates of the prototype orientation with a wearable sensor are required to apply the intended perturbations correctly, as specified by the user. An angular error of up to 5° is defined as the acceptable orientation uncertainty for perturbation, since differences below this level are not perceptible at the wrist [22].

III. Kinematic analysis

The requirements outlined in the previous section constrain the choice of suitable parallel manipulators. This section describes the parallel mechanism selected as most appropriate for the present application.

A. Mechanism

A 3-DoF translational parallel mechanism that offers relatively high stiffness, low mass, and rapid dynamic response is the Delta manipulator [23][24]. To prevent axial rotational perturbations, a ring is positioned around the wrist. A larger ring moves around this smaller inner ring, connected by kinematic chains. To constrain the platform and allow motion solely along three translational DoF, 3-RUU legs, as employed in the Delta manipulator, are sufficient. Here, 'R' denotes a revolute joint and 'U' denotes a universal joint. It is preferred that the outer and inner rings are concentric and coplanar in the home position.

Using only 3-RUU legs introduces unwanted singularities, which can reduce controllability [25]. Adding an additional RUU leg mitigates this issue, but introduces an overconstrained condition. At the risk of

generating critical internal forces in the mechanism, the addition of redundant members has often been shown to improve performance [26]. Overconstrained manipulators are significantly less sensitive to clearance and significantly stiffer than the exactly constrained 3-RUU manipulator. In this design, four actuators are mounted on the outer ring connected with a RUU-leg to the inner ring. In this configuration, the mass of the actuators contributes a significant mass necessary for generating perturbation forces during acceleration, following Newton's third law of motion. Studies have shown that the actuating torques needed to move the end-effector in a redundantly actuated 4-RUU parallel manipulator are lower compared with a non-redundant 3-RUU configuration, which is profitable for the mechanism [25]. The arm orientations are selected to alter in the Z-direction. In this configuration, the mechanism has singularities at the limit of the workspace, but it is not sensitive while performing (near) in-plane trajectories than other possible directions.

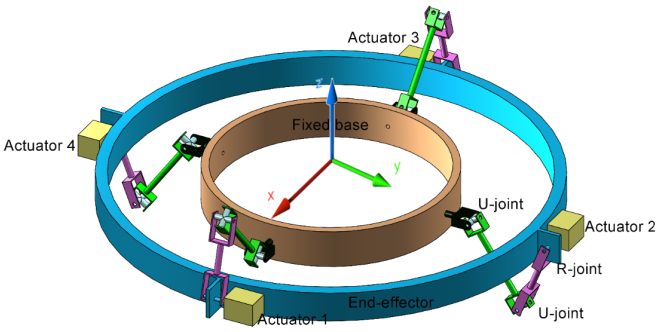


Fig. 2: Simplified concept of the 4-RUU parallel manipulator. The inner ring functions as the fixed base and is intended for wrist mounting. The outer ring, the end-effector, which carries additional mass from the actuators, is capable of perturbations along three translational DoF.

B. *Inverse kinematics*

The inverse kinematics of the parallel mechanism is defined based on the coordinate systems illustrated in Figure 3. The fixed origin O is located in the geometric center of the inner ring. Local frames, shown in Figure 4 define coordinate systems for the four legs (S_i, with $i = 1, 2, 3, 4$), arranged at 90° intervals, and for the center of the outer ring, which serves as the end-effector (E).

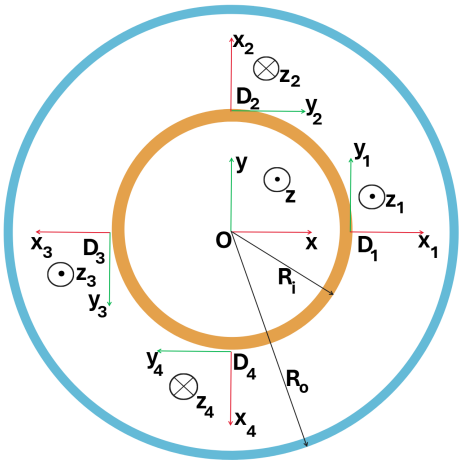


Fig. 3: The fixed frame's origin, O, is located at the center of the inner ring. Accelerations of the outer ring (the end-effector) produce forces that are conveyed through joints D_i connected to the inner ring.

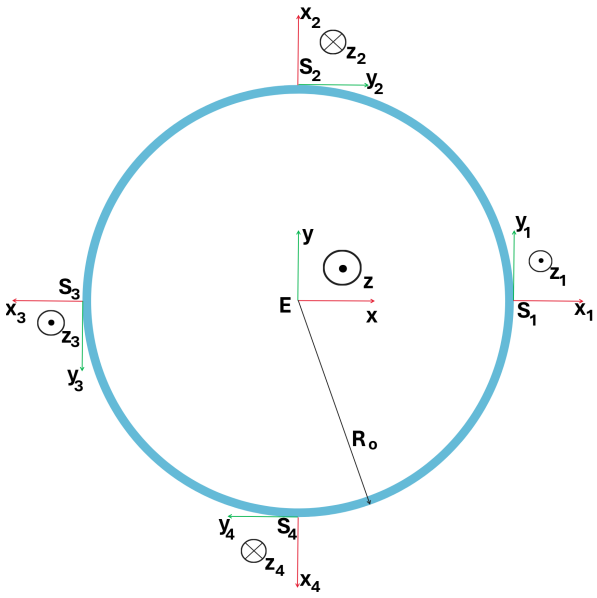


Fig. 4: The moving frame is attached to the outer ring, with the local coordinate origin E located at the center of the end-effector. The four actuated legs are denoted S_i ($i = 1, ..., 4$). Since adjacent legs alternate between upward and downward orientations, the local Z-axis direction varies with the mechanism configuration.

The geometric model of a general leg connected to both inner and outer rings is shown in Figures 5 and 6. The position of point E represents the location of the end-effector, and its relationship to the fixed origin O is described using a vector loop [27]. A pair of connected universal joints is replaced by a parallelogram composed of spherical joints, R-(S-S)^2, because of the unavailability of small universal joints for such a small mechanism. The configurations are mechanically equivalent to each other, only the spherical

joints are underconstrained and can rotate freely along their longitudinal axes, which makes no difference in the inverse kinematics.

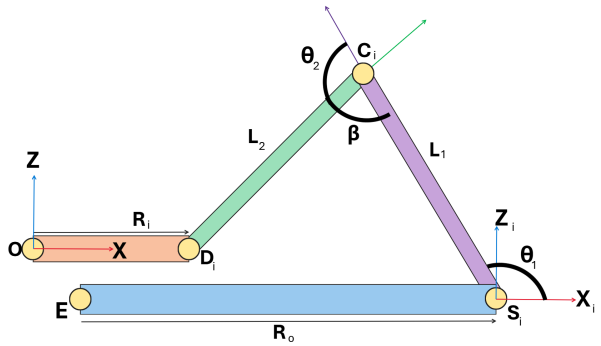


Fig. 5: Side view of arm i used to construct the vector loop. The inner and outer ring halves (R_i and R_o) are shown in orange and blue, respectively. The upper and lower arms are labeled L_1 and L_2. θ_1 indicates the actuation angle; remaining variables used in the analysis are indicated.

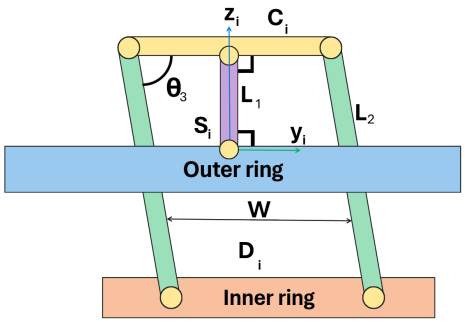


Fig. 6: Front view of arm i illustrating the spatial parallelogram R-(S-S)^2 with spherical joints and its coupling to motion along the local y_i-axis. Because the upper arm L_1 cannot move in the y_i direction, θ_3 maps directly to displacement along that axis. The parallelogram width W is indicated; it does not affect the inverse kinematics but can be increased to reduce mechanical play.

$$\overrightarrow{ES_i} + \overrightarrow{S_iC_i} + \overrightarrow{C_iD_i} = \overrightarrow{OD_i} + \overrightarrow{EO} \tag{1}$$

In matrix format, the individual terms are expressed as follows, where $c\theta_i = \cos\theta_i$ and $s\theta_i = \sin\theta_i$.

$$\overrightarrow{ES_i} = \begin{bmatrix} R_o \\ 0 \\ 0 \end{bmatrix} \tag{2}$$

$$\overrightarrow{S_iC_i} = \begin{bmatrix} L_1 c\theta_1 \\ 0 \\ L_1 s\theta_1 \end{bmatrix} \tag{3}$$

$$\overrightarrow{C_iD_i} = \begin{bmatrix} L_2 c(\theta_1+\theta_2) s\theta_3 \\ L_2 c\theta_3 \\ L_2 s(\theta_1+\theta_2) s\theta_3 \end{bmatrix} \tag{4}$$

$$\overrightarrow{OD_i} = \begin{bmatrix} R_i \\ 0 \\ 0 \end{bmatrix} \tag{5}$$

The vector $\overrightarrow{EO}$ constructs the rotation matrix for the different legs from the displacement of the end-effector (E) with respect to the origin (O), where $c\phi_i = \cos\phi_i$ and $s\phi_i = \sin\phi_i$. This connects the moving frame to the fixed frame for every leg with a 90° leg arrangement. Where legs 1 and 3 are pointing upwards and legs 2 and 4 are pointing downwards. It results in the factor 2 in the lower-right entry of the matrix.

$$\overrightarrow{EO} = \begin{bmatrix} c\phi_i & s\phi_i & 0 \\ s\phi_i & c\phi_i & 0 \\ 0 & 0 & c2\phi_i \end{bmatrix} \begin{bmatrix} E_x \\ E_y \\ E_z \end{bmatrix} \tag{6}$$

For the actuated angles around the outer ring: $\phi_1 = 0°$, $\phi_2 = 90°$, $\phi_3 = 180°$, $\phi_4 = 270°$

$$\overrightarrow{EO} = \begin{bmatrix} E_x c\phi_i + E_y s\phi_i \\ E_x s\phi_i + E_y c\phi_i \\ E_z c2\phi_i \end{bmatrix} \tag{7}$$

For displacements along the local y_i-axis, the angle θ_3 for each leg can be calculated from the X- and Y-coordinates of the inner ring:

$$\theta_3 = \cos^{-1}\left(\frac{E_x \sin\phi_i + E_y \cos\phi_i}{L_2}\right) \tag{8}$$

A new angle β, also presented in Figure 5, is defined for computing θ_2 using the cosine rule: $\cos\beta = -\cos\theta_2 \sin\theta_3$ forming a triangle between points S_i, C_i, and D_i [28]:

$$\overrightarrow{SD}^2 = L_1^2 + L_2^2 - 2L_1L_2\cos\beta \tag{9}$$

with

$$SD_x = R_i + E_x\cos\phi_i + E_y\sin\phi_i - R_o \tag{10}$$

$$SD_y = E_x\sin\phi_i + E_y\cos\phi_i \tag{11}$$

$$SD_z = E_z\cos 2\phi_i \tag{12}$$

$$\theta_2 = \cos^{-1}\left(\frac{SD_x^2 + SD_y^2 + SD_z^2 - L_1^2 - L_2^2}{2L_1L_2\sin\theta_3}\right) \tag{13}$$

The angle θ_1 is obtained by solving:

$$L_1 s\theta_1 + L_2 s\theta_1 c\theta_2 s\theta_3 + L_2 c\theta_1 s\theta_2 s\theta_3 = E_z c2\phi_i \tag{14}$$

To solve this equation, two relations are defined [29]:

$$\sin\theta_1 = \frac{2\tan\left(\frac{\theta_1}{2}\right)}{1+\tan^2\left(\frac{\theta_1}{2}\right)} \tag{15}$$

$$\cos\theta_1 = \frac{1-\tan^2\left(\frac{\theta_1}{2}\right)}{1+\tan^2\left(\frac{\theta_1}{2}\right)} \tag{16}$$

and substituting them, the resulting quadratic equation can be solved:

$$(G+H)\tan^2\left(\frac{\theta_1}{2}\right) - 2F\tan\left(\frac{\theta_1}{2}\right) - G + H = 0 \tag{17}$$

where

$$F = L_1 + L_2\cos\theta_2\sin\theta_3 \tag{18}$$

$$G = L_2\sin\theta_2\sin\theta_3 \tag{19}$$

$$H = E_z\cos 2\phi_i \tag{20}$$

Finally, θ_1 is obtained using the standard quadratic formula:

$$\theta_1 = 2\tan^{-1}\left(\frac{F \pm \sqrt{F^2+G^2-H^2}}{G+H}\right) \tag{21}$$

The θ_1 angle derived for each specific leg with this analytical inverse kinematics is sufficient for angle-controlled actuation to make the right perturbations. This will contribute to the succeeding section to evaluate the mechanism in a simulation with dimension optimization.

IV. Simulation

The inverse kinematics produces discrete motor angles for a specified trajectory over time based on the required force profile. The full mechanism is modeled in the Simulink SimScape environment in MATLAB, which enables analysis of the system's behavior. Because the mechanism is overconstrained, one arm is modeled with a flexible element of very high stiffness to ensure the solver can find a solution. The simulation was solved using a fixed-step size of 2.4×10^{-4} s with the backward Euler method.

Although inertial elements are simplified in the model, the essential geometric and inertial parameters are preserved. The model is driven by inverse kinematics derived motor trajectories for the four servos, and the simulated outputs are used to verify that the kinematic profiles are tracked over time. In addition, the forces and torques computed at the base provide an estimate of the perturbations the haptic device will deliver to a subject.

The dimensions of the previous design were used to establish reference values for the velocities and accelerations required to generate the desired forces; these kinematic requirements were compared against the specifications of candidate servos. To avoid unduly restricting the wearer's natural motion, the device mass was kept low. A target total mass of approximately 0.30 kg (comparable to a modern mobile phone) was chosen. The majority of this mass should be movable, since a moving mass is necessary to produce the required perturbation forces.

TABLE I: Parallel mechanism parameters showing the main geometric dimensions for the old design alongside those of the optimized design, enabling direct comparison of changes introduced by the optimization.

Dimensions	Parameters	Old design	New design
Inner ring radius	R_i	35.7 mm	35.7 mm
Outer ring radius	R_o	64.0 mm	62.0 mm
Upper arm length	L_1	19.0 mm	20.0 mm
Lower arm length	L_2	22.5 mm	20.0 mm
Width	W	16.4 mm	20.0 mm
Moving mass	M_m	0.146 kg	0.25 kg
Total mass	M_t	0.175 kg	0.30 kg

Arm lengths for mechanisms that move in-plane achieve optimal force transmission when the upper- and lower arm ratio is 1:1 [30]. The inner ring radius is fixed; otherwise, it is no longer suitable for a human wrist. The outer ring radius can be reduced slightly to make the prototype more compact without reducing the required maximum displacement capacity of 20.0 mm. With this set, the arm lengths are based on the torque and angular velocity profiles that match the best achievable servo performances. Using these dimensions and the desired force profile (simulated in Figure 7), the maximum torque required from a single servo is 0.12 N m (see Figure 8), and the maximum absolute angular velocity is 10 rad/s, see Figure 10. Consequently, the simulated torques acting on the inner ring are below 0.15 N m, which is considered imperceptible to a human subject, see Figure 9.

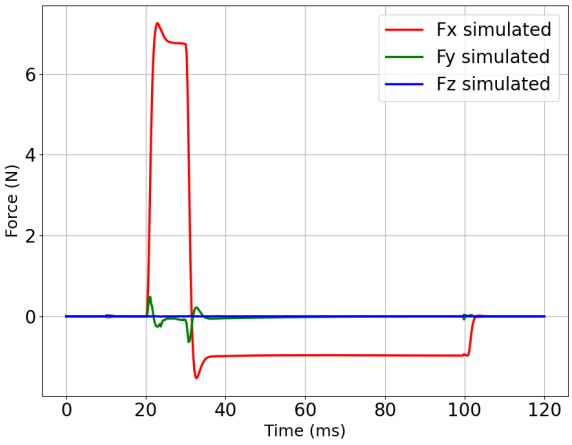


Fig. 7: Force simulation of the Simulink model for a perturbation in the positive X-direction. The simulation follows the force profile specified in the requirements.

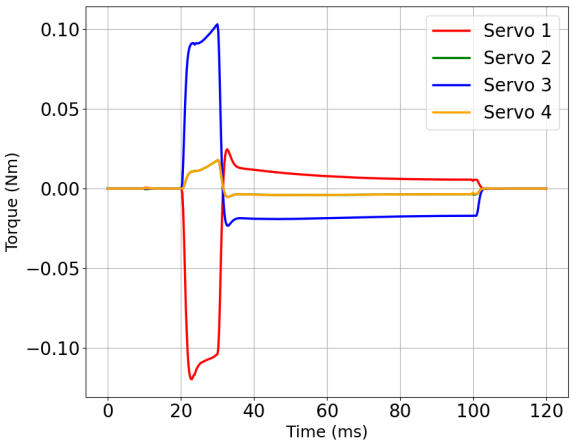


Fig. 8: Simulated servo torques from the Simulink model for a perturbation along the positive X-axis. Torques for servos 2 and 4 overlap due to the symmetry of the mechanism.

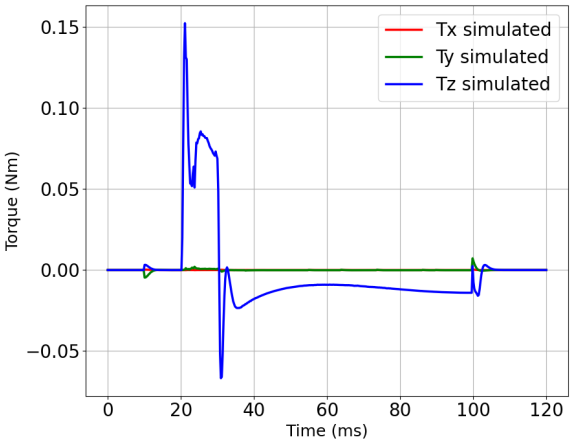


Fig. 9: Simulated torque from the Simulink model for a perturbation along the positive X-axis. The resulting torque magnitude falls below the perceptual threshold specified in the Requirements section.

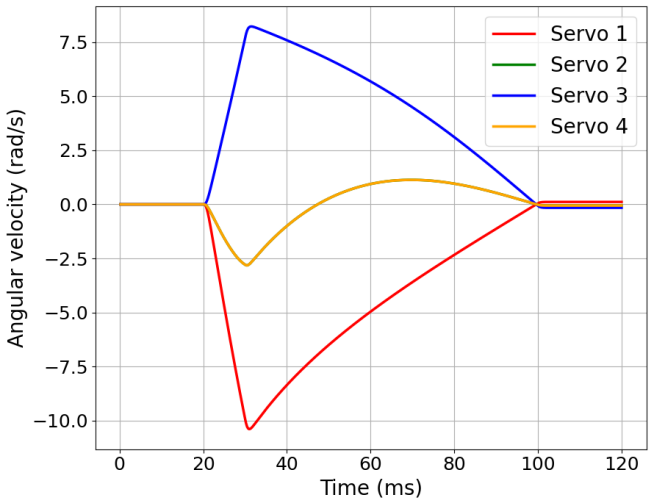


Fig. 10: Simulated angular velocity responses from the Simulink model for a perturbation along the X-axis. The angular velocities of servos 2 and 4 coincide due to the system's symmetry.

V. Prototype

A. Perturbator design

Wherever possible, off-the-shelf components were used in the prototype. Rather than the universal joints described in Section III-A, micro ball joints were implemented to keep the device compact. Consequently, all links connected to the inner ring are under-constrained and can rotate freely about their longitudinal axes. This rotation is physically limited and does not affect the motion of the outer ring, so it does not impair the device's functionality.

Both rings were fabricated from Polylactic Acid (PLA) filament using 3D printing. Additive manufacturing was chosen for rapid prototyping of lightweight structures and for easy design iteration. A stiffer material than PLA would, however, be advantageous to improve overall rigidity. The inner ring is split into two separable halves so it can be mounted around an adapter for experimentation purposes.

The main dimensions and mechanism configuration from the previous sections were applied to the prototype. Actuation is provided by four servomotors mounted on the outer ring; these actuators contribute to the device mass, and dedicated routing channels for wiring were integrated into the CAD design. The resulting SolidWorks model is shown in Figure 11.

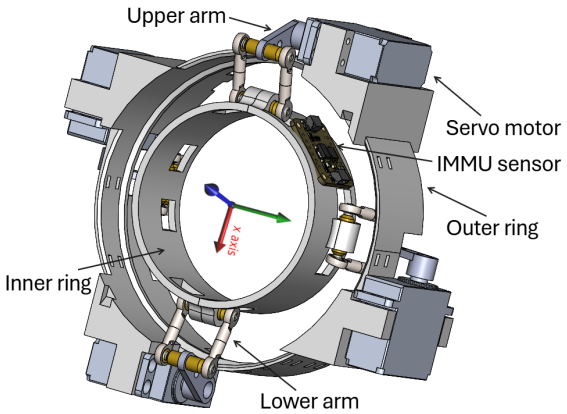


Fig. 11: SolidWorks model of the prototype wearable wrist perturbator with optimized dimensions and an integrated inertial-magnetic sensor.

B. Orientation estimation

To achieve precise three-dimensional orientation estimation, a 9-axis Inertial and Magnetic Measurement Unit (IMMU) sensor is required. The BNO-055 sensor integrates accelerometers, magnetometers, and gyroscopes in a single unit. The relevant noise parameters are provided in the sensor datasheet: $\sigma_w = 0.3\,^\circ$/s $\sigma_a = 0.1$ g $\sigma_m = 0.6\,\mu$T [31][32].

A stable three-axis orientation output can be obtained by fusing these three data streams in real time. For the application of the wrist perturbator, a practical and precise orientation filter was selected, the Madgwick filter. This algorithm is capable of handling nonlinear motions by processing multi-sensor data in quaternion representation with relatively high real-time accuracy. The performance is evaluated in Section VII.

The Madgwick filter [33] computes a quaternion estimate by combining gyroscope, accelerometer, and magnetometer data using a gradient descent optimization (GDA). The optimization problem minimizes the difference between the estimated orientation q and the reference direction in the Earth frame ${}^E d$ with its corresponding measurement in the sensor frame ${}^S s$:

$$f(q, {}^E d, {}^S s) = q^* {}^E dq - {}^S s \tag{22}$$

$$\min f(q, {}^E d, {}^S s) \tag{23}$$

The GDA updates the quaternion estimate as:

$$q_{k+1} = q_k - \mu \frac{\nabla f(q_k, {}^E d, {}^S s)}{|\nabla f(q_k, {}^E d, {}^S s)|} \tag{24}$$

with

$$\nabla f(q_k, {}^E d, {}^S s) = J(q_k, {}^E d)^T f(q_k, {}^E d, {}^S s) \tag{25}$$

The algorithm is applied separately for accelerometer and magnetometer references, using:

$$f_{g,b}(q, {}^S a, {}^E b, {}^S m) = \begin{bmatrix} f_g(q, {}^S a) \\ f_b(q, {}^E b, {}^S m) \end{bmatrix} \tag{26}$$

$$J_{g,b}(q, {}^E b) = \begin{bmatrix} J_g^T(q) \\ J_b^T(q, {}^E b) \end{bmatrix} \tag{27}$$

Finally, the quaternion update is given by:

$$q_t = q_{t-1} + \left(\dot{q}_t - \beta \frac{J_{g,b}^T f_{g,b}}{||J_{g,b}^T f_{g,b}||} \right) \Delta t \tag{28}$$

The gain β represents the balance between fast convergence to the real orientation and noise rejection. The usual gain rule depends on the noise in the gyroscope of the sensor. In a later experiment in this paper, the gain β is further investigated for the wrist perturbator application.

$$\beta = \sqrt{\frac{3}{4}} \sigma_w \approx 0.27 \text{s}^{-1} \tag{29}$$

C. Electronics

For actuation, four KST DS565X digital servos are used; their specifications for different operating voltages are listed in Table II. Each servo integrates a gearbox, potentiometer, and internal controller, which simplifies the actuation hardware. Angular commands are generated by an Arduino Nano and sent as Pulse Width Modulation (PWM) signals. This configuration, however, does not provide direct feedback of torque, current, or the servos' true positions. The servos are powered by a 7.2 V NiMH battery pack. Their small footprint and low mass are sufficient for the required torques and accelerations.

TABLE II: KST DS565X digital servo specifications at various operating voltages.

Operating voltage	6.0 V	7.4 V	8.4 V
Torque (N m)	0.39	0.49	0.59
No load speed (rad/s)	17.45	20.94	26.18
Working frequency (Hz)	560		
Dimensions (mm)	35.5 × 15.0 × 32.5		
Weight (gram)	40		

Communication is handled via the I2C protocol, which uses a serial clock line (SCL) and a serial data line (SDA) [34]. A PCA9685 16-channel, 12-bit PWM

driver uses this interface to generate PWM signals for the four servos at the required operating frequency. IMMU data are also transmitted over the same data line and processed as described in Section V-B to estimate orientation. The complete wiring diagram of the prototype is shown in Figure 14.

Servo angular trajectories are obtained from the analytical inverse kinematics presented in Section III-B. A Python script computes these trajectories from the required end-effector motion. The end-effector path is defined by the user-specified perturbation force profile and direction. To convert the force profile into motion, the acceleration profile (scaled by mass to produce force) is integrated to velocity and displacement using the composite trapezoidal rule.

Because there is no general closed-form mapping from desired servo angle to PWM value, a calibration procedure is required for each servo. A range of PWM commands is applied, and the resulting angles are measured with a digital inclinometer (see Figure 12 for the calibration setup). The digital inclinometer has a precision of 0.2°. The measured angle vs PWM data are recorded and fitted with a trend line to produce an angle-PWM conversion function for each servo. Calibrations were repeated several times: the servos consistently exhibit the same slope, although their zero points can shift over time. Therefore, calibration before use is important.

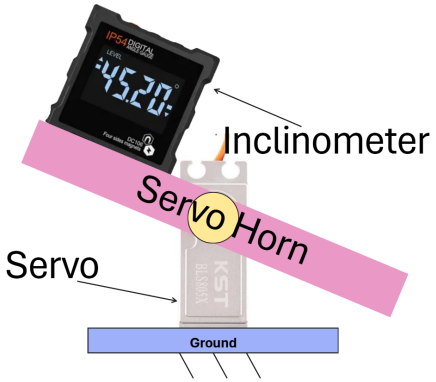


Fig. 12: Schematic of the servo calibration setup. Each servo is clamped to the ground, and a 3D-printed adapter on each servo horn facilitates inclination measurements with an inclinometer.

After calibration, the perturbation trajectory is converted into PWM signals and stored in arrays. Due to performance limitations, the servos reliably follow only a finite number of setpoints per perturbation. In practice, the system can execute up to 40 data points with a 2 ms interval between successive points.

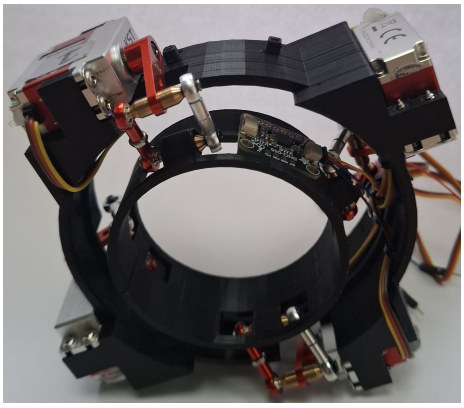


Fig. 13: Fully assembled perturbator prototype, constructed to the dimensions specified by the simulation and ready for actuation.

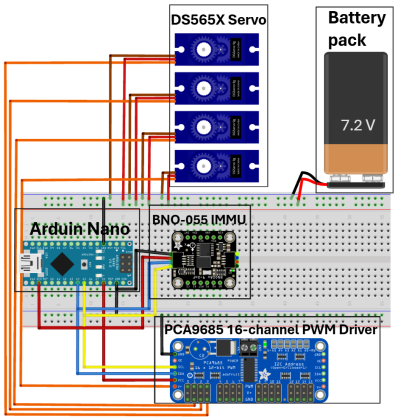


Fig. 14: Wiring diagram for servo actuation and measurement using an IMMU sensor. An Arduino Nano performs control and computation and is connected to a laptop for data acquisition.

VI. Perturbation Experiment

The constructed prototype was tested to evaluate its perturbation forces. Using the actuation protocol described previously, a variety of perturbations were applied to the inner ring and the resulting forces were measured.

A. Experimental setup

The prototype was mounted to an OnRobot HEX-H QC six-axis force-torque sensor using a 3D-printed adapter (see Figure 15). The sensor's specifications are adequate for this application: a maximum sampling rate of 500 Hz and an approximate noise level of 0.2 N in the X- and Y-axes; noise in the Z-axis is notably higher ($\approx$ 0.8 N). The sensor assembly was clamped to the ground to ensure a rigid measurement reference.

The HEX-H QC communicates via an Ethernet cable with User Datagram Protocol, permitting data acquisition at up to 500 Hz. Automatic zeroing was used

before measurements; no further filtering was applied during acquisition, and all filtering was performed in post-processing. Data was recorded and contains force and torque components along the three main axes.

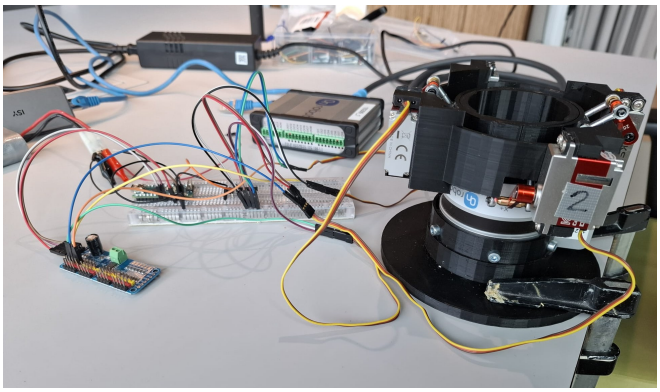


Fig. 15: Experimental setup for measuring forces during a perturbation. The prototype is mounted to the HEX-H QC force/torque sensor, which is clamped to the ground. The four servos are actuated with the wiring diagram as presented earlier.

Because the sensor's Z-axis noise was relatively high, a second 3D-printed adapter (rotated 90°) was used to redirect prototype motion in the Z-direction to the sensor's X- or Y-axis. This approach reduces reliance on the sensor's Z-axis readings.

Perturbation trajectories computed and stored as PWM arrays were sent to the four servos as described in the previous section. Measurements were started with a button press. Each measurement lasted one second; the sensor recorded forces and torques for the full measurement duration. Because actuation and force-sensor recording were not hardware-synchronized, alignment of the actuation and measurement traces was performed in post-processing.

The prototype can translate along the three principal axes and therefore generate perturbations in an effectively continuous space of directions. For practical evaluation, testing was restricted to perturbations along the principal axes and their combinations, yielding 18 distinct perturbation directions. Each direction was repeated ten times to compute mean forces and standard deviations.

B. Arm optimization

Two lower arm variants with identical dimensions are available for constructing the parallelogram of the mechanism. The red-anodized arms exhibit relatively large backlash, whereas the silver-anodized arms show relatively high friction. Increased friction reduces servo speed because the servos must deliver greater torque to maintain end-effector accelerations. Conversely, backlash degrades mechanism stability and generates forces in undesired directions. Force-perturbation measurements on the prototype are used to identify which arm arrangement most closely follows the theoretical force profile. From the many possible arrangements, five configurations that are not symmetry-equivalent were selected for detailed study. Each configuration is perturbed with a 10.0 N force peak, because preliminary tests revealed that the end-effector accelerations are not reached with the selected servos. The resulting force measurements in the X-direction for the five configurations are plotted. Each plot includes a schematic of the corresponding lower arm configuration.

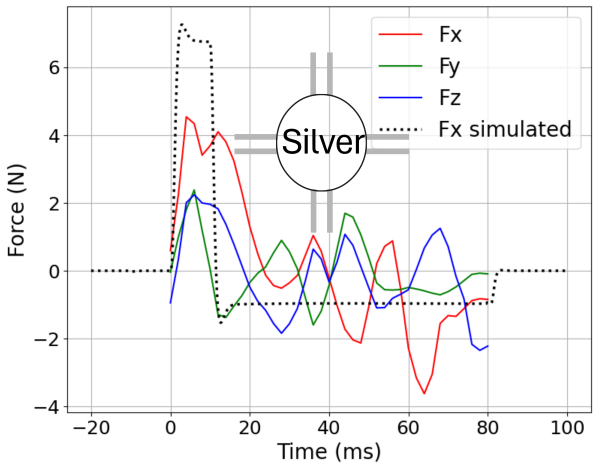


Fig. 16: Configuration using only silver-anodized arms. High friction in this configuration reduces the peak amplitude $\leq$ 5 N and increases the peak width.

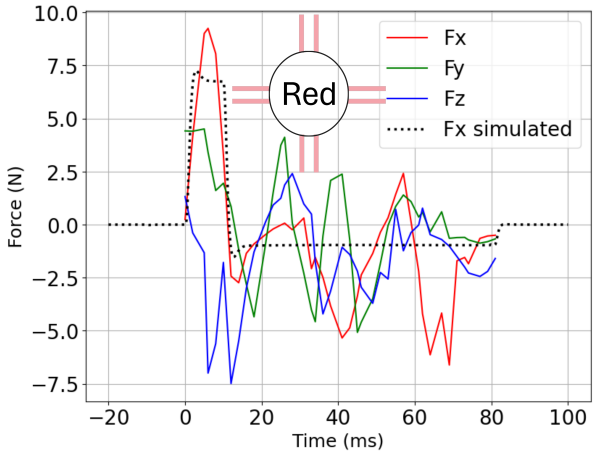


Fig. 17: Configuration with only red-anodized arms. Excessive play in this configuration causes dynamic instability. A perturbation along a single axis can produce a high peak force and induce cross-axis coupling with forces on the other axes.

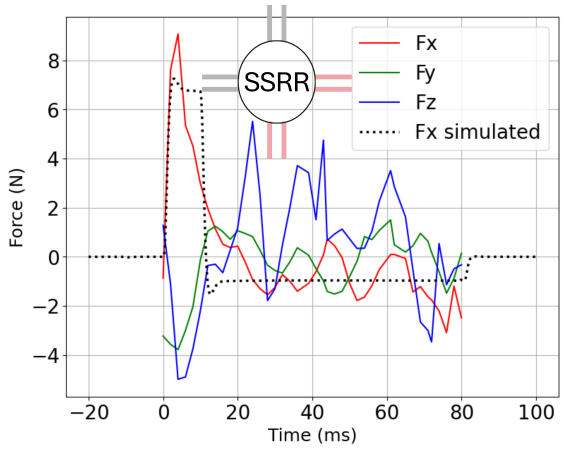


Fig. 18: Hybrid configuration produces a force peak, but forces measured along the other axes remain elevated. This hybrid configuration exhibits reduced stability under perturbation.

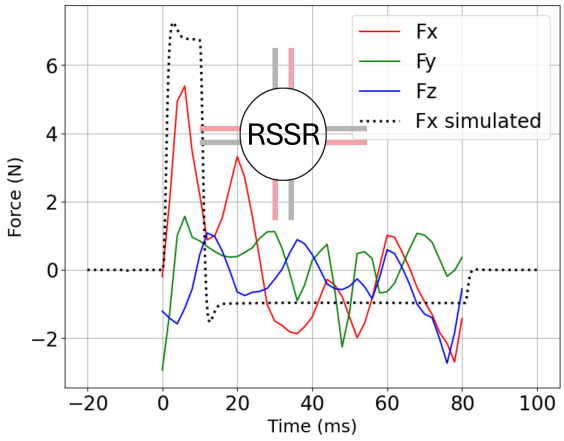


Fig. 19: The configuration shown is poorly controllable. The response exhibits a broad peak, and the peak force does not meet the specified requirements.

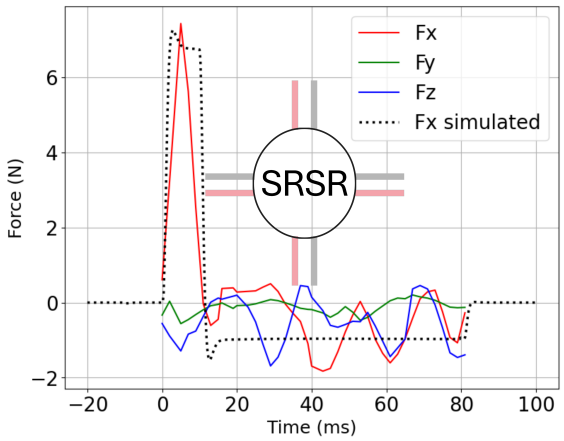


Fig. 20: The configuration provides the best stability and controllability in all perturbation directions. The peak force is achieved without significant cross-axis forces. This configuration produces the most favorable force measurements and is selected for further investigation in the perturbation experiments.

The different arm configurations produced notable differences in the measured force. With only silver-anodized arms (Figure 16), the servos must deliver high torques to overcome friction, which reduces end-effector accelerations and therefore the measured forces. A mechanism composed solely of red-anodized arms (Figure 17) does not contain such high torques; when actuated, the expected accelerations, and consequently the theoretical forces, are attained. However, this configuration exhibited relatively large unwanted forces in orthogonal directions during perturbation, caused by mechanical play. The hybrid mechanism (Figure 18) shows similar behaviour: the silver-anodized arms introduce additional friction that lowers the peak force, and asymmetric placement of arms generates forces in unintended directions. Although the configuration in Figure 19 attains a comparable peak force, it is less stable during the remainder of the perturbation. The SRSR configuration shown in Figure 20 provides the best combination of peak force without cross-axis measurement.

C. *Perturbation results*

During the tests, the start times of the perturbations relative to the measurement times of the sensor were stored in order to synchronize the perturbations with the measurements. The received data were filtered using a threshold wavelet denoising filter [35]. The wavelet chosen was 'sym4' with level 2 and a threshold of 0.2 N, corresponding to the noise level of the sensor. Level 2 preserved the short-term features while primarily removing noise. All filtered data for a particular perturbation were combined to compute a mean response and standard deviation. The resulting data were compared with the simulation results from MATLAB Simulink.

The perturbations were calculated with a peak force of 10.0 N and a moving mass of 0.25 kg to account for the lack of performance of the used servos. This required a force in the opposite direction of 1.43 N to decelerate the moving mass. Single servo validation tests demonstrated that this force profile most closely approximated the desired force profile along all axes, as defined in Section II. The resulting displacement of the moving mass was 16.0 mm, which was still achievable within the limits of the mechanism. For a single perturbation in the X-direction, it can be seen in Figure 21 that it has a nice overlap with the simulated forces.

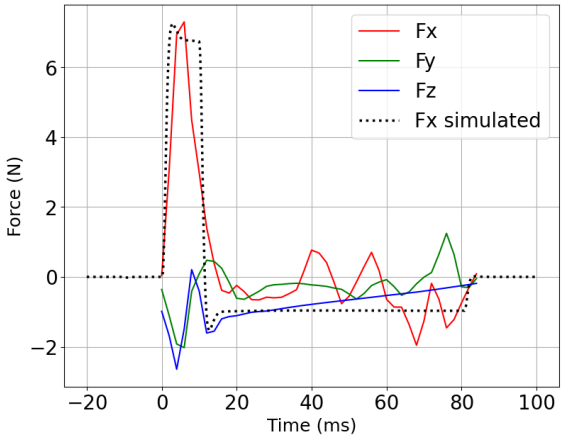


Fig. 21: Measured force response for a perturbation applied along the positive X-axis, following the previously described force profile. The dotted line denotes the Simulink-simulated X-direction force, while the remaining simulated components are negligible.

Perturbations in a single direction are measured ten times to check repeatability and evaluate the mean and standard deviations over time. See in Figure 22 the mean and deviations for a perturbation force in the negative Z-direction.

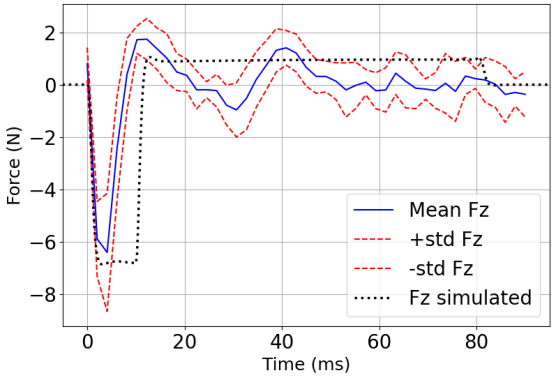


Fig. 22: Measured force responses (n = 10) for a perturbation applied in the negative Z-direction. The solid blue line denotes the mean; the red dashed lines indicate the mean ± one standard deviation. Experimental results are compared with the Simulink simulation force in the negative Z-direction. The other simulated components are approximately zero.

The plots show that a perturbation can also be realized along combinations of axes, as in Figure 23. Although some losses can be distinguished in the theoretical force profile put in to actuate the prototype with respect to the measurements. A certain force profile with peak can always be distinguished and the measurements are repeatable. The maximum absolute torque during a perturbation ranges in the order of 0.1–0.4 N m. The losses in force for perturbations in different directions with the theoretical peak force of 10.0 N, show a loss range of 14–50 %.

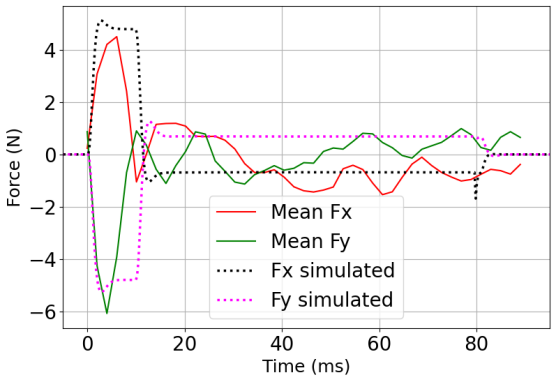


Fig. 23: Measured force responses (n = 10) for perturbation applied in the +X-/−Y-direction. The plot shows that the forces are divided well over the X- and Y-component forces closely follow the Simulink-simulated profiles. Standard-deviation traces are omitted for clarity.

VII. Orientation Experiment

The second experiment investigated the performance of the orientation measurement using an IMMU sensor, which gathers data from an accelerometer, gyroscope, and a magnetometer, all in three axes. Together with an incorporated Madgwick filter, orientations are estimated, as described in Section V-B. The BNO-055 sensor is placed on the inner ring of the prototype, closer to the rotation point, giving more reliable data [33]. The data input rate for the filter is the same as the output rate of the filter, with a rate of 50 Hz.

A. Orientation setup

To evaluate the accuracy of the orientation estimation, a test with the built prototype and the integrated sensor was constructed. The prototype was mounted on an adapter, and a reference block with multiple standard angles was clamped to the ground and aligned with the Earth's magnetic North.

First, the sensor was calibrated by rotating it in multiple directions and keeping it still at predefined angles. During the calibration procedure, the calibration status was monitored automatically by the BNO-055 sensor software. The calibration was completed once all internal sensors (system, gyroscope, accelerometer, and magnetometer) reached a full calibration state, indicated by the sensor software with a value of 3.

After calibration, random rotations and translations were applied manually for 10 seconds. Thereafter, a known orientation was imposed by positioning the prototype against the reference block. At that specific moment, the button was pressed, causing the Euler angles to be recorded for 5 seconds. The true orientation in Euler angles can be compared directly to the

real output Euler angles estimated with the Madgwick filter. This procedure quantified both the accuracy of the orientation estimation and the convergence rate to the true angle.

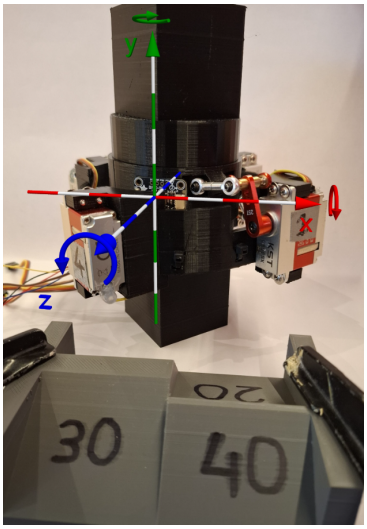


Fig. 24: Experimental setup used to determine prototype orientation. The prototype with an integrated BNO-055 IMMU sensor is mounted to an adapter. The adapter is pressed against a reference block, rotated to predefined standard angles, and aligned with Earth's magnetic north to minimize alignment errors and ensure repeatability. Euler angles are defined as rotation about the X-axis (roll), Y-axis (pitch), and Z-axis (yaw).

B. Orientation results

The Euler angles are compared with the real Euler angles in a plot over time. A rotation matrix represents the orientation unambiguously; the relative rotation $R_{err} = R_{est}R_{true}^{T}$ is the rotation that takes the true attitude into the estimated attitude. The angle of that rotation is the minimal rotation required. Then the total angle error is computed with the trace of the error rotation matrix and cosine.

$$\alpha = \cos^{-1}\left(\frac{\mathrm{tr}(R_{err}) - 1}{2}\right) \tag{30}$$

The gain β for the Madgwick filter is crucial for the right balance between fast convergence to the real orientation and noise rejection. The bigger β, the bigger the noise in the estimation, but the faster the convergence is to the real orientation. The usual gain rule depends on the noise in the gyroscope of the sensor, as explained earlier. In this case, the gain should be $\beta = 0.27\,\mathrm{s}^{-1}$. This specific gain shows the best trade-off between fast convergence and noise rejection in the estimation.

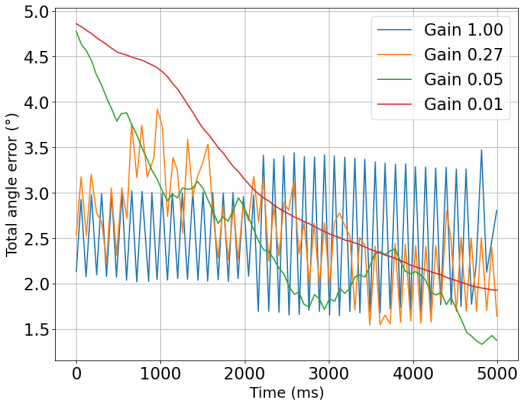


Fig. 25: Total angular error over time for a fixed orientation (roll = 50°), measured with different Madgwick filter gains (β). Increasing β reduces convergence time at the expense of increased estimation noise. The remaining steady-state error persists because of sensor noise, bias, and model inaccuracies.

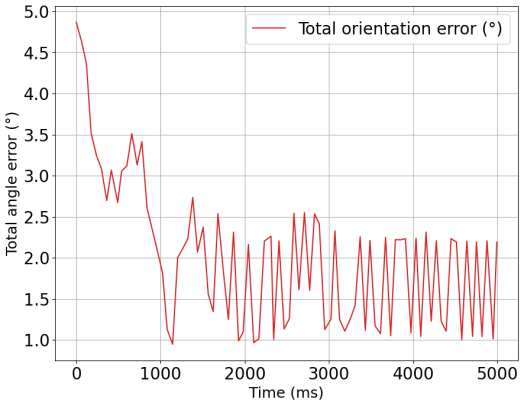


Fig. 26: Total angular error of a single-trial orientation estimate using Madgwick gain $\beta = 0.27\,\mathrm{s}^{-1}$. The plot demonstrates convergence toward the true orientation, which is set to roll = 40°, pitch = 0°, and yaw = 180°.

VIII. Discussion

Thanks to the developed inverse kinematics for the four-leg RUU parallel mechanism, the design has improved substantially. Forward kinematics was not analyzed further because of the system's complexity. The inverse kinematics was sufficient to help with optimizing the dimensions and improving motion planning. The fourth redundant leg helps the mechanism maintain stable and controllable with motions along the three translational degrees of freedom without the problem of singularities. As a result, the prototype can generate ungrounded motion along the three principal axes and along combinations of those axes.

An extra test was performed to evaluate a single servo. The angle of this servo was tracked with a camera during the actuation, with the profile that was

also used during all the other perturbations. Various constant torques were applied to the single servo, this shows that the actual servo performance is substantially lower than was given in the datasheet. The torque needed for a single servo for the defined perturbation is approximately 0.12 N m. The maximum angular velocity tracked with this torque was 40 % lower than needed. That is the reason that a theoretical force profile with a peak of 10 N was used to reach the right acceleration of the end-effector, and thereby the right force of 7 N was measured. Quantification of how the end-effector moves in time during perturbation is also interesting to investigate to validate the measured forces, but was kept outside the scope of this study.

Adding a dedicated feedback wire from the servo potentiometer to the control module would enable more precise control and allow implementation of a custom controller, for example, a proportional–integral–derivative (PID) controller. At present, the servos use their built-in controllers and follow the supplied PWM signals. An initial test with an additional feedback wire produced unwanted results, such as a short circuit on the servo board and unrealistic rotation patterns, likely caused by poor soldering. That is why the test was aborted to avoid damaging the servos. The concept of a direct feedback connection remains a good idea in theory, further testing with proper wiring and precise soldering is recommended.

The orientation experiment produced encouraging results for estimating the prototype's orientation. Using the implemented Madgwick filter and a combined gain study, a maximum total orientation error of 5° was measured with a reference block. In this prototype, the IMMU was mounted on the inner ring, which makes magnetometer readings susceptible to interference from the magnetic servos. When the outer ring is stationary, this interference can be removed from the data, but when the outer ring moves, the reference frame changes, and cancellation is not possible. For that reason, it is preferable to mount the IMMU on the outer ring so that the interference from the servos remains constant and can be compensated for. With the current mounting and sensor accuracy, the IMMU is adequate for determining the direction of intended perturbations, but it is insufficient for accurately measuring the arm's motion during perturbation, because pure translations are not estimated with the incorporated Madgwick filter. Deployment outside the laboratory has therefore not yet been achieved.

IX. Conclusion

This work optimized the primary dimensions of a parallel mechanism capable of translation in three degrees of freedom. The optimization employed a newly derived analytical inverse kinematics formulation for a four-legged RUU parallel mechanism. As a result, the mechatronic system's control and its ability to generate ungrounded forces on the human arm for a prescribed force profile improved. However, the measured forces remained 14–50 % below theoretical and simulated values. A theoretical force profile with a short-duration peak of 10.0 N was applied to the prototype, from which a repeatable short peak of approximately 7.0 N was measured in all the tested directions.

The prototype's total mass is 0.30 kg, which is considered a minimal hindrance to movement during experiments. Measured parasitic torques reach a maximum of 0.4 N m, equal to the threshold of perceptibility for a human limb. The prototype's maximum producible force is 10 N, but operating at that level increases cross-axis coupling and produces larger parasitic torques.

Real-time orientation estimation using an IMMU and an implemented Madgwick filter was feasible: the maximum total angular error remained below 5°, which is sufficient to determine when a direction-specific perturbation should be applied. Collectively, these developments bring the wearable wrist perturbator design closer to enabling system identification on a human arm during daily activities.

References

[1] R. Kearney and I. Hunter, "System identification of human joint dynamics," *Critical reviews in biomedical engineering*, vol. 18, pp. 55–87, 02 1990.

[2] Y. Huang, X. Wang, D. Liang, J. Xiong, A. Xie, and J. Gu, "Modeling and Parameter Identification for Human-robot Coupled Systems in Powered Lower Limb Prostheses." *2023 WRC Symposium on Advanced Robotics and Automation, WRC SARA 2023*, vol. 57243, pp. 14–20, 8 2023. [Online]. Available: https://doi.org/10.1109/wrcsara60131.2023.10261834

[3] A. Schwarz, M. M. C. Bhagubai, G. Wolterink, J. P. O. Held, A. R. Luft, and P. H. Veltink, "Assessment of Upper Limb Movement Impairments after Stroke Using Wearable Inertial Sensing," *Sensors*, vol. 20, no. 17, p. 4770, 8 2020. [Online]. Available: https://doi.org/10.3390/s20174770

[4] G. Venture, K. Yamane, Y. Nakamura, and T. Yamamoto, "Identification of Human Limb Viscoelasticity using Robotics Methods to Support the Diagnosis of Neuromuscular Diseases," *The International Journal of Robotics Research*, vol. 28, no. 10, pp. 1322–1333, 5 2009. [Online]. Available: https://doi.org/10.1177/0278364909103786

[5] S. Kloiber, V. Settgast, C. Schinko, M. Weinzerl, J. Fritz, T. Schreck, and R. Preiner, "Immersive analysis of user motion in VR applications," *The Visual Computer*, vol. 36, no. 10-12, pp. 1937–1949, 8 2020. [Online]. Available: https://doi.org/10.1007/s00371-020-01942-1

[6] A. Schouten, E. De Vlugt, J. Van Hilten, and F. Van Der Helm, "Quantifying Proprioceptive Reflexes During Position Control of the Human Arm," *IEEE Transactions on Biomedical Engineering*, vol. 55, no. 1, pp. 311–321, 12 2007. [Online]. Available: https://doi.org/10.1109/tbme.2007.899298

[7] W. Wang, Z.-G. Hou, L. Tong, Y. Chen, L. Peng, and M. Tan, "Dynamics modeling and identification of the human-robot interface based on a lower limb rehabilitation robot," *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 5 2014. [Online]. Available: https://doi.org/10.1109/icra.2014.6907745

[8] M. Van De Ruit, L. L. Van Der Velden, B. Onneweer, J. L. Benner, C. J. W. Haarman, G. M. Ribbers, and R. W. Selles, "System identification: a feasible, reliable and valid way to quantify upper limb motor impairments," *Journal of NeuroEngineering and Rehabilitation*, vol. 20, no. 1, 5 2023. [Online]. Available: https://doi.org/10.1186/s12984-023-01192-x

[9] H. M. Hondori and N. A. W. Tech, "Smart mug to measure hand's geometrical mechanical impedance," *Annu Int Conf IEEE Eng Med Biol Soc.*, vol. 7, pp. 4053–4056, 8 2011. [Online]. Available: https://doi.org/10.1109/iembs.2011.6091007

[10] Y. Xu, I. Hunter, J. Hollerbach, and D. Bennett, "An airjet actuator system for identification of the human arm joint mechanical properties," *IEEE Transactions on Biomedical Engineering*, vol. 38, no. 11, pp. 1111–1122, 1 1991. [Online]. Available: https://doi.org/10.1109/10.99075

[11] H. Gurocak, S. Jayaram, B. P. And, and U. Jayaram, "Weight sensation in virtual environments using a haptic device with air jets," *Journal of Computing and Information Science in Engineering*, vol. 3, no. 2, pp. 130–135, 6 2003. [Online]. Available: https://doi.org/10.1115/1.1576808

[12] J. Belden, W. L. Staats, A. Mazumdar, and I. W. Hunter, "A portable air jet actuator device for mechanical system identification," *Review of Scientific Instruments*, vol. 82, no. 3, p. 035106, 3 2011. [Online]. Available: https://doi.org/10.1063/1.3562894

[13] K. N. Winfree, J. Gewirtz, T. Mather, J. Fiene, and K. J. Kuchenbecker, "A high fidelity ungrounded torque feedback device: The iTorqU 2.0," *World Haptics 2009 - Third Joint EuroHaptics conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, 1 2009. [Online]. Available: https://doi.org/10.1109/whc.2009.4810866

[14] J. M. Walker, M. Raitor, A. Mallery, H. Culbertson, P. Stolka, and A. M. Okamura, "A dual-flywheel ungrounded haptic feedback system provides single-axis moment pulses for clear direction signals," *2016 IEEE Haptics Symposium (HAPTICS)*, pp. 7–13, 4 2016. [Online]. Available: https://doi.org/10.1109/haptics.2016.7463148

[15] T. Amemiya, I. Kawabuchi, H. Ando, and T. Maeda, "Double-layer slider-crank mechanism to generate pulling or pushing sensation without an external ground," *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2101–2106, 10 2007. [Online]. Available: https://doi.org/10.1109/iros.2007.4399211

[16] E. Kakuda, S. Yokota, A. Matsumoto, D. Chugo, and H. Hashimoto, "Concept Verification of Ungrounded Force Display Using Cam," *2018 11th International Conference on Human System Interaction (HSI)*, pp. 432–437, 7 2018. [Online]. Available: https://doi.org/10.1109/hsi.2018.8431319

[17] R. Koene, J. Meijaard, M. Van De Ruit, W. Mugge, and V. Van Der Wijk, *Design and validation of a 3-DOF wrist perturbator based on an inverted spatial redundant 4-RUU parallel manipulator*. Springer Nature Switzerland, 1 2023. [Online]. Available: https://doi.org/10.1007/978-3-031-45705-0_76

[18] H. Kim and A. T. Asbeck, "Just noticeable differences for elbow joint torque feedback," *Scientific Reports*, vol. 11, no. 1, p. 23553, 12 2021. [Online]. Available: https://doi.org/10.1038/s41598-021-02630-3

[19] C. G. Welker, S. H. Collins, and A. M. Okamura, "Human perception of wrist flexion and extension torque during upper and lower extremity movement," *IEEE Transactions on Haptics*, vol. 15, no. 4, pp. 741–752, 10 2022. [Online]. Available: https://doi.org/10.1109/toh.2022.3219031

[20] R. D. Trumbower, M. A. Krutky, B.-S. Yang, and E. J. Perreault, "Use of Self-Selected Postures to Regulate Multi-Joint Stiffness During Unconstrained Tasks," *PLoS ONE*, vol. 4, no. 5, p. e5411, 4 2009. [Online]. Available: https://doi.org/10.1371/journal.pone.0005411

[21] M. A. Krutky, R. D. Trumbower, and E. J. Perreault, "Influence of environmental stability on the regulation of end-point impedance during the maintenance of arm posture," *Journal of Neurophysiology*, vol. 109, no. 4, pp. 1045–1054, 12 2012. [Online]. Available: https://doi.org/10.1152/jn.00135.2012

[22] S. Allin, Y. Matsuoka, and R. Klatzky, "Measuring just noticeable differences for haptic force feedback: implications for rehabilitation," *Proceedings 10th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems. HAPTICS 2002*, 6 2003. [Online]. Available: https://doi.org/10.1109/haptic.2002.998972

[23] T. H. T. Tran, D. S. Nguyen, N. T. Vo, and H. N. Le, "Design of Delta Robot Arm based on Topology optimization and Generative Design Method," *2020 5th International Conference on Green Technology and Sustainable Development (GTSD)*, 11 2020. [Online]. Available: https://doi.org/10.1109/gtsd50082.2020.9303083

[24] A.-A. Ayazbay, G. Balabyev, S. Orazaliyeva, K. Gromaszek, and A. Zhauyt, "Trajectory planning, kinematics, and experimental validation of a 3D-Printed Delta robot manipulator," *International Journal of Mechanical Engineering and Robotics Research*, vol. 13, no. 1, pp. 113–125, 1 2024. [Online]. Available: https://doi.org/10.18178/ijmerr.13.1.113-125

[25] H. Qu, S. Guo, and Y. Zhang, "Kinematics analysis of a redundantly actuated 4-RUU translational parallel manipulator and its nonredundant 3-RUU counterpart," *Proceedings of the Institution of Mechanical Engineers Part C Journal of Mechanical Engineering Science*, vol. 231, no. 17, pp. 3238–3249, 4 2016. [Online]. Available: https://doi.org/10.1177/0954406216642798

[26] D. Corbel, M. Gouttefarde, O. Company, and F. Pierrot, "Towards 100G with PKM. Is actuation redundancy a good solution for pick-and-place?" *2010 IEEE International Conference on Robotics and Automation*, pp. 4675–4682, 5 2010. [Online]. Available: https://doi.org/10.1109/robot.2010.5509921

[27] H. S. Kim, "Design of a novel 4-DOF High-Speed Parallel Robot," *International Journal of Mechanical Engineering*

and Robotics Research, pp. 500–506, 1 2018. [Online]. Available: https://doi.org/10.18178/ijmerr.7.5.500-506

[28] C. Urrea, C. Domínguez, and J. Kern, "Modeling, design and control of a 4-arm delta parallel manipulator employing type-1 and interval type-2 fuzzy logic-based techniques for precision applications," *Robotics and Autonomous Systems*, vol. 175, p. 104661, 2 2024. [Online]. Available: https://doi.org/10.1016/j.robot.2024.104661

[29] J. Stewart, *Calculus, Early Transcendentals, International Metric Edition*, 8th ed. Cengage Learning, Inc, 8 2015.

[30] J. P. Merlet, "Jacobian, manipulability, condition number, and accuracy of parallel robots," *Journal of Mechanical Design*, vol. 128, no. 1, pp. 199–206, 6 2005. [Online]. Available: https://doi.org/10.1115/1.2121740

[31] K. Townsend and E. Herrada. (2015, 4) AdaFruit BNO055 Absolute Orientation Sensor. Retrieved March 12, 2025. [Online]. Available: https://learn.adafruit.com/adafruit-bno055-absolute-orientation-sensor/overview

[32] B. Sensortec, "BNO055 Intelligent 9-axis absolute orientation sensor," 10 2021. [Online]. Available: https://www.bosch-sensortec.com/media/boschsensortec/downloads/datasheets/bst-bno055-ds000.pdf

[33] S. O. H. Madgwick, A. J. L. Harrison, and R. Vaidyanathan, "Estimation of IMU and MARG orientation using a gradient descent algorithm," *IEEE International Conference on Rehabilitation Robotics*, pp. 1–7, 6 2011. [Online]. Available: https://doi.org/10.1109/icorr.2011.5975346

[34] F. Leens, "An introduction to I2C and SPI protocols," *IEEE Instrumentation Measurement Magazine*, vol. 12, no. 1, pp. 8–13, 1 2009. [Online]. Available: https://doi.org/10.1109/mim.2009.4762946

[35] C. Li, D. Cao, and Y. Yuan, "Research on Improved Wavelet Denoising Method for sEMG Signal," *Proceedings - 2019 Chinese Automation Congress, CAC 2019*, pp. 5221–5225, 11 2019. [Online]. Available: https://doi.org/10.1109/cac48633.2019.8996945

3
Discussion

In the literature review, non-intrusive wrist-orientation measurements were discussed. The wearable sensors commonly used for orientation estimation are the accelerometer, gyroscope, and magnetometer. Each sensor has advantages and limitations; therefore, an inertial and magnetic measurement unit (IMMU), which integrates these three sensors on all three axes, is the preferred choice. With appropriate sensor fusion algorithms, it is possible to estimate nonlinear orientations in real time. Two suitable filter methods are the Extended Kalman Filter (EKF) and the Madgwick filter. With periodic calibration (every 15 minutes in this work) and validation against optical tracking systems, wrist orientation can be estimated reliably during daily activities, and the resulting orientation data can later be validated against optical tracking of the wrist.

The orientation data is important for the system identification of the human arm. System identification can be performed using a wearable perturbator attached to the wrist that applies ungrounded forces in multiple directions to observe the arm's response. A lightweight, compact, four-legged RUU parallel manipulator was designed to provide perturbations in 3-DoF. An additional non-redundant arm was included for extra controllability; however, this produced an overconstrained system with four servo actuators. Inverse kinematics was derived analytically to compute servo actuation angles over time for a prescribed theoretical force profile.

During experimental characterization, it was found that the servos' short-time performance was substantially lower than the datasheet specifications. As a result, end-effector accelerations and, therefore, achievable perturbation force (mass times acceleration), were lower than predicted. This discrepancy will be taken into account in subsequent real-world perturbation experiments. Prior to prototype fabrication, the main mechanism dimensions were optimized using MATLAB Simulink simulations. A new prototype was built with a total mass of 0.30 kg and a total movable mass of 0.25 kg.

Short, fast perturbations were applied to the prototype while it was mounted on a six-axis force/torque sensor. Measured forces were substantially lower than simulated values, an effect attributed primarily to the lower-than-expected servo performance. The angular-velocity performance was approximately 40 % lower when a constant torque of 0.12 N m was applied to the servo. To compensate, a theoretical force profile increased by 40 % was tested and produced improved force measurements: force peaks became higher and more repeatable across measured directions, with a minimum peak of 4.42 N. Overall, measured force losses across directions range from 14 % to 50 % relative to the theoretical applied forces. These results indicate that a dedicated control architecture and higher-performance actuators are required to achieve the intended perturbation profiles.

An initial attempt to implement direct feedback control involved adding an extra wire to a servo for potentiometer readout. That modification inadvertently caused short-circuiting and unwanted rotations; nevertheless, a robust direct-feedback implementation could theoretically improve control precision once the electrical issues are resolved.

The literature findings informed the sensor choice and filtering approach for the prototype: an integrated IMMU combined with a Madgwick filter provided real-time orientation estimates with a maximum total angle error of approximately 5°. However, the optical tracking remains necessary for high-quality validation and full kinematic measurements.

The improved prototype and the insights gained into the behavior of a four-legged, overconstrained

RUU parallel mechanism are relevant beyond the immediate application in biomechanical system identification. The mechanism can be enhanced and adapted for research in robotics, precision mechanics, and agricultural automation, provided that actuator selection and control are revised to meet the dynamic requirements of target applications.

4

Conclusion

This master's thesis improves a mechatronic proof-of-concept device for use as a wrist perturbator in the system identification of the human arm. Because changes in arm orientation affect the system identification of applied perturbations, a literature survey was carried out on orientation estimation with available sensors, filtering, calibration, and validation techniques. An inertial and magnetic measurement unit (IMMU) was selected for the application and incorporated into the design. The mechatronic device is based on a delta perturbator parallel mechanism with an overconstrained four-leg RUU configuration that provides three translational DoF. A new analytical inverse kinematics solution for this mechanism was derived and implemented in a Simulink model. Simulation results enabled refinement of the main dimensions to satisfy the defined requirements and the capabilities of the available servo actuators.

A physical prototype was built with a total mass of 0.30 kg and a movable mass of 0.25 kg to generate ungrounded perturbations in 3-DoF. Perturbation tests demonstrate that the prototype can produce a repeatable peak force of at least 5.0 N and can reproduce distinguishable force profiles in all translational directions, without generating significant parasitic forces and torques that would obstruct the human subject. Orientation estimation with a single IMMU using an implemented Madgwick filter yielded a maximum total angle error 5°. This level of accuracy is sufficient for real-time prototype control.

Overall, the improved prototype demonstrates relevant contributions to robotic manipulation and positioning for an overconstrained parallel mechanism through the derived inverse kinematics and validated perturbation capabilities and orientation estimation.

A

Kinematic trajectory planning

The inverse kinematics algorithm is implemented in Python to compute the target angles for the four actuator servos. Mechanism dimensions and mass parameters are defined, and the outer ring mass is used to convert the prescribed force profile into resulting velocities and displacements via numerical integration. The computed end-effector displacements are projected onto the desired perturbation direction, and the inverse kinematics are evaluated at those mapped end-effector positions to obtain the corresponding joint angles.

Joint angles are converted into pulse-width modulation (PWM) commands using calibration equations determined from prior servo calibration tests. To avoid overloading the Arduino's communication and processing capacity, PWM trajectories are down-sampled to 40 evenly spaced points per trajectory. Please take a look at the Python code for this algorithm below.

```python
import matplotlib.pyplot as plt
import numpy as np
from scipy.integrate import cumtrapz
from scipy.spatial.transform import Rotation as R

# Constants
Ro = 62.00
Ri = 35.72
L1 = 20.00
L2 = 20.00
toRad = np.pi / 180
toDeg = 1 / toRad
phi = np.array([0, 90, 180, 270]) * toRad  # Angles in radians

# Inverse kinematics function
def inverse(Ex, Ey, Ez):
    theta3 = np.arccos((Ex * np.sin(phi) + Ey * np.cos(phi)) / L2) * toDeg

    SDx = Ri + Ex * np.cos(phi) + Ey * np.sin(phi) - Ro
    SDy = Ex * np.sin(phi) + Ey * np.cos(phi)
    SDz = Ez * np.cos(2 * phi)

    theta2 = np.arccos((SDx**2 + SDy**2 + SDz**2 - L1**2 - L2**2) / (2 * L1 * L2 * np.sin(
        theta3 * toRad))) * toDeg

    F = L1 + L2 * np.cos(theta2 * toRad) * np.sin(theta3 * toRad)
    G = L2 * np.sin(theta2 * toRad) * np.sin(theta3 * toRad)

    theta1 = 2 * np.arctan((F + np.sqrt(F**2 + G**2 - SDz**2)) / (G + SDz)) * toDeg

    return theta3, theta2, theta1

# Generate force and compute displacement
time = np.linspace(0, 80, 1000)  # Time in ms
force = np.piecewise(time,
                     [time < 0, (time >= 0) & (time < 10), (time >= 10) & (time < 80)],
                     [0, 10, -10/7])

mass = 0.25  # kg
acceleration = force / mass
```

```
velocity = cumtrapz(acceleration, time / 1000, initial=0)
displacement = cumtrapz(velocity, time / 1000, initial=0)
displacement_avg = displacement*1000 - displacement[-1]*1000/2  # in mm, centered

quaternion = [0, 0, 0, 1]
rotation_quat = R.from_quat(quaternion)
base_axis = np.array([1, 0, 0])
direction_quat = rotation_quat.apply(base_axis)
unit_direction_quat = direction_quat / np.linalg.norm(direction_quat)

# Compute servo angles for each displacement sample
theta1_values = []
theta2_values = []
theta3_values = []
for s in displacement_avg:
    Ex, Ey, Ez = s * unit_direction_quat
    theta3, theta2, theta1 = inverse(Ex, Ey, Ez)
    theta1_values.append(theta1)
    theta2_values.append(theta2)
    theta3_values.append(theta3)

theta1_values = np.array(theta1_values)
theta2_values = np.array(theta2_values)
theta3_values = np.array(theta3_values)

# Convert angles to PWM signals
PWM1_values = np.round(10.27749229*(theta1_values[:,0] - 90.00) + 1473, 0).astype(int)
PWM2_values = np.round(-10.4381336*(theta1_values[:,1] - 90.00) + 2041, 0).astype(int)
PWM3_values = np.round(10.31991744*(theta1_values[:,2] - 90.00) + 1383, 0).astype(int)
PWM4_values = np.round(-10.47120419*(theta1_values[:,3] - 90.00) + 2068, 0).astype(int)

# Downsample to 40 points evenly distributed over the trajectory
def downsample_pwm_array(pwm_array, num_points=40):
    indices = np.linspace(0, len(pwm_array) - 1, num_points, dtype=int)
    return pwm_array[indices]

PWM1_ds = downsample_pwm_array(PWM1_values)
PWM2_ds = downsample_pwm_array(PWM2_values)
PWM3_ds = downsample_pwm_array(PWM3_values)
PWM4_ds = downsample_pwm_array(PWM4_values)
```

The down-sampled PWM arrays for each servo are uploaded to Arduino and output to the servos at a nominal frequency of 560 Hz, with a 2 ms delay between successive values. The perturbations are initiated by pushing a button.

B

Simulink model

The mechanism is modeled in MATLAB Simulink using the SimScape environment. A four-legged RUU parallel mechanism is constructed from the available SimScape building blocks. The inner and outer rings are simplified in SolidWorks and exported as a Simulink block. The inner ring serves as the fixed base, while the outer ring functions as the end-effector. The complete mechanism is highlighted in Figure B.1 with blue rectangles.

The mechanism contains four actuators, the selected servos are modeled as revolute joints in Simulink. Because the mechanism is overconstrained, one of the upper arms is modeled as a compliant element with high stiffness so that the kinematic configuration can be solved without significantly altering the system dynamics.

Model inputs consist of discrete angular positions over time. The prescribed force profile is converted to an end-effector trajectory via cumulative trapezoidal integration in MATLAB. The inverse kinematics are implemented in MATLAB to compute the discrete servo angles required to realize the desired trajectory given the mechanism's primary dimensions. Using these angles together with the prescribed force profile and perturbation direction, the simulation is executed. The model is solved with a fixed time step of 2.4×10^{-4} s using the backward Euler solver.

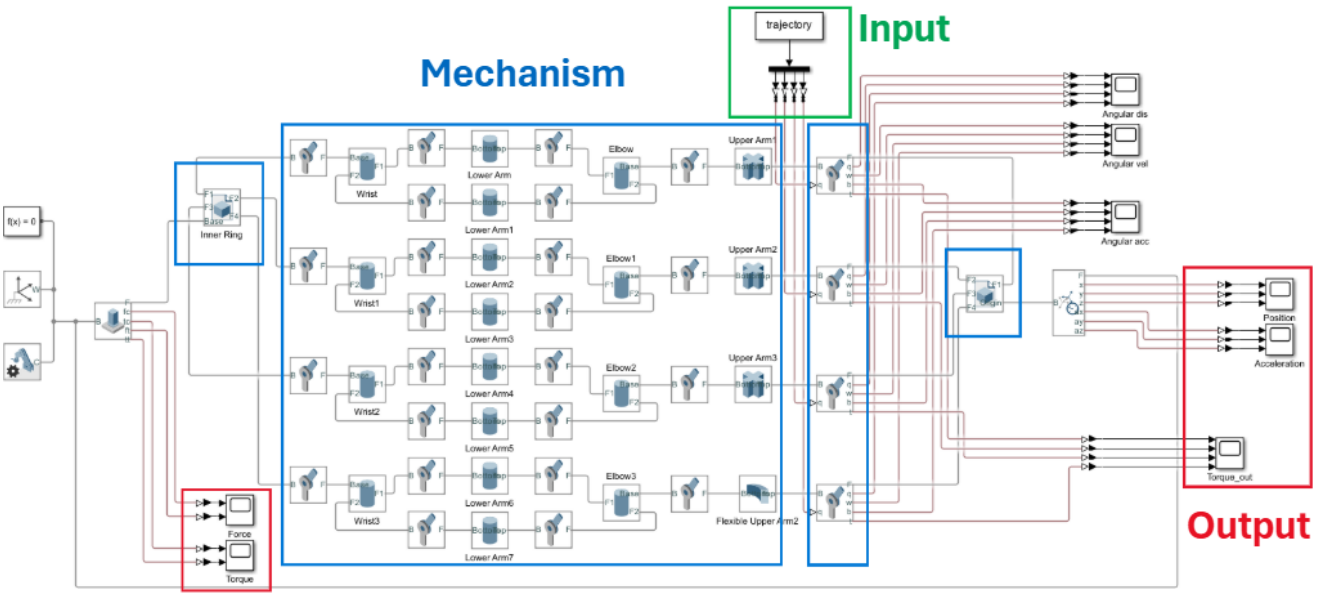


Figure B.1: Simulink block diagram. Blue rectangles denote the four-legged RUU parallel mechanism. The green highlight marks the computed servo-angle input signals for each leg over time. Red rectangles indicate simulation outputs used to evaluate mechanism performance and optimization of the main dimensions.

Several outputs are extracted from the model (highlighted in red in Figure B.1). The force generated at the fixed inner ring is compared to the prescribed input profile. The simulation reproduces the input with only minor overshoot and delay, which is expected because forces cannot change instantaneously

in a physical system. Off-axis reaction forces are quantified and minimized. Torques at the inner ring are simulated to verify that they remain below the required thresholds. Actuator angle trajectories are inspected to validate the inverse kinematics, and the torque sensors on the servo joints are used to quantify the torques required during perturbations. Multiple simulations are performed with varied dimensions to identify an optimal design that satisfies the available servo performance (torque and angular speed), minimizes parasitic forces and torques, and respects constraints on total mass and the end-effector's available range of motion.

C

Servo validation

Measured perturbation peak forces were 14–50 % lower than the theoretical simulations. In this ungrounded system, reaction forces are transmitted through the mechanism and are expected to be captured by the sensor; therefore, measured ungrounded forces depend directly on the actual end-effector accelerations. If commanded accelerations are not achieved in the intended direction, the result is reduced force magnitudes in the perturbation direction and increased force components in other directions. This shortfall in peak acceleration, therefore, limits overall perturbation performance and motivates analysis of the servo dynamics.

Servo performance was evaluated using the standard perturbation profile: a theoretical maximum force of 10 N applied for 10 ms, followed by an opposing force of −1.43 N for 70 ms. For the single-servo test, a pulley and mass were mounted to the servo, and a constant torque was applied while the servo received the same angle inputs used in the perturbation experiment. Servo rotation was recorded in slow motion at 240 Hz and analyzed with the optical tracking software Kinovea, see Figure C.1 for the setup. High-rate recording enabled accurate time-resolved tracking of the servo angle.

Figure C.1: Experimental setup for single-servo testing. The servo is rigidly mounted to the ground, and a mass-pulley assembly connected to the servo horn applies a constant torque.

Experimental results show that actual servo performance is substantially lower than the KST DS565X datasheet specifications. Angle-over-time traces recorded under multiple torque loads exhibit markedly reduced angular velocities compared to expectations, which contributes to the observed reduction in peak accelerations (and therefore forces). The Simulink simulation indicates a theoretically required torque of 0.12 N m for this force profile, corresponding to a maximum angular velocity of 6.0 rad/s. With that angular-velocity profile, the maximum achievable force is approximately 7.0 N. Figure C.2 com-

pares the input of the theoretical servo angle with the measured angle obtained through optical tracking for different applied constant torques.

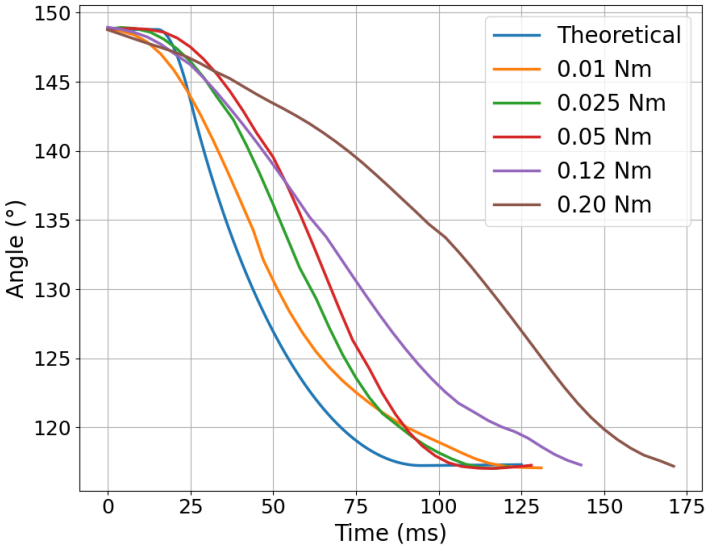


Figure C.2: Measured servo angular velocity deviates from the theoretical profile, indicating that the commanded end-effector accelerations are not achieved. This explains the reduced force peak observed in experiments.

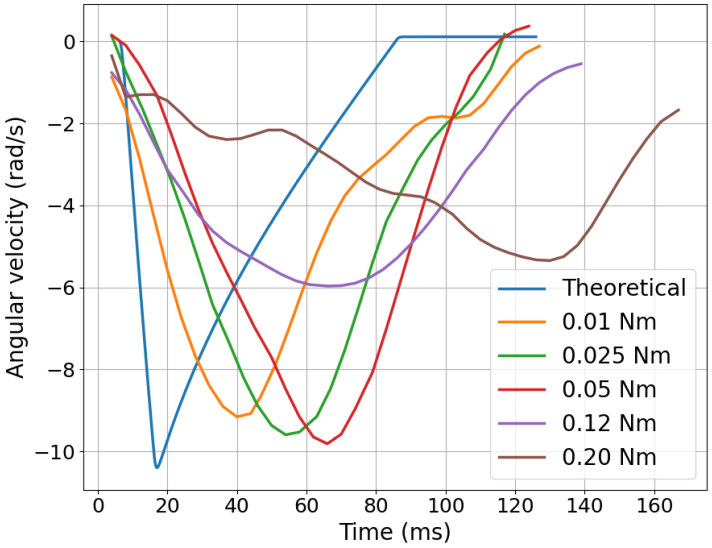


Figure C.3: The angular velocity of the servo is logically also not following the theoretical line. This indicates the accelerations of the end-effector are not reached and concludes why the force peak during experiments is not reached.

A higher force profile needs to be given for the servos to reach the right accelerations for the end-effector, and thereby the right forces. For a torque of 0.12 N m, which is simulated as needed for the perturbation a maximum angular velocity of 6 rad/s, this is 40 % lower than the maximum 10 rad/s needed. That is why a perturbation profile of 40 % higher is chosen.

D

Data communication HEX-H sensor

The prototype is mounted to an OnRobot HEX-H QC six-axis force/torque sensor using a 3D-printed adapter. The HEX-H QC communicates over Ethernet using the User Datagram Protocol (UDP) to a dedicated IP address and port. Data acquisition is performed at 500 Hz, the sensor's maximum sampling rate, which is well suited to the short-duration perturbations applied. Automatic zeroing is performed by the sensor software. No filtering is applied during acquisition, so that raw signals are recorded for offline processing. Logged measurements, forces (Fx, Fy, Fz) and torques (Mx, My, Mz), are saved to a CSV file for one full second. The entire acquisition routine is implemented in Python; the code is shown below.

```python
import socket, struct, time, csv
DEVICE_IP = '192.168.1.1'
DEVICE_PORT = 49152

def send_command(sock, cmd, data):
    header = 0x1234
    packet = struct.pack('>HHI', header, cmd, data)
    sock.sendto(packet, (DEVICE_IP, DEVICE_PORT))

sock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
sock.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
sock.bind(('', 0))
sock.settimeout(1.0)
send_command(sock, 0x0042, 255) #Zero: 255 no-zero: 0
send_command(sock, 0x0081, 0)   #Filtering: 0-6, No - 1.5 Hz
send_command(sock, 0x0082, 2)   #Sampling: ms
send_command(sock, 0x0002, 0)   #Start reading: 0
csv_file = open('hex_h.csv', 'w', newline='')
writer = csv.writer(csv_file)
writer.writerow(['time_ms', 'Fx_N', 'Fy_N', 'Fz_N', 'Tx_Nm', 'Ty_Nm', 'Tz_Nm'])
print("Logging voor 1 seconden gestart...")
start_time = time.perf_counter()

try:
    while True:
        if (time.perf_counter() - start_time) > 1:
            break
        packet, addr = sock.recvfrom(1024)
        if len(packet) < 36:
            continue
        data = struct.unpack('>3I6i', packet[:36])
        _, _, _, fx, fy, fz, tx, ty, tz = data
        Fx_N  = fx / 10000.0
        Fy_N  = fy / 10000.0
        Fz_N  = fz / 10000.0
        Tx_Nm = tx / 100000.0
        Ty_Nm = ty / 100000.0
        Tz_Nm = tz / 100000.0
        timestamp_ms = int((time.perf_counter() - start_time) * 1000)
        writer.writerow([timestamp_ms, Fx_N, Fy_N, Fz_N, Tx_Nm, Ty_Nm, Tz_Nm])

except Exception as e:
    print(f"Fout tijdens logging: {e}")

```

```
45 finally:
46     send_command(sock, 0x0000, 0)
47     csv_file.close()
48     sock.close()
49     print("Logging gestopt. Bestand opgeslagen.")
```

E

Orientation uncertainty

The orientation of a rigid body in three-dimensional space can be fully represented either by a rotation matrix or by a quaternion. Consider two orthonormal vectors that define two of the three body-fixed axes; the third axis can be obtained through the cross product to preserve orthogonality[1]. Formally, $||v_1|| = ||v_2|| = 1$, and $v_1 \cdot v_2 = 0$. For each reference vector, a cone acceptance with angle θ can be defined. This cone specifies the region of acceptable deviation[2]. A vector v_q lies within the cone if $\cos^{-1}(v_q \cdot v) \leq \theta$, which is equivalently expressed as $v_q \cdot v \geq \cos\theta$.

In practice, orientation estimates are usually provided in quaternion form, rather than as vectors. Using the rotation matrix $R(q)$ associated with the quaternion q, the corresponding body-frame vectors can be extracted as:
$v_{q,1} = R(q) \cdot \hat{x}$
$v_{q,2} = R(q) \cdot \hat{y}$.

These computed vectors can then be directly compared with the reference vectors to evaluate alignment within the defined cones of acceptance.

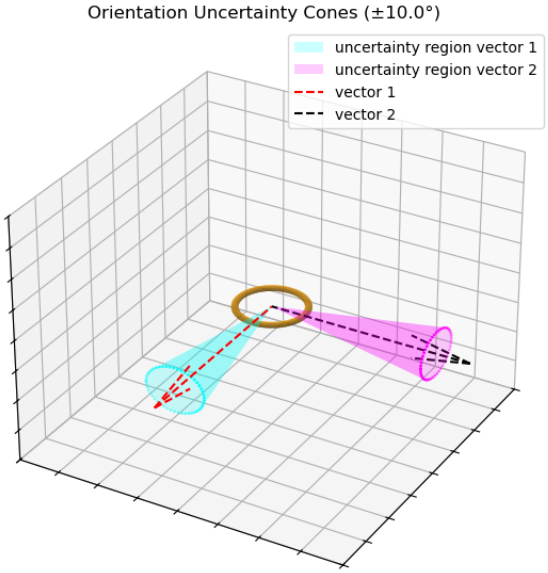


Figure E.1: Uncertainty cones of 10° in all directions for the two given orthonormal direction vectors along the principal X-, and Y-axis representing the orientation of the prototype animated with the orange ring.

[1] Vutukuri, S., & Padhi, R. (2025). Robust adaptive predefined time prescribed performance attitude control for spacecraft. *Control Engineering Practice*, *157*, 106271. https://doi.org/10.1016/j.conengprac.2025.106271

[2] Hashmall, J. (2006). Accommodating Sensor Uncertainty in the Cones Method: Polycones and Fuzzycones. *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*. https://doi.org/10.2514/6.2006-6162

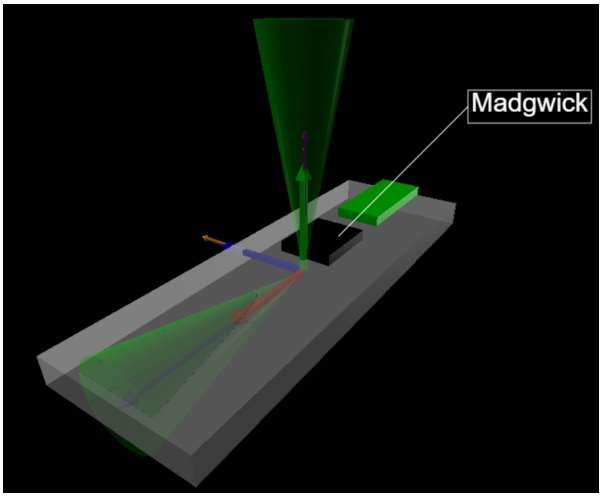


Figure E.2: Uncertainty cones of 10° in all directions for the two given orthonormal direction vectors along the principal X-, and Y-axis representing the orientation of the IMMU sensor.

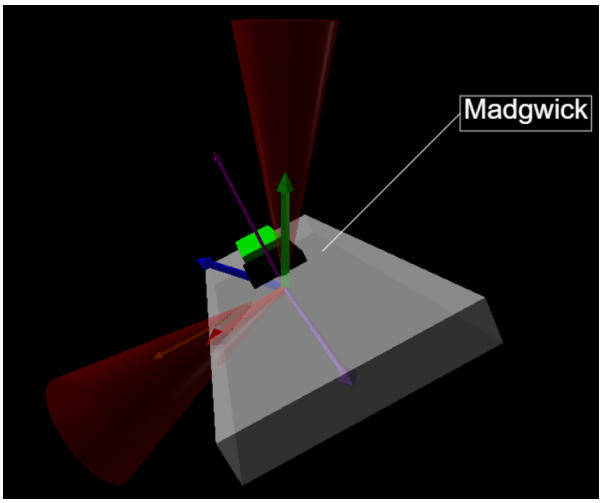


Figure E.3: Uncertainty cones of 10° in all directions for the two given orthonormal direction vectors along the principal X-, and Y-axis representing the orientation of the IMMU sensor.

The cones of acceptance can be used by the controller to evaluate whether the estimated orientation is within the desired range. If yes, the perturbation can get started by actuating the servos and applying the forces. By actuating the servos, the estimated orientation also needs to be stored to get more valuable data for the system identification of the human arm. These algorithm steps bring the prototype further towards the real-world application as a medical measurement device.