A Novel Triad Twisted String Actuator for Controlling a Two Degrees of Freedom Joint: Design and Experimental Validation

Damian Crosby, Joaquin Carrasco, *Member, IEEE,* William Heath, *Member, IEEE,* and Andrew Weightman

***Abstract*—Actuated universal joints, or equivalent joint systems, are found in a number of robotic applications, in particular mobile snake robots, snake-arm robots and robotic tails. Such joints use a variety of actuation methods, including direct drive motors, linear screw drives, cable based systems, and hydraulics/pneumatics. In this paper the authors present a novel design for a mechanism that utilises the TSA! in an antagonistic triad to actuate the universal joint, using orientation sensors and load cells to create a robust cascading closed loop control system. This design realises a light, compact, high-performance actuation system that avoids the extra mass and hardware complexity that alternative actuation methods present, with the additional challenge of nonlinearity. The authors were able to develop a closed loop control system that could track a universal joint orientation setpoint to within $\pm 1.8°$ within a $\pm 11°$ range for a single axis and $\pm 6°$ range for dual axis.**

***Index Terms*—???**

I. Introduction

AUJ! (**AUJ!**) mechanisms are found in a wide range of robotic applications, such as confined space inspection using snake-arm robots **buckingham2012nuclear**, **mohammad2021efficient**, **dong2014design**, **walker2013continuous**, highly manoeuvrable mobile snake robots **luo2018**, and biomimetic robot tails for stability **rone2018**. Mobile snake robots must usually incorporate electric actuators inline with their joints. This results in an **AUJ!** having to shift the mass of the follower segments and all the actuators inside the follower segments, which results in high torque requirements. Snake-arm robots and robotic tails can reduce the mass and size of the **AUJ!** by moving their actuators away from the **AUJ!**s and use cables to transfer the force to the joints, or use hydraulic or pneumatic actuators which tend to be lighter than equivalent electric motors. This comes at the expense of increased mass and bulk at the base of the arm or tail.

First developed by **wurtz2010 wurtz2010** in **wurtz2010**, the **TSA!** (**TSA!**) uses two or more strings between two fixtures as a linear actuator. When one fixture is rotated (typically by an electric motor), the looped string twists into a helix, decreasing the distance between them. **TSA!** actuators have been used for a hand orthosis **muehlbauer2021twisted**, elbow joint **park2020control** and foldable robot arm **suthar2021design** among other functions.

The primary advantage of **TSA!** over similar linear actuators such as a leadscrew is the reduction the **TSA!** provides is not proportional (or even slightly inversely proportional) to the mass of the actuator. By decreasing the string cross-section radius the reduction increases given a constant unwound length and motor angle. While the reduction in a leadscrew can be increased by decreasing the lead on the thread, which also has no increase in mass, this has a limited range and can quickly run up against manufacturing tolerances or material strength requirements. In order to achieve greater or more robust reductions, the screw radius has to be increased, or the driving motor has to have a larger reduction before driving the screw, both of which usually result in more material (typically steel) and therefore more mass and bulk. This would reduce the utility of several applications of **AUJ!** mechanisms, such as each segment being required to support all its follower segments, or being able to operate in confined spaces.

However, **TSA!** does have some disadvantages, the most significant of which is a non-linear reduction equation, which is also dependent on the motor angle θ_s (and therefore actuator position). The reduction decreases in a non-linear fashion as the angle increases, with the derivative decreasing as the angle increases. There is also the compliance of the strings to consider, depending on the thickness and material chosen, which becomes a significant factor under high forces. Both of these issues can be addressed with accurate modelling **nedelchev2020** and/or a robust control strategy, as demonstrated in **wurtz2010**. What is more of an issue is the unidirectional force of the **TSA!**, which can only impart force in tension. This means that for an **AUJ!**, which is a two degree of freedom joint, a minimum of three **TSA!** are required for unless spring return mechanisms are used, which would impart additional force on the **TSA!** and therefore reduce performance. However, the potential high force to mass ratio of the **TSA!** due to the non-proportional reduction may adequately compensate for the additional actuator requirement.

The focus of this research is to investigate if the **TSA!** is a suitable candidate for control of an **AUJ!** considering both the benefits and drawbacks. To this end, the first objective is to perform a simulation and construct a physical experimental prototype to validate the proposed

0000–0000/00$00.00 © 20XX IEEE

control system. Then, the second objective is to compare the results from both the simulation and experimental prototype with other inline actuation methods for an **AUJ!** to analyse and discuss the potential advantages and disadvantages of **TSA!** actuation against alternatives.

The use of **TSA!** as an actuator for an **AUJ!** is an understudied area of research. **bombara2021twisted** have proposed a similar design using a flexible core with continous curvature as opposed to a rigid universal joint, however currently this research has not demonstrated control of both axes of motion with multiple **TSA!**. For the first time the authors demonstrate a robust closed loop control of an **AUJ!** in both axes of motion using three **TSA!** in an "antagonistic triad" confugration. The result is a light, compact **AUJ!** design that has the potential to significantly improve upon exiting inline actuation options.

This publication will first give an outline of the **TSA!** based on the existing literature, and the concept of an antagonistic triad. Then a detailed explanation of the control system is given, followed by results from the simulation and experimental system. Finally, a theoretical anaylsis of the **TSA! AUJ!** compared to a similar **AUJ!** using leadscrews is conducted, followed by a discussion and conclusion.

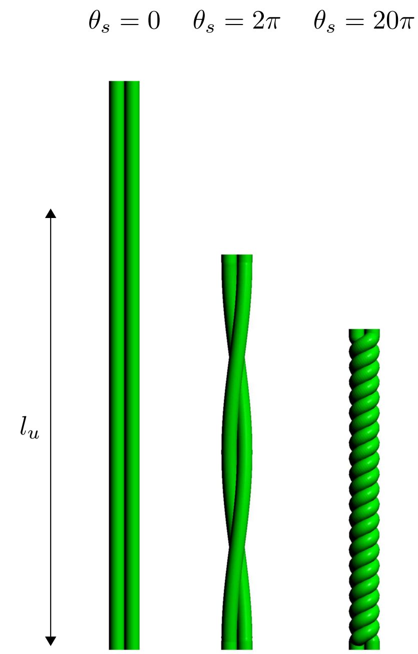


Fig. 1: The value of θ_s increases the number of twists in a string bundle with a string length l_u.

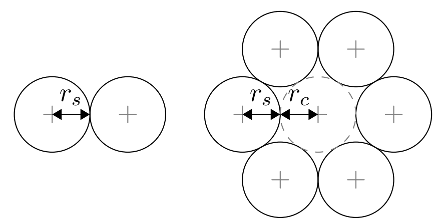


Fig. 2: The location of r_s and optionally r_c in a string bundle.

A. A ***TSA!***

First developed by **wurtz2010 wurtz2010** in 2010, **TSA!** uses two or more strings between two fixtures as a 1 **DOF!** (**DOF!**) linear actuator. When one fixture is rotated (typically by an electric motor), the strings twist into a helix, decreasing the distance between the fixtures, as shown in figure **??**. Given the unwound length l_u, and the cross-section radius of the string r_s (or $r_s + r_c$ when there are more than two strings) as shown in figure **??**, the actuator length is given by

$$l_s(\theta_s) = \sqrt{l_u^2 - \theta_s^2 r_s^2} \tag{1}$$

where θ_s is the motor angle, as shown in figure **??**.

This equation assumes an infinite string stiffness, so is only reasonably accurate under low tension. Although theoretically the stroke of the **TSA!** can be the entire domain of $[0, l_u]$, in reality the thickness of the string prevents a geometric helix from forming once the helix pitch $q < 4r_s$ (or $q < 2nr_s$ for n strings). This limits the lower bound of the stroke as follows,

$$l_{min} = \frac{l_u}{\sqrt{\frac{\pi^2}{2} + 1}} \approx 0.46\ l_u \tag{2}$$

or approximately 46% of l_u for a two string **TSA!**.

B. Antagonistic Triad

As mentioned in the introduction, because the **TSA!** provides only tensile force, a minimum of three actuators are required for a 2 **DOF!** actuation system. These can be arranged in a triangular configuration to create an "antagonistic triad", akin to the antagonistic pairs of muscles found in animals. Where a revolute joint would be found between the connecting ends of the actuator, a universal joint is found instead. The geometric structure of the system as shown in figure **??** can be described with two equilateral triangles of inradius r on two planes separated in the z axis. The centroids are then connected via a universal joint from each plane normal to an intersecting point, let the vector $\boldsymbol{\theta} = \begin{bmatrix}\theta_1 & \theta_2\end{bmatrix}$ denote the rotation of the second plane relative to the first, in the x and y axes around the intersecting point, and let l_1 and l_2 denote the normal distance from the intersection to the first and second plane centroids respectively. When $\boldsymbol{\theta} = \begin{bmatrix}0 & 0\end{bmatrix}$ the triangles are parallel to each other. The distance between the vertex pairs of each triangle is then denoted as $\begin{bmatrix}\lambda_1 & \lambda_2 & \lambda_3\end{bmatrix}$ for the "top", "left" and "right" vertices of the triangles. When $\boldsymbol{\theta}$ is changed, this will change λ_1, λ_2 and λ_3 respectively.

To calculate the lengths of the strings for a given $\boldsymbol{\theta}$ of the universal joint, we define a vector function $\Lambda(\boldsymbol{\theta}) = \begin{bmatrix}\lambda_1(\boldsymbol{\theta}) & \lambda_2(\boldsymbol{\theta}) & \lambda_3(\boldsymbol{\theta})\end{bmatrix}$ as follows.

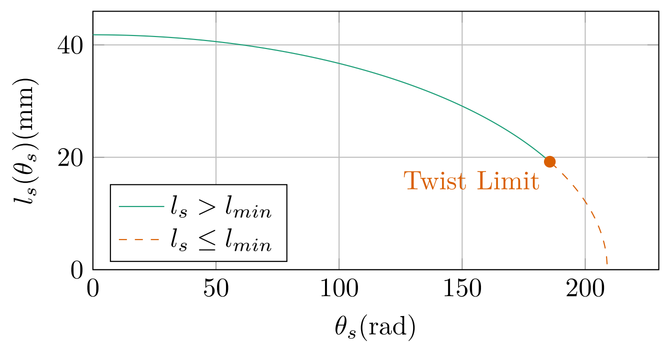


Fig. 3: **TSA!** string length against motor angle with coefficients from table **??**.

$$
\begin{aligned}
\lambda_1(\boldsymbol{\theta}) &= \sqrt{\begin{aligned}&(l_1 + l_2\cos\theta_1\cos\theta_2 + r\cos\theta_1\sin\theta_2)^2\\ &+ (r - r\cos\theta_2 + l_2\sin\theta_2)^2\\ &+ (l_2\cos\theta_2\sin\theta_1 + r\sin\theta_1\sin\theta_2)^2\end{aligned}}\\
\lambda_2(\boldsymbol{\theta}) &= \sqrt{(a-b+c)^2 + (l_1-d)^2 + e^2}\\
\lambda_3(\boldsymbol{\theta}) &= \sqrt{(a+b-c)^2 + (l_1+d)^2 + e^2}\\
&\text{where:}\\
&\quad a = -\tfrac{\sqrt{3}r(\cos\theta_1 - 1)}{2}\\
&\quad b = l_2\cos\theta_2\sin\theta_1\\
&\quad c = \tfrac{r\sin\theta_1\sin\theta_2}{2}\\
&\quad d = \tfrac{\sqrt{3}r\sin\theta_1}{2} + l_2\cos\theta_1\cos\theta_2 - \tfrac{r\cos\theta_1\sin\theta_2}{2}\\
&\quad e = \tfrac{r\cos\theta_2}{2} - \tfrac{r}{2} + l_2\sin\theta_2
\end{aligned}
\tag{3}
$$

The output of this is shown in figure **??** for a domain of $\left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$.

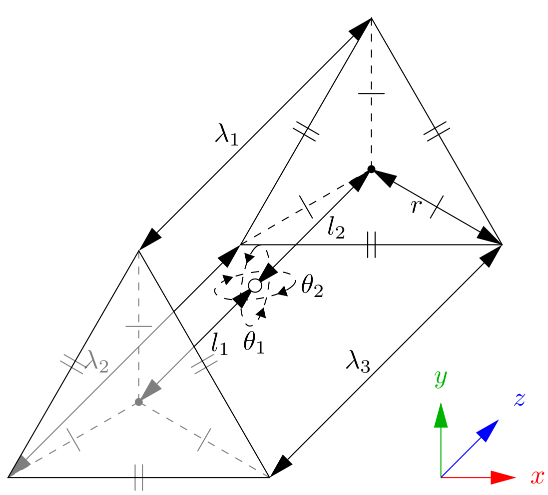


Fig. 4: Kinematic diagram of the antagonistic triad, where the universal joint rotation is defined by $\theta_{1,2}$ on the x and y axes respectively, and the actuator lengths are defined by $\lambda_{1,2,3}$ for the "Top", "Left" and "Right" actuators. r and $l_{1,2}$ define the anchor points of the actuators.

II. Control System

The control system is a four layer cascade design, joining an inverse dynamic control system **spong2020robot**, to the triad force controller in **dessen1986**, to a proprtional controller for each **TSA!**. It uses feedback signals of the joint position from the accelerometers and **TSA!** force from the load cells. A second order setpoint trajectory $\boldsymbol{q}$ is used as the input, which can either be pre-defined or generated dynamically from user input. Feedback is provided by the **AUJ!** angular position $\boldsymbol{\theta}$ as shown in figure **??**, angular velocity $\dot{\boldsymbol{\theta}}$, and **TSA!** tension force $\hat{\boldsymbol{f}}$ as shown in figure **??**. Figure **??** shows a complete block diagram of the control system.

1) C_1 **AUJ!** Position PID Controller
2) C_2 Inverse Dynamics
3) C_3 **TSA!** Force Optimisation Algorithm
4) C_4 **TSA!** Force P Controller

Functions $C_{1\ldots4}$ are then combined into a cascade function $C_4\left(C_3\left(C_2\left(C_1\left(\ldots\right),\ldots\right),\ldots\right),\ldots\right)$.

A. *AUJ! Position PID Controller with Acceleration Feedforward*

Firstly, a PID controller is used to generate a control signal $\boldsymbol{u}$ with the input $\boldsymbol{q}$ as the setpoint, and the **AUJ!** angular position $\boldsymbol{\theta}$ and velocity $\dot{\boldsymbol{\theta}}$ as feedback, plus the addition of a feedforward term for the input acceleration $\ddot{\boldsymbol{q}}$, i.e.

$$\boldsymbol{u} = \boldsymbol{k}_p(\boldsymbol{q} - \boldsymbol{\theta}) + \boldsymbol{k}_i\left(\int_0^t \boldsymbol{q} - \boldsymbol{\theta}\right)dt + \boldsymbol{k}_d\left(\dot{\boldsymbol{q}} - \dot{\boldsymbol{\theta}}\right) + \ddot{\boldsymbol{q}}. \tag{4}$$

In the discrete implementation used for fixed step simulation and experimental system control, the integral term is replaced by the trapezoidal rule.

$$K_I\int_0^t \boldsymbol{\epsilon} \approx \sum_{i=0}^{N}\frac{\boldsymbol{\epsilon}(t_i) + \boldsymbol{\epsilon}(t_{i-1})}{2}\Delta t \tag{5}$$

B. *Inverse Dynamics*

The control signal $\boldsymbol{u}$ from the PID controller is then converted to the desired **AUJ!** torque $\boldsymbol{\tau}$ as follows

Firstly the affine transformation matrix T for a coordinate frame between the **AUJ!** pivot and the centre of mass of the follower segment can be defined. The order of R_x and R_y can be reversed, but this requires other terms to be reversed as well.

$$T(\boldsymbol{\theta}) = R_x(\theta_1)\, R_y(\theta_2)\, P_z(l_2) \tag{6}$$

Then the linear velocity jacobian J_v is simply the jacobian of the translation vector of T.

$$J_v(\boldsymbol{\theta}) = \begin{bmatrix} \frac{\partial t_{14}}{\partial\theta_1} & \frac{\partial t_{24}}{\partial\theta_1} & \frac{\partial t_{34}}{\partial\theta_1} \\ \frac{\partial t_{14}}{\partial\theta_2} & \frac{\partial t_{24}}{\partial\theta_2} & \frac{\partial t_{34}}{\partial\theta_2} \end{bmatrix} \tag{7}$$

The angular velocity jacobian J_ω is calculated using the first joint angle relative to the base frame, and the second

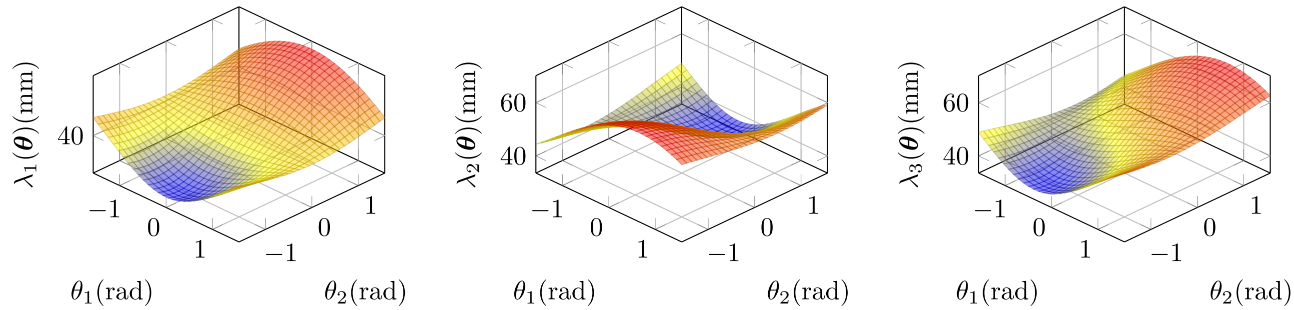


Fig. 5: Surface plots of each element of the vector function $\Lambda(\boldsymbol{\theta})$, assuming coefficient values from table **??**.

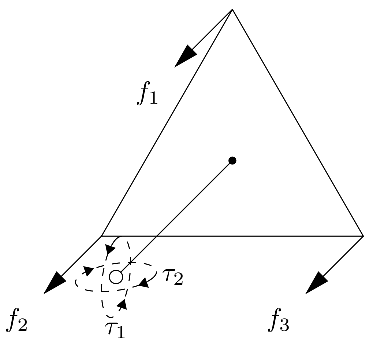


Fig. 6: Dynamics of an antagonistic triad based on figure **??**, where the forces on each anchor point are $f_{1,2,3}$ and the universal joint torque is $\tau_{1,2}$.

in the frame of the first. If R_x is the first rotation in T then the first column of the jacobian is $\begin{bmatrix}100\end{bmatrix}^\intercal$ (to represent the pitch angle) and the second column is $R_x\begin{bmatrix}010\end{bmatrix}^\intercal$.

$$J_\omega(\boldsymbol{\theta}) = \begin{bmatrix} 1 & r_{x_{11}} \\ 0 & r_{x_{21}} \\ 0 & r_{x_{31}} \end{bmatrix} \tag{8}$$

Then the mass matrix D can be created from the jacobian, the mass for the follower segment m and its inertia tensor $I \in \mathbb{R}^{3\times3}$, and R_x and R_y to express the inertia in the correct frame.

$$D(\boldsymbol{\theta}) = mJ_v^\intercal J_v + J_\omega^\intercal (R_x R_y) I (R_x R_y)^\intercal J_\omega^\intercal \tag{9}$$

The centrifugal/coriolis matrix C is created from the christoffel symbols of D, along with the **AUJ!** velocity vector $\dot{\boldsymbol{\theta}}$.

$$C\left(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}\right)_{k,j} = \sum_{i=1}^{N} \frac{1}{2}\frac{\partial d_{kj}}{\partial \theta_i} + \frac{\partial d_{ki}}{\partial \theta_j} - \frac{\partial d_{ij}}{\partial \theta_k}\dot{\theta}_i \tag{10}$$

Then the gravity term G. As the gravity vector direction is the same as the z axis as in figure **??**, the height is equal to $-l_2\cos\theta_1\cos\theta_2$, therefore the potential energy is $mg(-l_2\cos\theta_1\cos\theta_2)$. The jacobian of this then becomes the gravity term.

$$G(\boldsymbol{\theta}) = \begin{bmatrix} \frac{\partial mg - l_2\cos\theta_1\cos\theta_2}{\partial\theta_1} \\ \frac{\partial mg - l_2\cos\theta_1\cos\theta_2}{\partial\theta_2} \end{bmatrix} \tag{11}$$

D, C and G are then combined to form the dynamics equation C_2, along with the **AUJ!** position and velocity vectors. C_1 is used as the acceleration term. This results in a setpoint joint torque which can be used in the optimisation algorithm.

$$\boldsymbol{\tau} = D(\boldsymbol{\theta})\boldsymbol{u} + C\left(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}\right)\dot{\boldsymbol{\theta}} + G(\boldsymbol{\theta}). \tag{12}$$

C. *TSA! Force Optimisation Algorithm*

This uses a modified algorithm from **dessen1986** to select an optimal force vector from the desired joint torque. A force matrix F is created from the torque input $\boldsymbol{\tau}$, jacobian J_Λ from the vector function Λ as defined in equation **??**, and minimum force constant f_{min}. f_{ii} is equal to f_{min}, while the other elements in the column are based on a calculation using $J_{\Lambda_{-i,*}}$ where $-i$ is a row removed from the matrix.

$$J_\Lambda = \begin{bmatrix} \frac{\partial\lambda_1}{\partial\theta_1} & \frac{\partial\lambda_2}{\partial\theta_1} & \frac{\partial\lambda_3}{\partial\theta_1} \\ \frac{\partial\lambda_1}{\partial\theta_2} & \frac{\partial\lambda_2}{\partial\theta_2} & \frac{\partial\lambda_3}{\partial\theta_2} \end{bmatrix}$$
$$\boldsymbol{F}(\boldsymbol{\tau}, \boldsymbol{\theta}) = \begin{cases} f_{i,i} = f_{min} \\ f_{-i,i} = -J_{\Lambda_{-i,*}}^{-\intercal}\left(J_{\Lambda_{i,*}}^\intercal f_{min} + \boldsymbol{\tau}\right) \end{cases} \tag{13}$$
$$= \begin{bmatrix} f_{min} & f_{12} & f_{13} \\ f_{21} & f_{min} & f_{23} \\ f_{31} & f_{32} & f_{min} \end{bmatrix}$$

The following algorithm then selects one column of $\boldsymbol{F}$ to be the output force vector $\boldsymbol{f}$.

1: $\boldsymbol{s} \leftarrow \begin{bmatrix}\mathbf{T} & \mathbf{T} & \mathbf{T}\end{bmatrix}$
2: **if** $f_{23} > f_{min}$ **then** $s_2 \leftarrow$ **F else** $s_3 \leftarrow$ **F end if**
3: **if** $f_{31} > f_{min}$ **then** $s_3 \leftarrow$ **F else** $s_1 \leftarrow$ **F end if**
4: **if** $f_{12} \geq f_{min}$ **then** $s_1 \leftarrow$ **F else** $s_2 \leftarrow$ **F end if**
5: **for** $i = 1$ to 3 **do**
6: **if** $s_i \rightarrow$ **T then** $\boldsymbol{f} \leftarrow \boldsymbol{f}_{*,i}$ **end if**
7: **end for**

D. *TSA! Force Proportional Controller*

The selected forces are then used as an input to a Proportional controller using the measured load cell forces

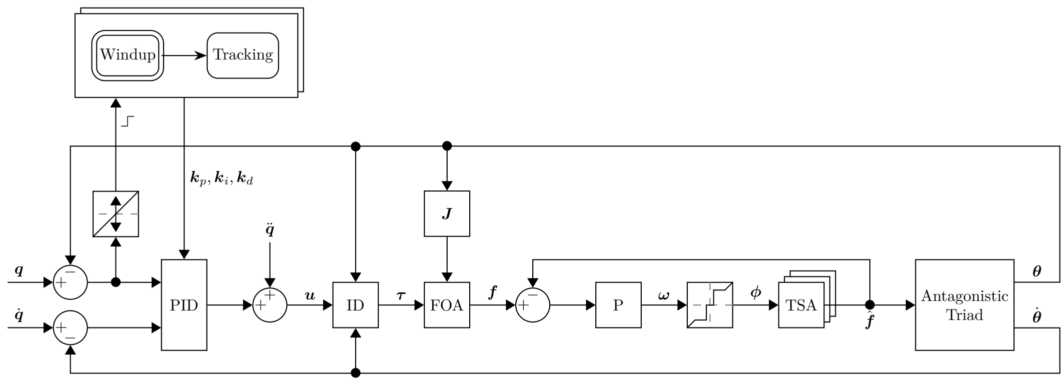


Fig. 7: Block diagram of the complete experimental control system, excluding the hardware velocity controllers for the motors.

$\hat{\boldsymbol{f}}$ as feedback. The output from this can then be used to control the top, left and right **TSA!** motors.

1) Simulation Current Control: In the simulation, each **TSA!** was modelled as a state space system from **wurtz2010** which takes motor current u as an input and outputs y as the **TSA!** tension force. **wurtz2010** defines it as such, where J is the motor inertia, C is the motor coulomb friction (modified from viscous friction as the 1724TSR only has dry friction), K_t is the motor torque constant, and K_L is the load stiffness. As the original definition is for a fixed load l_u distance from the motor a modified model is required which takes into account the varying length between the motor and load defined by $\Lambda(\boldsymbol{\theta})$. A saturation function is used to prevent incorrect compression forces when the string is slack. All of the motor coefficients were taken from the Faulhaber 1724TSR datasheet **1724tsr** as this is the motor used in the experimental system. An estimated value is used for the load stiffness, this was chosen to be a high number as the model is expected to be very stiff.

$$\begin{aligned} h(\theta_s) &= \frac{\theta r_s^2}{\sqrt{l_u^2 - \theta_s^2 r_s^2}} \\ k(\theta_s, \boldsymbol{\theta}) &= \lambda_n(\boldsymbol{\theta}) - \sqrt{l_u^2 - \theta_s^2 r_s^2} \\ \dot{\boldsymbol{x}} &= \begin{bmatrix} x_2 \\ -\frac{K_L}{J} h(x_1) k(x_1, \boldsymbol{\theta}) - \frac{C}{J}\operatorname{sgn}(x_2) \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{K_t}{J} \end{bmatrix} u \\ y &= K_L \operatorname{sat}_0^\infty k(x_1, \boldsymbol{\theta}) \end{aligned} \tag{14}$$

The state space model was then adapted to include constraints on motor velocity and acceleration set by the motor controller in order to keep the motor within design limits, by replacing $\dot{\boldsymbol{x}}$ with $\dot{\boldsymbol{x}}'$ which contains saturation functions for maximum motor velocity v_s and acceleration α_s.

$$\dot{\boldsymbol{x}}' = \begin{bmatrix} \operatorname{sat}_{\omega_s} \dot{x}_1 \\ \operatorname{sat}_{\alpha_s} \dot{x}_2 \end{bmatrix} \tag{15}$$

2) Experimental Velocity Control with Deadband Compensation: In the experimental system, current control did not result in a stable output, so instead the P controller output would be the motor velocity as $\boldsymbol{\omega}$ (each motor controller has a hardware velocity PI controller). Due to a motor deadband within $\pm 10\,\mathrm{min}^{-1}$, an adjustable deadband compensator is used,

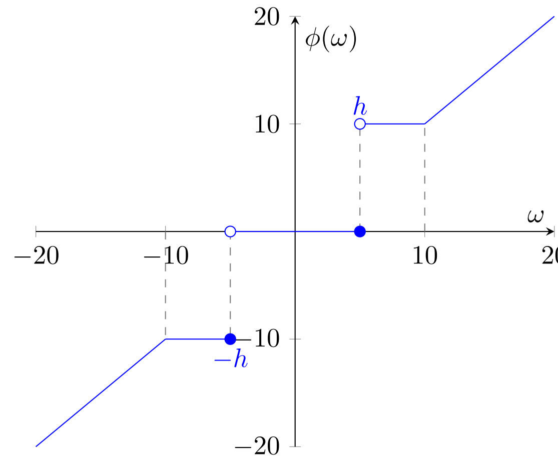


Fig. 8: Graph plotting the function of equation **??**.

$$\phi_i(\omega_i) = \begin{cases} 10 & 10 \leq \omega_i < h \\ -10 & -h < \omega_i \leq -10 \\ 0 & h \leq \omega_i \leq -h \\ \omega_i & \text{otherwise} \end{cases} \tag{16}$$

where ϕ_i is the compensator for the controller i. An adjustment value $h \in [0, 10]$ changes the threshold at which the compensator switches on and off, allowing for a small deadband to remain. A graph of this function is shown in figure **??**.

The result from the **TSA!** is then a compressive force acting between each of the three **TSA!** and its corresponding endpoint on the Antagonistic Triad, imparting a torque on the axes of the universal joint.

*3) **DCC! (DCC!)**:* This mode takes a preset velocity $\dot{\theta}_{set} \in [0, 4220]$ and uses a hardware PI controller with velocity feedback $\dot{\theta}_{act}$ to generate a control voltage. This voltage $\boldsymbol{V}$ is then multiplied by the signum of the result from the cascade function in order to ensure the motor spins in the right direction, and then passed to a current limiter with the current error (the result with the actual current I_{act} subtracted) as the limit, before being sent to the motor. This ensures the motor stops spinning when the target current is reached.

$$\boldsymbol{\epsilon_c} = \dot{\theta}_{set} - \boldsymbol{\dot{\theta}_{act}}$$
$$\boldsymbol{V} = \left(K_{P_c}\boldsymbol{\epsilon_c} + K_{D_c} \int_0^t \boldsymbol{\epsilon_c} \right) \operatorname{sgn}\left(C_4\left(\ldots\right)\right)\omega\left(C_4\left(\ldots\right) - \boldsymbol{I_{act}}\right) \tag{17}$$

Where $\omega\left(\ldots\right) \in [0, 1]$ is an unknown hardware limiting function that controls the motor speed depending on the current error.

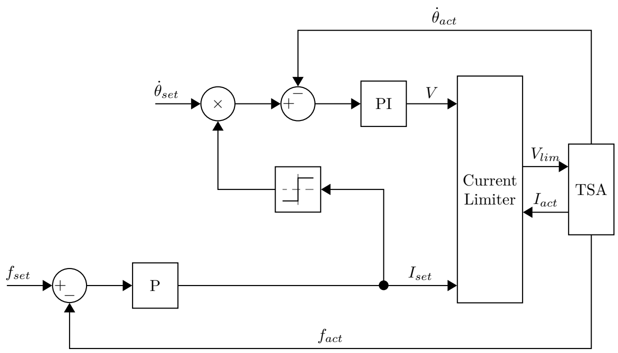


Fig. 9: **DCC!**

4) Proportional Current Controller: This strategy is a more direct method of current control, using a software P controller to directly set the voltage of the motor, using the MCDC 3002 as simply a passive amplifier. In this case the current error is passed directly to a P controller which has its output limited to prevent damage to the motors.

$$\boldsymbol{V} = K_{P_c} C_4\left(\ldots\right) - \boldsymbol{I_{act}} \tag{18}$$

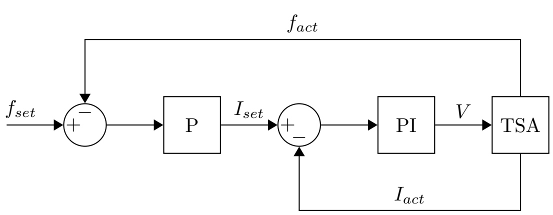


Fig. 10: Proportional Current Controller

5) Velocity Control: This strategy simply uses the result from the cascade function as a velocity setpoint using the hardware velocity PI controller as in the **DCC!**.

$$\boldsymbol{V} = K_{P_c}\boldsymbol{\epsilon_c} + K_{D_c} \int_0^t \boldsymbol{\epsilon_c} \tag{19}$$

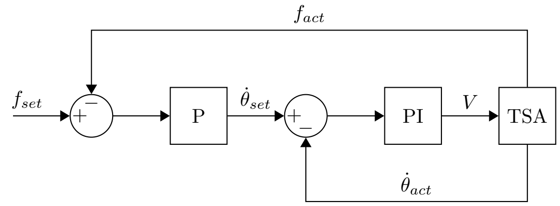


Fig. 11: Velocity Controller

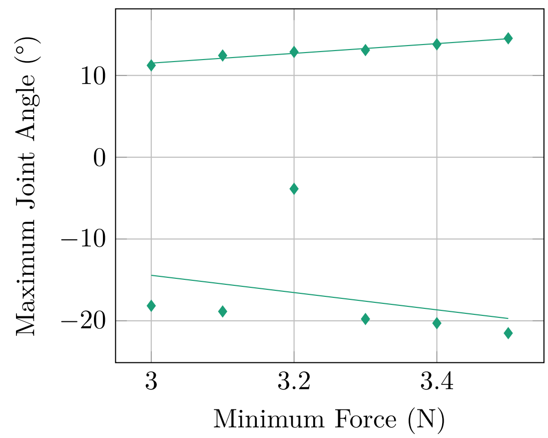


Fig. 12

III. **AUJ!** Performance Limits

A. Angular Range

B. Angular Velocity

C. Follower Payload

IV. Simulation & Experimental Results

To design and refine the parameters of the control system, a Simscape Multibody™model of the antagonistic triad and control system was created in MATLAB®/Simulink™. This allowed for model design coeffecients $l1, 2$ and controller gains k_p, k_i, k_d, k_{p_s} to be modified in order to have the most stable control within design limits.

A. Experimental Setup

For the experimental validation, a physical prototype of the mechanism was constructed with coefficients from ta-

TABLE I: Model coefficients.

Coefficient	Value	Coefficient	Value
l_1	41.8 mm	J	1×10^{-6} kg m^{-2}
l_2	0 mm	K_L	1 000 N m^{-1}
r	13 mm	f_{min}	3 N
l_u	41.8 mm	ω_s	441.9 rad s^{-1}
r_s	200 µm	I_s	0.19 A
m	72.619 13 g	K_t	0.026 3 N m A^{-1}
C	0.131 5 N mm	τ_s	4.5 mN m
α_s	1×10^5 rad s^{-2}		

Coefficient	Value
I	$\begin{bmatrix} 3 \times 10^{-5} & 0 & 0 \\ 0 & 3.2 \times 10^{-5} & 0 \\ 0 & 0 & 1.4 \times 10^{-5} \end{bmatrix}$ kg m^{-2}

TABLE II: PID gains in the simulation and experiment.

Gain	Value	
	Simulation	Experiment*
k_p	800	3×10^4
k_i	3000	350
k_d	50	50
k_{p_s}	19	100

* Tracking mode, see section **??**.

ble **??** as design parameters. This was mounted vertically, in order for the **IMU!** (**IMU!**) to measure the orientation of the universal joint. The **TSA!** mechanisms consist of a compact high torque motor (Faulhaber 1724T024SR) attached to the base segment, with a string clamp attached to the motor shaft. On the follower segment, a load cell is mounted on top of a universal joint to ensure a purely axial load, with a capstan bolt attached to the load cell. The string itself is attached to the clamp at both ends using two grub screws for extra security and easy adjustment, and looped through the hole in the capstan bolt. Figures **??** and **??** detail the construction of the experiment with all the consituent parts.

The motors were controlled by Fauhaber MCDC3002 motor controllers, which could interface with a National Instruments MyRIO via the USB port, using a USB to serial converter. The load cells were Futek LCM100 minature load cells, selected for their small size. The signals from these were amplified using Flyde FE-359-TA instrumentation amplifiers and decoded using an external AD7606 ADC before being fed into the MyRIO using SPI. The orientation of the **AUJ!** was measured using a Bosch Sensortec BNO080 **IMU!** using the accelerometer data.

B. Windup & Tracking States

When the mechanism is started with the **TSA!** in a completely unwound state, before it can begin tracking a motion trajectory, the **TSA!** strings must "wind up" to closely match the intial state of $\boldsymbol{f}$. During this phase, the outer PID gains $\boldsymbol{k}_p, \boldsymbol{k}_i$ are unsuitable and can result in damage to the mechanism. To mitigate this, two sets of PID gains are chosen, one for the windup state ($\boldsymbol{k}_p = 800$, $k_i = 3000$), and another for the tracking state ($\boldsymbol{k}_p = 3 \times 10^4$, $k_i = 350$), which the windup state transitions to once suitable stability is achieved. This transition trigger is defined on a per-axis basis, as the first zero crossing of the angle error (as $\boldsymbol{q} = 0$ this is effectively $\boldsymbol{\theta}$). A graph showing the difference this state change makes to the **AUJ!** orientation is shown in figure **??**.

C. Results

1) Force Proportional Controller & Motor Characterisation: Before verifying robust control of the **AUJ!**, the control of each individual **TSA!** needed to be tested in order to characterise the performance of the motors and to ensure the inner loop control system was robust. This would involve selecting the control strategy that gave the best performance. A test trajectory consisting of a smooth ramp followed by a sine wave was fed into the inner loop of the cascade function as $\begin{bmatrix} f(t) & f(t) & f(t) \end{bmatrix}$. Each control strategy was tested, and in the end the velocity control strategy proved most optimal, as is shown in figure **??**. The results of a sinusoidal force trajectory for the velocity control strategy is shown in figure **??**. As can be seen in the in the inset, there is a stick-slip friction effect whch causes a sawtooth like effect on the measured force. This means the motor angle has to be contantly adjusted even with a constant force input, in order to mantain the setpoint force.

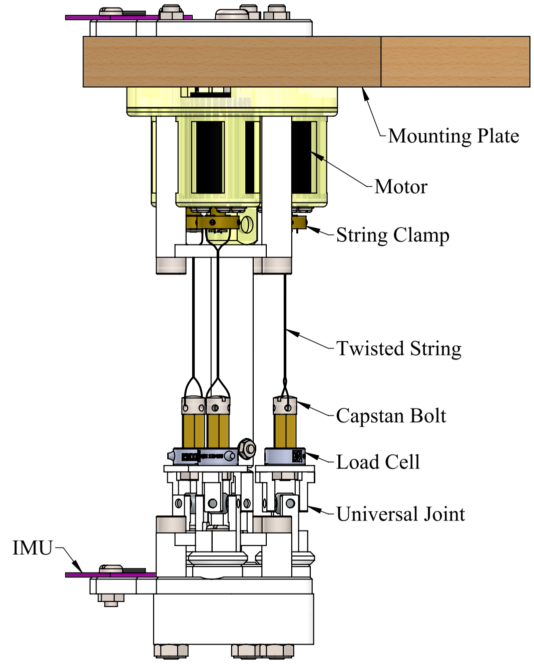


Fig. 13: Schematic of the experimental system with labelled components.

*2) **AUJ!** Position PID Controller:* Figure **??** plots the tracking response of both the simulation and experiment. Three trajectories were created to test the capabilites of the mechanism. Two were only in one axis of the Universal Joint, and the third was in both axes. The deflection angle range was limited due to the low value of l_u, but can easily be extended by increasing this value. A low l_u was chosen as it resulted in easier installation.

V. Performance Comparison

To compare the performance of a **TSA! AUJ!** against alternatives, we can measure two metrics, the maximum tension force f_{max} and maximum stroke velocity $\dot{p}_{max}$. This corresponds to the equivalent of maximum torque and maximum velocity in a rotary motor, a larger f_{max} would be able to actuate a larger follower mass, and a larger $\dot{p}_{max}$ would be able to rotate the **AUJ!** more quickly. The alternatives chosen for comparison are lead-screws of various rod diameters d_m and pitches λ, and a "direct drive" where the motor is rotating the universal

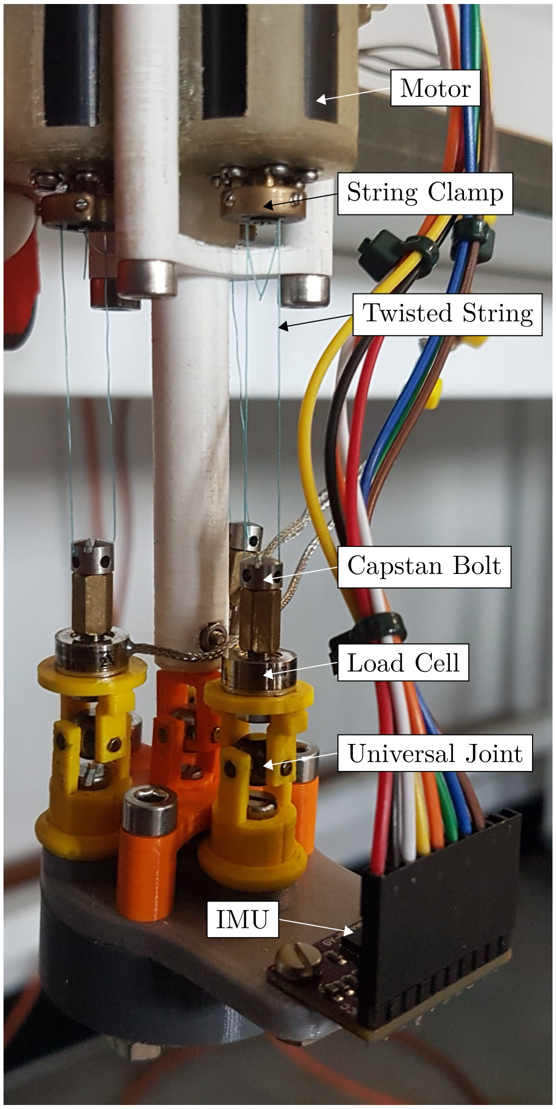


Fig. 14: Annotated photograph of the experimental system.

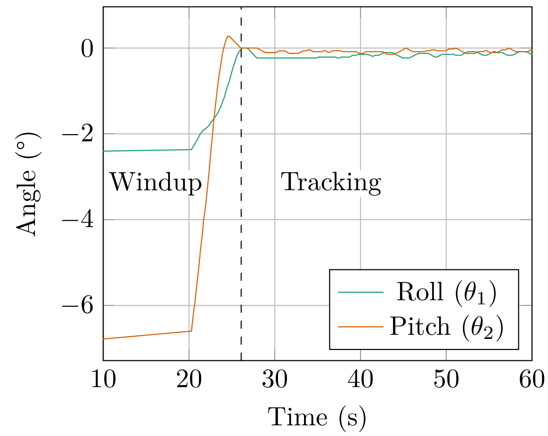


Fig. 15: The inital startup of the mechanism showing the transition between “windup” and “tracking” states.

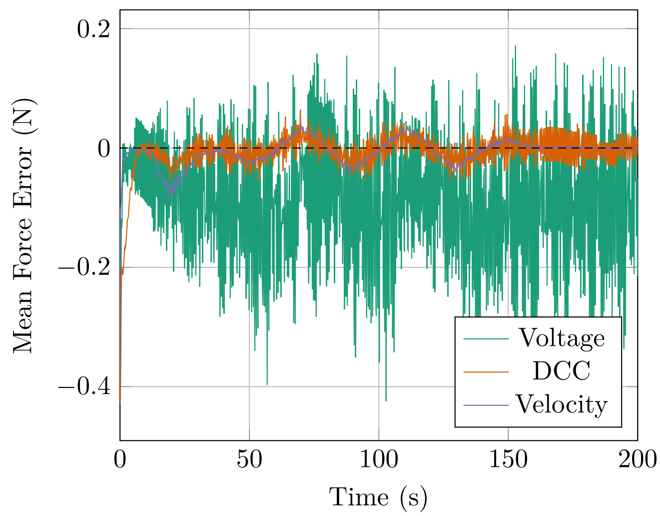


Fig. 16: Mean tracking error for each control strategy.

joint directly without any reduction or motion transformation.

A. *TSA!*

For the **TSA!** metrics, the equations from **wurtz2010**, in particular $h(\theta)$ and $k(\theta)$ as used for the state space, which can be used to determine f_{max} and $\dot{p}_{max}$. By extracting coefficient r_s as an input to make $f(p, r_s)$ and $\dot{p}(\dot{\theta}, p, r_s)$ the performance of different string thicknesses can be compared for a given unwound length l_u and $\tau_{max}, \dot{\theta}_{max}$ over the range of the contraction length p.

$$
\begin{aligned}
k(\theta) &= l_u - \sqrt{l_u^2 - \theta^2 r_s^2} \\
k^{-1}(p) &= \pm\frac{\sqrt{p(2l_u - p)}}{r_s} \\
h^{-1}(\theta) &= \frac{\sqrt{l_n^2 - r_s^2\theta^2}}{r_s^2\theta} \\
f(p) &= h^{-1}\left(k^{-1}(p)\right) = \pm\frac{\sqrt{(l_u - p)^2}}{r_s\sqrt{p(2l_u - p)}} \\
f_{max} &= f(p)\tau_{max}
\end{aligned}
\tag{20}
$$

$$
\begin{aligned}
\dot{k}\left(\dot{\theta}, \theta\right) &= \frac{\dot{\theta} r_s^2 \theta}{\sqrt{l_n^2 - r_s^2\theta^2}} \\
\dot{p}\left(\dot{\theta}, p\right) &= \dot{k}\left(\dot{\theta}, k^{-1}(p)\right) = \pm\frac{\dot{\theta} r_s\sqrt{p(2l_n - p)}}{\sqrt{(l_n - p)^2}} \\
\dot{p}_{max} &= \dot{p}\left(\dot{\theta}_{max}, p\right)
\end{aligned}
\tag{21}
$$

B. *Leadscrew*

For the leadscrew metrics, the raising torque calculation **shigley2004mechanical** can be used as the absolute value of f_{max}, since the **TSA!** only operates in tension, which can be used to determine the same metrics. The performance of different screw diameters d_m and leads λ can then be compared for a given τ_{max} and coefficient of friction μ. $\dot{p}_{max}$ is then calculated by multiplying λ with $\dot{\theta}_{max}$. The performance of different λ can then be compared for a given $\dot{\theta}_{max}$.

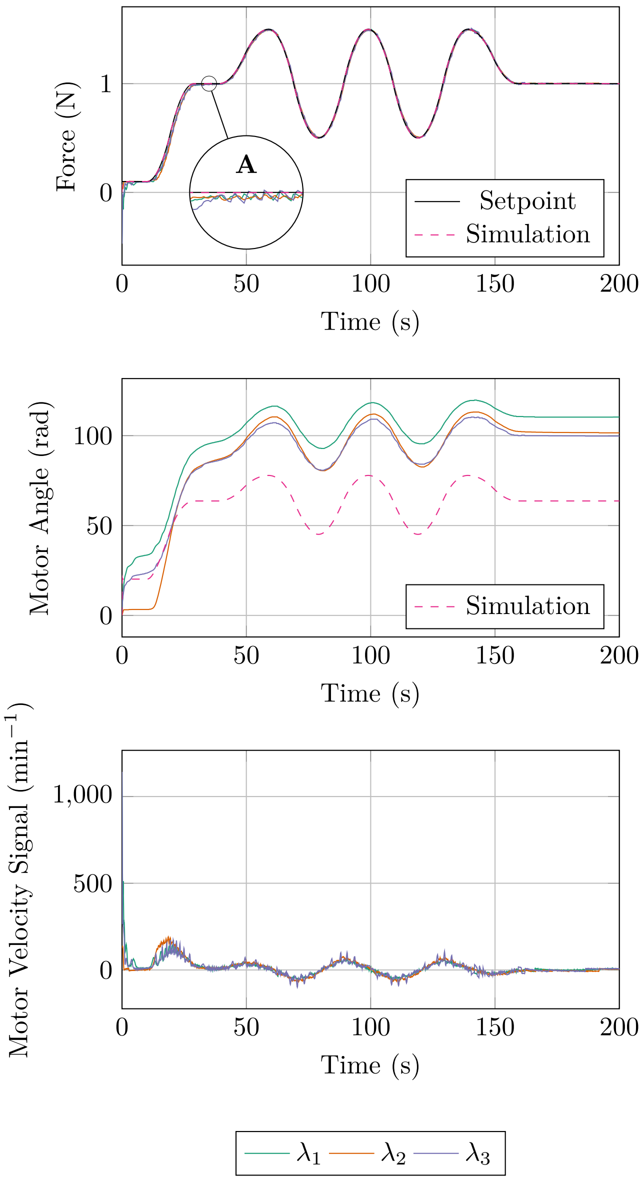


Fig. 17: Load cell, motor angle, and motor velocity signal plot from each string using the velocity control strategy for an experimental trajectory. Inset **A** shows a zoomed in portion of the graph, illustrating the stick-slip phenomenon noticed in the load cell measurements.

$$
\begin{aligned}
|\tau(f)| &= \frac{d_m f(\lambda + \pi d_m \mu)}{2(\pi d_m - \lambda\mu)} \\
|f(\tau)| &= \frac{2\tau(\pi d_m - \lambda\mu)}{d_m(\lambda + \pi d_m \mu)} \\
f_{max} &= |f(\tau_{max})|
\end{aligned} \tag{22}
$$

$$
\begin{aligned}
\dot{p}\left(\dot{\theta}\right) &= \lambda\frac{\dot{\theta}}{2\pi} \\
\dot{p}_{max} &= \dot{p}\left(\dot{\theta}_{max}\right)
\end{aligned} \tag{23}
$$

C. Direct Drive

The direct drive metrics are trivially calculated using the lever force and tangential velocity that would be generated at the endpoint of a linear actuator able to impart equivalent angular velocity and torque on the universal joint.

$$f_{max} = \frac{\tau_{max}}{\sqrt{l_2^2 + r^2}} \tag{24}$$

$$\dot{p}_{max} = \dot{\theta}_{max}\sqrt{l_2^2 + r^2} \tag{25}$$

D. Comparison between *TSA!* and Leadscrew

As the values for τ_{max} and $\dot{\theta}_{max}$ for the **TSA!** depend on p, but remain constant for the leadscrew, the performance of the **TSA!** is going to be better or worse than a given leadscrew depending on the value p. Figure **??** compares the **TSA!** configuration using the coefficients from table **??** against a number of common leadscrew configurations that are practical for the dimensions of the **AUJ! empty citation** The **TSA!** outperforms or underperforms different leadscrew configurations depending on p. In simpler terms, the performance of the **TSA!** is dependent on the contraction length. The maximum linear velocity increases with the contraction length, and the maximum tension force decreases with contraction length, both in an non-linear fashion.

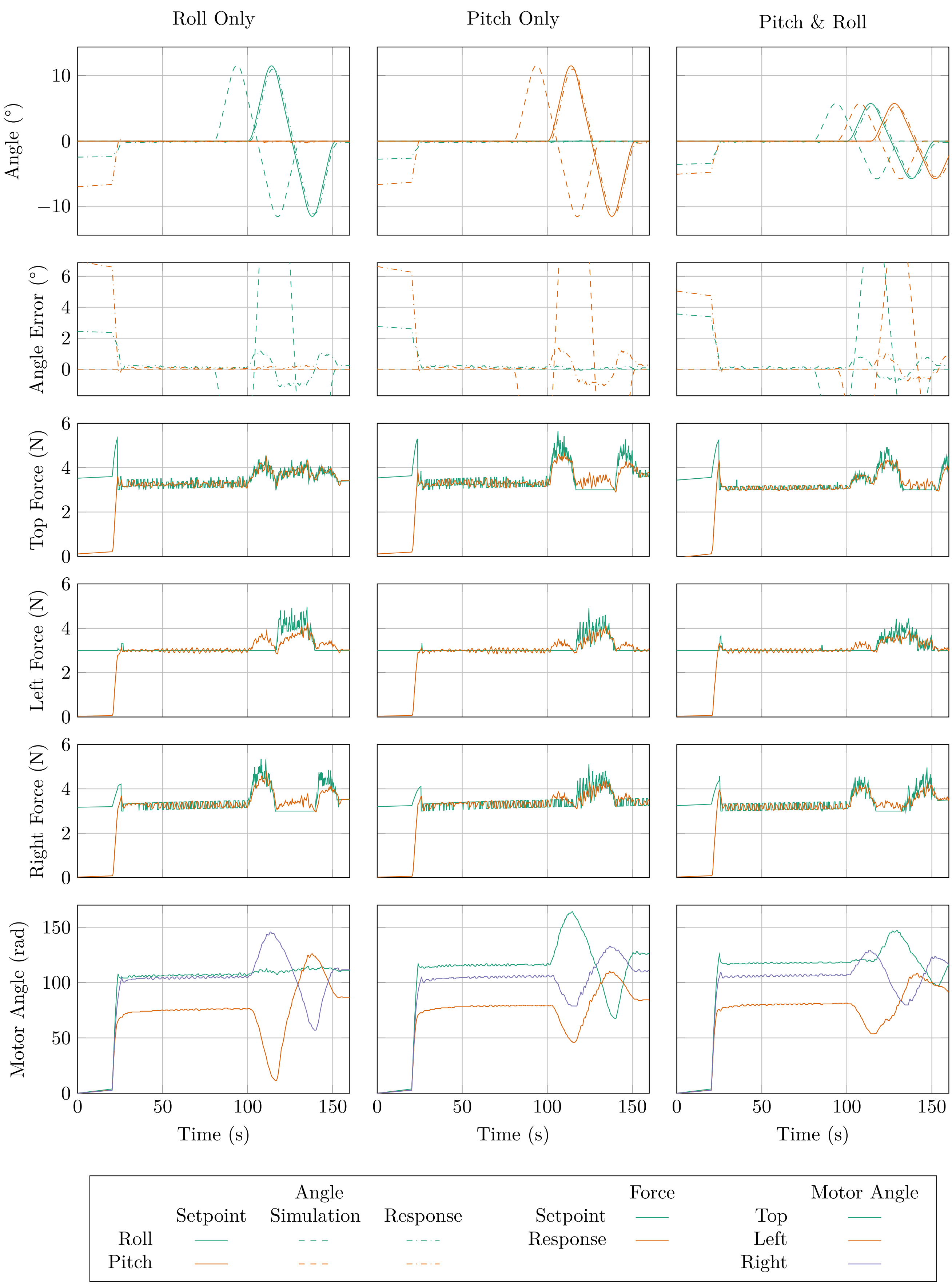
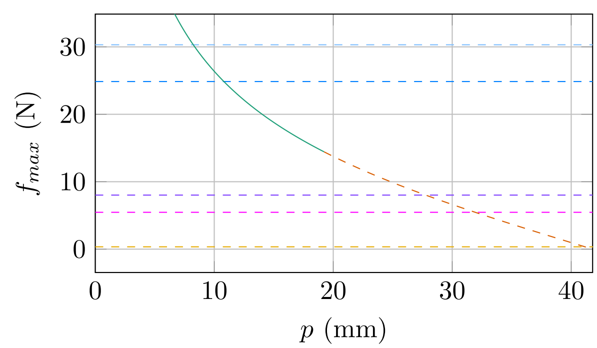
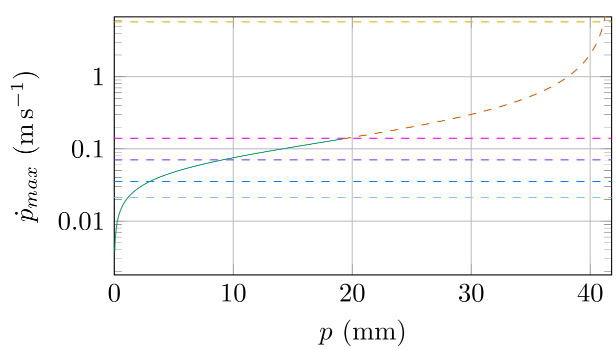


Fig. 18: Plots of the setpoint and response for three different trajectories, one on only the roll axis (column 1), one on only the pitch axis (column 2), and one on both axes (column 3). Plots include **AUJ!** orientation, forces at the top, left and right **TSA!**, and the motor positions.





??

Fig. 19: Performance comparison of the **TSA!** configuration using coefficients from table **??** to various leadscrew configurations with different d_m and λ, and the direct drive, where $0.1 = 0.1$ for the leadscrews.

VI. Discussion & Conclusion

A. Discussion

Upon reflection of the research conducted and the results obtained, future developments would include improvements to the orientation sensors, the data from the **IMU!** proved to be unreliable and of poor resolution (2.0° static, 3.5° dynamic). The magnetomoter was unusable within the indoor experimental environment even after several calibration attempts, so the universal joint angle would not be able to be calculated when the gravity vector is not orthogonal to the unviersal joint **DOF!**. An alternative method for sensing the universal joint orientation should be investigated, such as **LVDT!**s (**LVDT!**s), hall effect sensors or potentiometers. Alternative control algorithms that are less or not dependant on the dynamic properties of the follower segments (mass and inertia) will also be considered, in order to improve the robustness of the control system to external forces. The 1724TSR motors also had poor performance at low speeds, necessitating the deadband compensator. Investigating the use of similar size and torque brushless motors with sinusoidal commutation for stable low speed control could eliminate the need for the compensator, and allow for smoother **AUJ!** motion. Development of a system comprised of multiple segments is the eventual goal, to demonstrate its suitability for applications such as mobile snake robots or snake-arm robots. This can use a embedded controller for each segment, controlled by a master controller for individual joint control or inverse kinematics, as in figure **??**.

B. Conclusion

This research has successfully demonstrated the robust control of the orientation of a universal joint using **TSA!** in an antagonistic triad configuration. It has also compared the performance of the system to alternative actuation methods, and found that its non-linear nature makes it difficult to directly compare to similar linear actuators in the same application.

Damian J. Crosby is a PhD student in the school of Mechanical, Aerospace and Civil Engineering, University of Manchester, Manchester, U.K. He received a B.Sc. in Special Effects Development from The University of Bolton, U.K., in 2010, and a M.Res. in Robotics from The University of Plymouth, U.K., in 2012. He worked as a Research Technician at The University of Manchester from 2013 to 2017, before commencing his PhD.

Joaquin Carrasco is a Lecturer at the School of Electrical and Electronic Engineering, University of Manchester, UK. He was born in Abarn, Spain, in 1978. He received the B.Sc. degree in physics and the Ph.D. degree in control engineering from the University of Murcia, Murcia, Spain, in 2004 and 2009, respectively. From 2009 to 2010, he was with the Institute of Measurement and Automatic Control, Leibniz Universitt Hannover, Hannover, Germany. From 2010 to 2011, he was a research associate at the Control Systems Centre, School of Electrical and Electronic Engineering, University of Manchester, UK.

William P. Heath is Chair of Feedback and Control in the School of Electrical and Electronic Engineering, University of Manchester, Manchester, U.K. He received the B.A. and M.A. degrees in mathematics from Cambridge University, U.K., in 1987 and 1991, and the M.Sc. and Ph.D. degrees in systems and control from the University of Manchester Institute of Science and Technology, U.K., in 1989 and 1992, respectively. He was with Lucas Automotive from 1995 to 1998 and was a Research Academic at the University of Newcastle, Australia from 1998 to 2004. His research interests include absolute stability, multiplier theory, constrained control, and system identification.

Dr. Andrew Weightman graduated in 2006 with a PhD in Mechanical Engineering form the University of Leeds. Whilst at the University of Leeds he developed rehabilitation robotic technology for improving upper limb function in adults and children with neurological impairment which was successfully utilised in homes, schools and clinical settings. In 2013 he moved to the University of Manchester, School of Mechanical, Aerospace and Civil Engineering as a Lecturer in Medical Mechatronics. Dr Weightman has research interests in biomimetic mobile robotics, rehabilitation robotics, robotics for nuclear decommissioning and collaborative robotics.

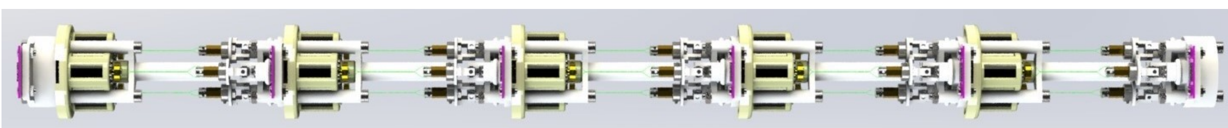


Fig. 20: Visualisation of several **TSA! AUJ!** chained together to form a robotic snake or tail.