DESIGN GUIDELINES FOR SHOULDER DESIGN OF AN ANTHROPOMORPHIC ROBOTIC ARM

Leroux, Martin; Achiche, Sofiane; Beaini, Dominique; Raison, Maxime
École Polytechnique de Montréal, Canada

Abstract

The development of biomechanically-accurate robotic arms is of high interest; in this paper we investigate an actuated spherical mechanism for biofidelic shoulder design. Given the high inertia of an extended arm, serial mechanisms were quickly discarded in favour of a parallel design. One particular design, the agile eye, was studied in depth due to multiple intrinsic advantages. However, no work was done regarding its capacities with higher inertia loads. Our objective is to characterise the behaviour of the agile eye in terms of speed, precision and input efforts when there is a load on the end effector. We developed the dynamic model of the robot to implement torque control with a PD controller. The system was then simulated for a few point to point trajectory conditions. The characterisation of the behaviour of the robot with a load on its end effector revealed that it reaches similar precision, with slightly slower speed depending on design choices but requires motors with power proportional to the expected load in comparison to when there is no mass on the end effector. These results can serve as design guidelines for anthropomorphic robotic arm/shoulder development.

Keywords: Mechatronics, Bio-inspired design / biomimetics, Biomedical design, Adaptation robotics

Contact:
Prof. Sofiane Achiche
École Polytechnique de Montréal
Mechanical Engineering
Canada
sofiane.achiche@polymtl.ca

Please cite this paper as:
Surnames, Initials: *Title of paper*. In: Proceedings of the 21st International Conference on Engineering Design (ICED17), Vol. 4: Design Methods and Tools, Vancouver, Canada, 21.-25.08.2017.

1 INTRODUCTION

Developing a biomechanically accurate robotic manipulator for adaptation purposes is of great interest. Indeed, anthropomorphic robots can allow more intuitive and natural looking trajectories during the motion of the robot. In order to reproduce the behaviour of a human upper limb, we aim to design an anthropomorphic robotic arm to serve as a base for myoelectric prostheses. The developed system is based on early results, of our research group, based on the biofidelic biomechanical model presented in (Laitenberger et. al, 2015) which models the arm with seven degrees-of-freedom and includes a closed cinematic loop to properly reproduce the movement of pronation and supination which are generally not found in traditional robotic prostheses. To elaborate the design of the robot accurately, a mechanism acting as an actuated ball-joint is required for the shoulder. In the present paper, we investigate such a mechanism.

Spherical parallel mechanisms (SPM) proved to be useful for 3 degree of freedom (3DOF) applications where only rotations are required, as they have a wide workspace with a really low moment of inertia and stiffness (Tao, 2013. Gosselin, 1996. Gosselin, 1994) They are used in numerous domains such as parallel robotics (Hamlin, 1994), surgical robotics (Lum, 2006), flight simulators (Pouliot, 1998), action origami (Bowen, 2014), haptic technologies (Antolini, 2013), and many more.

Research in spherical mechanisms have led to the development of the agile eye, illustrated in Figure 1 in the early 90s by Gosselin & al., which is recognised for its higher angular velocity, higher angular accelerations and a workspace consisting of a 140° cone (Gosselin, 1996. Bonev, 2003). It is designed to minimise the interference between links (Tao, 2013. Gosselin, 1996) and its singularities consist of curves at the boundaries of the workspace instead of surfaces like most 3DOF SPM, when the second joint of any leg is at an angle of $\alpha_i = 0, \pi$, in addition to a single point at the origin (Bonev, 2003. Srivatsan, 2013) and therefore maximises its workspace volume. Hence, the agile eye appears to be one of the best spherical mechanisms available.

The agile eye was developed as a mechanism that would be mounted by a camera to allow really fast eye-like movements (Gosselin, 1994. 1996). Therefore, most applications of this mechanism orient optical devices, like cameras and mirrors, to mimic the human eye and to allow faster active vision. In these applications, the load to be moved by the agile eye is negligible compared to the inertia of the robot. Consequently, no study was done to analyse the capacities of the agile eye when a high inertia load is attached to its end-effector. This lead to very few integrations of the agile eye into more complex mechanisms, which is regrettable considering all of the above advantages that the agile eye has by design compared to other SPMs.

The main objective of this paper is to compare the performance of the agile eye in terms of speed, precision and input efforts when there is a high inertia load or a negligible mass compared to the robot attached to its end-effector to validate its use as a design for the shoulder of an anthropomorphic robotic arm. As a secondary objective, we take the opportunity to fill a gap in the literature regarding the control of the robot. The control of such a device was already mentioned in the literature (Gosselin, 1996), but no method for control is developed in detail. Non-linear control algorithms for a rather complex device can be very cumbersome to implement, therefore, in this paper, the dynamic model of the mechanism is developed in detail based on a Lagrangian formalism in order to implement a computed torque proportional-derivative (PD) control algorithm (Benon, 2000). Using the developed model and controller in a simulation, we then explore the capacities of the agile eye as a design option in more complex systems.

2 METHODOLOGY

In this section, we develop the non-linear computed torque control algorithm for the agile eye that was implemented in our simulation. Firstly, the kinematic model of the agile eye is described shortly. Secondly, the dynamic model of the system is developed using the Lagrangian formalism. Thirdly, a computed torque control algorithm with a PD controller is implemented. Finally, the simulation parameters used to obtain the results illustrated in the following section are detailed and can be used as design guidelines for further work.

2.1 Kinematic model of the Agile eye

The inverse and direct kinematics is briefly presented. The loop-closure equations are then presented since they were used to reduce the number of equations in the dynamic model.

2.1.1 Inverse and direct kinematics

Since the agile eye is a spherical robot, the natural output of the system is its orientation. In order to compute the reference angles (θ_{ref}) from a desired output, the inverse kinematics of the robot has to be calculated. This was done in previous work (Gosselin, 1996) and can be used as is. That algorithm yields solutions separated by π rad, leading to two solutions for each angle θ_i and thus a total of $2^3 = 8$ combinations for a given orientation (2 solutions per $\theta_i$with i=1,2,3). By defining $\theta_i \in [-\pi, \pi]$ and knowing that it represents the offset angle from the rest position, these configurations can be narrowed down to a single one by choosing the smallest absolute values of between the 2 solutions of each θ_i.
For simulation purpose, it was also necessary to obtain the direct kinematics of the robot in order to verify the output orientation and to observe the behaviour of the controlled system. Assuming the loop-closure equations were solved correctly, this could have been done using transformation matrices to express the coordinates of a point given in the end-effector local coordinate system into the global coordinate system. However, we found more convenient to numerically inverse the inverse kinematic function using a non-linear least-square algorithm.
This optimisation process was computed for each integration step using the previous value as an initial guess to start the optimisation.

2.1.2 Loop-closure equations

Since the agile eye is a parallel robot, its configuration must undergo certain constraints to close the kinematic loops. The mathematical representation of those constraints is not unique. We chose to express the location of a point in the global coordinate system as a transformation of a point expressed in the local coordinate system of the end of each of the kinematic chains as shown in Equation (1).

$$\begin{aligned} &{}^{O}\boldsymbol{T}_{O_{A1}}(\theta_1)\,{}^{OA}\boldsymbol{T}_{OB}(\alpha_1)\,{}^{OB}\boldsymbol{T}_{OE}(\beta_1)\cdot \boldsymbol{CM_E} = \\ &{}^{O}\boldsymbol{T}_{O_{A2}}(\theta_2)\,{}^{OA}\boldsymbol{T}_{OB}(\alpha_2)\,{}^{OB}\boldsymbol{T}_{OE}(\beta_2)\cdot \boldsymbol{CM_E} = \\ &{}^{O}\boldsymbol{T}_{O_{A3}}(\theta_3)\,{}^{OA}\boldsymbol{T}_{OB}(\alpha_3)\,{}^{OB}\boldsymbol{T}_{OE}(\beta_3)\cdot \boldsymbol{CM_E} \end{aligned} \tag{1}$$

Where:

- $\boldsymbol{CM_E}$ is the centre of mass of the end-effector expressed in local coordinates identical in all kinematic chains because of the symmetry of the agile eye.
- $\theta_i, \alpha_i, \beta_i$are the angles of rotation illustrated in Figure 2 for one kinematic chain.
- O, O_A, O_B, O_E are respectively the global, the first link, the second link and the end-effector coordinate systems.

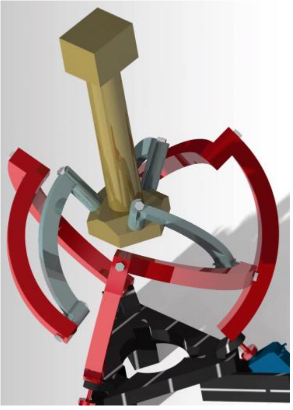


Figure 1. The Agile eye

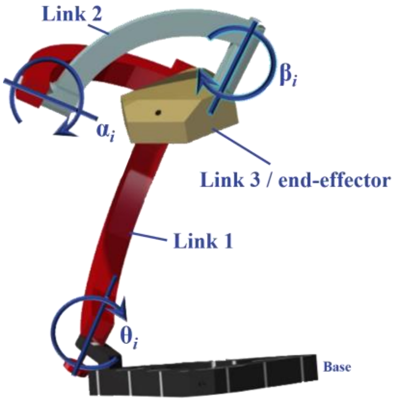


Figure 2. Generalized coordinates on link i

These non-linear expressions allow for three expressions of $\alpha_i$and β_i (the passive angles) as a function of θ_i (the active angles) to be found.

Furthermore, the speed of $\boldsymbol{CM_E}$ must be the same if expressed from any coordinate system. The time derivative of each expression in Equation (3) can be taken in order to obtain expressions for $\dot{\alpha}_\iota$ and $\dot{\beta}_\iota$as functions of $\dot{\theta}_\iota$and the position variables. They will be useful to substitute the passive angle velocities in the following section. These expressions, along with the expressions derived from Equation (3) are non-linear and must be solved numerically.

2.2 Dynamic model of the agile eye

We obtained the equations of motion using a Lagrangian approach. This method was preferred to a Newtonian approach to reduce the number of equations to solve by eliminating the internal forces (Taylor, 2005). The chosen generalised coordinates $\boldsymbol{q}$ were all the angles between de links $\theta_i, \alpha_i, \beta_i$ illustrated in Figure 2.

The Lagrangian ($\mathcal{L}$) for an arbitrary system, is always written as (Taylor, 2005) - Equation (2):

$$\mathcal{L} = K - V \tag{2}$$

Where K and V are respectively the kinetic and potential energy of the system.

For the agile eye, the kinetic energy $K_j$of a link j ($j \in [1,2,\dots,7]$), including the end-effector, assuming all local coordinate systems were chosen so that the $\hat{z}$ axis is along the joint rotation axis, is given by Equation (3):

$$K_j = \frac{1}{2} m_j \sum_q (({}^{O_j}\boldsymbol{T}_{O_q}\boldsymbol{CM}_j)' \,\boldsymbol{M}\, ({}^{O_j}\boldsymbol{T}_{O_q}\boldsymbol{CM}_j))\, \dot{\boldsymbol{q}}^{\boldsymbol{2}} + \frac{1}{2}\, (\dot{\boldsymbol{q}}'({}^{O_j}\boldsymbol{T}_{O_q}\boldsymbol{I}_j)\dot{\boldsymbol{q}}) \tag{3}$$

Where:

- m_j [kg] is the mass of link j
- O_j is the local coordinate system of link j
- $\boldsymbol{CM}_j$ [m] is a $4{\times}1$ vector containing the coordinates of the center of mass of link j in O_j
- $\boldsymbol{I}_j[kg \cdot m^2]$ is the 4×4 inertia tensor of link j in O_j relatively to $\boldsymbol{CM}_j$
- $\boldsymbol{q}$ are 4×1 vectors containing the generalised coordinates that can physically influence link j, replacing the notation $\theta_i, \alpha_i, \beta_i$ only here for a more concise equation
- O_q is the local coordinate system of the link directly influenced by q
- ${}^{O_j}\boldsymbol{T}_{O_q}$ is a $4{\times}4$ linear transformation converting vectors from $O_q$to O_j
- $\boldsymbol{M}$ is a 4×4 projection matrix keeping the $\hat{x}$ and $\hat{y}$ values of a vector only

Take note that all the above matrices and vectors are expressed in homogeneous coordinates to simplify the multiple calculations of linear transformations.

In the simulation described in the following sections, the values of m_j, $\boldsymbol{CM}_j$, and $\boldsymbol{I}_j$were given by a developed CAD model of the agile eye, assuming a constant full density of ABS plastic. ABS plastic was chosen as it is a common material for rapid prototyping.

The potential energy $V_j$of each link j including the end-effector, is given by Equation (4):

$$V_j = m_j({}^{O}\boldsymbol{T}_{O_j}\boldsymbol{CM}_j \cdot \boldsymbol{g}) \tag{4}$$

Where:

- O is the global coordinate system
- $\boldsymbol{g}$ is the gravity vector

Hence, the Lagrangian of the complete agile eye is given by the following function of the nine general coordinates given by Equation (5):

$$\mathcal{L}(\boldsymbol{q}) = \sum_j (K_j - V_j) \tag{5}$$

Using the expressions found using the loop-closure Equations (1), the number of generalised coordinates, and hence the number of equations necessary to model the system is reduced to three. The equations of motion of the system are given by the Euler-Lagrange formalism as in Equation (6) (Taylor, 2005) for the actuated angles. The results from the loop closure equations were then substituted to obtain a dynamic model depending only on the θ_i values and their respective time derivatives - Equation (6).

$$\frac{d}{dt}\left(\frac{d\mathcal{L}}{d\dot{\theta}_i}\right) - \frac{d\mathcal{L}}{d\theta_i} = \tau_i \tag{6}$$

Where τ_i [Nm] is the joint torque applied on link 1 of the kinematic chain i. It can be rewritten - Equation (7):

$$\boldsymbol{\tau} = \boldsymbol{D}(\boldsymbol{\theta})\ddot{\theta} + \boldsymbol{h}(\dot{\boldsymbol{\theta}}, \boldsymbol{\theta}) \tag{7}$$

Where:

- $\boldsymbol{D}(\boldsymbol{\theta})$ is a generalised 3×3 inertia matrix
- $\boldsymbol{h}(\dot{\boldsymbol{\theta}}, \boldsymbol{\theta})$ is a 3×1 vector containing the non-linear Coriolis, centrifugal and gravitational forces

2.3 Implementation of a non-linear computed torque algorithm

For the purpose of control, the vector $\boldsymbol{\tau}$ from Equation (7) was used as an input since the motors can apply torques on the actuated joints. It was then necessary to linearise the system in order to use the PD controller. A bloc diagram of the control feedback loop is illustrated in Figure 3. The following section describes the content of the controller box.

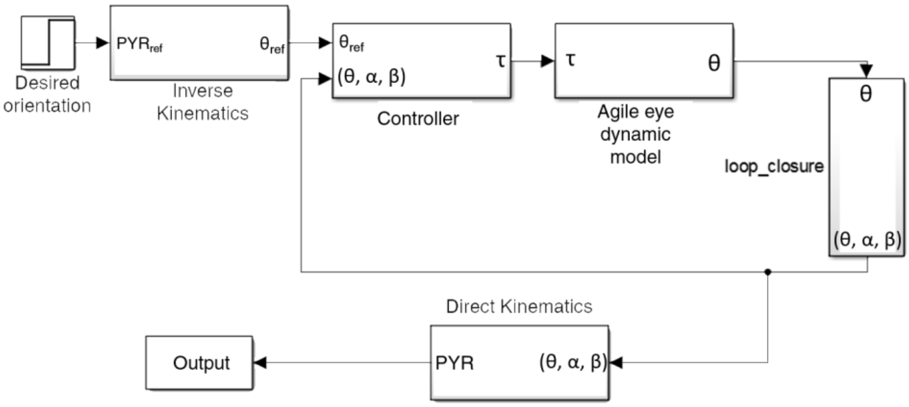


Figure 3. Simulink bloc diagram of the agile eye control simulation

A natural choice of non-linear controller for the agile is a computed torque algorithm because it should behave uniformly over the workspace and it is easy to implement once the dynamic modelling of the robot is completed. It was done by first linearising the state-space model by dynamic inversion of the system, then a PD controller can be used to stabilise the output.

By choosing $x_{2i-1} = \theta_i$ and $x_{2i} = \dot{\theta_i}$ for i=1..3 as state variables, the state-space model of the system that was used for a computed torque PD controller can be expressed as in Equations (8-9).

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} + \begin{bmatrix} 0 \\ f_1(\boldsymbol{x}, \tau_1) \\ 0 \\ f_2(\boldsymbol{x}, \tau_2) \\ 0 \\ f_3(\boldsymbol{x}, \tau_3) \end{bmatrix} \tag{8}$$

$$f_i(\boldsymbol{x}, \tau_i) = \boldsymbol{D}^{-1}(x_i)[\tau_i - \boldsymbol{h}(x_i, x_{i+1})] \tag{9}$$

We used the input τ_i written in Equation (10), using a simple PD controller to stabilise the system, with $K_P$and K_D as proportional and derivative gains respectively.

$$\tau_i = \boldsymbol{D}(\boldsymbol{\theta})\left[\ddot{\theta}_{iref} + K_{D_i}\left(\dot{\theta}_{iref} - \dot{\theta}_i\right) + K_{P_i}\left(\theta_{iref} - \theta_i\right)\right] + \boldsymbol{h}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) \tag{10}$$

By inserting the input τ_i from Equation (10) subsequently in Equations (9) and (8), that the state-space model is simplified and can be rewritten as:

$$\dot{\boldsymbol{x}} = \begin{bmatrix} A_1 & 0 & 0 \\ 0 & A_2 & 0 \\ 0 & 0 & A_3 \end{bmatrix} \boldsymbol{x}, \quad A_i = \begin{bmatrix} 0 & 1 \\ -K_{P_i} & -K_{D_i} \end{bmatrix} \tag{11}$$

Since the agile eye is symmetrical, the controller gains should also be symmetrical, hence the further simplification:

$$A_1 = A_2 = A_3 = A \quad (12)$$

meaning the controller gains are identical on each kinematic chain.
The error dynamic of the system, leading to the correct trajectory, is then stable if the eigenvalues of A have negative real parts. This leads to the following condition on the PD controller gains:

$$K_D^2 > 4K_P \quad (13)$$

2.4 Simulation parameters

To be able to compute the mass and inertia of every link, as mentioned above, a 3D CATIA (Dassault Systemes, France) model was made by using the specifications mentioned by Gosselin & al. (1994, 1996), assuming densities corresponding to 3D printing ABS plastic. We chose the controller gains $K_P = 50$ and $K_d = 30$ arbitrarily as an example so that in Equation (13) was respected. We chose to simulate point to point trajectories, so $\ddot{\theta}_{iref}$ and $\dot{\theta}_{iref}$ are always zero. The initial configuration of the agile eye in all our simulations is at rest such that $\theta_{1,2,3} \sim 0$ just avoiding the singularity at the origin. A saturation of the input torque was set to 10 Nm.
The dynamic model and the control strategy described previously were simulated using Simulink with the diagram illustrated on Figure 3. The controller calculates the torque using Equation (10). This torque allows determining the subsequent $\theta_i, \alpha_i, \beta_i$ through integration and the loop-closure Equations (1). These values will then close the loop with the controller to allow feedback for the PD controller.
In the following section, we present simulation results for two systems and trajectories. First, a validation of the dynamic model and control algorithm is obtained by simulating the agile eye with negligible mass on its end-effector, responding to a step input with a delay of 1s as a reference orientation with Pitch, Yaw and Roll PYR_{ref} values of $(\frac{1}{4}, -0.34, \frac{\pi}{7})$ randomly chosen inside the workspace of the agile eye (Gosselin, 1996). Then, our model of the agile eye was modified to include an arbitrary high inertia load on its end-effector, roughly five times heavier than the rest of the robot, as seen in Figure 1. To emulate a more realistic input and demonstrate that our control algorithm works for another reference orientation, but still yield easy to read and printer friendly graphics, the simulation of this model was run with $PYR_{ref} = (\frac{1}{4}, -0.34, \frac{\pi}{9})$ and a saturated ramp as an input.
More simulation conditions have been tested and will be briefly mentioned in the following section.

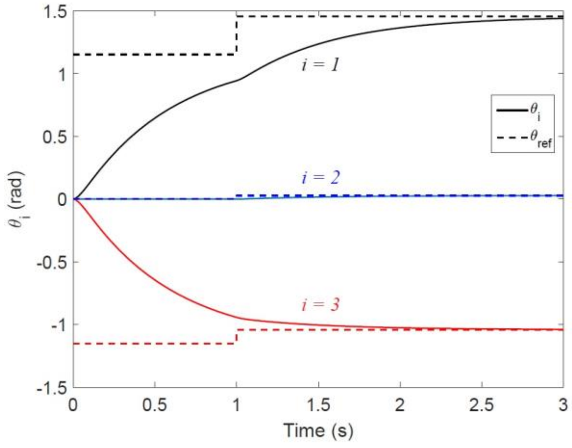


Figure 4. Joint path following results, no load

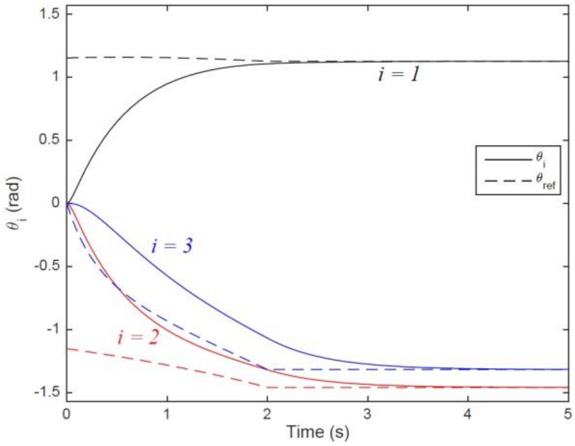


Figure 5. Joint path following results, load

3 SIMULATION RESULTS

3.1 Dynamic model and controller validation

Using the parameters described previously, given the reference states obtained with the desired output and the inverse kinematics, our simulation resulted in the response illustrated in Figure 4 for the values of the θ_i.The state angles converge smoothly toward their respective references over time.

The output orientation (PYR) of the system are illustrated in Figure 6. They were calculated by considering the direct kinematics as the numerical inverse of the inverse kinematics.

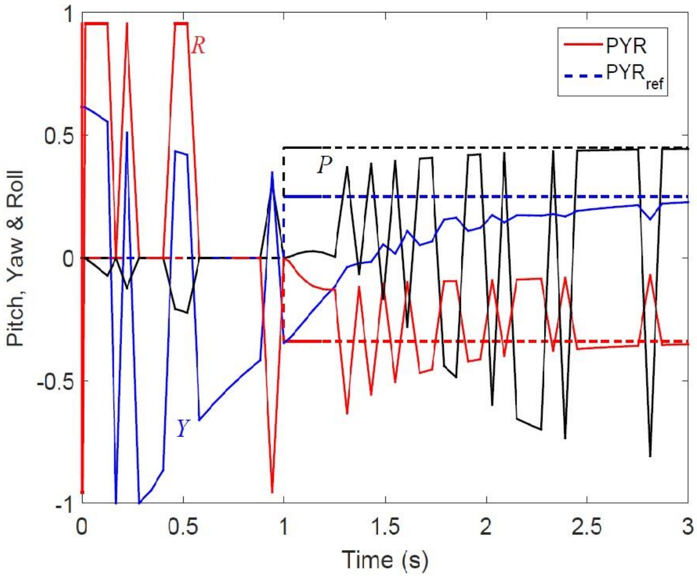
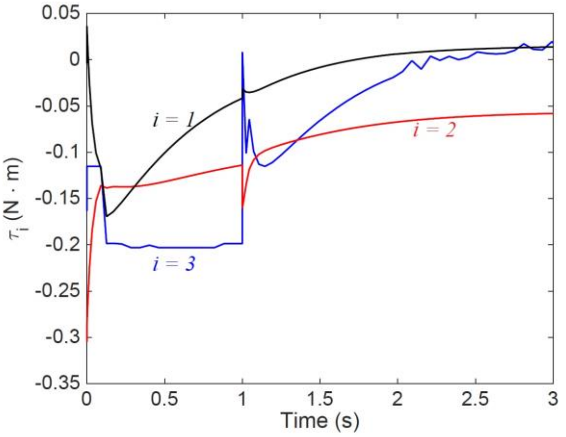


Figure 6. Output orientation path following simulation with computed torque PD controller



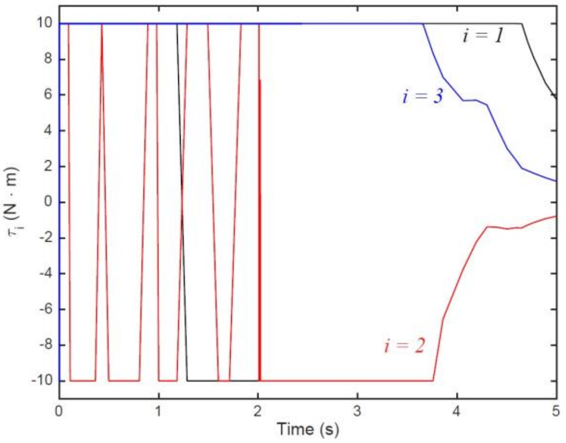


Figure 7. Output torque of PD with no load

Figure 8. Output torque of PD with load

Although some erratic patterns are displayed, which will be discussed later, the general behaviour is that the PYR angles converge toward their reference values.

Figure 7 illustrates the torque output for each motor τ_i during our simulated point to point trajectory.

More simulations were run to determine how the controller gains (K_P and K_D) would influence the results. We found that the gains could be up to 100 times larger than the values used in the illustrated simulation without requiring higher torques than 10 Nm. The gains can be further increased when including a saturation of the input in the simulation. The response time of the system does not significantly increase when the saturation is applied, but input efforts saturate for longer duration as the gains increase. Those results are not illustrated here for brevity considering that, other than the mentioned local differences, they are very similar to the results illustrated in Figures 5-7.

3.2 The agile eye moving a high inertia load

Figure 5 illustrates the behaviour of the motor angles θ_i over time compared to their respective reference values θ_{ref} describing the new trajectory for the agile eye with a high inertia load attached to its end-effector.

A plot of the output path following results is not displayed here for brevity considerations since it would display similar erratic behaviour as in Figure 6 and is thus rather unnecessary.
The input torques τ_i of the simulation with a high inertia load on the end-effector are illustrated of Figure 8.

4 DISCUSSION

The objective of this paper was to compare the performance in terms of speed, precision and input efforts of the agile eye with and without a non-negligible load on its end-effector as it is generally described solely with no load in literature. We developed the complete dynamic model for a computed torque PD controller that we implemented in simulation for different point to point trajectories. The detailed equations can serve for any designer who aims to use this type of robot.

4.1 Implementation of the computed torque control algorithm

As mentioned earlier in this paper, the control of the agile eye was previously treated in the literature (Gosselin, 1994), however with only very little details given. Nevertheless, we felt that the agile eye presented enough unique challenges in its dynamic modelling, such as the loop-closure constraints and the $very$ non-linear nature of its dynamic equations, to deserve a detailed demonstration to help future researchers using the agile eye spend less efforts, time and resources in the modelling and more on their innovations. Equations (5) and (6) can be used as plug-and-play functions for geometrical and inertia parameters assuming rudimentary knowledge of transformation matrices. The subsequent steps of the calculation are straightforward and can be performed with a symbolic calculation software.

4.2 Simulation analysis - Validation of the dynamic model

Simulation shows, as seen in Figure 5 that dynamic inversion of the system as a linearisation input along with a PD controller as a stabilising input will, as expected, make the motor angles converge smoothly towards their reference value. This convergence is pretty slow considering the speed that the agile eye is known to be able to generate (Gosselin, 1996), however this is easily explained by the low values of our controller gain that were chosen as an example rather than after an optimisation process.
Although a single simulated path was illustrated previously, many of them were actually run with similar results. One of the main concerns when using inverse dynamics in the design of a controller is the existence of singularities. This is not as important of an issue in the agile eye since, as demonstrated in (G Bonev, 2006. I Bonev, 2006), its design not only reduces the singularity manifolds from planes to curves, but those curves are also located so that it is unlikely to cross them in an arbitrary trajectory, notably on the edge of the workspace. Knowing this, an effective workspace can be generated by reducing the actual workspace by applying a security factor, which can be chosen depending on the accuracy of the actuator used for the implementation.
When comparing the simulated output to the desired output, in Figure 6, a lot of large oscillations can be seen. Since the curves of the simulated motor angles are all smooth, this behaviour must be a simulation artefact and could be explained by the fact that the output is computed numerically. The solution of the minimisation problem created to obtain the direct kinematic must be, for some time samples, converging towards a local but incorrect minimum. This explanation looks very plausible since for most of the time samples the simulated output aligns with the desired output, just as the desired motor angles do align with their respective references. This would therefore not be an issue in an actual implementation since the direct kinematics does not have to be computed.
Finally, Figure 7 shows that the input torque τ_i is smooth for every actuator, except on the very moment where the desired output changes. At this moment, the input torque varies quickly without however jumping by an important value. Indeed, it can be seen, still on Figure 7 that, when spiking, the input efforts barely go beyond values that are necessary to stay stationary on any position that is not gravitationally stable. This is due to the low inertia of the agile eye itself; its spherical shape around its centre of rotation makes it inherently easy to accelerate.

4.3 Simulation analysis - Agile eye moving a high inertia load

It can be observed in Figure 5 that the pre-computed torque algorithm still stabilises the motor angles to the values dictated by the input and the inverse kinematic even if a saturation on the input torque has to be implemented. The stabilisation time is slightly higher, about twice as long, but since again no optimisation was done to choose the controller gains, this is not an issue. It could indeed be anticipated that moving a load will slow the agile eye, but since said load was so much heavier (5x) than the agile eye, this increase in time seems very reasonable. Design choices between the choice of actuators and the desired speed will have to be made for implementations. Moreover, since the stabilising input of the controller works correctly, one can assume that the precision of the agile eye with a heavy load will not decrease.

The output signal was not illustrated in the previous section, however the the motor angles stabilise to their respective desired values, one can safely assume that the real output, as opposed to the numerically estimated one, behaves correctly.

Finally, many interesting facts can be noticed regarding the input torques in Figure 9. First of all, the inputs are set to their saturation value for most of the simulation time. This could be expected since the inertia and saturation values were chosen arbitrarily as this simulation was only meant to be a demonstration. Secondly and most importantly, it can be observed that the input efforts are well distributed between all the motors. The input torques τ_i are indeed all at their saturated values and decrease at very close instants. Similar observation can be done in Figure 7. This is an important feature of the parallel architecture of the agile eye. Since the effort is redistributed, among all the motors, it can be expected that the agile eye mechanism should be able to move loads with less efforts than any of his serial counterparts. The most important conclusion from these results is that point to point trajectories with a high inertia load are realisable assuming motors with power proportional to the load. Using the same motors as the ones designed for a negligible mass on the end-effector will most likely not suffice.

Many other spherical/universal parallel mechanisms have been proposed in the literature. The work of Kong (2016), Sokolov (2007) and Dunlop (1997) are a few examples and share this characteristic of effort distribution. However, those devices do not share all the optimisation by design characteristics of the agile eye discussed at the beginning of this article and elsewhere in the literature (wide workspace, simple singularities, etc.) (Tao, 2013. Gosselin, 1994. Bonev, 2003). Unless some application specific characteristic, such as an asymmetric workspace, is required, the agile eye could always be a go to solution. Further, serial spherical mechanisms, such as those presented in (Hummel, 2000. Tanik, 2015. Mazzei, 1999. Zhang, 2014), do not share the efforts between all the actuators. A custom made actuator can of course be designed with a large power margin if needed, so this might not always be an issue, but most design problems would strongly benefit from the use of a more complex parallel mechanism that has the detailed characteristic. Lastly, electromagnetic spherical actuators have also been designed (Yan, 2006, 2012), for large spherical workspace designs, but their output power is typically rather low compared to their mechanical counterparts.

For all the above reasons, we feel that our choice to consider the agile eye as a spherical joint mechanism to be used as a shoulder is the right design choice and it can also be included into other more complex systems.

5 CONCLUSION

This paper presented the complete development of a computed torque PD controller for the agile eye to simulate and characterise the performance of the robot when carrying a high inertia load on its end-effector compared to its usual implementation with a negligible load to validate the its use for the design of a robotic shoulder. Our simulation results illustrated (see Figure 5-7) the behaviour of the agile eye for point to point trajectory with negligible mass on its end-effector and showed already promising features for heavy load manipulations. Further results when moving a high inertia load suggest that the agile eye mechanism can be used as a spherical actuator in more complex systems with similar precision, slightly slower speed depending on design choices but requiring motors with power proportional to the expected load (see Figure 8-9). These obtained results as well as the detailed modelling of the spherical robot can serve as design guidelines for shoulder design of an anthropomorphic robotic arm.

REFERENCES

Antolini, M., Kose, O. & Gurocak, H., (2013), “Haptic device with spherical mr-brake for wrist rehabilitation”, *ASME 2013 International Design Engineering Technical Conferences and Computer and Information,* p. V06AT07A003–V06AT07A003.

Benon, L., (2000), *Étude expérimentale du contrôle en force/position pour un manipulateur robotique*, s.l.: Master's thesis, École Polytechnique Montréal.

Bonev, I., (2006), “Analytical determination of the workspace of symmetrical spherical parallel mechanisms”, *IEEE Transactions on Robotics,* 22(5), pp. 1011-1017.

Bonev, I., Chablat., D. & Wenger, P., (2006), “Working and assembly modes of the agile eye”, *Proceedings of the 2006 International Conference on Robotics and Automation,* pp. 2317-2322.

Bonev, I., Zlatanov, D. & Gosselin, C., (2003), “Singularity analysis of 30dof planar parallel mechanisms via screw theory”, *Journal of Mechanical Design,* 125(3), pp. 573-581.

Bowen, L., Baxter, W., Magleby, S. & Holwell, L., (2014), “A position analysis of coupled spherical mechanisms found in action origami”, *Mechanism and Machine Theory,* Volume 77, pp. 13-24.

Gosselin, C. & Hamel, J.-F., (1994), “The agile eye : a high-performance three-degree-of-freedom camera-orienting device”, *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pp. 781-786.

Gosselin, C., St-Pierre, E. & Gagne, M., (1996), “On the development of the agile eye”, *Robotics & Automation magazine,* 3(4), pp. 29-37.

Gunlop, G. & Jones, T., (1997), “Position analysis of a 3-dof parallel manipulator”, *Mechanism and Machine Theory,* 32(8), pp. 903-920.

Hamlin, G. & Sanderson, A., (1994), “A novel concentric multilink spherical joint with parallel robotics applications”, *Proceedings of the 1994 International Conference on Robotics and Automation,* pp. 1267-1272.

Hummel, S. & Chassapis, C., (2000), “Configuration design and optimization of universal joints with manufacturing tolerances”, *Mechanism and Machine Theory,* 35(3), pp. 463-476.

Kong, X. & Jin, Y., (2016), “Tyoe synthesis of 3-dof multi-mode translational/spherical parallel mechanisms with lockable joints”, *Mechanism and Machine Theory,* Volume 96, pp. 232-333.

Laitenberger, M., Raison, M., Périé, D. & Begon, M., (2015), “Refinement of the upper limb joint kinematics and dynamics using a subject-specific closed-loop forearm model”, *Multibody System Dynamics,* 33(4), pp. 413-438.

Lum, M., Rosen, J., Sinanan, M. & Hannaford, B., (2006), “Optimization of a spherical mechanism for a minimally invasive surgical robot: theoretical and experimental approaches”, *IEEE Transactions on Biomedical Engineering,* 53(7), pp. 1440-1445.

Mazzei, A., (1999), “Dynamic stability of a rotating shaft drivn through a universal joint”, *Journal of sound and vibration,* 222(1), pp. 19-47.

Pouliot, N. & Gosselin, C., (1998), “Motion simulation capabilities of three-degree-of-freedom flight simulators”, *Journal of Aircraft,* 35(1), pp. 9-17.

Sokolov, A. & Zirouchakis, P., (2007), “Dynamics analysis of a 3-dof parallel manipulator with r-p-s joint structure”, *Mechanism and Machine Theory,* 42(5), pp. 541-557.

Srivatsan, R., Bandyopadhyay, S. & Ghosal, A., (2013), “Analysis of the degrees-of-freedom of spatial parallel manipulators in regular and singular configurations”, *Mechanism and Machine Theory,* Volume 69, pp. 127-141.

Tanik, M., Parlaktas, V., Tanik, E. & Kadioglu, S., (2015), “Steel compliant cardan universal joint”, *Mechanism and Machine Theory,* Volume 92, pp. 171-183.

Tao, Z. & An, Q., (2013), “Interference analysis and workspace optimization of 3-rrr spherical parallel mechanism”, *Mechanism and Machine Theory,* Volume 69, pp. 62-72.

Taylor, J., (2005), *Classical Mechanics.* 1rst ed. s.l.:University Science books.

Yan, L. et al., (2006), “Torque modeling of spherical actuators with double-layer poles”, *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems,* pp. 5447-5452.

Zhang, G., Du, J. & To, S., (2014), “Study of the workspace of a class universal joints”, *Mechanism and Machine Theory,* Volume 73, pp. 244-258.