

# Adaptive Neural Network Dynamic Surface Control: An Evaluation on the Musculoskeletal Robot Anthrob

Michael Jäntsch\*, Steffen Wittmeier\*, Konstantinos Dalamagkidis\*, Guido Herrmann\*\* and Alois Knoll\*

**Abstract**—The soft robotics approach is widely considered to enable robots in the near future to leave their cages and move freely in our modern homes and manufacturing sites. Musculoskeletal robots are such soft robots which feature passively compliant actuation, while leveraging the advantages of tendon-driven systems. Even though these robots have been intensively researched within the last decade, high-performance feedback control laws have only very recently been developed. In [1], a controller was developed utilizing Dynamic Surface Control (DSC), an extension to *backstepping*, with an adaptive neural network compensator for joint as well as muscle friction. We compare these novel control strategies to Computed Force Control (CFC), an existing technique from the field of tendon-driven control, yielding highly improved trajectory tracking. The musculoskeletal robot Anthrob [2] serves as a benchmark.

**Keywords**—Compliant actuation, musculoskeletal robots, non-linear control, adaptive control, backstepping

## I. INTRODUCTION

Innovations of the last years make it ever more clear that robots are at the verge of leaving their cages, not only in manufacturing sites [3], but also to move into our modern homes as service robots. Today, even the most impressive soft and safe robots are still outperformed by humans in almost every aspect. The goal of mimicking the extremely smooth and effortless movements of humans could possibly be achieved by incorporating more of our inner mechanisms. Musculoskeletal robots feature compliant actuation and leverage the advantages of tendon-driven systems, where actuators can be placed with more freedom, to e. g. improve the weight distribution. This interesting field of research has in the last decade produced robots like Kenshiro [4] or the ECCES [5].

The robot arm Anthrob [2] has been developed to investigate provably stable control strategies for the class of musculoskeletal robots. It presents a replica of the human upper limb (see Fig. 1) and is actuated by eleven muscles which utilize DC motors to wind a tendon onto a spindle to realize muscle contraction. Each muscle is equipped with a Series Elastic Element. While eight muscles actuate the spherical shoulder and two the revolute elbow, there is an additional bi-articular muscle spanning both joints. Each muscle is equipped with the necessary sensors which include force sensors and actuator position sensors. This setup was developed to focus the challenges of musculoskeletal robot control onto a single well defined system. These include spherical joints, compliant and multi-articular muscles, as well as bio-inspired muscle routing which leads to collisions

\*These authors are with the chair for Robotics and Embedded Systems, Department of Informatics, Technische Universität München, Munich

\*\*This author is with University of Bristol, United Kingdom  
michael.jaentsch@in.tum.de

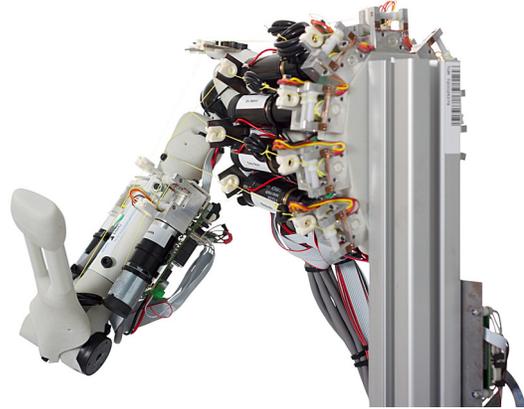


Fig. 1: Photograph of the robot Anthrob. [2]

between muscles and the skeleton and is therefore difficult to model.

Due to their complexity, musculoskeletal robots have so far failed to produce insights into possible control strategies. In this work, we compare the feedback control technique of Computed Force Control (CFC), which has been developed for tendon-driven robots without additional compliance and well defined muscle routing [6], [7], to a novel control approach based on Dynamic Surface Control (DSC), an extension to *backstepping*, which takes inspiration from [8] and was presented in [1]. The Anthrob serves as an ideal testbed for those control strategies, due to its sensorization and relatively simple skeletal kinematics.

## II. MODELING

A generic model for the class of musculoskeletal robots was presented in [1], proposing a subdivision into skeletal dynamics, muscle dynamics and muscle kinematics. The resulting submodels are presented in the following.

### A. Skeletal Dynamics

A model for the skeletal dynamics can be given by expressing the joint torque  $\tau$ :

$$\tau = H(\alpha)\ddot{q} + C(\alpha, \dot{q})\dot{q} + \tau_G(\alpha) + \tau_F(\dot{q}, \alpha, f) \quad (1)$$

where  $H(\alpha)$  is the mass matrix,  $C(\alpha, \dot{q})$  accounts for Coriolis and centrifugal effects,  $\tau_G(\alpha)$  is the vector of gravity terms, and  $\tau_F(\dot{q}, \alpha, f)$  is the vector of joint frictions. This friction component depends to a large degree on the joint velocities  $\dot{q}$ . Due to normal forces on the sliding surfaces inside the joint, it is additionally affected by the muscle forces  $f$  and the pose of the robot, given by  $\alpha$ .

In (1) the joint angles are expressed by  $\alpha$  which consists of the unit quaternion representation for parametrizing the

rotation of spherical joints and the angular representation for the other, e. g. revolute, joints. This parametrization of rotation was chosen, due to its lack of singularities [9]. The relationship between the derivative of  $\alpha$  and the rotational velocities  $\dot{q}$  can be defined such that:

$$\dot{\alpha} = A(\alpha) \cdot \dot{q} \quad (2)$$

where  $A(\alpha)$  is defined as a diagonal matrix in which each joint is represented by the mapping between the rotational velocity and the derivative of the angular representation. This mapping is equal to 1 for revolute joints or the following relation for a quaternion  $Q$  [10]:

$$\dot{Q} = \frac{1}{2} \cdot \begin{bmatrix} -\epsilon_1 & -\epsilon_2 & -\epsilon_3 \\ \eta & -\epsilon_3 & \epsilon_2 \\ \epsilon_3 & \eta & -\epsilon_1 \\ -\epsilon_2 & \epsilon_1 & \eta \end{bmatrix} \cdot \dot{q} \quad (3)$$

where  $\eta$  denotes the real part and  $\vec{\epsilon} = [\epsilon_1 \ \epsilon_2 \ \epsilon_3]^T$  the three dimensional imaginary part of the quaternion. Note that this definition assumes that joint velocities are expressed in the coordinate frame of the moving part of the joint, i. e. the distal rigid body.

### B. Muscle Kinematics

The muscle kinematics describe the force transmission of the muscles and is hence a geometric representation of how muscles interact with the skeleton. This can be given by the muscle Jacobian  $L(q)$  which is defined as the partial derivative of the muscle lengths  $l$  with respect to the joint angles and can be transformed into the force-torque domain by the principle of virtual work<sup>1</sup> [11]:

$$L(q) = \frac{\partial l}{\partial q} \quad \longleftrightarrow \quad \tau = -L^T(q) \cdot f \quad (4)$$

In the presence of spherical joints this definition can be extended by introducing (2), such that [1]:

$$L(\alpha) = \frac{\partial l}{\partial \alpha} \cdot \frac{\partial \alpha}{\partial q} = \frac{\partial l}{\partial \alpha} \cdot A(\alpha) \quad (5)$$

Hence, it is possible to obtain the muscle Jacobian by first acquiring the function of the muscle lengths with respect to the joint angles and subsequent differentiation. However, analytic solutions are not always feasible, especially when relationships become highly complex, due to collisions between muscles and skeleton in certain poses. Approximation of the muscle Jacobian through Artificial Neural Networks (ANNs) has been extensively studied by Schmalzer [12]. In this work, samples of joint angles and corresponding muscle lengths are generated and presented offline to an ANN, posing a supervised learning problem. The resulting ANN approximation is differentiated with respect to the joint angles, using the difference quotient to obtain the muscle Jacobian.

This method was evaluated on the spherical shoulder joint of the Anthrob and the eight muscles spanning it. Samples of corresponding joint angles, actuator positions, and forces

<sup>1</sup>The minus sign arises from the definition of a positive force being associated with muscle shortening.

were generated by utilizing a muscle force controller to maintain a force of 2 N. The difficulty here was to cover the full working range of the joint. Subsequently, samples were removed, where measured forces were outside of a threshold of  $\pm 1$  N from the reference. Due to this boundary, spring expansions could be neglected in calculating the muscle lengths from the actuator positions. This led to a total of 42208 samples which were split into a training set and a validation set and used to train a Multilayer Perceptron (MLP) with 30 hidden neurons in a single hidden layer. Validation errors as low as  $3.46 \times 10^{-4}$  m (for the shoulder) show that the muscle length function was approximated with sufficient accuracy. The same method was utilized to obtain the muscle length functions for the elbow joint and the bi-articular Biceps, with similar results.

In addition to friction in the joints, musculoskeletal robots can exhibit noteworthy friction effects within the transmission system of the tendons. The effects on the control performance depend largely on the position of the muscle force sensor. The force is either measured at the driving or at the load end. In the latter case, the force measured is equal to the force acting on the skeleton and is therefore the preferred scheme from a control point of view. From an integration point of view, however, it might be preferable to implement an actuator unit which is fully sensorized, including the force sensor. In the Anthrob, force sensors are always located on the driving end [2]. In this case, (1) may be rewritten to include the muscle friction  $f_F(f, \dot{q}, \alpha)$ , by introducing (4):

$$\begin{aligned} H(\alpha)\ddot{q} + C(\alpha, \dot{q})\dot{q} + \tau_G(\alpha) + \tau_F(\dot{q}, \alpha, f) = \\ = -L^T(\alpha)[f - f_F(f, \dot{q}, \alpha)] \end{aligned} \quad (6)$$

The muscle friction component depends on the transmitted force, the angle under which the tendon is routed and the velocity at which the tendon is running over the routing point.

### C. Muscle Dynamics

The muscle dynamics of a musculoskeletal robot with electromagnetic actuators are governed by the actuator dynamics, which can be derived by instantiating a model of the DC motor and gearbox, and a model for the muscle compliance. The latter is represented by a possibly non-linear spring stiffness  $K(f)$  as follows, by introducing (4):

$$\begin{aligned} \dot{f} &= K(f)(\dot{l} + r_s \dot{\theta}) \\ &= K(f) \left[ L(\alpha)\dot{q} + r_s \dot{\theta} \right] \\ &= f_1(\alpha, \dot{q}, f) + g_1(f)\dot{\theta} \end{aligned} \quad (7)$$

where the change in the spring expansion is expressed as the change in the muscle length  $l$  plus the change in the tendon length, given by the actuator velocity  $\dot{\theta}$  times the radius of the spindle winding up the tendon  $r_s$ . The actuator dynamics are of the following form:

$$\ddot{\theta} = f_2(f, \dot{\theta}) + g_2 v_A \quad (8)$$

where the system state is given by the actuator velocity, the system input is the actuator voltage  $v_A$  and  $f_2$ , and  $g_2$  are defined as possibly non-linear functions [1]. More elaborate

models include the motor current as a system state which is neglected in this work.

### III. COMPUTED FORCE CONTROL

Most tendon-driven controllers use feedback linearization by linearizing (6) and introducing some linear control law [7], [13]. Hence, a tendon force  $f_d$  is computed for a given movement and subsequently controlled by a low-level force controller. A trajectory tracking form can be given [14]:

$$-L^T f_d = H [\ddot{q}_d + D(\dot{q}_d - \dot{q}) + P \cdot \Delta q] + C\dot{q} + \tau_G \quad (9)$$

where  $\Delta q$  is the angular error used for control, as explained later. In tendon-driven robots, there are typically more actuators than joints. We call the number of skeletal Degrees of Freedom (DoF)  $n$  and the number of muscles  $m$ . Hence there is a redundancy ( $m > n + 1$ ) and a null-space exists in the muscle forces that does not affect joint motion. This can be resolved in several different ways, where the most common one is to solve a quadratic program of the following form to obtain the muscle forces [14]:

$$\min_{f_d} \|f_d\|^2 \quad \text{s.t.} \begin{cases} -L^T(\alpha) \cdot f_d = \tau_d \\ f_d \geq f_{\min} \end{cases} \quad (10)$$

where the objective function minimizes the internal forces. Other objective functions, e.g. minimum energy, may be used instead. This method is an extension of computed torque control [9] towards tendon-driven robots and hence neglects the dynamics of low-level force controllers, including the elasticity. When applied to musculoskeletal robots, the dynamics of the low-level control are governed by passive compliance and are therefore not negligible.

### IV. DYNAMIC SURFACE CONTROL

In robotics, several methods of non-linear control have been utilized to obtain provably stable control laws. For flexible-joint robots, especially the method of passivity based control has been used [15]. However it does not provide a systematic approach to synthesizing control laws. The method of *backstepping* gives a very specific synthetization procedure, if the system model can be reformulated to form a *strict-feedback* system. For an exact definition of *strict-feedback* and the *backstepping* method, the reader is referred to [16]. In this method, the state of the second subsystem  $x_1$  is assumed as a virtual control input to the first subsystem, finding an associated control law  $\phi_1$ . Subsequently, the next system needs to be controlled such that  $x_1$  converges to  $\phi_1$ , assuming the state of the third subsystem  $x_2$  as a virtual control signal, iteratively yielding a set of control laws  $\{\phi_1, \dots, \phi_N\}$  that take the full system dynamics of depth  $N$  into account. However, due to the fact that each subsystem comprises an integration, the *backstepping* procedure requires the time derivative of the previous control law. This accounts for the interaction dynamics between control laws on each depth, but creates an explosion of complexity, due to continuous differentiation. For this purpose Swaroop et. al. [17] developed Dynamic Surface Control (DSC) which

introduces a first order low-pass filter in between each level control to realize stable numeric differentiation. Each of these filters is applied to the control signal  $\phi_i$  such that:

$$\mu_i \dot{s}_i + s_i = \phi_i \quad i = \{1, \dots, N-1\} \quad (11)$$

where  $\mu_i$  is the filter constant and  $s_i$  the filter output. For the system considered here, there are three subsystems. Hence, the skeletal dynamics (6) are controlled first, while we step back through the muscle force dynamics (7) and ultimately through the actuator dynamics (8).

Oh and Lee [18] designed a non-linear controller for trajectory tracking of flexible-joint robots which can be extended for musculoskeletal robots. For a detailed derivation as well as a proof of stability the reader is referred to [1]. In the following, this controller is introduced shortly:

$$\phi_1 = -L^{T+} \{H(\ddot{q}_d + \Lambda_1(\dot{q}_d - \dot{q})) + C(\dot{q}_d + \Lambda_1 \Delta q) + \tau_G - K_d r + \hat{\tau}_F\} + \hat{f}_F \quad (12)$$

while  $\hat{\tau}_F$  and  $\hat{f}_F$  denote estimations of the unknown joint and muscle friction terms, respectively,  $\Lambda_1$  is a positive definite control gain, and  $L^{T+}$  denotes the pseudo inverse of the muscle Jacobian, which is associated to the result of an algorithm that resolves muscle redundancy (see Section III). The error for control is denoted by  $\Delta q$ . It is defined as the  $n \times 1$  vector of errors in each of the joints, which is simply the difference for revolute joints, and defined as follows for a single spherical joint:

$$\Delta q = \eta \bar{e}_d - \eta_d \bar{e} - S(\bar{e}_d) \bar{e} \quad (13)$$

where  $S(\cdot)$  denotes the skew symmetric operator. This definition arises from the stability analysis of quaternion control. In conjunction with a tracking error  $r$  such that:

$$r = \dot{q} - \dot{q}_d - \Lambda_1 \Delta q$$

exponential convergence of  $\alpha$  to  $\alpha_d$  can be proven, while  $\|r\| < \epsilon_r$  for a small positive scalar  $\epsilon_r$  implies bounded tracking [19].

It has been shown that friction terms can be well approximated by Radial Basis Function Networks (RBFNs) which are defined as a vector of RBFs  $\Phi$  and a vector of weights  $\Theta$  such that  $\phi = \Theta^T \Phi$  [20]. Due to the fact that  $\tau_F \in \mathbb{R}^n$  and  $f_F \in \mathbb{R}^m$ , the two compensators are defined as vectors of RBFNs, where  $\phi_{JFj}$  and  $\phi_{MFj}$  denote the  $j^{\text{th}}$  entry of each of these vectors, respectively. A general definition for each RBF can be given as follows:

$$\Phi_{ijk} = \exp - \frac{(x_{ij} - c_{ik})^T \cdot (x_{ij} - c_{ik})}{\sigma_{ik}^2} \quad (14)$$

where  $i \in \{JF, MF\}$  and  $j \in \{1, \dots, o\}$  with  $o$  as the number of RBFNs per friction compensator which is  $n$  for the joint friction and  $m$  for the muscle friction. Finally  $k \in \{1, \dots, l\}$ , where  $l$  denotes the number of neurons. Each RBF is parametrized by a center  $c_{ik}$ , a standard deviation  $\sigma_{ik}$ , and has  $x_{ij}$  as an input. The  $l \times 1$  vector of neurons  $\Phi_{ij}$  is pre-multiplied with the  $1 \times l$  vector of weights  $\hat{\Theta}_{ij}^T$ :

$$\hat{\tau}_{Fj} = \hat{\Theta}_{JFj}^T \Phi_{JFj} \quad (15)$$

$$\hat{f}_{Fj} = \hat{\Theta}_{MFj}^T \Phi_{MFj} \quad (16)$$

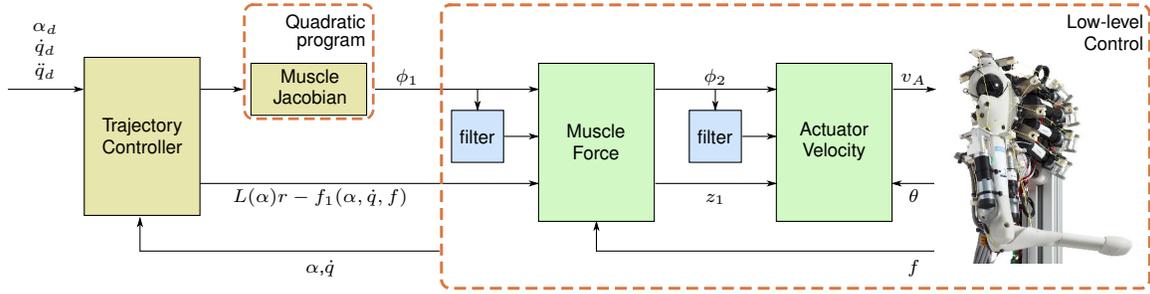


Fig. 2: Block Diagram of DSC. The controller consists of three levels: the (i) controller for the skeletal dynamics, comprising the trajectory controller and the muscle Jacobian, the (ii) muscle force and the (iii) actuator velocity controllers.

while the RBF parameters of each neuron are static, the weights are used to adapt the output of the RBFN. A learning rule of the following form is given [21]:

$$\dot{\hat{\Theta}}_{ij} = -\Gamma_{ij} \cdot \Phi_{ij} \cdot r_{ij} - \Sigma_i \cdot \hat{\Theta}_{ij} \quad (17)$$

where  $\Gamma_{ij}$  is the respective learning factor and  $\Sigma_i$  can be seen as a robustifying damping term on the weights. This is necessary, as omitting it may cause network weights to grow indefinitely, due to small perturbations [21]. The driving value for the adaptation rule is the error  $r_i$  which is defined as follows:

$$r_{JF} = r \quad (18)$$

$$r_{MF} = -L \cdot r \quad (19)$$

It has been shown that an ANN of the given form can approximate any continuous function, depending on the number of neurons and the parameters of the RBFs, up to an arbitrarily small bounded error [22].

The methods of *backstepping* and DSC yield low-level muscle controllers which utilize system models (7) and (8) such that [1]:

$$\phi_2 = g_1^{-1} [\dot{s}_1 + Lr - \Lambda_2(f - s_1) - f_1] \quad (20)$$

$$\phi_3 = g_2^{-1} [\dot{s}_2 - g_1(f - s_1) - \Lambda_3(\dot{\theta} - s_2) - f_2] \quad (21)$$

This two-staged control law takes the filtered reference of the high-level controller  $s_1$  as an input and computes an actuator voltage  $v_A = \phi_3$  (see also Fig. 2). In conjunction with the trajectory tracking controller, given in (12), uniformly ultimately bounded stability can be proven for this control scheme, as long as control parameters adhere to certain bounds [1].

## V. RESULTS

The control law developed in Section IV, with and without adaptive friction compensators, was compared to the method of CFC from Section III. All approaches utilized skeletal dynamics, which were derived from the CAD model, the learned muscle Jacobian, and the nominal actuator characteristics. Both, CFC as well as DSC were implemented as separate high-level trajectory tracking control laws and low-level muscle controllers. However, CFC utilized (9) and a state space tendon force controller, while the novel DSC control law implemented the controllers (12), (20) and (21).

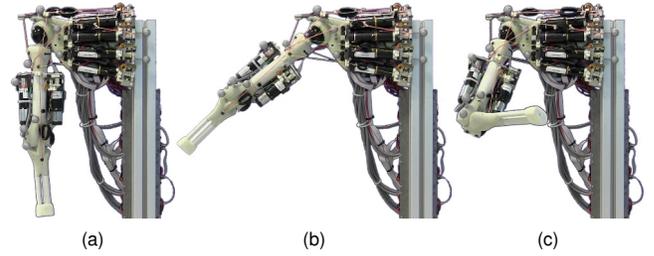


Fig. 3: A reference trajectory was generated by interpolation between three poses which were (a) the shoulder adduction, (b) the shoulder abduction and (c) the shoulder anteversion. In a second experiment, the elbow joint additionally performed a flexion from (b) to (c).

Separation into a high and a low-level controller yields the possibility of distributed muscle control. In this case only the result of  $\phi_1$  and a feedforward part was needed for computing  $\phi_2$ , eliminating the need for cross dependency between muscle controllers. This greatly improves scalability, as muscle controllers can be executed with only a minimum of communication between a central controller and the distributed nodes. Distributed control was facilitated by Anthrob through a control architecture which implemented distributed control units for low-level muscle control and communication via CAN bus to a central PC [2]. To remain comparable throughout the experiments, high-level control was executed at a frequency of 200 Hz and low-level control at 1 kHz.

Joint angles were measured by an analog potentiometer in the elbow joint and a stereo camera system with infrared markers on the fixed scapula and the humerus to obtain the state of the spherical shoulder joint [23]. Even though this motion capture system featured comparably low latencies of 4.3 ms in average, the frame rate was limited to 60 fps which complicated the numerical differentiation of the joint angle to obtain joint velocities. For this purpose a high-gain observer [16] was utilized to improve the quality of both the angular position measurements, as well as the angular velocities. This observer utilizes the non-linear system model from (6) to estimate the states, based on system inputs and measurements of the system output. The system states  $\alpha$  and  $\dot{q}$  are redefined to  $x_1$  and  $x_2$ , respectively, where the first is equal to the measurement from the camera system. Hence,

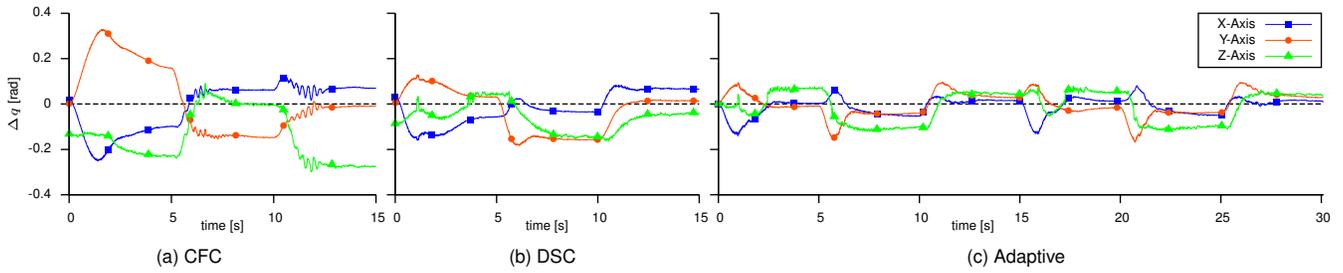


Fig. 4: The joint angle error  $\Delta q$  is depicted over the trajectory for three different control scenarios.

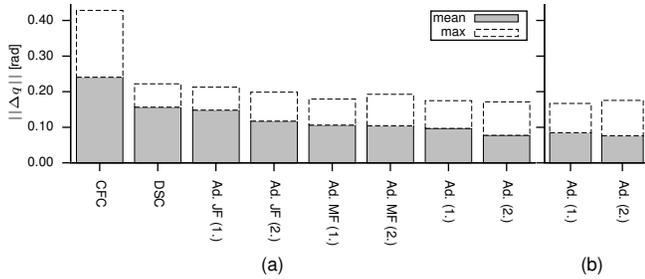


Fig. 5: Joint angle errors  $\|\Delta q\|$  are depicted, both for mean and maximum values. Results are shown for (a) only the shoulder joint and (b) the full Anthrob. Controllers include CFC, DSC and the adaptive (Ad.) controllers with only joint (JF), only muscle (MF) and finally both friction compensation terms, while the adaptive controllers are executed twice (cycle (1.) and (2.)) to show convergence.

the high-gain observer was defined as follows to serve as an estimator for  $\alpha$  and  $\dot{q}$  [16]:

$$\begin{aligned} \hat{x}_1 &= A(\hat{x}_1) [\hat{x}_2 + h_1 \cdot \Delta q] \\ \hat{x}_2 &= \gamma_0(x_1, x_2, f) + h_2 \cdot \Delta q \end{aligned} \quad (22)$$

where  $\hat{x}_{1/2}$  denote estimates of the respective system states and  $\gamma_0$  the nominal system model. The angle error  $\Delta q$  according to (13) was utilized with the positive definite gains  $h_1$  and  $h_2$  to estimate the system states. Since the controller utilizes the same model of the system dynamics to compensate for any non-linearities, the result of  $\gamma_0$  is essentially equal to the reference acceleration  $\ddot{q}_d$ .

First, the spherical shoulder joint of the Anthrob with the eight muscles of the rotator cuff was considered, subsequently moving on to the full arm, including the elbow joint and the bi-articular Biceps.

### A. Spherical joint control

A reference trajectory was obtained by choosing three poses and subsequently interpolating. The resulting trajectory was therefore described by reference accelerations, velocities and angles. The three different poses (see Fig. 3) were (a) the resting position, which presents the shoulder adduction, (b) an abduction (extended to the side) of 0.9 rad ( $\sim 51.6^\circ$ ), and (c) an anteversion (drawn to the front) of 0.67 rad ( $\sim 38.4^\circ$ ). This trajectory was chosen, as it presents substantial coverage of the arm's movement capabilities. It was executed in several experiments by the control laws obtained

by the techniques of CFC and DSC and finally different combinations of adaptive controllers.

The CFC controller resulted in large trajectory deviations, as well as steady state errors (see Fig. 4a), and even oscillations during the transition from the abduction to the anteversion ( $\sim 5$  to 7 s) and from the anteversion to the adduction ( $\sim 10$  to 12 s). This is a consequence of neglecting the low-level controller dynamics which are comparably slow due to the muscle compliance. The simple muscle force controller is not able to track the reference force, resulting in the undesired oscillations. Both steady state as well as overall tracking errors were reduced by DSC and oscillations were fully eliminated (see Fig. 4b). This was due to acquiring a set of control laws for the complete system dynamics, including the muscle compliance.

Performance of both developed adaptive terms was evaluated by conducting three experiments, starting with (i) the joint friction compensation as stated in (15) to (ii) the muscle friction compensation as stated in (16) and finally going to (iii) the combination of both. The overall tracking error was reduced for any of the three experiments, while the combination of both friction compensators performed best (see Fig. 4c). Here, the steady state offset for the abduction was reduced to 0.07 rad ( $\sim 4.1^\circ$ ). The highest steady state error remained for the anteversion which amounted to 0.12 rad ( $\sim 6.8^\circ$ ). A comparison, displaying the tracking errors for the different control laws, is shown in Fig. 5a. It can be seen that both maximum as well as mean tracking errors were significantly reduced by utilizing DSC and even further by the adaptive friction terms. For any of the adaptive control experiments, the mean absolute trajectory tracking error was reduced for the second iteration.

Fig. 4c shows clearly that steady state errors are mostly in the Z-Axis (rotation of the arm around itself). From the placement of the muscles we know that this rotation is the most difficult to achieve. We assume that the anteversion, which shows the largest steady state offset, requires more effort to achieve, as humans facilitates this pose by an additional rotation of the scapula which is not present in the Anthrob.

### B. Anthrob control

Both DSC control laws, with and without adaptive friction compensation were evaluated for controlling the full Anthrob, including the elbow joint and the bi-articular Biceps. A comparison was not possible, as we were not even able

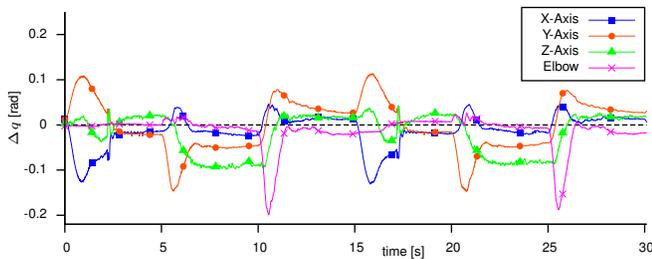


Fig. 6: For adaptive control of the full Anthrob, the three dimensional quaternion error and the elbow joint error are shown.

to stabilize CFC when the elbow was attached. Here, the same trajectory as for the shoulder was utilized, while an elbow flexion was executed during the transition between the abduction and the anteversion (see Fig. 3). The resting pose of the elbow was set to  $0.2\text{rad}$  ( $\sim 11.45^\circ$ ) and the flexed pose to  $0.5\pi$  ( $90^\circ$ ). The elbow extension was performed during the shoulder adduction (see Fig. 6). It can be seen that accurate tracking was achieved for both joints, with similar performance as for the pure shoulder control and negligible steady state errors for the elbow joint.

In the complimentary video, some minor oscillations can still be observed during trajectory tracking. These can be reduced by increasing the frequency of the angle measurements in the shoulder joint ( $> 60\text{Hz}$ ) and are not inherent to the control scheme.

## VI. CONCLUSIONS

A novel control strategy for the class of musculoskeletal robots, based on *backstepping* and DSC, was evaluated on the robot arm Anthrob. This robot focusses the challenges of musculoskeletal robot control in a single well defined system. While the bio-inspired muscle routing was addressed by learning the muscle kinematics through ANNs, it could be shown that muscle compliance, as well as bi-articular muscles and spherical joints, could be controlled with improved performance, compared to CFC. Control performance was further improved by introducing adaptive friction compensation terms, both for joint, as well as muscle friction.

Interesting challenges for future work include the investigation of online learning, both for the muscle kinematics, as well as the skeletal dynamics. The tracking error contains information about the system, which can be stimulated by motor commands to learn the system dynamics. However, robust techniques have to be found to not damage the robot during learning. Another interesting direction is the extension of the existing control framework towards operational space control, to possibly resolve muscle and joint space redundancy in a single optimization step.

## VII. ACKNOWLEDGMENTS

The research leading to these results has received funding from the European Community's Seventh Framework Programme FP7/2007-2013 - Challenge 2 - Cognitive Systems, Interaction, Robotics - under grant agreement no. 288219 - MYROBOTICS.

## REFERENCES

- [1] M. Jäntsch, S. Wittmeier, K. Dalamagkidis, G. Herrmann, and A. Knoll, "Adaptive Neural Network Dynamics Surface Control for Musculoskeletal Robots," in *Proc. IEEE Conference on Decision and Control*, 2014.
- [2] M. Jäntsch, S. Wittmeier, K. Dalamagkidis, A. Panos, F. Volkart, and A. Knoll, "Anthrob - A Printed Anthropomorphic Robot," in *Proc. IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2013.
- [3] A. Albu-Schäffer, O. Eiberger, M. Grebenstein, S. Haddadin, C. Ott, T. Wimbock, S. Wolf, and G. Hirzinger, "Soft robotics," *IEEE Robotics & Automation Magazine*, vol. 15, no. 3, pp. 20–30, 2008.
- [4] T. Kozuki, H. Mizoguchi, Y. Asano, M. Osada, T. Shirai, U. Junichi, Y. Nakanishi, K. Okada, and M. Inaba, "Design Methodology for the Thorax and Shoulder of Human Mimetic Musculoskeletal Humanoid Kenshiro -A Thorax structure with Rib like Surface -," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems IROS*, 2012, pp. 3687–3692.
- [5] S. Wittmeier, C. Alessandro, N. Bascarevic, K. Dalamagkidis, D. Devereux, A. Diamond, M. Jäntsch, K. Jovanovic, R. Knight, and H. G. Marques, "Towards Anthropomorphic Robotics: Development, Simulation, and Control of a Musculoskeletal Torso," *Artificial Life*, vol. 19, no. 1, pp. 171–193, 2013.
- [6] S. C. Jacobsen, H. Ko, E. K. Iversen, and C. C. Davis, "Antagonistic control of a tendon driven manipulator," in *Proc. IEEE International Conference on Robotics and Automation ICRA*, 1989, pp. 1334–1339.
- [7] J. K. Salisbury and J. J. Craig, "Articulated Hands - Force Control And Kinematic Issues." *The International Journal of Robotics Research*, vol. 1, no. 1, pp. 4–17, Mar. 1982.
- [8] J. Na, X. Ren, G. Herrmann, and Z. Qiao, "Adaptive neural dynamic surface control for servo systems with unknown dead-zone," *Control Engineering Practice*, vol. 19, no. 11, pp. 1328–1343, Nov. 2011.
- [9] B. Siciliano and O. Khatib, *Springer Handbook of Robotics*. Springer New York, 2007.
- [10] A. L. Schwab and J. P. Meijaard, "How to draw Euler angles and utilize Euler parameters," in *Proceedings of IDETC/CIE*, 2008.
- [11] V. M. Zatsiorsky, *Kinetics of Human Motion*. Human Kinetics Publishers, 2002.
- [12] C. Schmalzer, "Extracting the Muscle Jacobian for Anthropomorphic Robot Control using Machine Learning," Master thesis, Technische Universität München, 2011.
- [13] M. E. Abdallah, R. Platt, and C. W. Wampler, "Decoupled torque control of tendon-driven fingers with tension management," *The International Journal of Robotics Research*, vol. 32, no. 2, pp. 247–258, Feb. 2013.
- [14] D. G. Thelen, F. C. Anderson, and S. L. Delp, "Generating dynamic simulations of movement using computed muscle control," *Journal of Biomechanics*, vol. 36, no. 3, pp. 321–328, 2003.
- [15] A. Albu-Schäffer, C. Ott, and G. Hirzinger, "A Unified Passivity-based Control Framework for Position, Torque and Impedance Control of Flexible Joint Robots," *International Journal of Robotics Research*, vol. 26, no. 1, pp. 23–39, 2007.
- [16] H. K. Khalil, *Nonlinear systems*, 3rd ed. Prentice Hall New Jersey, 2002, vol. 3.
- [17] D. Swaroop, J. K. Hedrick, P. P. Yip, and J. C. Gerdes, "Dynamic surface control for a class of nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 45, no. 10, pp. 1893–1899, 2000.
- [18] J. H. Oh and J. S. Lee, "Control of flexible joint robot system by backstepping design approach," in *Proc. IEEE International Conference on Robotics and Automation ICRA*, vol. 4, Apr. 1997, pp. 3435–3440.
- [19] O.-E. Fjellstad and T. I. Fossen, "Quaternion feedback regulation of underwater vehicles," in *Proc. IEEE Conference on Control Applications*, Aug. 1994, pp. 857–862.
- [20] S. S. Ge, T. H. Lee, and S. X. Ren, "Adaptive friction compensation of servo mechanisms," *International Journal of Systems Science*, vol. 32, no. 4, pp. 523–532, Jan. 2001.
- [21] S. S. Ge, T. H. Lee, and C. J. Harris, *Adaptive neural network control of robotic manipulators*. World Scientific Publishing Company, 1998, vol. 19.
- [22] T. Poggio and F. Girosi, "A theory of networks for approximation and learning," MIT Artificial Intelligence Laboratory, Cambridge, MA, USA, Tech. Rep., 1989.
- [23] S. Wittmeier, A. Gaschler, M. Jäntsch, K. Dalamagkidis, and A. Knoll, "Calibration of a physics-based model of an anthropomorphic robot using evolution strategies," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems IROS*, Oct. 2012, pp. 445–450.