

Relaxed Individual Control of Skeletal Muscle Forces via Physical Human-robot Interaction

William Gallagher · Ming Ding · Jun Ueda

Received: date / Accepted: date

Abstract This study develops a relaxed formulation of a method for controlling individual muscle forces using exoskeleton robots. Past studies have developed a muscle-force control method with very strict limitations on the conditions. These conditions will be removed, and the problem will be reformulated as a constrained optimization of several parameters. The optimization algorithm recognizes when a solution to the muscle control problem cannot be exactly realized, and finds the solution that minimizes the mean errors of the individual muscles between expected and desired muscle activation. This is demonstrated in a computer simulation of human arm dynamics and compared against the prior method to demonstrate its wider applicability. In addition, the control method is extended to resolve issues associated with a non-ideal exoskeleton with incomplete torque application to the joints. A quasi-optimized motor-task that minimizes the errors in target muscles and non-target muscles can be obtained. This paper presents theoretical analysis, simulation, and experimental results on the performance of the relaxed individual muscle control.

Keywords exoskeleton · motor task · muscle isolation · physical human-robot interaction · musculoskeletal model

1 Introduction

Wearable exoskeleton robots, which can apply torques directly to the wearer's joints, are capable of either assisting the motion of the wearer, by applying torques provide which aid those of their muscles, or resisting the motion of the wearer, by applying torques that oppose those of their muscles. This makes them an ideal platform for designing systems for the purposes of exercise and rehabilitation of human muscles or diagnosis of muscle related

William Gallagher and Jun Ueda
Bio-Robotics and Human Modeling Laboratory, Georgia Institute of Technology 771 Ferst Drive, Atlanta, Georgia, 30332-0405, USA, Tel: 404-385-3900, Fax: 404-894-8496,
E-mail: bgallagher@gatech.edu, E-mail: jun.ueda@me.gatech.edu

Ming Ding
RIKEN-TRI Collaboration Center for Human-Interactive Robot Research, Advanced Science Institute, RIKEN, 2271-130, Anagahora, Shimoshidami Moriyama-ku, Nagoya, Aichi 463-0003, Japan,
E-mail: mingding@naogya.riken.jp

disorders due to their ability to apply torques directly to human joints [9, 13, 16, 18, 21, 22]. In these tasks, it is often desired to induce certain target muscles to generate a desired force level. This is useful in exercise or rehabilitation to strengthen the muscle, and for diagnostic cases to induce some expected muscle activation pattern and observe any deviation from it. However, targeting a specific muscle or group of muscles can be difficult, due to the large number of muscles compared to the relatively small number of joints where torque may be applied.

Several studies have presented methods for predicting the activation pattern of human muscles under a known applied loading [3, 8, 15]. These are usually based on the Optimality Principle in the neuromotor system, which was developed from empirical data on how the human neuromuscular system optimizes the activation pattern of muscles. This muscle force prediction allows a mathematical formulation that has been reliably demonstrated in several studies [6, 8, 10, 17].

With a mathematical formulation for a muscle activation pattern for a given applied loading, it should be possible to perform the inverse calculation and find the required applied loading to achieve a certain activation pattern for some target muscles. To accomplish this objective, the mechanisms of redundant muscle coordination during human-robot physical interaction need to be rigorously formulated and understood. Such a method has been developed by [7, 19–21], wherein it is possible to provide target forces for a certain subset of arm muscles.

The previous method could exactly achieve the desired target muscle forces without inducing activity in otherwise inactive muscles. Thus, the algorithm was able to calculate the necessary joint torques to be applied by an exoskeleton. However, this method had several key limitations. It included several feasibility criteria that had to be met in order for the muscle activation pattern to be physically feasible [21]. These criteria required that target muscle forces be achieved exactly, which may not always be possible or necessary. For many exercise or rehabilitation situations, approximate forces may be acceptable. In addition, it required that inactive muscles be maintained as inactive, another requirement that may not be strictly necessary. Finally, the formulation did not address the upper limit of muscle forces, requiring only that the calculated forces be non-negative. In reality, using such a system on a person could result in excess applied force, causing muscle strain. These limitations were accepted to simplify the resultant formulation. This study will relax these requirements and reframe the feasibility criteria as optimization problems to achieve the target muscle pattern with as little error as possible, as well as account for the upper limit of muscle forces.

Furthermore, an ideal exoskeleton should be able to apply force to all joints of the wearer, but practical difficulties in the design of exoskeletons leads to non-ideal devices that may be able assist/resist movement of certain joints. For example, the shoulder joint is more complicated than the elbow joint in terms of the number of degrees of freedom. The application of torques to the torso and the neck joints is almost impossible, due to the complex structure of the vertebral column and limitations of wearable designs. Inadequate application of joint torques of such a non-ideal exoskeleton may result in errors in individual muscle forces. Similar problems can widely be observed in existing exoskeletons, nearly all of which are non-ideal in this fashion: the wearer's joints have more degrees-of-freedom (DOF) than the device can control. The hip joint has three DOF. However, some leg exoskeletons only control movements in the frontal plane. Therefore, robustness to the lack of actuators needs to be examined and incorporated with the computational algorithm for practical use. Instead of finding a solution that perfectly realizes a muscle activation pattern in targeted muscles, a quasi-optimized solution that minimizes errors in individual muscles is obtained.

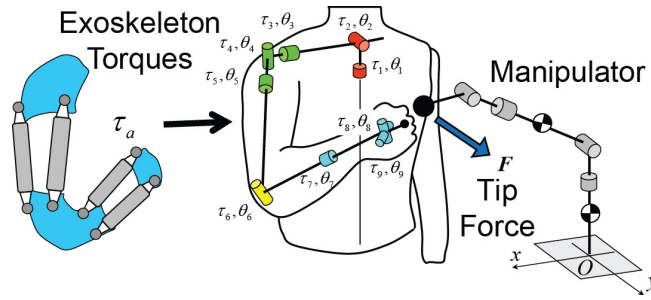
2 Problem Formulation

2.1 System Equation

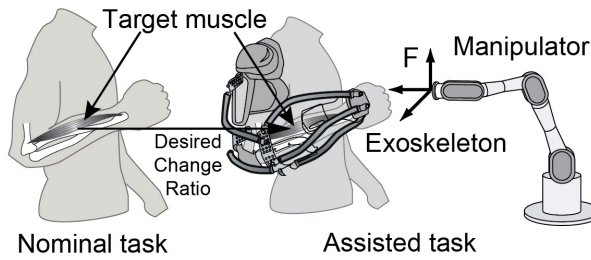
The methods explained in this article will consider the individual muscle force control for static tasks, where it is assumed that a subject does not change his or her posture during a task and all muscle contractions are isometric. The dynamics of the body and robot are neglected. Figure 1(a) shows a schematic diagram where a human operator is performing a motor task, which requires the human to exert their muscles to accomplish some target force or motion, while interacting with a robotic manipulator and wearing an exoskeleton device. The robot is capable of applying force to the human's hand in three dimensions, while the exoskeleton is assumed to be ideal and designed such that it is capable of applying torques to all nine of the human's arm joints. The goal, as shown by Figure 1(b), is to modulate the muscle forces applied by the human through these forces and torques applied by the manipulator and exoskeleton. For a human musculoskeletal model that has M joints and N muscles, the net torques in the human joints is given by (1).

$$\tau_h = g(\theta) + J(\theta)^T F - \tau_a \quad (1)$$

Here, $\tau_h \in \mathfrak{R}^M$ is a vector of human joint torques, $\theta = [\theta_1, \dots, \theta_M]^T \in \mathfrak{R}^M$ is a vector of joint angles, $F = [F_x, F_y, F_z]^T$ is the translational force at the tip (i.e., the force the human applies to the environment or the force applied by the manipulator to the human), $J(\theta)$ is



(a) Human operator performs a motor task against a manipulator while wearing an exoskeleton



(b) Desired muscle activation pattern is expected to be induced in the operator's muscles of interest

Fig. 1 Concept of individual muscle control via physical human-robot interaction

the Jacobian between the joints and end-point, $\mathbf{g}(\boldsymbol{\theta})$ is the gravity force, and $\boldsymbol{\tau}_a \in \mathfrak{R}^M$ is the joint torque applied by the exoskeleton. The static equation of the musculoskeletal system is then given by (2). In order to achieve the desired muscle control, the muscle forces should be induced such that the joint torques are those applied externally by (1).

$$\boldsymbol{\tau}_h = \mathbf{A}(\boldsymbol{\theta})\mathbf{f} = \begin{bmatrix} a_{11} & \cdots & a_{1N} \\ \vdots & \ddots & \vdots \\ a_{M1} & \cdots & a_{MN} \end{bmatrix} \begin{bmatrix} f_1 \\ \vdots \\ f_N \end{bmatrix} \quad (2)$$

In this relation, $\mathbf{A} \in \mathfrak{R}^{M \times N}$ is the moment-arm matrix of the muscles and $\mathbf{f} = [f_1, \dots, f_N]^T \in \mathfrak{R}^N$ is the human muscle force vector. The element a_{ij} of \mathbf{A} represents the moment arm of muscle j for joint i , so $a_{ij} = 0$ is given if f_j does not affect joint i . Note $f_j \geq 0$ ($j = 1, \dots, N$) because muscle contraction cannot generate negative forces. $\mathbf{g}(\boldsymbol{\theta})$, $\mathbf{J}(\boldsymbol{\theta})$, and $\mathbf{A}(\boldsymbol{\theta})$ for a given posture $\boldsymbol{\theta}$ can be calculated by the musculoskeletal model developed in [19–21]. This model of the upper-right limb with 51 muscles has been developed to analyze the physical interaction between the exoskeleton and the wearer at the level of individual muscles. It model consists of 5 rigid links and 9 joints. For more detail, see [21]. To simplify the problem, no upper-bound is given for $\boldsymbol{\tau}_a$.

2.2 Optimization Principle in the Neuromuscular System and Muscle Force Prediction

The human body has a much larger number of muscles than the number of joints, i.e., $N \gg M$, leading to muscle redundancy. This fact makes the prediction of muscle forces \mathbf{f} by knowing joint torques $\boldsymbol{\tau}_h$ an ill-posed problem. Various optimization approaches have been proposed to model the Optimization Principle in the neuromuscular system [2, 3, 14, 15] and to solve this problem by minimizing a cost function. The main difference among the approaches is the structure of cost functions that represent performance criteria on which the neuromuscular system optimizes the activation of muscle forces. In the literature that deals with isometric or relatively slow motions, the cost functions have a general form comprised of the sum of muscular stress or force raised to a power. The static optimization method can be formulated as follows.

$$\text{Minimize } u(\mathbf{f}) = \sum_{j=1}^N c_j f_j^r \quad (3)$$

$$\text{subject to } \begin{cases} \boldsymbol{\tau}_h = \mathbf{A}\mathbf{f} \\ 0 \leq f_j \leq f_{\max_j} (j = 1, \dots, N) \end{cases} \quad (4)$$

The cost function $u(\mathbf{f})$ is determined by the weighting factors c_j 's, which are determined based on the physical characteristics of each muscle, and the integer power r . It should be noted that arguments still exist on the choice of the weighting factors c_j and the integer r of the power [2, 3, 14, 15]. There are still arguments and criticism of the neurological background of this muscle force prediction; however, this approach has been implemented in publicly or commercially available software [4, 5]. The effectiveness of this approach for predicting stereotyped motor performances has been reported in the literature [6, 8, 10].

2.3 Individual Muscle Control

Individual muscle control requires solving the inverse problem of the aforementioned muscle force prediction in (3) to control N muscles by adjusting M joint torque inputs. Note that this inverse solution is not straightforward since any induced muscle activation pattern must satisfy the physiology-based optimality criterion. The first condition in (4) represents the balance of joint torques. The second condition means that each muscle can only “pull”, exerting positive contracting force limited by its maximum voluntary force. Mathematically, this can be treated as finding an adequate equality constraint for the joint torques such that desired muscle forces are obtained as a result of the cost function minimization. As described earlier, the exoskeleton merely modifies human joint torques, which is equivalent to the modification of the first equality condition in (4).

Let \mathbf{f}_0 be the nominal muscle forces when a subject is performing a nominal motor task. The human force vector \mathbf{f}_0 may be permuted according to three groups of muscles: target muscles, which will be controlled to achieve a desired force level; non-target muscles, which are active but not directly controlled; and inactive muscles, which provide no force. Let $\tilde{N} \leq N$ be the number of the active muscles, and $N - \tilde{N}$ be the number of the inactive muscles. The active muscles have elements with nonzero values in \mathbf{f}_0 , and the inactive muscles have zeros. The active muscle group consists of target muscles $\mathbf{f}_t \in \mathfrak{R}^{N_t}$ and non-target muscles $\mathbf{f}_n \in \mathfrak{R}^{N_n}$ where $N_t + N_n = \tilde{N}$. Without the loss of generality, the order of the N muscles may be permuted according to these three groups for the simplicity of description.

$$\mathbf{f} \triangleq \begin{bmatrix} \mathbf{f}_t \\ \mathbf{f}_n \\ \mathbf{0} \end{bmatrix} \begin{array}{l} \cdots \text{target muscles} \\ \cdots \text{non-target muscles} \\ \cdots \text{inactive muscles} \end{array} \quad (5)$$

The above permutation for \mathbf{f} is also applied to the moment-arm matrix \mathbf{A} accordingly:

$$\mathbf{A}^T = \begin{bmatrix} \mathbf{A}_t \\ \mathbf{A}_n \\ \mathbf{A}_v \end{bmatrix} \begin{array}{l} \cdots \text{target muscles} \\ \cdots \text{non-target muscles} \\ \cdots \text{inactive muscles} \end{array} \quad (6)$$

Hereafter these permuted vectors and matrices will be used.

The desired target muscle forces \mathbf{f}_{td} are given as follows by explicitly specifying the ratio of change for each of the target muscles:

$$\mathbf{f}_{td} = \text{diag}[\gamma_1, \gamma_2, \dots, \gamma_{N_t}] \mathbf{f}_{t0} \quad (7)$$

where $\gamma_j (> 0)$ is the ratio of change of the j -th target muscle. The subscript d denotes the desired muscle forces, and 0 denotes the nominal muscle forces. Let the total external torque $\boldsymbol{\tau}_{ex} \in \mathfrak{R}^M$ be

$$\boldsymbol{\tau}_{ex} = \mathbf{J}^T \mathbf{F} - \boldsymbol{\tau}_a. \quad (8)$$

Equation (8) shows how the joint toques can be adjusted by applying $\boldsymbol{\tau}_a$ from the exoskeleton and by exerting \mathbf{F} at the tip either by performing a motor task or through a robot manipulator. Note that there is a certain freedom in choosing \mathbf{F} and $\boldsymbol{\tau}_a$ to achieve a given $\boldsymbol{\tau}_{ex}$. This will be discussed in a later section. The problem to realize \mathbf{f}_{td} can be formulated as follows.

Perfect Individual Muscle Control:

Find \mathbf{F} and $\boldsymbol{\tau}_a$ that will achieve a $\boldsymbol{\tau}_{ex}$ such that the solution of (3) includes a desired muscle activation pattern for the target muscles, \mathbf{f}_{td} , i.e.,

$$[\mathbf{f}_{td}^T, \mathbf{f}_{nd}^T, \mathbf{0}^T]^T = \underset{\mathbf{f}(\tau_{ex})}{\operatorname{argmin}} u(\mathbf{f}) \quad (9)$$

with minimum changes in non-target muscle forces, i.e., $|\mathbf{f}_{nd} - \mathbf{f}_{n0}| \rightarrow \min$.

2.4 Solution for Perfect Muscle Force Control

The previous work by Ueda, Ming, et al, [7, 19–21] provided a solution to exactly realize the desired target muscles forces, which will be referred to as perfect muscle control. Performing the minimization in (9) to find a solution to (2) results in the external torque τ_{ex} for the perfect control given by

$$\tau_{ex} = [\mathbf{A}_t^T \ \mathbf{A}_n^T] w^{-1} \left(\begin{bmatrix} \mathbf{A}_t \\ \mathbf{A}_n \end{bmatrix} \boldsymbol{\alpha} \right) \quad (10)$$

where $\boldsymbol{\alpha}$ is the control input vector and represents in a moment arm independent manner the change in torques that must be applied to achieve the desired muscle activation pattern. The transformation $w(\ast)$ is a function that converts the muscle force vector \mathbf{f} to a new vector \mathbf{q} as $\mathbf{q} = w(\mathbf{f})$ where the j -th elements of \mathbf{f} and \mathbf{q} are given as

$$q_j \triangleq \frac{\partial u(\mathbf{f})}{\partial f_j} = rc_j f_j^{r-1}, (j = 1, \dots, N), \quad (11)$$

and $w^{-1}(\mathbf{q}) = \mathbf{f}$ is the inverse function of $w(\ast)$ and the c_j 's are as in (3). \mathbf{q} can be permuted in a similar manner to \mathbf{f} :

$$\mathbf{q} = \begin{bmatrix} \mathbf{q}_t \\ \mathbf{q}_n \\ \mathbf{q}_v \end{bmatrix} \begin{matrix} \cdots \text{target muscles} \\ \cdots \text{non-target muscles} \\ \cdots \text{inactive muscles} \end{matrix} \quad (12)$$

The control input vector $\boldsymbol{\alpha}$ is the main element in the solution and is given as

$$\boldsymbol{\alpha} = \mathbf{A}_t^+ [w(\mathbf{f}_{td}) - w(\mathbf{f}_{t0})] + (\mathbf{I} - \mathbf{A}_t^+ \mathbf{A}_t) \boldsymbol{\beta}, \quad (13)$$

where \mathbf{I} is the identity matrix, and $\boldsymbol{\beta}$ is a free parameter that represents the remaining redundancy for controlling the non-target muscles as the second priority. This solution holds regardless of the choice of the parameter r in (3) and (11), which was verified in [15] (See Appendix for further details). In [21], the existence of this solution relied on three criteria:

1. The target muscles forces must all be linearly independent of one another:

$$\operatorname{rank}(\mathbf{A}_t) = \operatorname{rank}([\mathbf{A}_t \ w(\mathbf{f}_{td}) - w(\mathbf{f}_{t0})]) \quad (14)$$

2. The inactive muscles must be maintained as exactly inactive:

$$-\mathbf{A}_v \begin{bmatrix} \mathbf{A}_t \\ \mathbf{A}_n \end{bmatrix}^+ \begin{bmatrix} \mathbf{q}_{t0} \\ \mathbf{q}_{n0} \end{bmatrix} - \mathbf{A}_v \boldsymbol{\alpha} > 0 \quad (15)$$

3. The non-target active muscles must be maintained as active:

$$\mathbf{A}_n \boldsymbol{\alpha} + w(\mathbf{f}_{n0}) > 0 \quad (16)$$

Due to space limitation, only the final criteria are given here. Please see [21] for detailed derivation. The method attempted to find a solution by minimizing the change in non-target active muscles while satisfying the three criteria. If this was not possible, then no solution existed that could exactly realize the target muscles forces within physiological constraints. This resulted in a very narrow range of feasibility that could be practically expanded by relaxing the required assumptions.

In addition, it was assumed that the exoskeleton is ideal, and therefore has means to adjust all of the joints, i.e., τ_h is *fully* modifiable by τ_{ex} , and a combination of F and τ_a always exists that realizes a given τ_{ex} via the physical human-robot interaction.

4. The exoskeleton is *ideal*, and therefore τ_a exists (as well as F in the case where a manipulator arm is also used) and can realize τ_{ex} in (8).

This paper shows theoretical results that relax these requirements and reframe the feasibility criteria to achieve the target muscle pattern with as little error as possible.

3 Relaxing the Feasibility Conditions

3.1 Problem Reformulation

The above limitations on the use of the perfect muscle control solution motivated generalizing the solution to eliminate assumptions and make the control scheme more widely applicable. To do so, it is necessary to examine the three key assumptions, criteria 1-3 discussed in Section 2.4.

First, the solution ignores the upper bound on an individual muscle's force. While mathematically sound, this assumption has serious physiological consequences. Should an exoskeleton attempt to apply the resultant torques, and one or more muscles are required to supply more force than physically possible, the wearer of the device could sustain serious injury. Accounting for this limit requires adding the right side of the inequality condition, $f_j \leq f_{\max j} (j = 1, \dots, N)$, into the Karush-Kuhn-Tucker (KKT) equation, which will result in a second set of λ 's. For clarity, the λ 's in the initial solution, which correspond to the lower bound $0 \leq f_j$, will be denoted as λ_l , while the λ 's corresponding to the upper bound will be denoted as λ_u .

The second assumption is that it is necessary and possible to match the desired target muscle forces exactly. In reality, it is never possible to exactly match all muscle forces, and it is likely not absolutely necessary, either. Therefore, instead of requiring $f_j = f_{jd}$, a small error tolerance e_j is allowed such that $f_j = f_{jd} + e_j$. When performing the transformation $\mathbf{q} = w(\mathbf{f})$, this becomes $\mathbf{q}_t = \mathbf{q}_{td} + \boldsymbol{\epsilon}_t$, where $\boldsymbol{\epsilon}_t$ is the allowable error tolerance in \mathbf{q}_t and takes the same units as \mathbf{q} .

Finally, the third assumption states that inactive muscles must stay inactive, and non-target active muscles must stay active. Physically, this may be difficult to realize. Therefore, in the interest of finding a control scheme that is as widely application as possible, it is desired to allow initially active non-target muscles to become inactive, while allowing initially inactive muscles to generate a small amount of force. The former requires including the Lagrange multipliers in the KKT condition as with the generalization applied for the first assumption. The latter requires the introduction of an error tolerance term, $\boldsymbol{\epsilon}_v$, as with the generalization for the second assumption.

3.1.1 New KKT Condition

Combining the generalizations for all three assumptions, a new feasibility equation is obtained from the KKT condition [1, 12]. For convenience, the inequality constraint $0 \leq f_j \leq f_{\max j}$ ($j = 1, \dots, N$) was split into two conditions, with the lower and upper bounds corresponding to g_{jl} and g_{ju} , respectively. The updated KKT condition is shown in (17) and is subject to the conditions in (18), (19), and (20).

$$\nabla u(\mathbf{f}) + \sum_{i=1}^M \mu_i \nabla h_i(\mathbf{f}) + \sum_{j=1}^N (\lambda_{jl} \nabla g_{jl}(\mathbf{f}) + \lambda_{ju} \nabla g_{ju}(\mathbf{f})) = 0 \quad (17)$$

$$h_i(\mathbf{f}) = 0 \quad (18)$$

$$\lambda_{jl} g_{jl}(\mathbf{f}) = 0, \quad \lambda_{jl} \geq 0, \quad g_{jl}(\mathbf{f}) \leq 0 \quad (19)$$

$$\lambda_{ju} g_{ju}(\mathbf{f}) = 0, \quad \lambda_{ju} \geq 0, \quad g_{ju}(\mathbf{f}) \leq 0 \quad (20)$$

$$\text{where } (i = 1, \dots, M), \quad (j = 1, \dots, N)$$

The constraints h_i , g_{jl} , and g_{ju} are given by (21), (22), and (23).

$$h_i(\mathbf{f}) = \tau_i - \mathbf{a}_i^T \mathbf{f} \quad (21)$$

$$g_{jl}(\mathbf{f}) = -f_j \quad (22)$$

$$g_{ju}(\mathbf{f}) = f_j - f_{\max j} \quad (23)$$

The derivatives of the three constraints are then given by (24), (25), and (26).

$$\frac{\partial h_i}{\partial f_j} = -a_{ij} \quad (24)$$

$$\frac{\partial g_{j_1 l}}{\partial f_{j_2}} = \begin{cases} -1, & j_1 = j_2 \\ 0, & j_1 \neq j_2 \end{cases} \quad (25)$$

$$\frac{\partial g_{j_1 u}}{\partial f_{j_2}} = \begin{cases} 1, & j_1 = j_2 \\ 0, & j_1 \neq j_2 \end{cases} \quad (26)$$

These, combined with the definition of \mathbf{q} in (11), which remains unchanged, give the updated equation given by (27).

$$\mathbf{q} = w(\mathbf{f}) = \mathbf{A}^T \boldsymbol{\mu} + \boldsymbol{\lambda} \quad (27)$$

$$\text{where } \mathbf{q} = \begin{bmatrix} q_1 \\ \vdots \\ q_N \end{bmatrix} \quad \boldsymbol{\mu} = \begin{bmatrix} \mu_1 \\ \vdots \\ \mu_M \end{bmatrix}$$

$$\boldsymbol{\lambda} = \boldsymbol{\lambda}_l - \boldsymbol{\lambda}_u = \begin{bmatrix} \lambda_{1l} - \lambda_{1u} \\ \vdots \\ \lambda_{Nl} - \lambda_{Nu} \end{bmatrix}$$

Permutating this equation as done for (12), but without making any simplifications or assumptions, results in (28).

$$\begin{bmatrix} \mathbf{q}_t \\ \mathbf{q}_n \\ \mathbf{q}_v \end{bmatrix} = \begin{bmatrix} \mathbf{A}_t \\ \mathbf{A}_n \\ \mathbf{A}_v \end{bmatrix} \boldsymbol{\mu} + \begin{bmatrix} \lambda_{tl} - \lambda_{tu} \\ \lambda_{nl} - \lambda_{nu} \\ \lambda_{vl} - \lambda_{vu} \end{bmatrix} \quad (28)$$

It is possible to make a few simplifications regarding the inequality constraint multipliers. Target muscles will necessarily always be active, and therefore will supply a non-zero force, so the lower constraint is always satisfied ($\lambda_{tl} = \mathbf{0}$). In addition, inactive muscles by definition supply zero force, so the upper constraint is satisfied ($\mathbf{q}_v = 0$ and $\lambda_{vu} = \mathbf{0}$). This yields the result in (29).

$$\begin{bmatrix} \mathbf{q}_t \\ \mathbf{q}_n \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_t \\ \mathbf{A}_n \\ \mathbf{A}_v \end{bmatrix} \boldsymbol{\mu} + \begin{bmatrix} -\lambda_{tu} \\ \lambda_{nl} - \lambda_{nu} \\ \lambda_{vl} \end{bmatrix} \quad (29)$$

The result can be evaluated for both the nominal muscle forces, \mathbf{f}_0 , and the desired muscle forces, \mathbf{f}_d . For the nominal case, by definition, the active non-target muscles will have a positive force, so the lower bound is automatically satisfied ($\lambda_{0nl} = \mathbf{0}$), giving (30). For the desired case, it can be assumed that the desired values of the target muscles will never be the maximum force for safety reasons, while there is no assumption that the non-target active muscles will stay active. Also, the allowable error tolerance terms, $\boldsymbol{\epsilon}$, are incorporated, which gives (31).

$$\begin{bmatrix} \mathbf{q}_{0t} \\ \mathbf{q}_{0n} \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_t \\ \mathbf{A}_n \\ \mathbf{A}_v \end{bmatrix} \boldsymbol{\mu}_0 + \begin{bmatrix} -\lambda_{0tu} \\ -\lambda_{0nu} \\ \lambda_{0vl} \end{bmatrix} \quad (30)$$

$$\begin{bmatrix} \mathbf{q}_{dt} + \boldsymbol{\epsilon}_t \\ \mathbf{q}_{dn} \\ \boldsymbol{\epsilon}_v \end{bmatrix} = \begin{bmatrix} \mathbf{A}_t \\ \mathbf{A}_n \\ \mathbf{A}_v \end{bmatrix} \boldsymbol{\mu}_d + \begin{bmatrix} \mathbf{0} \\ \lambda_{dnl} - \lambda_{dnu} \\ \lambda_{dvl} \end{bmatrix} \quad (31)$$

3.1.2 Muscle Control

Using the generalized KKT conditions given in (30) and (31), it is possible to derive a control law that can be used to find the value of $\boldsymbol{\tau}_h$ that will induce the desired muscle activation pattern. Since $\boldsymbol{\mu}$ corresponds to the vector of human joint torques, it is the ideal quantity to control, as finding an appropriate value for $\boldsymbol{\mu}_d$ will lead to finding $\boldsymbol{\tau}_{ex}$, and subsequently \mathbf{F} and $\boldsymbol{\tau}_a$, to realize the desired muscle forces. Therefore, let $\boldsymbol{\mu}_d = \boldsymbol{\mu}_0 + \boldsymbol{\alpha}$, where $\boldsymbol{\alpha}$ is the control input vector, similar to that presented in (13) for the case of perfect muscle control. Through the derivation below, this leads to (32).

$$\begin{aligned} \mathbf{q}_d &= \mathbf{A}^T (\boldsymbol{\mu}_0 + \boldsymbol{\alpha}) + \boldsymbol{\lambda}_d \\ &= \mathbf{A}^T \boldsymbol{\mu}_0 + \mathbf{A}^T \boldsymbol{\alpha} + \boldsymbol{\lambda}_d \\ &= \mathbf{A}^T \boldsymbol{\mu}_0 + \boldsymbol{\lambda}_0 + \mathbf{A}^T \boldsymbol{\alpha} + \boldsymbol{\lambda}_d - \boldsymbol{\lambda}_0 \\ &= \mathbf{q}_0 + \mathbf{A}^T \boldsymbol{\alpha} + \boldsymbol{\lambda}_d - \boldsymbol{\lambda}_0 \\ \mathbf{q}_d - \mathbf{q}_0 &= \mathbf{A}^T \boldsymbol{\alpha} + \boldsymbol{\lambda}_d - \boldsymbol{\lambda}_0 \end{aligned}$$

$$\begin{bmatrix} \mathbf{q}_{dt} + \boldsymbol{\epsilon}_t \\ \mathbf{q}_{dn} \\ \boldsymbol{\epsilon}_v \end{bmatrix} - \begin{bmatrix} \mathbf{q}_{0t} \\ \mathbf{q}_{0n} \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_t \\ \mathbf{A}_n \\ \mathbf{A}_v \end{bmatrix} \boldsymbol{\alpha} + \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\lambda}_{dnl} - \boldsymbol{\lambda}_{dnu} \\ \boldsymbol{\lambda}_{dvl} \end{bmatrix} - \begin{bmatrix} -\boldsymbol{\lambda}_{0tu} \\ -\boldsymbol{\lambda}_{0nu} \\ \boldsymbol{\lambda}_{0vl} \end{bmatrix} \quad (32)$$

The unknown to be solved for in (32) is the control input for the joint torques, $\boldsymbol{\alpha}$. The \mathbf{q}_0 's are known from calculating the nominal case and \mathbf{q}_{dt} is known from the desired muscle activation pattern, while \mathbf{A}^T is known from the geometry and configuration of the muscles. This leaves \mathbf{q}_{dn} , both $\boldsymbol{\epsilon}$'s, and all $\boldsymbol{\lambda}$'s as parameters that can be chosen to make the solution feasible. There are so many parameters due to the large difference between the number of muscles, N , and the number of joints, M ($N \gg M$).

Examining (32), it can be seen that each term has a physical meaning. The left hand side, from (11), gives the desired change in muscle forces. The first term on the right hand side provides the effect of the control input on the muscles, since \mathbf{A}^T gives the mapping from joint torques to muscles forces, while the remaining terms account for any muscles which are exactly equal to zero or the muscle's maximum force. It is important to note that the inequality constraint multiplier vectors, $\boldsymbol{\lambda}$'s, will tend to be sparse. The elements of these are only non-zero if that particular muscle is exactly on the constraint multiplied by that $\boldsymbol{\lambda}$. For example, if non-target muscle j is providing maximum force in the nominal case, but is between zero and maximum in the desired case, then $\lambda_{j0nu} > 0$, $\lambda_{jdnu} = 0$, and $\lambda_{jdnl} = 0$. In general, many of the entries in these vectors will be zero.

3.1.3 Solution

The goal of this algorithm is to match the desired target muscle forces as exactly as possible. Since the number of outputs (muscle forces), N , is significantly larger than the number of inputs (joint torques), M , a priority based approach is used to find a solution. Therefore, the solution for $\boldsymbol{\alpha}$ can be found from the first row of (32): $\mathbf{q}_{dt} + \boldsymbol{\epsilon}_t - \mathbf{q}_{0t} = \mathbf{A}_t \boldsymbol{\alpha} + \boldsymbol{\lambda}_{0tu}$. Rearranging slightly, this becomes (33).

$$\mathbf{q}_{dt} - \mathbf{q}_{0t} + \boldsymbol{\epsilon}_t - \boldsymbol{\lambda}_{0tu} = \mathbf{A}_t \boldsymbol{\alpha} \quad (33)$$

Assuming the solution exists (the existence of a solution is discussed in Section 3.1.4), it is given by (34), which is analogous to (13) for the perfect control case.

$$\boldsymbol{\alpha} = \mathbf{A}_t^+ (\mathbf{q}_{dt} - \mathbf{q}_{0t} + \boldsymbol{\epsilon}_t - \boldsymbol{\lambda}_{0tu}) + (\mathbf{I} - \mathbf{A}_t^+ \mathbf{A}_t) \boldsymbol{\beta} \quad (34)$$

This new solution is applied to (10) to obtain external torques. The free parameter $\boldsymbol{\beta}$ indicates the redundancy of the solution, and has an effect on the values of the remaining non-target (active and inactive) muscles. The choice of $\boldsymbol{\beta}$ is generally arbitrary, and will be discussed further with the choice of the other parameters in Section 3.1.4.

3.1.4 Feasibility and Choice of Parameters

Due to the way in which the matrices were permuted, each row of (32) deals with a different set of muscles. The first row is concerned with the target muscles, the second row with the active (in the nominal case) non-target muscles, and the last row with the inactive (again, in the nominal case) non-target muscles. Therefore, each row can be extracted from the matrix equation to determine a set of feasibility criteria for the existence of the solution discussed in Section 3.1.3.

The first row, previously extracted to get (33), is the most important, since it deals with the target muscles. In order to solve (33) using linear algebra, a matrix composed of A_t augmented with the left hand side of the equation must have the same number of linearly independent rows as A_t itself. This is concisely expressed in (35).

$$\text{rank}(A_t) = \text{rank}\left(\begin{bmatrix} A_t & (\mathbf{q}_{dt} - \mathbf{q}_{0t} + \boldsymbol{\epsilon}_t - \lambda_{0tu}) \end{bmatrix}\right) \quad (35)$$

When this condition is satisfied, it is possible to obtain the desired target muscle forces while also satisfying the bounds on muscle force. The error tolerance $\boldsymbol{\epsilon}_t$ is a parameter that can be chosen to make this condition hold when it is not possible to exactly realize the target forces. In this case, a small value can be given to some or all elements of $\boldsymbol{\epsilon}_t$ to make an approximation of the desired \mathbf{q}_{dt} realizable. The multiplier λ_{0tu} is known from the nominal muscle distribution, and is only non-zero for muscles that are initially at their maximum force.

The next priority in choosing the solution is to maintain inactive muscles as inactive (or keep them minimally active if this is not possible). The third row of (32), given by (36), provides the necessary condition for this.

$$\lambda_{dvl} - \boldsymbol{\epsilon}_v = \lambda_{0vl} - A_v \boldsymbol{\alpha} \quad (36)$$

For inactive muscles to remain inactive, λ_{dvl} must be positive and $\boldsymbol{\epsilon}_v$ will be zero, requiring the right hand side of the equation to be greater than zero. If, however, it is not possible (or not necessary) to meet this exactly, then elements of $\boldsymbol{\epsilon}_v$ can be chosen to be nonzero, leading to the condition given by (37).

$$A_v \boldsymbol{\alpha} + \lambda_{0vl} > -\boldsymbol{\epsilon}_v \quad (37)$$

The final priority is to minimize the change in the forces of the non-target muscles. The second row of (32) can be written as in (38).

$$\mathbf{q}_{dn} - \mathbf{q}_{0n} = A_n \boldsymbol{\alpha} + \lambda_{dnl} - \lambda_{dnu} + \lambda_{0nu} \quad (38)$$

In the ideal case, none of the non-target muscles are at maximum force, and it is desired to keep them from becoming inactive, so all the λ_n 's are zero. For this to be the case, (39) must be true, where $\mathbf{q}_{\max} = w(\mathbf{f}_{\max})$.

$$0 < A_n \boldsymbol{\alpha} + \mathbf{q}_{0n} < \mathbf{q}_{\max} \quad (39)$$

Substituting the solution for $\boldsymbol{\alpha}$ from (34) into (38), then making the same decisions that lead to (39), the only remaining free parameter is β . Therefore, a reasonable way to choose β is to avoid a change in the non-target muscles, or to minimize $\|\mathbf{q}_{dn} - \mathbf{q}_{0n}\|$, where $\mathbf{q}_{dn} - \mathbf{q}_{0n}$ is given by (40).

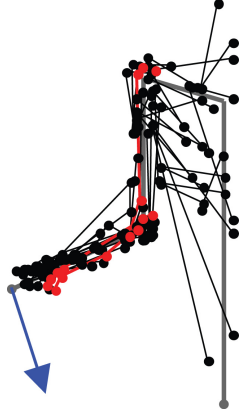


Fig. 2 Nominal task, $F = [10, 10, 10]$

$$\begin{aligned}
 \mathbf{q}_{dn} - \mathbf{q}_{0n} &= \mathbf{A}_n [\mathbf{A}_t^+ (\mathbf{q}_{dt} - \mathbf{q}_{0t} + \boldsymbol{\epsilon}_t - \boldsymbol{\lambda}_{0tu}) + \\
 &\quad (\mathbf{I} - \mathbf{A}_t^+ \mathbf{A}_t) \boldsymbol{\beta}] \\
 &= \mathbf{A}_n \mathbf{A}_t^+ (\mathbf{q}_{dt} - \mathbf{q}_{0t} + \boldsymbol{\epsilon}_t - \boldsymbol{\lambda}_{0tu}) - \\
 &\quad [-\mathbf{A}_n (\mathbf{I} - \mathbf{A}_t^+ \mathbf{A}_t)] \boldsymbol{\beta}
 \end{aligned} \tag{40}$$

In general, this is consistent with the condition given by (39). However, due to the need to use the pseudoinverse of \mathbf{A} (since \mathbf{A} is not square), it is possible for numerical inaccuracies to arise. In this case, if the condition is not satisfied, then one or more non-target muscles is at the upper or lower limit. To maintain feasibility, the appropriate elements of $\boldsymbol{\lambda}_{dmu}$ and $\boldsymbol{\lambda}_{dnl}$ must be chosen such that the condition is again satisfied (Note that $\boldsymbol{\lambda}_{0nu}$ is known from the nominal muscle distribution).

3.2 Validation

The relaxed formulation was validated and compared to the perfect muscle control solution using a computer model. The model, developed in prior work by Ueda, Ming, et al, [19–21], simulated the muscles of the human upper arm using a total of 9 joints ($M = 9$) from the torso to wrist joint with 51 muscles ($N = 51$). The Crowninshield’s cost function (see Appendix) is applied for the optimization criterion in the model. The joint torques an exoskeleton should apply for a variety of target muscle activation patterns, given in Table Table 1, were calculated using both the perfect control method and the relaxed control method, for comparison. The feasibility of each pattern for both methods is shown in Table Table 2. For the relaxed method solution, the norm of the error tolerances, $\boldsymbol{\epsilon}_t$ and $\boldsymbol{\epsilon}_v$, are presented normalized by the nominal muscle forces, \mathbf{q}_0 . For all three patterns, the arm is in held straight down at the side of the body with the elbow bent 90° and the forearm held out, as in Figure 2, with an external applied force $[F_x, F_y, F_z] = [10, 10, 0]$ for the nominal task.

To demonstrate that the relaxed method includes the solutions the perfect method could find, Figure 3(a) shows the results of both methods for pattern 1. Both solutions are identical, showing that a problem solvable by the perfect control method would yield the same

Table 1 Desired Muscle Activation Patterns for Testing

| # | Muscle 1 | Ratio | Muscle 2 | Ratio |
|---|-----------------|-------|------------------------|-------|
| 1 | Brachioradialis | x 0.8 | Flexor Carpi Ulnaris | x 1.2 |
| 2 | Brachioradialis | x 1.7 | Flexor Carpi Ulnaris | x 0.8 |
| 3 | Brachioradialis | x 1.5 | Extensor Carpi Ulnaris | x 1.5 |

Table 2 Desired Muscle Activation Pattern Feasibility Results

| # | Perfect | Relaxed | Error tolerance |
|---|----------------------|---------|-----------------|
| 1 | Pass | Pass | 0.0 |
| 2 | Fail (Cond. 2, (15)) | Pass | 0.12 |
| 3 | Fail (Cond. 3, (16)) | Pass | 0.05 |

solution using the relaxed. For patterns 2 and 3, however, the perfect method could not find a physiologically feasible solution, but the relaxed method could. Figure 3(b) shows the resulting muscle forces for some of the muscles returned by the relaxed method for target pattern 2. The relaxed method was able to find a feasible solution and match the target muscles by allowing inactive muscles to become active. For pattern 3, the results are shown in Figure 3(c). A feasible solution was found by not matching the target muscles exactly. The muscle forces are presented normalized by q_0 .

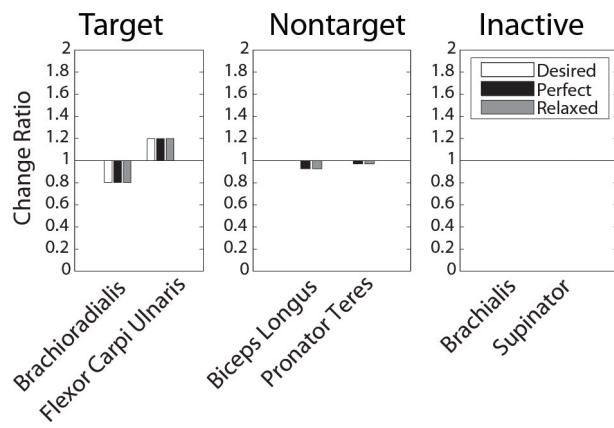
Figure 4 explores the feasibility of combinations between the muscles used in the three patterns for both the perfect and relaxed methods. The hatched areas in both plots represent the feasibility region of the perfect control method. Outside this region, only the relaxed control method was able to find a solution. The shading of the plot indicates the required magnitude of the error tolerance terms, ϵ_t and ϵ_v , normalized by the nominal muscle force, q_0 . The relaxed method demonstrated a much larger region in which a feasible solution could be found than the perfect method, with large regions requiring only small allowable errors. In both cases, however, as the limits of the muscle forces were approached, large allowable errors were required to find a feasible solution.

4 Solutions for a Non-Ideal Exoskeleton

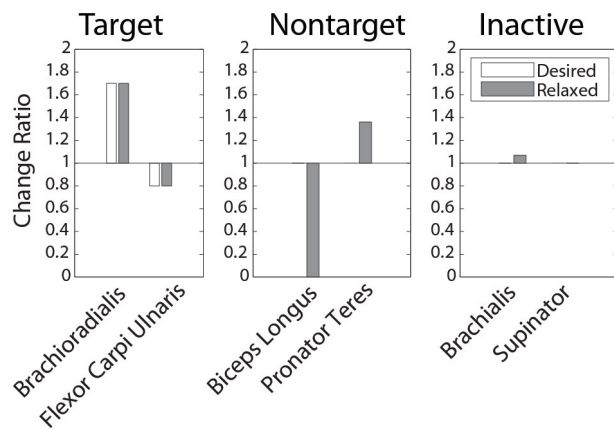
4.1 Limitation in Adjusting Joint Torques

The problem in realizing desired joint torques with incomplete torque application to the joints (i.e., with lack of means to apply torques to certain human joints) is considered. Criterion 4 discussed in Section 2.4 is relaxed. Instead of finding a solution that perfectly realizes a muscle activation pattern in targeted muscles, a quasi-optimized solution that minimizes errors in individual muscles is obtained.

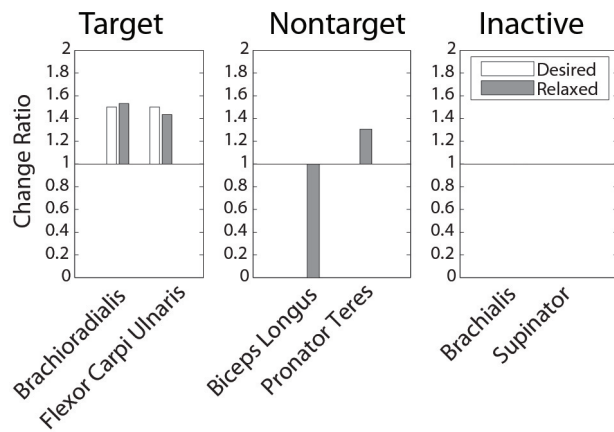
Consider the individual muscle control shown in Figure 1(a) where the exoskeleton is *not* capable of controlling all of the joints torques involved in a task due to a lack of means to apply torques. Even if f_{id} is realizable and a solution of τ_{ex} exists mathematically, a combination of F and τ_a that satisfies (8) may not exist. In other words, τ_{ex} may not be realizable due to the inability to control all of the joint torques. Such use of a *non-ideal* exoskeleton may result in errors in target muscle forces. Recall that F is projected onto the torque space through J^T . The number of elements of F is three (i.e. F_x, F_y, F_z), but that of τ_{ex} is much larger than three in general. Therefore, the control of the tip-force via a manipulator arm F could mitigate the issue but would not completely resolve the problem. Note that



(a) Target pattern 1



(b) Target pattern 2, error tolerance of 0.12



(c) Target pattern 3, error tolerance of 0.05

Fig. 3 Simulation results of relaxed method

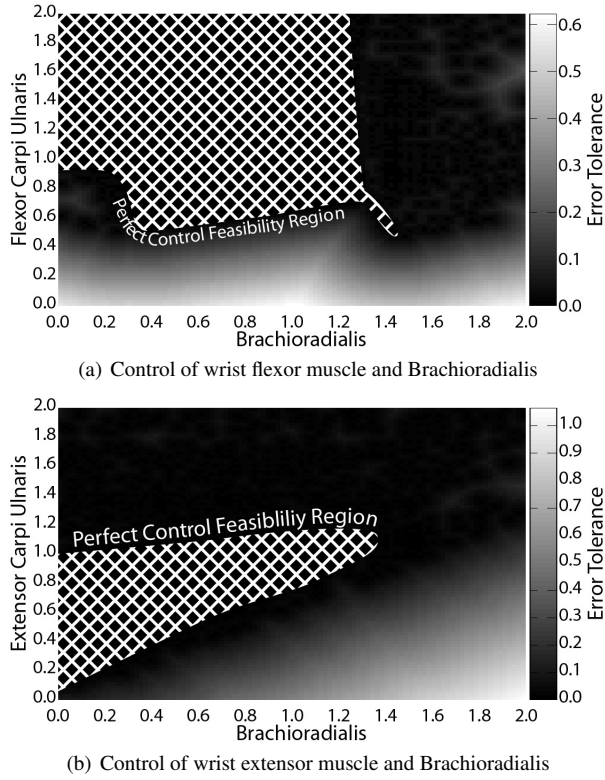


Fig. 4 Comparison of feasibility region of both methods: Hatched region shows perfect method feasibility, color indicates required error tolerance to obtain feasible result using relaxed method.

other controllability issues, such as misalignment of or insufficient torque from exoskeleton actuators are not considered here and may be addressed in future work.

To overcome such a *non-ideal* exoskeleton problem, the realizability of (8) is further analyzed. The equation (8) is rewritten as follows by considering the controllable degrees of freedom of the exoskeleton:

$$\begin{aligned}\tau_{ex} &= \mathbf{J}^T \mathbf{F} - \tau_a \\ &= \mathbf{J}^T \mathbf{F} - \mathbf{E} \tau_{ac}\end{aligned}\quad (41)$$

$$= [\mathbf{J}^T \quad -\mathbf{E}] \begin{bmatrix} \mathbf{F} \\ \tau_{ac} \end{bmatrix} \triangleq \mathbf{J}_e^T \mathbf{F}_e \quad (42)$$

where $\mathbf{J}_e = [\mathbf{J}^T \quad -\mathbf{E}]^T$ and $\mathbf{F}_e = [\mathbf{F}^T \quad \tau_{ac}^T]^T$. τ_{ac} is a vector that only contains the torque components controllable by the exoskeleton's actuators. Matrix \mathbf{E} maps these components onto the joint torque space. For example, if $\tau_{ex} \in \mathbb{R}^{9 \times 1}$ (i.e., a total of 9 joints are involved in a movement) and the exoskeleton robot can control three torques, τ_3, τ_5, τ_9 , then $\tau_{ac} =$

$[\tau_3, \tau_5, \tau_9]^T$ and

$$\mathbf{E} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ \vdots & & \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}^{9 \times 3}.$$

Substituting (42) for (10) gives

$$[\mathbf{A}_t^T \ \mathbf{A}_n^T] w^{-1} \begin{pmatrix} \mathbf{A}_t \\ \mathbf{A}_n \end{pmatrix} \boldsymbol{\alpha} = \mathbf{J}_e^T \mathbf{F}_e. \quad (43)$$

4.2 Perfect Individual Muscle Control with a Non-Ideal Exoskeleton

While previous sections have left r unspecified, this section explicitly utilizes Crowninshield's cost function with $r = 2$ (see Appendix) to obtain closed-form solutions. Generalization with respect to r will be considered in future work. The Crowninshield's cost function is quadratic:

$$u(\mathbf{f}) = \mathbf{f}^T \mathbf{S}^2 \mathbf{f} \quad (44)$$

$$\mathbf{S} = \begin{bmatrix} \frac{1}{PCSA_1} & & \\ & \ddots & \\ & & \frac{1}{PCSA_N} \end{bmatrix} \quad (45)$$

$$\mathbf{q} = w(\mathbf{f}) = 2\mathbf{S}^{-2} \mathbf{f}, \text{ or } \mathbf{f} = \frac{1}{2} \mathbf{S}^2 \mathbf{q}. \quad (46)$$

Although the range of feasible f_{td} is limited, perfect control of target muscles is still feasible if the following condition holds.

Theorem: Perfect individual muscle control with a non-ideal exoskeleton

Assume that either the feasibility conditions presented in [21] or the relaxed conditions presented in the previous sections hold; therefore $\boldsymbol{\alpha}$ exists. If the following condition also holds, a solution for \mathbf{F}_e exists for a given $\boldsymbol{\alpha}$ that perfectly realizes the target muscle forces:

$$\text{rank}(\mathbf{G}) = \text{rank}([\mathbf{G} \ \boldsymbol{\alpha}]), \quad (47)$$

where

$$\mathbf{G} = 2([\mathbf{A}_t^T \ \mathbf{A}_n^T] \mathbf{S}^2 \begin{bmatrix} \mathbf{A}_t \\ \mathbf{A}_n \end{bmatrix})^{-1} \mathbf{J}_e^T. \quad (48)$$

The solution of \mathbf{F}_e is obtained by

$$\mathbf{F}_e = \mathbf{G}^+ \boldsymbol{\alpha} + (\mathbf{I} - \mathbf{G}^+ \mathbf{G}) \boldsymbol{\psi}. \quad (49)$$

where $\boldsymbol{\psi}$ is a free vector parameter mapped onto the null space of \mathbf{G} , defined by $(\mathbf{I} - \mathbf{G}^+ \mathbf{G})$. The parameter $\boldsymbol{\psi}$ can be chosen to tune the solution given by $\boldsymbol{\alpha}$.

Proof:

From (43) and (46),

$$\frac{1}{2} [\mathbf{A}_t^T \ \mathbf{A}_n^T] \mathbf{S}^2 \begin{bmatrix} \mathbf{A}_t \\ \mathbf{A}_n \end{bmatrix} \boldsymbol{\alpha} = \mathbf{J}_e^T \mathbf{F}_e. \quad (50)$$

If a motor task induces activities in a number of muscles greater than or equal to M , the matrix $[\mathbf{A}_t^T \ \mathbf{A}_n^T]$ is full row rank. This assumption most likely holds for a whole-arm task involving multiple joints. Therefore, the square matrix $[\mathbf{A}_t^T \ \mathbf{A}_n^T] \begin{bmatrix} \mathbf{A}_t \\ \mathbf{A}_n \end{bmatrix}$ is invertible, which yields

$$\boldsymbol{\alpha} = \mathbf{G} \mathbf{F}_e. \quad (51)$$

Hence, the existence of \mathbf{F}_e can be checked by (47). Under this condition, \mathbf{F}_e is given by (49). Note that this condition guarantees perfect realization of target muscle forces although condition (47) is more restrictive than (35).

Remark 1: Reducing changes in non-target muscles

If $(\mathbf{I} - \mathbf{G}^+ \mathbf{G})$ is a non-zero square matrix, $\boldsymbol{\psi}$ can be chosen to reduce the changes in non-target muscles, e.g., $\|\mathbf{q}_m - \mathbf{q}_{0m}\| \rightarrow \min$. Roughly speaking, using an exoskeleton able to apply more joint torques is expected to lead to a wider choice of $\boldsymbol{\psi}$, resulting in smaller changes in non-target muscles. This additional controllable DOF provided by such an exoskeleton can also be used to modify the distribution between \mathbf{F} and $\boldsymbol{\tau}_a$. In some cases, where the rank of $(\mathbf{I} - \mathbf{G}^+ \mathbf{G})$ is large enough, the application of $\boldsymbol{\tau}_a$ using an exoskeleton may not be necessary; performing a task by only exerting an end point force \mathbf{F} using a manipulator arm would appropriately adjust target muscles.

4.3 Relaxed Individual Muscle Control with a Non-Ideal Exoskeleton

The rank condition (47) may not always hold. In other words, \mathbf{F}_e may not exist that perfectly realizes a given $\boldsymbol{\alpha}$. It still is possible to obtain desired target muscle forces with a certain range of error.

Remark 2: Relaxing (47)

For simplicity, consider the unrelaxed criterion (14), a special case of (35) where $\boldsymbol{\epsilon}_r = \mathbf{0}$ and $\boldsymbol{\lambda}_{0ru} = \mathbf{0}$. Assume that (14) holds but (47) does not hold. This means that the individual muscle control problem has a solution with $\boldsymbol{\alpha}$. However, there is no combination of \mathbf{F} and $\boldsymbol{\tau}_a$ to realize $\boldsymbol{\alpha}$ due to a limitation of the hardware. It is still possible to induce changes in target muscles by using

$$\mathbf{F}_e = \mathbf{G}^+ \boldsymbol{\alpha}. \quad (52)$$

Table 3 Target muscles and desired ratios of change for simulation

| Muscle name | Ratio |
|------------------------------|--------|
| Biceps long (BIC) | x 2.0 |
| Flexor Carpi Ulnaris (FCU) | x 0.25 |
| Extensor Carpi Ulnaris (ECU) | x 0.5 |

This solution does not guarantee perfect realization of f_{td} , but minimizes errors in the target muscles in the sense of least mean square. Note that this solution does not have any redundancy, i.e., there is no free parameter than can be tuned to adjust non-target muscles.

4.4 Simulation

First, the association between the controllability of target muscle forces and the number of joints the exoskeleton can apply torques to is investigated by simulation.

Consider a static posture shown in Figure 2 where the angle of the elbow joint angle is 90° . Similar to Section 3.2, the musculoskeletal system has a total of 9 joints ($M = 9$) from the torso to wrist joint with 51 muscles ($N = 51$). The Crowninshield's cost function (see Appendix) is applied for the optimization criterion. In the nominal task, a subject exerts a tip-force of $F = [F_x, F_y, 0]^T = [10, 10, 0]^T$ [N] to a handle in the horizontal plane. Among 51 muscles, 26 muscles are active and remaining 25 muscles are inactive. The Biceps Longus muscle (BIC), Flexor Carpi Ulnaris muscle (FCU), and Extensor Carpi Ulnaris muscle (ECU) are selected as target muscles and controlled as shown in Table Table 3.

Five types of force/torque application shown in Figure 5 are considered. In Type 1, a wearable device can apply torques to *all* 9 joints of the skeletal model. The mathematical formulation presented in [21] assumed this ideal mechanism. In Type 2, the wearable robot applies torques to a total of 7 joints from the shoulder to the wrist, excluding the torso and neck joints. This type uses a manipulator that can apply 3-dimensional forces to the tip. In Type 3, the wearable robot applies torques to a total of 4 joints from the elbow to the wrist joint. The device used in the authors' past research [7, 19–21] has the same actuator configuration. Type 4 applies torque to the elbow joint. In Type 5, wearable actuators are not used but only a manipulator is used.

Table Table 4 shows computed tip-forces and assist-torques for types 1 to 5. Only types 1 and 2 satisfied (47), realizing perfect individual muscle control. Equation (52) for relaxed muscle control was used for types 3 to 5. As can be observed in the results, Types 1 and 2 had no error in desired target muscle forces. Type 3 could not accomplish perfect control, but it still achieved the target muscle forces at a maximum error of 25%, recorded from the Biceps muscle. The forces for the Flexor Carpi Ulnaris muscle and Extensor Carpi Ulnaris muscle were still accurate. The table also shows average errors in 23 non-target muscles. Errors in the target muscles and changes in non-target muscles tend to increase as the number of controllable joints decreases toward Type 5. Type 5 particularly exhibited large errors. This implies that the sole exertion of a tip-force is insufficient to induce a desired muscle activation pattern, justifying the use of a wearable exoskeleton that can apply joint torques. This result also indicates that an ideal exoskeleton with a capability of applying all the joints is not always necessary. An exoskeleton with a reduced number of controllable joints would suffice for muscle control tasks with a certain allowable error range.

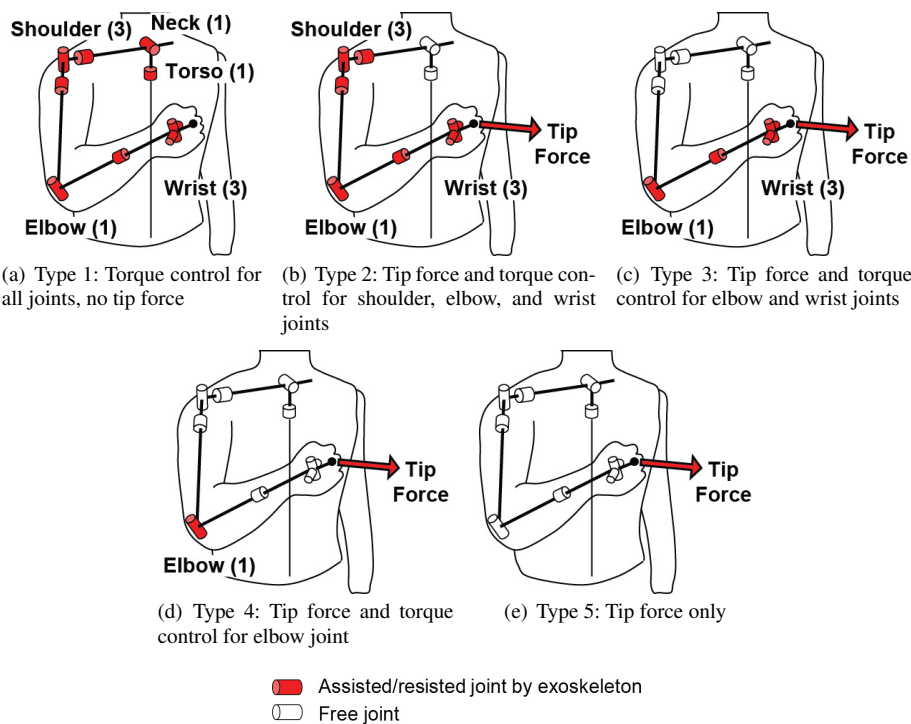


Fig. 5 Exoskeleton configurations

Table 4 Simulation results

| | | Exoskeleton type | | | | |
|----------------------------------|-----------------------|------------------|---------------|---------------|---------------|---------------|
| | | 1 | 2 | 3 | 4 | 5 |
| Tip-force | F_x [N] | – | 10.19 | 7.69 | 8.02 | 3.48 |
| | F_y [N] | – | 10.07 | 10.68 | 9.92 | 9.83 |
| | F_z [N] | – | 0.39 | -0.02 | -0.01 | 3.72 |
| Torso | τ_1 [Nm] | -0.05 | – | – | – | – |
| Neck | τ_2 [Nm] | 0.04 | – | – | – | – |
| | τ_3 [Nm] | -0.83 | -0.98 | – | – | – |
| | τ_4 [Nm] | -1.99 | -1.97 | – | – | – |
| | τ_5 [Nm] | -0.82 | -0.80 | – | – | – |
| Elbow | τ_6 [Nm] | 0.66 | 0.77 | 0.66 | -1.66 | – |
| | τ_7 [Nm] | 0.15 | 0.15 | 0.15 | – | – |
| Wrist | τ_8 [Nm] | 0.14 | 0.15 | 0.14 | – | – |
| | τ_9 [Nm] | 0.31 | 0.32 | 0.34 | – | – |
| | Target muscle changes | | | | | |
| Nontarget muscle average changes | BIC | $\times 2.00$ | $\times 2.00$ | $\times 2.51$ | $\times 3.68$ | $\times 2.75$ |
| | ECU | $\times 0.50$ | $\times 0.50$ | $\times 0.50$ | $\times 0.74$ | $\times 0.66$ |
| Perfect control condition (47) | | Pass | Pass | Fail | Fail | Fail |

4.5 Experiments

Two experiments were conducted to confirm the efficacy of an exoskeleton with a reduced set of controllable joints. These experiments made use of a manipulator arm equipped with a force sensor to apply an end point force to the human's arm.

Table 5 End-effector force and applied joint torque for experiment 1: Biceps (BIC) $\times 0.7$

| Task | F_x [N] | F_y [N] | F_z [N] | τ_6 [Nm] (Elbow) |
|-------------|-----------|-----------|-----------|-----------------------|
| Nominal (A) | 0.00 | 0.00 | -30.00 | |
| Type 4h | 0.92 | -25.94 | -3.36 | 0.96 |
| Type 5h | -0.41 | -18.13 | -10.84 | - |

4.5.1 Exoskeleton Joints and Changes in Non-Target Muscles

A nominal task applies an external downward force of 30 [N] (i.e., $[F_x, F_y, F_z] = [0, 0, -30]$). The Biceps Longus muscle (BIC) was selected as a target muscle and the desired change was given as $\times 0.7$. The both cases satisfied (47); therefore, ψ in (49) can reduce changes in non-target muscles.

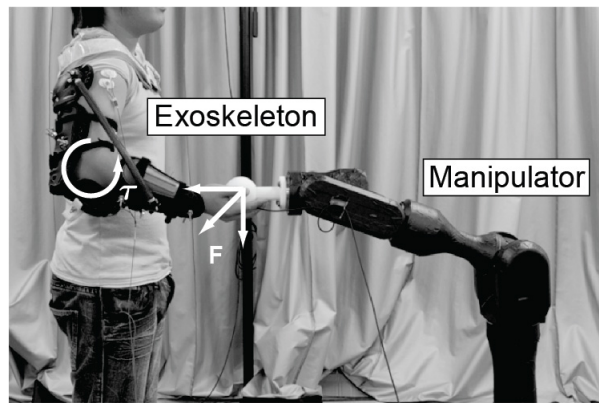
Two configurations shown in Figure 6 were considered. The system shown in Figure 6(a), named Type 4h, uses an exoskeleton that applies torque to the elbow joint and a manipulator that applies 3-dimensional forces to the tip, totaling 4 controllable DOF. The device shown in Figure 6(b), named Type 5h, only uses a manipulator, totaling 3 controllable DOF. The device types 4h and 5h correspond to types 4 and 5 in Figure 5, respectively. It is expected that a configuration with more controllable DOF has a higher capability of controlling non-target muscles than a mechanism with a fewer controllable DOF.

Electromyographic signals (EMG) during tasks were processed in the same manner as [21]. The results are shown in Figure 7 for 8 healthy subjects. Each subject performed five trials of the assigned task. White bars show desired changes in the two target muscles and gray bars show means of collected EMG signals with error bars indicating standard deviations (SD). As shown in Figure 7(b), the expected/simulated average change in the non-target muscle activations increased from 21% to 24% as the number of controllable DOF of the system decreased from 4 to 3. In the experiment, the average change increased from 12 to 15%.

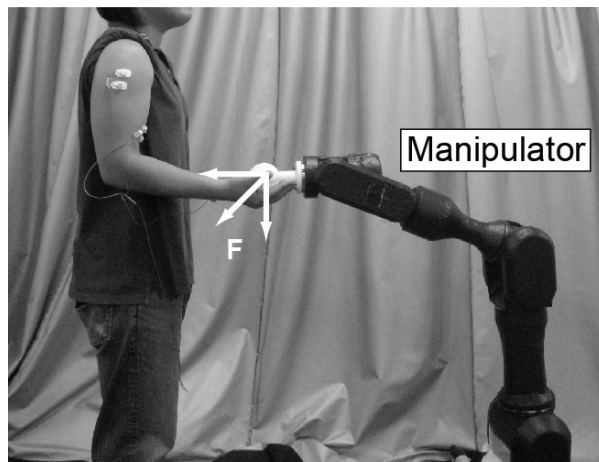
4.5.2 Individual Muscle Control Using Solely a Manipulator

If the objective of a motor task is primarily to adjust target muscles, with less emphasis on changes of non-target muscles, one might argue that the sole use of a manipulator would suffice. Such a system would, in practice, be advantageous in terms of reduced complexity of operation. It is likely that the condition (47) holds if the number of target muscles is not very large, for example, 1 or 2. Hence, motor tasks can be planned by using (49).

In the second experiment, tasks with the manipulator in Figure 6(b) are considered. The nominal task applied an external downward force of 30 N (i.e., $[F_x, F_y, F_z] = [0, 0, -30]$). The Anterior Deltoid and Biceps Longus muscles were chosen as the target muscles. The tasks satisfied (47). The results are shown in Figure 8. Each subject performed five trials. White bars show desired changes in the two target muscles and gray bars show means of collected EMG signals with error bars indicating standard deviations (SD). The observed tendencies to increase or decrease EMG signals, with respect to desired changes, were as expected overall. For example, a larger change in the Anterior Deltoid muscle was observed than that in the Biceps muscle where the Anterior Deltoid muscle was controlled to be $\times 2.0$ while maintaining (i.e., $\times 1.0$) the nominal force of the Biceps muscle. Similarly, both the two target muscle forces increased in task D where desired changes greater than 1.0 were given to the both muscles. Note that we cannot rule out the possibility that the magnitude of



(a) Type 4h: Manipulator and exoskeleton



(b) Type 5h: Manipulator only

Fig. 6 Experimental setup for individual muscle control**Table 6** Motor tasks using a robot manipulator

| Task | Anterior Deltoid (DEL) | Biceps Long (BIC) | F_x [N] | F_y [N] | F_z [N] |
|-------------|------------------------|-------------------|-----------|-----------|-----------|
| Nominal (A) | – | – | 0.00 | 0.00 | -30.00 |
| B | $\times 2.0$ | $\times 1.0$ | -38.87 | 1.47 | -32.99 |
| C | $\times 1.0$ | $\times 1.5$ | 19.87 | -0.73 | -48.17 |
| D | $\times 2.0$ | $\times 1.5$ | -18.95 | 1.70 | -51.33 |
| E | $\times 0.5$ | $\times 1.0$ | 19.35 | -2.56 | -28.18 |
| F | $\times 1.0$ | $\times 0.5$ | -19.92 | -0.30 | -11.64 |
| G | $\times 0.5$ | $\times 0.5$ | -0.54 | -2.34 | -9.92 |

EMG signals may not be strictly proportional to corresponding muscle force. Needle EMG signals will be used for further evaluation in the future.

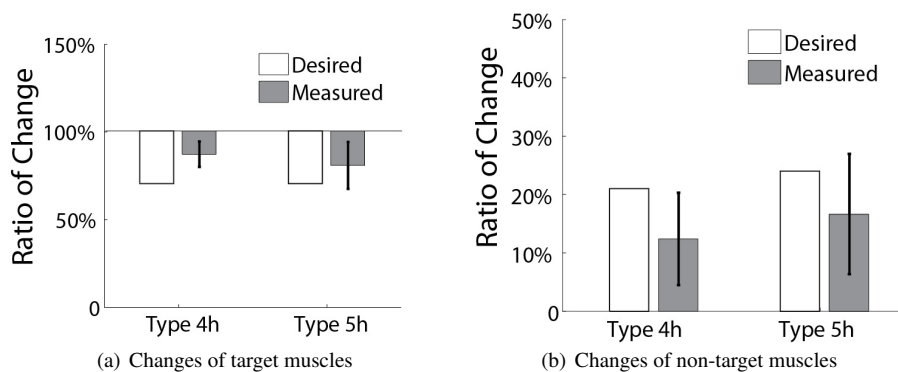


Fig. 7 Experimental result of individual muscle force control using a manipulator and exoskeleton: Biceps (BIC) $\times 0.7$.

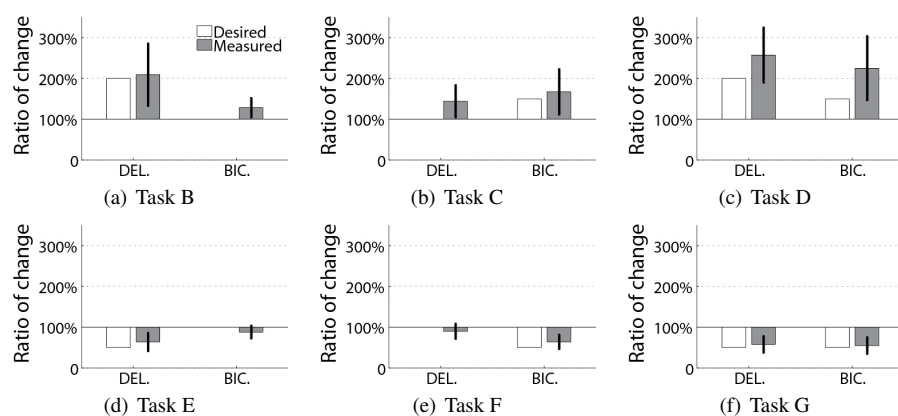


Fig. 8 Experimental result of individual muscle force control using only a manipulator

5 Conclusion

Prior work has developed a method by which the forces of human muscles can be controlled via an exoskeleton. However, the previous method relied on several limiting assumptions and requirements which simplified the mathematical formulation, but restricted its applicability. A relaxed method which removed several of these was presented. It reframed the problem from one attempting to exactly achieve a target muscle pattern to one in which the solution was optimized to minimize the error from a target pattern. This allowed patterns that were previously infeasible to be realized. The enhanced method was demonstrated using computer models and the improvement in feasibility region was shown. The efficacy was confirmed by simulation and experiments.

There are many situations in which such a method could be applied. Exoskeleton robot development is increasing, and this method would allow such a robot to apply torques to human joints to achieve some desired muscle pattern. This could be useful for exercise devices attempting to build muscle strength or for rehabilitation devices attempting to recover lost muscle abilities. This is due to the pinpointed nature of the method, targeting only the muscles that are desired and minimizing the changes in all others. In addition, it can be

used for diagnosing muscle related disorders. Since the expected muscle pattern due to the applied load is known, any deviation from the expected pattern would indicate a problem in a person's muscle control abilities. The relaxed method proposed here allows a much wider range of the applications than the previous method.

6 Acknowledgments

This work was supported in part by the National Science Foundation (NSF) under Grant: IIS 1142438 and Japan Science and Technology Agency (JST) - National Science Foundation (NSF) Strategic International Cooperative Program. The authors would like to thank Ellenor Brown reading the manuscript and for her helpful suggestions.

Appendix: Crowninshield's Static Optimization

Crowninshield's method [15] is a special case presented in (3) that predicts human muscle forces by minimizing a physiologically based criterion $u(\mathbf{f})$:

$$u(\mathbf{f}) = \sum_{j=1}^N \left(\frac{f_j}{PCSA_j} \right)^r \rightarrow \min \quad (53)$$

subject to $\begin{cases} \boldsymbol{\tau}_h = \mathbf{A}\mathbf{f} \\ f_{\min j} \leq f_j \leq f_{\max j} (j = 1, \dots, N) \end{cases}$,

where $PCSA_j$ is the physiological cross sectional area (PCSA), and $f_{\max j} = \varepsilon \cdot PCSA_j$ is the maximum muscle force of the j -th muscle. In this paper, $\varepsilon = 0.7 \times 10^6 [\text{N}/\text{m}^2]$ is given according to [8]. $PCSA_j$'s are given according to [11]. $f_{\min j} = 0, \forall j$ and $r = 2$ are used. See [15] for the choice of r .

References

1. Bertsekas, D.: Nonlinear programming. Athena Scientific Belmont, Mass (1999)
2. van Bolhuis, B., Gielen, C.: A comparison of models explaining muscle activation patterns for isometric contractions. *Biological Cybernetics* **81**(3), 249–261 (1999)
3. Buchanan, T., Shreeve, D.: An Evaluation of Optimization Techniques for the Prediction of Muscle Activation Patterns During Isometric Tasks. *Journal of Biomechanical Engineering* **118**, 565 (1996)
4. Damsgaard, M., Rasmussen, J., Christensen, S., Surma, E., de Zee, M.: Analysis of musculoskeletal systems in the anybody modeling system. *Simulation Modelling Practice and Theory* **14**(8), 1100–1111 (2006)
5. Delp, S., Anderson, F., Arnold, A., Loan, P., Habib, A., John, C., Guendelman, E., Thelen, D.: Opensim: open-source software to create and analyze dynamic simulations of movement. *Biomedical Engineering, IEEE Transactions on* **54**(11), 1940–1950 (2007)
6. Delp, S., Loan, J., Hoy, M., Zajac, F., Topp, E., Rosen, J., Center, V., Alto, P.: An interactive graphics-based model of the lower extremity to study orthopaedic surgical procedures. *Biomedical Engineering, IEEE Transactions on* **37**(8), 757–767 (1990)
7. Ding, M., Kurita, Y., Ueda, J., Ogasawara, T.: Pinpointed Muscle Force Control Taking Into Account the Control DOF of Power-assisting Device. In: *ASME Dynamic Systems and Control Conference (DSCC 2010)* (2010)
8. Karlsson, D., Peterson, B.: Towards a model for force predictions in the human shoulder. *Journal of biomechanics* **25**(2), 189–199 (1992)

9. Kazerooni, H., Racine, J., Huang, L., Steger, R.: On the Control of the Berkeley Lower Extremity Exoskeleton (BLEEX). In: IEEE International Conference on Robotics and Automation, vol. 4, p. 4353. IEEE; 1999 (2005)
10. Maurel, W., Thalmann, D.: A Case Study on Human Upper Limb Modelling for Dynamic Simulation. *Computer Methods in Biomechanics and Biomedical Engineering* **2**(1), 65–82 (1999)
11. MotCo Project: <http://motco.info/data/pcsa.html>
12. Pardalos, P., Rosen, J.: *Constrained global optimization: algorithms and applications* (1987)
13. Perry, J., Rosen, J.: Design of a 7 Degree-of-Freedom Upper-Limb Powered Exoskeleton. In: *Biomedical Robotics and Biomechatronics, 2006. BioRob 2006. The First IEEE/RAS-EMBS International Conference on*, pp. 805–810 (2006)
14. Prilutsky, B.: Coordination of two-and one-joint muscles: functional consequences and implications for motor control. *Motor Control* **4**(1), 48–52 (2000)
15. RD. Crowninshield, et al.: A physiologically based criterion of muscle force prediction in locomotion. *J. Biomechanics* **14**, 793–801 (1981)
16. Rosen, J., Fuchs, M., Arcan, M.: Performances of Hill-type and neural network muscle models-toward a myosignal-based exoskeleton. *Computers and Biomedical Research* **32**(5), 415–439 (1999)
17. Todorov, E.: Cosine tuning minimizes motor errors. *Neural Computation* **14**(6), 1233–1260 (2002)
18. Tsagarakis, N., Caldwell, D.: Development and Control of a Soft-Actuated Exoskeleton for Use in Physiotherapy and Training. *Autonomous Robots* **15**(1), 21–33 (2003)
19. Ueda, J., Ding, M., Matsugashita, M., Oya, R., Ogasawara, T.: Pinpointed control of muscles by using power-assisting device. In: *Robotics and Automation, 2007 IEEE International Conference on*, pp. 3621–3626 (2007)
20. Ueda, J., Hyderabadwala, M., Krishnamoorthy, V., Shinohara, M.: Motor task planning for neuromuscular function tests using an individual muscle control technique. *IEEE 11th International Conference on Rehabilitation Robotics* (2009)
21. Ueda, J., Ming, D., Krishnamoorthy, V., Shinohara, M., Ogasawara, T.: Individual muscle control using an exoskeleton robot for muscle function testing. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on* **18**(4), 339–350 (2010)
22. Veneman, J., Kruidhof, R., Hekman, E., Ekkelenkamp, R., Van Asseldonk, E., van der Kooij, H.: Design and Evaluation of the LOPES Exoskeleton Robot for Interactive Gait Rehabilitation. *IEEE Transactions On Neural Systems And Rehabilitation Engineering* **15**(3), 379 (2007)