

Decentralized Nonlinear Control of Redundant Upper Limb Exoskeleton with Natural Adaptation Law

Mahdi Hejrati¹, Jouni Mattila²

Abstract—The aim of this work is to utilize an adaptive decentralized control method called virtual decomposition control (VDC) to control the orientation and position of the end-effector of a 7 degrees of freedom (DoF) right-hand upper-limb exoskeleton. The prevailing adaptive VDC approach requires tuning of $13n$ adaptation gains along with $26n$ upper and lower parameter bounds, where n is the number of rigid bodies. Therefore, utilizing the VDC scheme to control high DoF robots like the 7-DoF upper-limb exoskeleton can be an arduous task. In this paper, a new adaptation function, so-called natural adaptation law (NAL), is employed to eliminate these burdens from VDC, which results in reducing all $13n$ gains to one and removing dependency on upper and lower bounds. In doing so, VDC-based dynamic equations are restructured, and inertial parameter vectors are made compatible with NAL. Then, the NAL adaptation function is exploited to design a new adaptive VDC scheme. This novel adaptive VDC approach ensures physical consistency conditions for estimated parameters with no need for upper and lower bounds. Finally, the asymptotic stability of the algorithm is proved with the virtual stability concept and the accompanying function. The experimental results are utilized to demonstrate the excellent performance of the proposed new adaptive VDC scheme.

I. INTRODUCTION

Generally speaking, exoskeletons are human-robot interfaces (HRIs) that ease interaction and communication between human operators and robots. In the context of teleoperation systems, a human operator needs to have an adequate sense of the remote site and control of the slave robot to accomplish a successful teleoperation task [1]. As a reasonable solution, an exoskeleton transfers human body movements to the slave robot and provides a sense of the remote site through haptic technology, allowing the operator to execute dexterous tasks.

The aim of our project is to study two upper-limb 7-DoF exoskeletons as master robots and two heavy-duty hydraulic mobile manipulators as slave robots within the concept of multi-master multi-slave (MMMS) teleoperation systems [2]. Like in our earlier work [3], we considered real-world dissimilar master and slave robots, where the range of motion and force amplification was 1:4 for motion and 1:800 for force amplification. This required us to consider the problem

of motion/force scaling which resulted complicated stability analysis tackled in [3]. However, in [3], a commercial 6-DoF desktop haptic manipulator was utilized for controlling a stationary heavy-duty hydraulic slave robot, whereas our current target is to utilize wearable 7-DoF exoskeletons to control multiple mobile slave robots in human-like object manipulation scenarios. Therefore, it is essential to design a control strategy with the capability of handling these highly complex MMMS systems for the required high performance control objectives, such as high transparency as in [3]. For this ambitious objective, in this paper, decentralized nonlinear control problem of the 7-DoF right-hand upper-limb exoskeleton is examined. In the future, this work will be expanded to the control of MMMS teleoperation systems.

The control problem of one-armed upper-limb exoskeletons has been widely examined. In [4]- [5], adaptive integral terminal sliding mode control (SMC) and integral second-order terminal SMC have been employed to control upper-limb exoskeletons, respectively. In [6], an integral second-order terminal SMC with quasi-time delay estimation (QTDE) was employed for force and position control of an upper-limb exoskeleton. Adaptive impedance control with a nonlinear disturbance observer with estimation of human-desired intended motion has been considered in [7] and [8]. Other control schemes, such as backstepping [9], backstepping-SMC [10], and intelligent approaches like fuzzy SMC [11] have also been applied to the exoskeleton control problem. Although the above-mentioned control schemes have shown good performance, they are not perfect choices for MMMS systems, especially when MMMS teleoperation systems work to manipulate a common object. In such a scenario, the levels of nonlinearity and complexity of the system are intensely high, and the aforementioned approaches can face difficulties in handling them. For example, backstepping can end up with an explosion of complexity, even for a simple system [12]. Moreover, most of the mentioned works are about the applications of exoskeletons in rehabilitation with one arm, while our goal is to use exoskeletons in MMMS teleoperation systems. However, MMMS systems encompass rehabilitation and telerehabilitation.

The virtual decomposition control (VDC) approach is a decentralized control scheme that breaks the entire complex system into multiple subsystems that are easy to deal with and design control action [13]. Virtual cutting points (VCPs) are points at which a system is broken down into subsystems. With the introduction of virtual power flows (VPFs) at VCPs as stability connectors, the stability of each subsystem, which is examined distinctly at the subsystem level, is expanded to

*The TITAN (Teaching human-like abilities to heavy mobile manipulators through multisensory presence) project is funded by the Technology Industries of Finland Centennial Foundation and the Jane and Aatos Erkkö Foundation Future Makers programme. 2020-2023.

¹M. Hejrati is with the Faculty of Engineering and Natural Science, Tampere University, 7320 Tampere, Finland, Corresponding Author mahdi.hejrati at tuni.fi

²J. Mattila is with the Faculty of Engineering and Natural Science, Tampere University, 7320 Tampere, Finland, jouni.Mattila at tuni.fi

the whole system and culminates in asymptotic convergence. Such an approach of modulation makes VDC a much more stable control algorithm in a way that altering the control action of one subsystem does not affect the others. Therefore, VDC is able to handle extremely high DoF dissimilar teleoperation systems and perfect choice for our future aims.

So far, the VDC method has been utilized in fields such as hydraulic manipulators [14]- [16], mobile manipulators [17], and underwater applications [18]. Nonetheless, for the exoskeleton application, only [19] has employed an adaptive VDC scheme to control a 7-DoF exoskeleton with estimation of the joint's moment of inertia and friction. In contrast to [19], which examined rehabilitation without motion/force scaling, our aim is to use the exoskeleton in dissimilar MMMS teleoperation systems with high motion/force scaling. However, because adaptive VDC utilizes a projection function [13] to estimate unknown parameters of the rigid body, it has two major drawbacks: *i*) it requires lower and upper bounds of each parameter, and *ii*) it designates an adaptation gain to each parameter. On the other hand, the inertial parameter vector of the VDC-based dynamic equation encompasses 13 parameters, which in general, the unique inertial parameter vector must contain 10 parameters [20]. This means that $13n$ adaptation gains and $26n$ lower and upper bounds are required for adaptive VDC, where n is the number of rigid bodies. Therefore, as the DoF of a system increases, like in [19] and MMMS teleoperation systems, preparing and tuning all these parameters can make VDC time-consuming, despite its benefit of being decentralized.

The aim of this paper is to eliminate the above-mentioned problems by proposing a novel adaptive VDC approach that utilizes NAL [21] for the estimation of unknown rigid body parameters. The NAL function is defined based on the Riemannian metric and is coordinate invariant with respect to coordinate frames or physical unit selection. This adaptation law only requires one tuning gain for the entire chain of rigid bodies with no need for upper and lower bounds. The contributions of this paper are as follows:

- Redesigning dynamic equations of VDC into the 10-form inertial parameter vectors.
- Proposing an asymptotic-stability guaranteed, novel VDC-NAL approach. By doing so, all the $26n$ lower and upper bounds are eliminated and $13n$ adaptation gains are reduced to 1, while the physical consistency of the estimated parameters is ensured.
- Applying the presented approach to control a 7-DoF exoskeleton. The commercial exoskeleton ABLE by Haption is used in the experiment to examine the performance of the modified controller.

II. PROPOSED VDC STRUCTURE

In this section, the prevailing form of VDC is explained and essential theorems and definitions are provided to prove stability of the system in the sense of virtual stability. Then, the dynamic equations of the system are reformulated to suit the presented adaptation law.

A. Virtual Decomposition Control Approach

Consider $\{A\}$ as a frame that is attached to a rigid body. Then, the 6D linear/angular velocity vector ${}^A V \in \mathbb{R}^6$ and force/moment vector ${}^A F \in \mathbb{R}^6$ can be expressed as follows [13]:

$${}^A V = [{}^A v, {}^A \omega]^T, \quad {}^A F = [{}^A f, {}^A m]^T \quad (1)$$

where ${}^A v \in \mathbb{R}^3$ and ${}^A \omega \in \mathbb{R}^3$ are the linear and angular velocities of frame $\{A\}$, and ${}^A f \in \mathbb{R}^3$ and ${}^A m \in \mathbb{R}^3$ are the force and moment expressed in frame $\{A\}$, respectively. The transformation matrix that transforms force/moment vectors and velocity vectors between frame $\{A\}$ and $\{B\}$, where $\{B\}$ is another frame attached to the rigid body, is [13]

$${}^A U_B = \begin{bmatrix} {}^A R_B & \mathbf{0}_{3 \times 3} \\ ({}^A r_{AB} \times) {}^A R_B & {}^A R_B \end{bmatrix} \quad (2)$$

where ${}^A R_B \in \mathbb{R}^{3 \times 3}$ is a rotation matrix between frame $\{A\}$ and $\{B\}$, (\times) operator is a skew-symmetric operator defined in [13], and ${}^A r_{AB}$ denotes a vector from the origin of frame $\{A\}$ to the origin of frame $\{B\}$, expressed in $\{A\}$. Based on the ${}^A U_B$, the force/moment and velocity vectors can be transformed between frames as [13]

$${}^B V = {}^A U_B^T {}^A V, \quad {}^A F = {}^A U_B {}^B F. \quad (3)$$

Then, the dynamic equation of the free rigid body expressed in frame $\{A\}$ can be derived as follows [13]:

$$\bar{M}_A \frac{d}{dt} ({}^A V) + \bar{C}_A ({}^A V) + \bar{G}_A = {}^A F^* \quad (4)$$

where $\bar{M}_A \in \mathbb{R}^{6 \times 6}$ is the mass matrix, $\bar{C}_A \in \mathbb{R}^{6 \times 6}$ is the centrifugal and coriolis matrix, $\bar{G}_A \in \mathbb{R}^6$ is the gravity vector, and ${}^A F^* \in \mathbb{R}^6$ is the net force/moment vector applied to the rigid body. These matrices are [13]

$$\begin{aligned} \bar{M}_A &= \begin{bmatrix} m_A I_3 & -m_A ({}^A r_{AB} \times) \\ m_A ({}^A r_{AB} \times) & I_A - m_A ({}^A r_{AB} \times)^2 \end{bmatrix} \\ \bar{C}_A &= \begin{bmatrix} m_A ({}^A \omega \times) & & & \\ m_A ({}^A r_{AB} \times) ({}^A \omega \times) & ({}^A \omega \times) I_A + & & \\ & -m_A ({}^A \omega \times) ({}^A r_{AB} \times) & & \\ I_A ({}^A \omega \times) - m_A ({}^A r_{AB} \times) ({}^A \omega \times) ({}^A r_{AB} \times) & & & \end{bmatrix} \\ \bar{G}_A &= \begin{bmatrix} m_A {}^A R_I g \\ m_A ({}^A r_{AB} \times) {}^A R_I g \end{bmatrix} \end{aligned} \quad (5)$$

where $m_A \in \mathbb{R}$ denotes the mass of the rigid body, I_3 is a 3×3 identity matrix, $I_A = {}^A R_I I_o^J R_A$ where $I_o \in \mathbb{R}^{3 \times 3}$ denotes the moment of inertia matrix around the center of mass, and $g = [0, 0, 9.8]^T \in \mathbb{R}^3$ is the gravity vector.

The dynamic equation (4) can be rewritten as a linear-in-parameter expression:

$$Y_A^T \theta_A = \bar{M}_A \frac{d}{dt} ({}^A V_r) + \bar{C}_A ({}^A V_r) + \bar{G}_A, \quad (6)$$

where $Y_A^T \in \mathbb{R}^{6 \times 13}$ is the regressor matrix generated from ${}^A V_r$, ${}^A \dot{V}_r$ is the required velocity vector, and $\theta_A \in \mathbb{R}^{13}$

is the rigid body inertial parameter vector. Furthermore, the required net force/moment vector can be designed as

$${}^A F_r^* = Y_A^r \hat{\theta}_A + K({}^A V_r - {}^A V), \quad (7)$$

where $\hat{\theta}_A$ is the estimation of the unknown inertial parameter vector, and K is the symmetric positive-definite matrix. To estimate the inertial parameter vector of each rigid body, the projection function is utilized as follows [13]:

$$\dot{\hat{\theta}}_{Ai}(t) = \rho_i s_i \kappa_i, \quad i = 1..13 \quad (8)$$

with

$$\kappa_i = \begin{cases} 0 & \text{if } \hat{\theta}_i \leq \underline{\theta}_i \text{ and } s_i \leq 0 \\ 0 & \text{if } \hat{\theta}_i \geq \bar{\theta}_i \text{ and } s_i \geq 0 \\ 1 & \text{otherwise} \end{cases} \quad (9)$$

and

$$s(t) = Y_A^{rT}({}^A V_r - {}^A V) \quad (10)$$

where $\rho_i > 0$ is the adaptation gain, and $\underline{\theta}_i$ and $\bar{\theta}_i$ are the lower and upper limits of the i^{th} parameter, respectively.

Each adjacent subsystem interacts with each other at VCPs with governing dynamics of (4) and a control law of (7). Such interactions are designated as VPFs. Considering the fixed frame $\{A\}$, VPF can be defined as follows [13]:

$$p_A = ({}^A V_r - {}^A V)^T ({}^A F_r - {}^A F) \quad (11)$$

where ${}^A F_r \in \mathbb{R}^6$ is the required force. VPFs act as stability connectors in the concept of virtual stability and extend the stability of each subsystem to the stability of entire system. The following lemma expresses the virtual stability concept of VDC:

Definition 1. [13] A single subsystem that is virtually decomposed from complex system with dynamics (4), control law (7) with known θ , and affiliated function $X(t)$ can be said to be virtually stable if and only if there exists a non-negative accompanying function $\nu(t)$

$$\nu(t) \geq \frac{1}{2} X(t)^T P X(t) \quad (12)$$

in a way that

$$\dot{\nu}(t) \leq -X(t)^T Q X(t) + p_A - p_{\bar{A}} \quad (13)$$

where P and Q are two block-diagonal positive-definite matrices, and \underline{A} and \bar{A} are two adjacent neighbors of A .

Theorem 1. [13] Consider a complex system that is virtually decomposed into subsystems. If all the decomposed subsystems are virtually stable in the sense of definition 1, then the entire system is stable.

As can be seen from (8) and (9), 13 adaptation gains must be tuned and 13 lower-bound and 13 upper-bound parameters must be designated for a single subsystem, which for a system with n rigid bodies, they become $13n$ and $26n$. The aim of this paper is to reduce the amount of required parameters by replacing the projection function with the NAL adaptation function. This replacement eliminates all the aforementioned burdens by reducing $13n$ gains to 1 and

eliminating $26n$ inertial parameter bounds. However, as can be seen from (6), $\theta_A \in \mathbb{R}^{13}$ contains 13 parameters, whereas NAL requires θ_A to be \mathbb{R}^{10} . Consequently, the next step is to reformulate the prevailing equations.

B. Redesigned Dynamic Equations

In this part, a new structure for the VDC scheme is proposed. This structure equips VDC with 10 unique representations of the rigid body in space with respect to frame $\{A\}$ and puts it in a natural form, making it compatible with NAL. By defining $\bar{I}_A = I_A - m_A ({}^A r_{AB} \times)^2$, we can rewrite (5) as

$$M_A = \begin{bmatrix} m_A I_3 & -m_A ({}^A r_{AB} \times) \\ m_A ({}^A r_{AB} \times) & \bar{I}_A \end{bmatrix} \quad (14)$$

$$C_A = \begin{bmatrix} m_A ({}^A \omega \times) & -m_A ({}^A \omega \times) ({}^A r_{AB} \times) \\ m_A ({}^A r_{AB} \times) ({}^A \omega \times) & ({}^A \omega \times) \bar{I}_A \end{bmatrix}.$$

Details are presented in Appendix A. It must be mentioned that gravity vector does not need to be reformulated. Consequently, the linear-in-parameter form of new matrices can be written as

$$W_A \phi_A = M_A \frac{d}{{dt}} ({}^A V_r) + C_A ({}^A V_r) + G_A, \quad (15)$$

where $W_A \in \mathbb{R}^{6 \times 10}$ is the new regressor matrix and $\phi_A \in \mathbb{R}^{10}$ is the 10-form inertial parameter vector. Details can be found in Appendix B. It is worth mentioning that the regressor matrix represented in (B.2) has a more compact and simple form than the one in [13]. Now, by using (15), it is possible to merge VDC and NAL.

III. NATURAL ADAPTATION LAW

In the previous section, we derived a 10-form inertial parameter vector for the VDC context. This vector, $\phi_A \in \mathbb{R}^{10}$ (detailed in Appendix B), is a unique representation of a rigid body in space that contains mass and inertia represented in the frame $\{A\}$. However, to have such a unique representation just a subset of \mathbb{R}^{10} can be assigned to ϕ_A . Such a subset must satisfy physical consistency conditions to be a unique representation of the rigid body.

Proposition 1. [22] Consider a non-negative mass density function $\mu: \mathbb{R}^3 \rightarrow \mathbb{R}_0^+$, such that

$$L_A = \int \begin{bmatrix} \mathbf{x}_A \\ 1 \end{bmatrix} \begin{bmatrix} \mathbf{x}_A \\ 1 \end{bmatrix}^T \mu(\mathbf{x}_A) dV_A = \begin{bmatrix} \Sigma_A & h_A \\ h_A^T & m_A \end{bmatrix} \quad (16)$$

where $L_A \in S(4)$ is a 4×4 symmetric matrix, $\Sigma_A = \int \mathbf{x}_A \mathbf{x}_A^T \mu(\mathbf{x}_A) dV_A \in S(3)$ is a second moment matrix, and $S(n)$ is the space of $n \times n$ real-symmetric matrices. The rigid body inertial parameter vector ϕ_A is physically consistent if there exists a one-to-one linear map $f: \mathbb{R}^{10} \rightarrow S(4)$,

$$f(\phi_A) = L_A = \begin{bmatrix} 0.5 \text{tr}(\bar{I}_A) \cdot \mathbf{1} - \bar{I}_A & h_A \\ h_A^T & m_A \end{bmatrix} \quad (17)$$

$$f^{-1}(\phi_A) = \phi_A(m_A, h_A, \text{tr}(\Sigma_A) \cdot \mathbf{1} - \Sigma_A) \quad (18)$$

such that L_A is positive-definite (i.e., $L_A \succ 0$). $\text{tr}(\cdot)$ is the trace operator and $\Sigma_A = 0.5 \text{tr}(\bar{I}_A) \cdot \mathbf{1} - \bar{I}_A$.

Based on proposition 1, the set of physically consistent inertial parameter vectors for a rigid body can be defined on manifold \mathcal{M} as

$$\mathcal{M} = \{\phi_A \in \mathbb{R}^{10} : f(\phi_A) \succ 0\} \subset \mathbb{R}^{10} \quad (19)$$

$$= \{L_A \in S(4) : L_A \succ 0\} = \mathcal{P}(4)$$

where $\mathcal{P}(4)$ is the space of all real-symmetric, positive-definite matrices.

A. Riemannian Geometry of $\mathcal{P}(n)$ and Bregman Divergence

$T_P\mathcal{P}(n)$ is the tangent space for a given $P \in \mathcal{P}(n)$. Therefore, Riemannian metric invariant under group action $G*P = GPG^T$, where $G \in GL(n)$ is any $n \times n$ nonsingular matrix, for given $P \in \mathcal{P}(n)$ and $\mathfrak{F}_1, \mathfrak{F}_2 \in T_P\mathcal{P}(n)$ can be defined as

$$\langle \mathfrak{F}_1, \mathfrak{F}_2 \rangle_P = \frac{1}{2} \text{tr}(P^{-1} \mathfrak{F}_1 P^{-1} \mathfrak{F}_2). \quad (20)$$

Then, the geodesic distance between two arbitrary points $L_1, L_2 \in \mathcal{P}(n)$ can be written in the sense of (20)

$$d_{\mathcal{P}(4)}(L_1, L_2) = \left(\sum_{i=1}^n (\log(\lambda_i))^2 \right)^{\frac{1}{2}} \quad (21)$$

where λ_i are the eigenvalues of $L_1^{-1/2} L_2 L_1^{-1/2}$. Thereafter, a distance metric on \mathcal{M} can be defined as

$$d_{\mathcal{M}}(\phi_A^1, \phi_A^2) = d_{\mathcal{P}(4)}(L_A^1, L_A^2). \quad (22)$$

One other possible way to define distance metric is Bregman divergence. Let $F : \Omega \rightarrow \mathbb{R}$ be a continuous-differentiable, strictly convex function defined on a closed convex set Ω . The Bregman distance is defined as

$$D_F(\mathbf{a}, \mathbf{b}) = F(\mathbf{a}) - F(\mathbf{b}) - \langle \nabla F(\mathbf{b}), \mathbf{a} - \mathbf{b} \rangle. \quad (23)$$

By considering $\Omega = \mathcal{P}(4)$ and assigning log-det function to F (i.e., $F(P) = -\log|P|$ for $P \in \mathcal{P}(4)$), the resulting Bregman divergence is,

$$D_{F(\mathcal{P}(4))}(L_1 \| L_2) = \log \frac{|L_2|}{|L_1|} + \text{tr}(L_2^{-1} L_1) - 4 \quad (24)$$

$$= \sum_{i=1}^4 (-\log(\lambda_i) + \lambda_i - 1),$$

where λ_i are the eigenvalues of $L_2^{-1} L_1$. Now, by utilizing a one-to-one mapping f from ϕ_A to $L_A = f(\phi_A)$, a distance metric on \mathcal{M} can be expressed as

$$d_{\mathcal{M}}(\phi_A^1, \phi_A^2) = D_{F(\mathcal{P}(4))}(L_A^1 \| L_A^2). \quad (25)$$

B. Extracted NAL

In this part, the NAL function is embedded into VDC and proves that asymptotic stability is ensured. Consider a non-negative accompanying function $\nu_T(t)$

$$\nu_T(t) = \nu_P(t) + \nu_A(t) \quad (26)$$

where $\nu_P(t)$ is in the sense of (12), and $\nu_A(t)$ is the accompanying function of the adaptive term. By taking the derivative of (26) and utilizing (15) and the control law in the sense of (7), the following is achieved:

$$\dot{\nu}_T(t) \leq -X(t)QX(t) + p_A - p_{\bar{A}} - \tilde{\phi}_A^T s(t) + \dot{\nu}_A(t) \quad (27)$$

where $\tilde{\phi}_A = \hat{\phi}_A - \phi_A$ and $s(t)$ is defined in (10). Most importantly, $\dot{\nu}_A(t)$ must possess the feature in the form of

$$\dot{\nu}_A(t) = \tilde{\phi}_A^T \mathcal{Z}(\dot{\phi}, \hat{\phi}). \quad (28)$$

The common quadratic function, which results in an adaptation function like the projection function, embodies such a feature. Another possible candidate for $\nu_A(t)$ can be geodesic distance in the form of (21). However, due to the nonlinearity of (21), its time derivative can not be written in the form of (28). Therefore, stability of the overall system cannot be guaranteed. Another choice is to consider $\nu_A(t)$ as the Bregman divergence distance metric [21]:

$$\nu_A(t) = \gamma D_{F(\mathcal{P}(4))}(L_A \| \hat{L}_A) \quad (29)$$

where $\gamma > 0$. From (24) and according to the fact that $-\log(x) + x - 1 \geq 0$, it can be concluded that (29) is a valid accompanying function. Taking the derivative of (29) leads to

$$\dot{\nu}_A(t) = \gamma \text{tr}([\hat{L}_A^{-1} \dot{\hat{L}}_A \hat{L}_A^{-1}] \tilde{L}_A) \quad (30)$$

where $\tilde{L}_A = \hat{L}_A - L_A$.

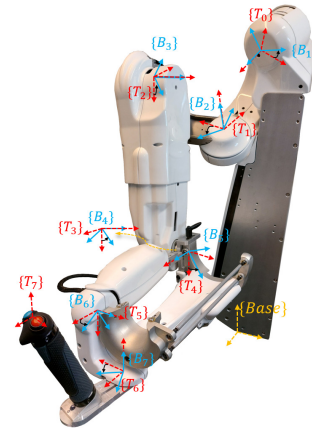


Fig. 1. Commercial ABLE exoskeleton and assigned VDC frames

Proposition 2. Suppose that the adaptive VDC control law in the sense of (7) results in a time derivative of the

accompanying function in the form of (27). Considering $\nu_A(t)$ as defined in (29) and its time derivative in (30), the NAL is defined as [21]

$$\dot{\hat{L}}_A = \frac{1}{\gamma} \hat{L}_A S \hat{L}_A \quad (31)$$

and results in asymptotic stability of the system. S is a unique symmetric matrix that satisfies $\tilde{\phi}_A^T s = \text{tr}(\tilde{L}_A S)$.

Proof. Substituting (30) in (27) and using (31) leads to

$$\begin{aligned} \dot{\nu}_T(t) &\leq -X(t)QX(t) + p_A - p_{\bar{A}} - \tilde{\phi}_A^T s(t) \\ &\quad + \gamma \text{tr}([\hat{L}_A^{-1} \dot{\hat{L}}_A \hat{L}_A^{-1}] \tilde{L}_A) \\ &\leq -X(t)QX(t) + p_A - p_{\bar{A}} - \text{tr}(\tilde{L}_A S) \\ &\quad + \gamma \text{tr}([\hat{L}_A^{-1} \dot{\hat{L}}_A \hat{L}_A^{-1}] \tilde{L}_A) \\ &\leq -X(t)QX(t) + p_A - p_{\bar{A}} \end{aligned}$$

which is in the sense of definition 1. Therefore, by using theorem 1, asymptotic stability of the whole system is established.

As demonstrated in (31), there is only one tuning gain for the estimation of n rigid bodies in the absence of lower and upper limits for the inertial parameters vector. In addition, the NAL function ensures that all the estimated parameters will remain physically consistent by ascertaining that \hat{L}_A will remain positive-definite [21].

IV. 7 DOF EXOSKELETON CONTROL

In this section, the modified VDC is employed to control the 7-DoF commercial ABLE exoskeleton. Assigned VDC frames are demonstrated in Fig. 1. Using these frames, the entire robot configuration is decomposed into subsystems of joints and rigid bodies. It is assumed that joints have negligible mass.

A. Rigid Body Control Equations

The velocity vector can be computed as

$${}^{B_i}V = {}^{T_{i-1}}U_{B_i}^T {}^{T_{i-1}}V + \varkappa_i \dot{q}_i \quad (32)$$

$${}^{T_i}V = {}^{B_i}U_{T_i}^T {}^{B_i}V \quad (33)$$

where $i = 1 \dots 7$, ${}^{T_0}V = {}^G V$ is the velocity vector of the ground, and ${}^{T_7}V$ is the velocity vector of the end-effector, q_i and \dot{q}_i are the joint angles and velocities, respectively, $\varkappa_i = z_\tau$ for $i = 1, 2, 3, 4, 6$, $\varkappa_i = y_\tau$ for $i = 7$, and $\varkappa_i = x_\tau$ for $i = 5$, which $z_\tau = [0, 0, 0, 0, 0, 1]^T$, $y_\tau = [0, 0, 0, 0, 1, 0]^T$, and $x_\tau = [0, 0, 0, 1, 0, 0]^T$. Moreover, ${}^{B_i}U_{T_i}$ can be computed with (2) by replacing B and A with T_i and B_i , respectively. The required velocities also can be expressed as

$${}^{B_i}V_r = {}^{T_{i-1}}U_{B_i}^T {}^{T_{i-1}}V_r + \varkappa_i \dot{q}_{ir} \quad (34)$$

$${}^{T_i}V_r = {}^{B_i}U_{T_i}^T {}^{B_i}V_r \quad (35)$$

where ${}^{T_0}V_r = {}^G V_r$ is the required velocity of the ground, and ${}^{T_7}V_r$ is the required velocity vector of the end-effector, and \dot{q}_{ir} can be defined as follows:

$$\dot{q}_{ir} = \mathbf{J}^{-1}(\dot{\chi}_d + \xi(\chi_d - \chi)) \quad (36)$$

where $\mathbf{J} \in \mathbb{R}^{6 \times 7}$ is the Jacobian matrix, $\dot{\chi}_d \in \mathbb{R}^6$ and $\chi_d \in \mathbb{R}^6$ are the desired velocity and pose of the end-effector in the Cartesian space, respectively, $\xi \in \mathbb{R}^{6 \times 6}$ is the diagonal, positive-definite matrix, and $\chi \in \mathbb{R}^6$ is the computed pose of the end-effector. Here, the defined end-effector's pose error is

$$e = \chi_d - \chi = \begin{bmatrix} e_p \\ e_o \end{bmatrix} \quad (37)$$

where $e_p = p_d - p$ is the common position error with $p = [x, y, z]^T$, and e_o is the quaternion-based orientation error, defined as

$$e_o = \eta(q)\epsilon_d - \eta_d\epsilon(q) - S(\epsilon_d)\epsilon(q) \quad (38)$$

where $\eta(q)$ and $\epsilon(q)$ are unit quaternions computed from the rotation matrix, η_d and ϵ_d are desired unit quaternions computed from the desired Euler angles of the end-effector with α, β, δ angles related to XYZ convention, and $S(\cdot)$ is a matrix operator. Now, the net force/moment vectors can be computed for each rigid body by replacing A with B_j in the (4) for $j = 7 \dots 1$. Then, the force/moment vectors can be expressed as

$${}^{B_j}F = {}^{B_j}U_{T_j}^T {}^{T_j}F + {}^{B_j}F^* \quad (39)$$

$${}^{T_{j-1}}F = {}^{T_{j-1}}U_{B_j} {}^{B_j}F \quad (40)$$

with required terms as

$${}^{B_j}F_r = {}^{B_j}U_{T_j}^T {}^{T_j}F_r + {}^{B_j}F_r^* \quad (41)$$

$${}^{T_{j-1}}F_r = {}^{T_{j-1}}U_{B_j} {}^{B_j}F_r \quad (42)$$

where ${}^{T_j}F$ is the interaction force between the end-effector and environment, and ${}^{T_0}F$ is the applied force from joint 1 to the ground. ${}^{B_j}F_r^*$ is the required net force/moment vector that is defined in (7).

$${}^{B_j}F_r^* = W_{B_j} \hat{\phi}_{B_j} + K_{B_j} ({}^{B_j}V_r - {}^{B_j}V). \quad (43)$$

As introduced in the previous section, each of $\hat{\phi}_{B_i}$ are updated by NAL (31) by replacing A with B_i in (31) with one adaptation gain γ for entire chain of rigid bodies.

B. Virtual stability

In this section, the virtual stability of the rigid body dynamics (15) under the control action of (43) is established.

Lemma 1. If the accompanying function for i^{th} links with the control of (43) and the adaptation law of (31) is chosen as

$$\begin{aligned} \nu_{T_i}(t) &= \frac{1}{2} ({}^{B_i}V_r - {}^{B_i}V)^T M_{B_i} ({}^{B_i}V_r - {}^{B_i}V) + \\ &\quad \frac{1}{2\gamma} \sum_{i=1}^n D_{F(\mathcal{P}(4))} (L_{B_i} \|\hat{L}_{B_i}) \end{aligned} \quad (44)$$

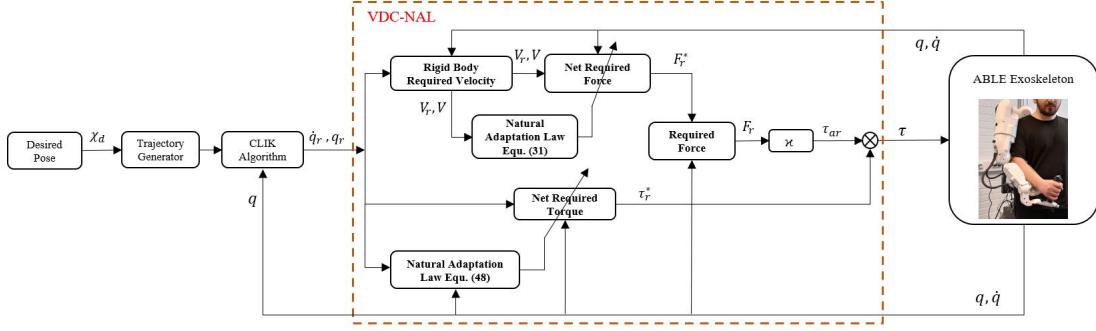


Fig. 2. Block diagram of control scheme

it results in

$$\dot{\nu}_{Ti}(t) = -(B_i V_r - B_i V)^T K_{B_i} (B_i V_r - B_i V) + (B_i V_r - B_i V)^T (B_i F_r^* - B_i F_r^*). \quad (45)$$

which shows the virtual stability of the link's dynamics in the sense of definition 1.

Proof. Proof can be established by means of proposition 2 and [13].

C. Joint Control Equations

The dynamic equations of joints in the absence of friction can be expressed as

$$I_{mi} \ddot{q}_i = \tau_i^* = \tau_i - \tau_{ai} \quad (46)$$

for $i = 1, \dots, 7$, I_{mi} are the moment of inertia of the i^{th} joint, τ_i^* denotes the net torque applied to the i^{th} joint, τ_i is the control torque of the i^{th} joint, and $\tau_{ai} = \mathcal{X}_i^T B_i F_r$. The required net torque for each joint can be defined as

$$\tau_{ir}^* = W_{ai} \hat{\phi}_{ai} + k_{ai} (\dot{q}_{ir} - \dot{q}_i) \quad (47)$$

where $k_{ai} > 0$, $W_{ai} = \mathcal{X}_i^T \ddot{q}_{ir}$, $\hat{\phi}_{ai}$ are estimations of inertial parameter vectors for each motor with the update function of

$$\dot{\hat{L}}_{ai} = \frac{1}{\gamma_a} \hat{L}_{ai} S_{ai} \hat{L}_{ai} \quad (48)$$

where \hat{L}_{ai} is corresponding pseudo inertia matrix. Therefore, the control torque can be designed as

$$\tau_i = \tau_{ir}^* + \tau_{air} \quad (49)$$

where $\tau_{air} = \mathcal{X}_i^T B_i F_r$.

D. Virtual stability

Lemma 2. If the accompanying function for the i^{th} joint with control action of (49) and adaptation law of (48) is chosen as

$$\nu_{ai}(t) = \frac{1}{2} I_{mi} (\dot{q}_{ir} - \dot{q}_i)^2 + \frac{1}{2\gamma_a} \sum_{i=1}^n D_F(L_{ai} \| \hat{L}_{ai}) \quad (50)$$

it results in

$$\dot{\nu}_{ai}(t) = -k_{ai} (\dot{q}_{ir} - \dot{q}_i)^2 + (\dot{q}_{ir} - \dot{q}_i) (\tau_{ir}^* - \tau_i^*). \quad (51)$$

which is virtually stable in the sense of definition 1.

Proof. Proof is the same as lemma 1.

Theorem 2. If the accompanying function of the entire system is considered as

$$\nu(t) = \sum_{i=1}^n \nu_{ai}(t) + \nu_{Ti}(t) \quad (52)$$

it follows,

$$\dot{\nu}(t) = - \sum_{i=1}^n ((B_i V_r - B_i V)^T K_{B_i} (B_i V_r - B_i V) + k_{ai} (\dot{q}_{ir} - \dot{q}_i)^2). \quad (53)$$

This shows that the entire system is asymptotically stable in the sense of theorem 1.

Proof. The procedure is the same as Appendix C of [23].

V. EXPERIMENTAL RESULTS

In this section, the experimental results are presented to exhibit the performance of the new adaptive VDC method. The commercial upper-limb exoskeleton ABLE is utilized for the experiments. The haptic technology that this robot contains makes it a perfect choice for bilateral teleoperation, which is the goal of our future work. All high-level processes are accomplished in a host computer, and computed torque signals are transmitted to the robot using SIMULINK and the Virtuoso interface, provided by Haption, with a sample time of 1 ms. Experiments are performed through the following scenario, as shown in Fig. 2. First, the desired Cartesian space targets are given to the 5th order trajectory generator [24], which generates smooth acceleration trajectories in Cartesian space. Then, the required joint velocities are derived by employing the Closed-Loop Inverse Kinematic (CLIK) algorithm [25], which is an online inverse kinematic solver. The computed required joint velocities are then fed to the proposed adaptive VDC method to compute the required net torques to execute Cartesian-space tasks.

To show that the new adaptive VDC has the same performance as the original one, simulation results related to the joint control of a 3-DoF spatial RRR robot utilizing both the original and proposed adaptive VDC are demonstrated in Fig. 3, where the solid lines are the tracking error of each joint with the projection function and the dashed lines correspond to the NAL function. The control objective is to

track the desire $\sin(t)$ trajectory for each joint, starting from the zero initial condition. All the adaptation and control gains are considered the same. As can be seen, the performance of the proposed adaptive VDC is as good as the original one, and the tracking error is almost zero. The difference is that the NAL function requires only one gain for tuning, but the original adaptive VDC requires 39 adaptation gains and 78 upper and lower limits. This shows that applying the proposed adaptive VDC to high-DoF systems can be highly time-efficient.

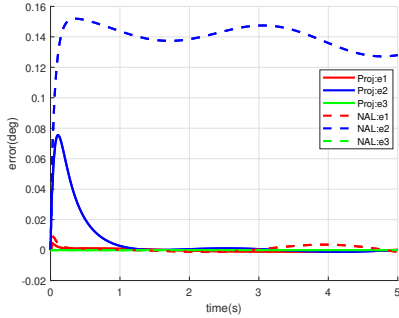


Fig. 3. Joint tracking error for 3 DoF robot with Projection and NAL function

In the following part, the experimental results of the pose control of the 7-DoF redundant exoskeleton are presented. It must be mentioned that to activate joint motors, the human operator must grasp the handle of the exoskeleton, which, as can be seen in Fig. 1, is the end-effector. Therefore, the intention force of the operator, which can be considered an unknown time-varying disturbance, affects the performance of the controller in reaching the desired pose. Moreover, the utilized exoskeleton is a redundant robot that can have different inverse kinematic solutions each time the robot performs a task, especially for different initial values of robot configuration. To have reliable results for the performance of the proposed controller, the experiment is repeated 6 times, and the means and standard deviations for position and orientation errors are computed. The utilized control and adaptation gains are as follows: $\xi = 25I_{6 \times 6}$, $K_{B_i} = 1.2$, $k_{a_i} = 0.1$, $\gamma = \gamma_a = 10$. The initial condition for the adaptation function of (31) is chosen in a way that satisfies the physical consistency condition and ${}^G V = {}^G V_r = {}^{T_r} F = {}^{T_r} F_r = \mathbf{0}_{6 \times 1}$. DH parameters and the Jacobian matrix are derived based on the convention explained in [26].

Fig. 4(a) demonstrates the mean and standard deviation of the position errors of the end-effector in each direction, and Fig. 4(b) shows a sample of desired path tracking. As can be seen in Fig. 4, the proposed adaptive VDC has excellent performance in tracking the desired path in Cartesian space. The results demonstrate that the adaptive VDC is robust against unknown time-varying disturbances and stably maintains the system throughout the working time. The average Root Mean Square Error (RMSE) of position errors for all experiments is $11.3 \pm 1mm$. Moreover, Fig. 5 shows the mean and standard deviation of the quaternion

orientation errors of the end-effector computed with (38). It can be seen that in the presence of unknown external time-varying disturbances, which are directly applied to the end-effector, adaptive VDC exhibits perfect tracking of the desired orientation. The average RMSE of orientation errors for all experiments is $1.06 \pm 0.25deg$.

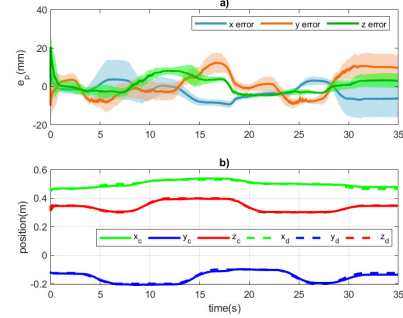


Fig. 4. End-effector position error a) The mean and standard deviation of position errors in 6 experiments b) Cartesian path following

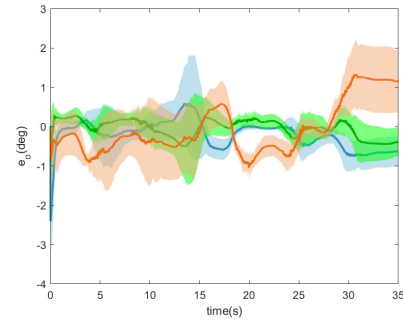


Fig. 5. End-effector orientation error

VI. CONCLUSIONS

In this paper, we proposed a new adaptive VDC approach that uses NAL to estimate unknown inertial parameters. The original adaptive VDC requires tuning of $13n$ adaptation gains and assigning $26n$ upper and lower parameter limits, which makes the process excessively tedious. However, the proposed adaptive VDC removes all the mentioned burdens and requires only one adaptation gain for all the rigid bodies. First, we reformulated the dynamic equations of the VDC context to put them into 10 unique inertial parameter vectors, making them compatible with the NAL function. Then, it was proved that employing the NAL function as the adaptation function for adaptive VDC culminates in asymptotic stability. Finally, the experimental results of applying the new adaptive VDC to the redundant upper-limb exoskeleton demonstrated the excellent performance of the VDC approach.

Our future work target is the control problem of two master exoskeleton robots that command heavy-duty hydraulic mobile manipulators as slaves for handling common objects in human-like manipulation scenarios with high scaling. As we showed in our earlier work [3], the high motion/force

scaling must be tackled in the controller design and stability analysis, which makes analysis much more complicated. Therefore, a novel adaptive VDC scheme developed in this paper will greatly lessen the implementation burden of this highly complex MMMS system.

REFERENCES

- [1] Sheridan, Thomas B. Telerobotics, automation, and human supervisory control. MIT press, 1992.
- [2] Azimifar, Farhad, et al. "Transparency performance improvement for multi-master multi-slave teleoperation systems with external force estimation." Transactions of the Institute of Measurement and Control 40.13 (2018): 3851-3859.
- [3] Lampinen, Santeri, et al. "Force-Sensor-Less Bilateral Teleoperation Control of Dissimilar Master-Slave System With Arbitrary Scaling." IEEE Transactions on Control Systems Technology (2021).
- [4] Riani, A., et al. "Adaptive integral terminal sliding mode control for upper-limb rehabilitation exoskeleton." Control Engineering Practice 75 (2018): 108-117.
- [5] Brahmi, Brahim, et al. "A new integral second-order terminal sliding mode control with time delay estimation for an exoskeleton robot with dynamics uncertainties." Proceedings of the 5th International Conference of Control, Dynamic Systems, and Robotics (CDSR'18). Vol. 113. 2018.
- [6] Brahmi, Brahim, et al. "Adaptive force and position control based on quasi-time delay estimation of exoskeleton robot for rehabilitation." IEEE Transactions on Control Systems Technology 28.6 (2019): 2152-2163.
- [7] Brahmi, Brahim, et al. "Novel adaptive impedance control for exoskeleton robot for rehabilitation using a nonlinear time-delay disturbance observer." ISA transactions 108 (2021): 381-392.
- [8] Brahmi, Brahim, et al. "Impedance learning control for physical human-robot cooperative interaction." Mathematics and Computers in Simulation 190 (2021): 1224-1242.
- [9] Zhang, Gao-Wei, et al. "Integrated observer-based fixed-time control with backstepping method for exoskeleton robot." International Journal of Automation and Computing 17.1 (2020): 71-82.
- [10] Wu, Qingcong, Bai Chen, and Hongtao Wu. "Rbf-based adaptive backstepping sliding mode control of an upper-limb exoskeleton with dynamic uncertainties." IEEE Access 7 (2019): 134635-134646.
- [11] Rahmani, Mehran, and Mohammad Habibur Rahman. "An upper-limb exoskeleton robot control using a novel fast fuzzy sliding mode control." Journal of Intelligent & Fuzzy Systems 36.3 (2019): 2581-2592.
- [12] Yip, P. Patrick, and J. Karl Hedrick. "Adaptive dynamic surface control: a simplified algorithm for adaptive backstepping control of nonlinear systems." International Journal of Control 71.5 (1998): 959-979.
- [13] Zhu, Wen-Hong. Virtual decomposition control: toward hyper degrees of freedom robots. Vol. 60. Springer Science & Business Media, 2010.
- [14] Shen, Youhao, et al. "An Adaptive Control Method for Electrohydrostatic Actuator Based on Virtual Decomposition Control." 2020 Asia-Pacific International Symposium on Advanced Reliability and Maintenance Modeling (APARM). IEEE, 2020.
- [15] Koivumäki, Janne, and Jouni Mattila. "Stability-guaranteed force-sensorless contact force/motion control of heavy-duty hydraulic manipulators." IEEE Transactions on Robotics 31.4 (2015): 918-935.
- [16] Humaloja, Jukka-Pekka, et al. "Decentralized Observer Design for Virtual Decomposition Control." IEEE Transactions on Automatic Control 67.5 (2021): 2529-2536.
- [17] Brahmi, Abdelkrim, et al. "Adaptive Control of Mobile Manipulator Robot based on Virtual Decomposition Approach." ICINCO (2). 2016.
- [18] Antonelli, Gianluca, Fabrizio Caccavale, and Stefano Chiaverini. "Adaptive tracking control of underwater vehicle-manipulator systems based on the virtual decomposition approach." IEEE Transactions on Robotics and Automation 20.3 (2004): 594-602.
- [19] Luna, Cristobal Ochoa, et al. "Virtual decomposition control of an exoskeleton robot arm." Robotica 34.7 (2016): 1587-1609.
- [20] Wensing, Patrick M., Günter Niemeyer, and Jean-Jacques E. Slotine. "Observability in inertial parameter identification." arXiv preprint arXiv:1711.03896 (2017).
- [21] Lee, Taeyoon, Jaewoon Kwon, and Frank C. Park. "A natural adaptive control law for robot manipulators." 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2018.
- [22] Wensing, Patrick M., Sangbae Kim, and Jean-Jacques E. Slotine. "Linear matrix inequalities for physically consistent inertial parameter identification: A statistical perspective on the mass distribution." IEEE Robotics and Automation Letters 3.1 (2017): 60-67.
- [23] Petrović, Goran R., and Jouni Mattila. "Mathematical modelling and virtual decomposition control of heavy-duty parallel–serial hydraulic manipulators." Mechanism and Machine Theory 170 (2022): 104680.
- [24] Jazar, Reza N. Theory of applied robotics. Springer Science+ Business Media, LLC, 2010.
- [25] Chiaverini, Stefano, and Bruno Siciliano. "The unit quaternion: A useful tool for inverse kinematics of robot manipulators." Systems Analysis Modelling Simulation 35.1 (1999): 45-60.
- [26] Sciacivico, Lorenzo, and Bruno Siciliano. Modelling and control of robot manipulators. Springer Science & Business Media, 2001.

APPENDIX

A. Restructured Dynamic Equations

To rewrite the C_A matrix, it is enough to rewrite $C_A(2, 2)$. By expanding $C_A(2, 2)^A \omega$, we get to

$$C_A(2, 2)^A \omega = ({}^A \omega \times) I_A ({}^A \omega) + I_A ({}^A \omega \times) ({}^A \omega) - m_A ({}^A r_{AB} \times) ({}^A \omega \times) ({}^A r_{AB} \times) ({}^A \omega) \quad (\text{A.1})$$

where it is obvious that $I_A ({}^A \omega \times) ({}^A \omega)$ is always zero. By using the fact that $a \times b = -b \times a$ along with

$$m_A ({}^A r_{AB} \times) ({}^A \omega \times) ({}^A r_{AB} \times) ({}^A \omega) = m_A {}^A \omega \times ({}^A r_{AB} \times)^2 {}^A \omega$$

(A.1) can be reformulated as

$$C_A(2, 2)^A \omega = ({}^A \omega \times) \bar{I}_A {}^A \omega.$$

Therefore, the C_A matrix can be written in the form of (14).

B. Deriving W_A and ϕ_A

By completing the multiplications on the right-hand side of (15), one can get,

$$\begin{bmatrix} m_A {}^A (\dot{v}_r) - m_A ({}^A r_{AB} \times) {}^A \dot{\omega}_r + m_A ({}^A \omega_r \times) \\ m_A ({}^A r_{AB} \times) {}^A (\dot{v}_r) + \bar{I} {}^A \dot{\omega}_r + m_A ({}^A r_{AB} \times) ({}^A \omega_r \times) {}^A v_r \\ - m_A ({}^A \omega_r \times) ({}^A r_{AB} \times) {}^A \omega_r + m_A {}^A R_{Ilg} \\ + ({}^A \omega_r \times) \bar{I} {}^A \omega_r + m_A ({}^A r_{AB} \times) {}^A R_{Ilg} \end{bmatrix} = {}^A F_r^* \quad (\text{B.1})$$

Then, by defining

$$vec I = [\bar{I}_{xx}, \bar{I}_{yy}, \bar{I}_{zz}, \bar{I}_{xy}, \bar{I}_{yz}, \bar{I}_{xz}]^T$$

$$({}^A \omega) = \begin{bmatrix} \omega_x & 0 & 0 & \omega_y & 0 & \omega_z \\ 0 & \omega_y & 0 & \omega_x & \omega_z & 0 \\ 0 & 0 & \omega_z & 0 & \omega_y & \omega_x \end{bmatrix}$$

and obtaining $a \times b = -b \times a$, (B.1) can be rewritten as

$$W_A ({}^A \dot{v}_r, {}^A \dot{\omega}_r, {}^A v_r, {}^A \omega_r, {}^A R_{Ilg}) \phi_A = {}^A F_r^*$$

where

$$W_A = \begin{bmatrix} {}^A \dot{v}_r + ({}^A \omega_r \times) {}^A v_r + {}^A R_{Ilg} \\ \mathbf{0}_{3 \times 1} \\ {}^A \dot{\omega}_r + ({}^A \omega_r \times)^2 \\ - (({}^A (\dot{v}_r) \times) + ({}^A \bar{\omega} v \times) + ({}^A R_{Ilg} \times)) \\ \mathbf{0}_{3 \times 6} \\ ({}^A \dot{\omega}_r) + ({}^A \omega_r \times) ({}^A \omega_r) \end{bmatrix} \quad (\text{B.2})$$

and $\phi_A = [m_A, h_A^T, vec I^T]^T$, $h = m_A {}^A r_{AB}$, and ${}^A \bar{\omega} v = ({}^A \omega_r \times) {}^A v_r$.