Proceedings of the ASME 2007 International Design Engineering Technical Conferences &
Computers and Information in Engineering Conference
IDETC/CIE 2007
September 4-7, 2007, Las Vegas, Nevada, USA

DETC2007-35393

LOAD-EFFECTIVE DYNAMIC MOTION PLANNING FOR REDUNDANT MANIPULATORS

Joo H. Kim* joo-kim@uiowa.edu
Jingzhou Yang jyang@engineering.uiowa.edu
Karim Abdel-Malek amalek@engineering.uiowa.edu

U.S. Army Virtual Soldier Research Program
Center for Computer-Aided Design
The University of Iowa, Iowa City, IA 52242, U.S.A.

ABSTRACT
The robotic motion planning criteria has evolved from kinematics to dynamics in recent years. Many research achievements have been made in dynamic motion planning, but the externally applied loads are usually limited to the gravity force. Due to the increasing demand for generic tasks, the motion should be generated for various functions such as pulling, pushing, twisting, and bending. In this presentation, a comprehensive form of equations of motion, which includes the general external loads applied at any points of the system, is derived and implemented. An optimization-based algorithm is then developed to generate load-effective motions of redundant manipulators (single-loop and tree-structured chains) that guarantee the execution of the generic tasks under limited actuator capacities. It is shown that if the external loads are not incorporated in the motion planning formulation, then the generated motions do not always guarantee the execution of the task, especially when a large load is desired. By using our algorithm, the load-effective motions can be found that are executable for given external loads. The proposed method is also applicable in predicting realistic dynamic human motions. Some dual-arm human tasks are simulated to show different motions to sustain different amounts of external loads. Our formulation for general external loads will further advance the current motion planning methods for redundant manipulators.
Key words: motion planning, equation of motion, load-effective motion, optimization, redundant manipulator, general external load.

1 INTRODUCTION
One of the major applications of robotic technologies lies in improving human life by utilizing robots in jobs that are hazardous, difficult, or undesirable for humans. Recently, dependency on robots for the tasks associated with these jobs has been growing very rapidly. The robots are also used for providing services in the public and private sectors. This trend will become more popular in the future as the realizable technologies grow and the cost of the products decreases. Broadening the applications of robots usually requires adding functions and increasing accuracies and capabilities. One way to achieve this is to increase the mobility of robots such that multiple alternate motions exist for performing an assigned task. In other words, higher-level robots must possess redundancy. The redundancy of robots will provide higher flexibility, dexterity, manipulability, and controllability. Good examples of these higher-level robots are seen in the recent developments of humanoids, bio-inspired robots, and space robots.

Despite great achievements in the research of redundant robots, there are still many unsolved problems in areas such as mechanics, control, design, and intelligence. In particular, the motion planning methods for redundant robots in the current literature do not extensively address the exertion of general external loads (forces and moments) other than gravity. Non-redundant manipulators have only one possible configuration at a time, and whether or not a task can be accomplished is determined by that configuration; redundant manipulators can possess an infinite number of configurations at a time, and successful task accomplishment depends on the proper choice of configurations. Thus, it is important to investigate appropriate methods of generating redundant manipulator motions that provide the desired results.

While most of the motion planning methods in the literature had been based on spatial kinematics [1, 2], the recent research on

dynamics-based motion planning has shown huge achievement. This is because motion planning based on kinematic constraints alone is not always satisfactory in real execution. Since the planned motions are supposed to be executed in a real physical environment, the selection of the motion should incorporate the dynamic constraints. The redundancy of the systems requires the selection of the best configuration among many admissible ones, and thus the majority of the studies used optimization methods. Some studies used gradient-based numerical optimization; some derived the pseudoinverse analytically from optimization; and others used a mixture of the pseudoinverse and numerical optimization [3]. Also, three- or four-degree-of-freedom (DOF) planar manipulators are widely used as a simple redundant system for demonstration purposes.

Due to the closed form and low-cost computations, the pseudoinverse, or generalized inverse, of the Jacobian matrices are popular tools for resolution of kinematic redundancy. Most of the Jacobian pseudoinverse matrices in literature were derived from minimum kinetic energy criteria [4, 5]. Nedungadi and Kazerouinian [6] optimized weighted torque and kinetic energy by the method of calculus of variation combined with the pseudo-Jacobian matrix, where local and global optimal forms were demonstrated for comparison. Chung *et al*. [7] introduced a Jacobian pseudoinverse that represents a minimization of the pseudo-elastic potential energy due to actuator stiffness. A control scheme for redundant manipulators is developed by optimizing a norm of actuator torques using the weighted generalized inverses of the Jacobian matrix [8]. In this method, the equation of motion that includes the contact forces due to target impedance was used, where the contact is modeled with inertia, damping, and stiffness.

Many methods on redundant manipulator motion planning are developed based on gradient-based numerical optimization. Various approximate forms of energy consumption or effort were minimized while the dynamic equations of motion were used either to derive the cost function or to impose dynamic constraints [9, 10, 11]. On the other hand, more exact forms of energy consumption of electric motors and hydraulic actuators were derived by combining the equations of motion, and these were minimized for optimal motion planning subject to dynamic constraints [12, 13]. The equations of motion used in these studies include the inertia, Coriolis, centrifugal, and gravity terms, and thus the external loads are limited to the weights of the links or the objects at the end-effector. In all the aforementioned studies, however, the general form of external loads such as pushing/pulling forces and bending/twisting moments are not incorporated in the dynamic motion planning formulation.

Some researchers investigated the methods of determining manipulator configurations to sustain general external loads (wrench) for static postures or quasi-static motions [14, 15, 16]. Papadopoulos and Gonthier [15, 16] developed a method of determining the base position and configuration of the manipulator that guarantees the execution of a large-force task under limited actuator torques. The maximum normalized torque to generate the manipulator postures is minimized subject to general external loads and gravity in static and quasi-static states, while the dynamic effects of the accelerations and velocities are not taken into account. Although the quasi-static evaluation of actuator torques gives reasonable approximate values for manipulators with relatively small link masses and low velocities, the inaccuracy of the calculation increases as the link masses, the velocities, or the accelerations of the motion become larger (e.g., manipulators used for construction) due to the significant contributions of the inertia forces.

Since human bodies can be modeled as redundant systems, many optimization-based methodologies have been shown to be valuable tools for human motion simulation [17, 18, 19, 20]. This is based on the underlying assumption that a human moves in a way that minimizes a cost function (e.g., energy consumption) subject to several constraints. As is the case for the manipulator motion planning, the implementation of general external loads is not addressed extensively in literature for the human motion prediction problem. For realistic physics-based human motion prediction, it is essential to consider all components of dynamics.

Based on the achievements and problems of previous works on motion planning, improvements can be made in the generalization of robotic tasks. In other words, the task categories should be extended from simple lifting and moving (against gravity) to generic tasks such as pulling, pushing, bending, and twisting. The objectives and contribution of this research are summarized below:

(1) The most general form of Lagrange's equations of motion will be derived for single open-loop chains and tree-structured systems. The equations of motion for robotic systems are well known and have been widely used in literature for a long time, and the external contact forces are implemented in some control-based articles [8]. However, in addition to the interaction with external contacts, manipulators are usually required to exert various general loads such as pulling forces and twisting moments. So far, the comprehensive form of dynamics that includes inertia, Coriolis, centrifugal, gravity, and general external loads applied at any points of interest on an open-loop system is rarely presented in literature as a single equation. In addition, although various formulations for the dynamics of tree structure were developed in the literature, each is based on its own kinematic representation [21, 22, 23, 24]. Therefore, it is desired to formulate the dynamics of tree-structured systems within the framework of our approach, i.e., based on the Lagrangian equations of motion developed for the Denavit-Hartenberg (DH) representation method [25, 26] and generalized coordinates. The complete form of the equations of

motion will be used as a critical subroutine for generation of redundant manipulator motions.

(2) As the main contribution of this article, the methodology of planning load-effective motions of redundant manipulators will be proposed. We define the load-effective motions as the generated motions that guarantee the execution of a task with given general external loads of a broad range of magnitudes. By implementing the comprehensive equations of motion, it is possible to generate optimal manipulator motions in which the general external loads are incorporated. In particular, it will be shown that the motions generated without taking the external loads into account may be unexecutable under given actuator capacity limits, especially when a large load is desired. A three-DOF planar manipulator will be used to demonstrate the formulation.

(3) The framework of dynamic motion planning for redundant manipulators with general external loads will be applied to predict realistic human motions. Different human motions according to different amounts of external loads will demonstrate how humans react effectively to perform a task. The examples using a dual-arm digital human model will illustrate our method in a three-dimensional space.

In the following sections, the DH kinematic modeling of serial and tree-structured chains is briefly described. Then a comprehensive form of dynamic equations of motion that include the general external loads will be derived. Next, the structure and components of the optimal motion planning problem are presented in detail. Finally, some examples of results of a redundant manipulator and a tree-structured human model will be discussed.

2 KINEMATIC MODELING OF OPEN-LOOP CHAINS

The analysis of general robotic systems is always concerned with the configurations, velocities, and accelerations of objects (e.g., robotic links, tools, environment) in three-dimensional space. Given an open-loop kinematic chain (Figure 1), the transform from frame $\{i\}$ to frame $\{i\text{-}1\}$ can be represented by the homogeneous transformation matrix ${}^{i-1}\mathbf{T}_i$.

$$ {}^{i-1}\mathbf{T}_i = \begin{pmatrix} \cos\theta_i & -\cos\alpha_i \sin\theta_i & \sin\alpha_i \sin\theta_i & a_i \cos\theta_i \\ \sin\theta_i & \cos\alpha_i \cos\theta_i & -\sin\alpha_i \cos\theta_i & a_i \sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (1) $$

If the joint is revolute, θ_i is called the joint variable, and the other three quantities, d_i, α_i, and a_i, are called link parameters. If the joint is prismatic, the joint variable is d_i, while the other three are the link parameters [27]. The convention is based on DH notation, which describes the configuration of a kinematic chain (Figure 2).

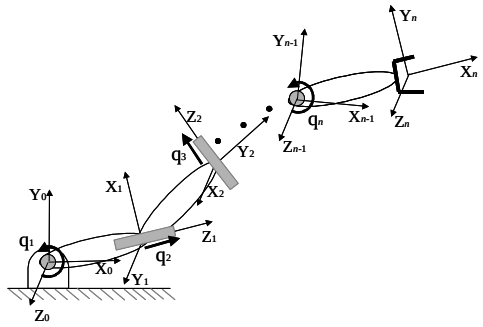


Figure 1. An *n*-DOF open-loop kinematic chain

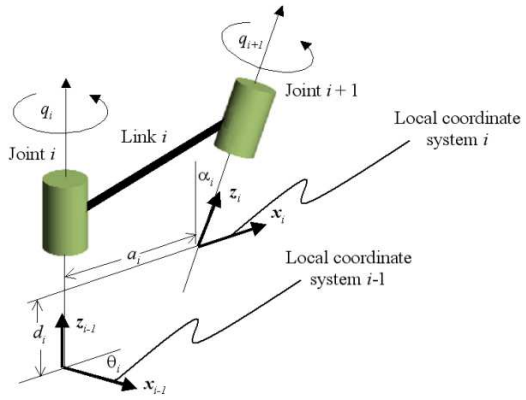


Figure 2. DH convention and parameters

The homogeneous transform includes the translation and rotation of one coordinate frame relative to another, where each frame is attached, respectively, to a rigid body in space. The homogeneous transformation matrix ${}^0\mathbf{T}_n$ that relates frame $\{n\}$ to frame $\{0\}$ can be obtained by multiplying all of the intermediate transforms.

$$ {}^0\mathbf{T}_n(q_1,...,q_n) = {}^0\mathbf{T}_1(q_1)\,{}^1\mathbf{T}_2(q_2)...\,{}^{n-1}\mathbf{T}_n(q_n) \quad (2) $$

where q_i is the joint variable of ${}^{i-1}\mathbf{T}_i$, and the set of joint variables $\mathbf{q} = [q_1, ..., q_n]^T \in \mathbf{R}^n$ is called the $n\times 1$ joint vector. Therefore, ${}^0\mathbf{T}_n$ is a function of all *n* joint variables. These joint variables uniquely determine the configuration of a manipulator system with *n* DOFs and are called the generalized coordinates. Then the position vector of a point of interest attached to the frame $\{n\}$ of the end-effector can be written with respect to the global frame $\{0\}$ using the joint variables.

Next, consider the kinematic modeling method of tree-structured systems using DH representation. A tree-structured kinematic chain is a connected group of multiple open-loop serial chains called branches (Figure 3). It has connection links, which connect the branches. The numbering method is the same as a regular serial chain for one loop (loop A, for example). For the other loop (B), the link that is attached to the connection link can be given any arbitrary number *j*, as long as

it is greater than or equal to the last link number of loop A plus two. In other words, $j \geq n+2$ in Figure 3. Then the numbering system of the joints and the local coordinate frames follow the usual convention. The mathematical expression for the configuration of a tree-structured kinematic chain can be described by using the homogeneous transformation matrices for each loop (loops A and B). Careful attention is required, however, when deriving the transformation matrices of the connection link for loops A and B. Since there are three joints at the connection link (Figure 4), different transformation matrices are assigned to each loop. For loop A, the usual transformation matrix ${}^{i-1}\mathbf{T}_i$ is used to relate the local coordinate frames {i-1} and {i}. However, to relate the local coordinate frames {i-1} and {j-1} for loop B, another transformation matrix ${}^{i-1}\mathbf{T}_{j-1}$ should be constructed. Note that the local coordinate frames {i} and {j-1} are all attached to the connection link (link i), whereas the local frame {i-1} is attached to link {i-1}. Since it is assumed that the links are all rigid, the kinematic relation between the {i} and the {j-1} local frames does not change, i.e., the two local coordinate frames are rigidly connected. Therefore, the transformation matrix ${}^{i-1}\mathbf{T}_{j-1}$ for loop B is also a function of the i^{th} joint variable.

$$ {}^{i-1}\mathbf{T}_{j-1} = {}^{i-1}\mathbf{T}_{j-1}(q_i) \tag{3} $$

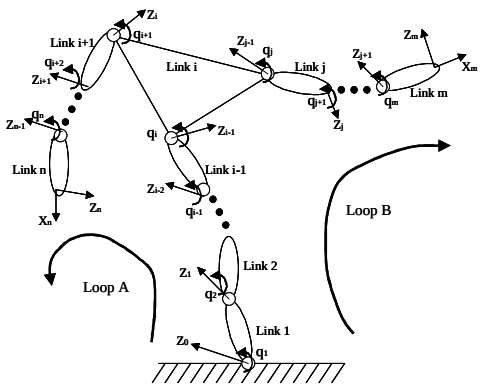


Figure 3. A tree-structured kinematic chain.

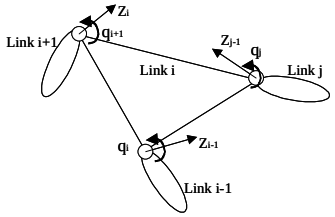


Figure 4. A connection link and its branches.

Using this matrix, the total transformation matrix that relates the global coordinate frame and the end-effector local coordinate frame through loop B is calculated as below. Whenever there are more than two branches connected with the connection link, a similar process can be used to derive the homogeneous transformation matrices to express the kinematic relationships.

$$ \begin{aligned} {}^{0}\mathbf{T}_m(\mathbf{q}) &= {}^{0}\mathbf{T}_1(q_1)\ldots{}^{i-2}\mathbf{T}_{i-1}(q_{i-1})\,{}^{i-1}\mathbf{T}_{j-1}(q_i)\,{}^{j-1}\mathbf{T}_j(q_j)\ldots{}^{m-1}\mathbf{T}_m(q_m) \\ &= {}^{0}\mathbf{T}_{i-1}\cdot{}^{i-1}\mathbf{T}_{j-1}\cdot{}^{j-1}\mathbf{T}_m \end{aligned} \tag{4} $$

The human body is a typical example of tree-structured systems with multiple limbs. The 30-DOF Santos™ digital human model of a human torso, right arm, and left arm is shown in Figure 5. There are twelve shared DOFs in the torso. The DH representation method will be used for the mathematical expression of end-effector coordinates.

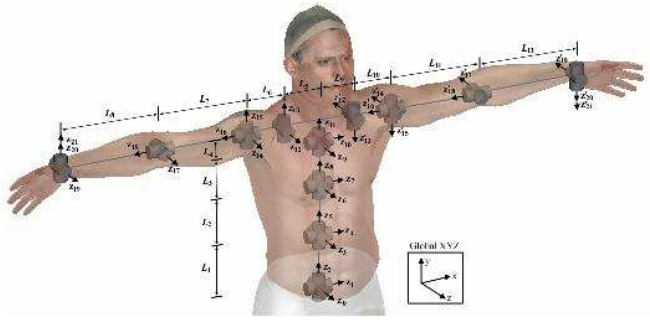


Figure 5. A dual-arm human model.

The global Cartesian position vectors $\mathbf{x}_R$ and $\mathbf{x}_L$ of the right-arm and left-arm end-effectors, respectively, can be calculated as shown below:

$$ \begin{aligned} \mathbf{x}_R(\mathbf{q}) &= (\mathbf{T}_{torso})\left({}^{11}\mathbf{T}_{12}\right)\left(\prod_{i=13}^{21}{}^{i-1}\mathbf{T}_i\right)\mathbf{x}_{Rn}; \\ \mathbf{x}_L(\mathbf{q}) &= (\mathbf{T}_{torso})\left({}^{11}\mathbf{T}_{22}\right)\left(\prod_{i=23}^{31}{}^{i-1}\mathbf{T}_i\right)\mathbf{x}_{Ln}; \\ \mathbf{T}_{torso} &= \prod_{i=1}^{11}{}^{i-1}\mathbf{T}_i \end{aligned} \tag{5} $$

where $\mathbf{x}_{Rn}$ and $\mathbf{x}_{Ln}$ are the 4×1 local frame position vectors for the right-arm and left-arm end-effectors, respectively. $\mathbf{T}_{torso}$ is the transformation matrix of the last local coordinate frame of the torso with respect to the global coordinate frame. ${}^{11}\mathbf{T}_{12}$ and ${}^{11}\mathbf{T}_{22}$ are the transformation matrices for the first local coordinate frame of the right arm and the left arm, respectively, in terms of the last local coordinate frame of the torso. $\prod_{i=13}^{21}{}^{i-1}\mathbf{T}_i$ and $\prod_{i=23}^{31}{}^{i-1}\mathbf{T}_i$ are the transformation matrices for the last local coordinate frame of the right arm and the left arm, respectively, in terms of the first local coordinate frame of the limb.

3 LAGRANGE'S EQUATIONS OF MOTION

To generate the motions of a manipulator where the externally applied forces and moments are taken into account, it is

essential to formulate a comprehensive expression of the equations of motion that govern the dynamics of open-loop kinematic chains. The implementation of the equations of motion is also necessary for control of the manipulators, although the control problems are not discussed here. In this presentation, we use Lagrangian dynamics, which provides a systematic method of formulating the equations of motion that are described in terms of the independent generalized coordinates. The use of Lagrangian dynamics also allows relatively easy implementation of any kinematic constraints and the corresponding constraint forces and moments.

For the motion of a mechanical system with finite DOFs, the Hamilton's principle in a given time interval $[t_0,\ t_1]$ is formulated as the following variational equation [28]:

$$\delta \int_{t_0}^{t_1} (T + W) dt = 0 \tag{6}$$

where T is the total kinetic energy of the system, W is the virtual work of the noninertial forces, and δ denotes the first variation. Note that T and W are functions of $\mathbf{q}$, $\dot{\mathbf{q}}$, and t. In theory, the variational form of the Hamilton's principle, Equation (6), can be solved for the generalized coordinates $\mathbf{q}(t)$ with respect to time, which is a classic problem of the calculus of variations. Several rigorous conditions should be imposed to define and solve this problem, but the details are omitted here. For a holonomic system, it can be shown that the solution $\mathbf{q}(t)$ of this problem satisfies the following general form of Lagrange's equations of motion (in vector-matrix form).

$$\frac{d}{dt}\frac{\partial (W+T)}{\partial \dot{\mathbf{q}}} - \frac{\partial (W+T)}{\partial \mathbf{q}} = \mathbf{0} \tag{7}$$

The virtual work is the sum of the virtual works done by conservative forces and non-conservative forces, i.e., $W = W_c + W_{nc}$. The variation of the conservative work can be expressed as the negative variation of the potential energy of the force system, i.e., $\delta W_c = -\delta V$. Thus we can write

$$\delta (W+T) = \delta T - \delta V + \delta W_{nc} = \delta L + \delta W_{nc} \tag{8}$$

where V is the total potential energy of the system and $L = T - V$ is the Lagrangian function. Then Equation (7) can be written as follows:

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\mathbf{q}}} - \frac{\partial L}{\partial \mathbf{q}} - \frac{d}{dt}\frac{\partial W_{nc}}{\partial \dot{\mathbf{q}}} - \frac{\partial W_{nc}}{\partial \mathbf{q}} = \mathbf{0} \tag{9}$$

Let us consider the case where a general form of external loads (sometimes called wrench) $\begin{bmatrix} \mathbf{F}_k^T & \mathbf{M}_k^T \end{bmatrix}^T$ is applied to the point at ${}^k\mathbf{r}_k$ location of link k, where $\begin{bmatrix} \mathbf{F}_k^T & \mathbf{M}_k^T \end{bmatrix}^T$ is a 6×1 vector comprised of a 3×1 force vector $\mathbf{F}_k$ and a 3×1 moment vector $\mathbf{M}_k$, and ${}^k\mathbf{r}_k$ is a 4×1 position vector expressed in terms of $\{k\}$ local coordinate frame attached to link k. Note that the constraint forces and moments due to external constraints from the environment can also be expressed in this manner. Equation (9) can be expanded using the kinetic energy, the potential energy, and the extended form of non-conservative work (for details, see [20]). Assuming that the velocity-dependent force does not exist, i.e., $d(\partial W_{nc}/\partial \dot{\mathbf{q}})/dt = 0$, the final vector-matrix form of the equations of motion for a general open-loop kinematic chain with general external loads is given below as a nonlinear, second-order ordinary differential equation.

$$\boldsymbol{\tau} = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{V}(\mathbf{q},\dot{\mathbf{q}}) + \sum_i \mathbf{J}_i^T m_i \mathbf{g} + \sum_k \mathbf{J}_k^T \begin{bmatrix} -\mathbf{F}_k \\ -\mathbf{M}_k \end{bmatrix} + \mathbf{T}(\mathbf{q},\dot{\mathbf{q}}) \tag{10}$$

where $\boldsymbol{\tau} = [\tau_1, \tau_2, ..., \tau_n]^T$ is the actuator torque vector, $\mathbf{M}(\mathbf{q})$ is the mass-inertia symmetric matrix, $\mathbf{V}(\mathbf{q},\dot{\mathbf{q}})$ is the Coriolis and centrifugal force vector, $\sum \mathbf{J}_i^T m_i \mathbf{g}$ is the joint torque vector due to gravity, $\mathbf{J}_i$ is the Jacobian matrix of the position vector for the center of mass of i^{th} link, and $\mathbf{J}_k$ is the augmented Jacobian matrix of the position vector ${}^k\mathbf{r}_k$ with respect to $\{k\}$ local coordinate frame. The vector $\mathbf{T}(\mathbf{q},\dot{\mathbf{q}})$ is the torque vector due to the joint stiffness and the dissipative forces such as viscous damping and Coulomb friction. The augmented Jacobian matrix $\mathbf{J}_k(\mathbf{q})$ is derived from the linear relationship between the tangent spaces of the joint variables and the Cartesian coordinates.

$$\mathbf{J}_k(\mathbf{q}) = \begin{bmatrix} \mathbf{J}_{k,1}(\mathbf{q}) & \cdots & \mathbf{J}_{k,i}(\mathbf{q}) & \cdots & \mathbf{J}_{k,k}(\mathbf{q}) \end{bmatrix}_{6\times k} \tag{11}$$

where the i^{th} column vectors for revolute and prismatic joints are, respectively,

$$\mathbf{J}_{k,i}^{revolute}(\mathbf{q}) = \begin{bmatrix} \dfrac{\partial\, {}^0\mathbf{T}_k(\mathbf{q})}{\partial q_i}\, {}^k\mathbf{r}_k \\ {}^0\mathbf{z}_{i-1}(\mathbf{q}) \end{bmatrix}_{6\times1} ;\ \mathbf{J}_{k,i}^{prismatic}(\mathbf{q}) = \begin{bmatrix} \dfrac{\partial\, {}^0\mathbf{T}_k(\mathbf{q})}{\partial q_i}\, {}^k\mathbf{r}_k \\ \mathbf{0}_{3\times1} \end{bmatrix}_{6\times1}.$$

Here, ${}^0\mathbf{z}_{i-1}$ $(i = 1, ..., k)$ is the local z-axis vector of $\{i\}$ local frame expressed in terms of the global coordinate frame. Note that in the above equation, only the first three elements of the 4×1 vector $(\partial\, {}^0\mathbf{T}_k(\mathbf{q})/\partial q_i)\, {}^k\mathbf{r}_k$ are used to assemble the first three rows of $\mathbf{J}_{k,i}(\mathbf{q})$.

Equation (10) is derived for a single-loop kinematic chain where the first joint is rigidly attached to the global reference frame. To apply the equations of motion subroutine that is coded for single-loop chains to tree-structured systems, some modifications are necessary. Consider a tree-structured system with three branches under a set of forces and moments (Figures 3 and 6). This tree structure is composed of three local branch chains, C_1, C_2, and C_3. The branch chain C_1 is composed of

links from link 1 to link *i*, C_2 from link *i* + 1 to link *n*, and C_3 from link *j* to link *m*.

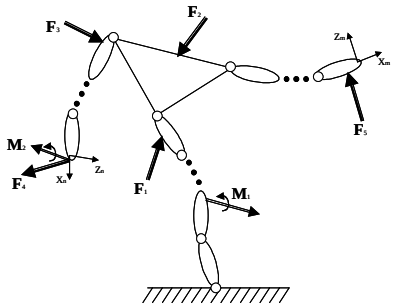


Figure 6. A tree-structured system with general external loads.

The Lagrangian function of the whole system is a scalar quantity and is thus calculated as the summation of the Lagrangian function of each branch. Therefore the Lagrange's equation of motion for the tree structure is written as the summation of the actuator torque vector at each branch chain.

$$\boldsymbol{\tau} = \sum_{i=1}^{3}\left(\frac{d}{dt}\frac{\partial L}{\partial \dot{\mathbf{q}}} - \frac{\partial L}{\partial \mathbf{q}} - \sum \mathbf{J}_k^T \begin{bmatrix} \mathbf{F}_k \\ \mathbf{M}_k \end{bmatrix} \right)_{C_i} + \mathbf{T}(\mathbf{q}, \dot{\mathbf{q}}) \,. \qquad (12)$$

Since the branch chains C_1 and C_2 constitute loop A, and C_1 and C_3 constitute loop B, Equation (12) can be written in the following form:

$$\boldsymbol{\tau} = \boldsymbol{\tau}\big|_{loopA} + \boldsymbol{\tau}\big|_{loopB} - \boldsymbol{\tau}\big|_{C_1} + \mathbf{T}(\mathbf{q}, \dot{\mathbf{q}}) \qquad (13)$$

where $\boldsymbol{\tau}\big|_{loopA}$ is the actuator torque vector with all elements zero except for those of the open-loop chain *A*, i.e., from 1st element to n^{th} element. The vectors $\boldsymbol{\tau}\big|_{loopB}$ and $\boldsymbol{\tau}\big|_{C_1}$ are defined in a similar manner.

$$\boldsymbol{\tau}\big|_l = \left(\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{V}(\mathbf{q}, \dot{\mathbf{q}}) + \sum_i \mathbf{J}_i^T m_i \mathbf{g} + \sum_k \mathbf{J}_k^T \begin{bmatrix} -\mathbf{F}_k \\ -\mathbf{M}_k \end{bmatrix} \right)_l \qquad (14)$$

$(l = loopA,\ loopB,\ C_1)$

For dual-arm human motion, let us assume that the only generalized torque vector due to the joint internal characteristics is the restoring torque vector $\boldsymbol{\tau}^{\text{Restoring}}$ from the joint stiffness.

$$\mathbf{T}(\mathbf{q}, \dot{\mathbf{q}}) = -\boldsymbol{\tau}^{Restoring} = \mathbf{K}\left(\mathbf{q} - \mathbf{q}^N\right) \qquad (15)$$

where $\mathbf{K}$ is a diagonal stiffness matrix, and $\mathbf{q}^N$ is the neutral joint variable vector [20]. Then the equations of motion for a human model subject to several external loads can be written as follows.

$$\boldsymbol{\tau} = \boldsymbol{\tau}\big|_{loopA} + \boldsymbol{\tau}\big|_{loopB} - \boldsymbol{\tau}\big|_{C_1} + \mathbf{K}\left(\mathbf{q} - \mathbf{q}^N\right) \qquad (16)$$

4 PROBLEM DEFINITION AND OPTIMIZATION FORMULATION

The problem of dynamic motion planning for redundant manipulators is defined as follows (Figure 7): The inputs to the algorithm are the link parameters of the manipulator, dynamic parameters (such as mass, centers of mass, moments and products of inertia, joint stiffness, and damping coefficients), joint variable limits, actuator torque limits (possibly as functions of joint velocity), points of application and components of external loads, and task-based constraints (such as the time desired to perform the task, end-effector path and orientations). Then it is desired to generate the joint profiles that guarantee the execution of the task, where the external loads can have broad ranges of magnitudes. To resolve the redundancy, the problem is formulated as an optimization problem, where the outputs are the joint variable profiles, the required actuator torques, and the energy rates as functions of time. The proposed optimal motion planning problem is stated as:

Find: Joint control points ($\mathbf{P}_{(nc\times n)}$)

to minimize: Energy consumption (E)

subject to constraints:

Joint limits ($\mathbf{q}^L \le \mathbf{q} \le \mathbf{q}^U$)

Actuator torque limits ($\boldsymbol{\tau}^L \le \boldsymbol{\tau} \le \boldsymbol{\tau}^U$)

Path constraints ($\| \mathbf{x}(\mathbf{q}(t)) - \mathbf{path}(t) \| \le \varepsilon$)

It should be emphasized that the general external loads term, as well as the inertia and gravity, must be included in the calculation of actuator torques to guarantee the execution of the planned motion. Then the load-effective motions are the feasible motions where the general external loads are taken into account for the calculation of required actuator torques that are used for the constraints (torque limits) and/or the cost function. The energy consumption or the norm of torque vector used as the cost function will result in the most efficient motion. If the optimization is solved while the external loads are not considered for the dynamics calculation, then the actual required actuator torques may exceed the maximum torque limits that the actuators can provide; thus, the planned motion may not be executable (i.e., uncontrollable) in reality, especially when large external loads are desired. This will be illustrated later in our examples.

Due to the redundancy of the human body, the approach for human motion prediction is basically the same as that described for the redundant manipulators. This is based on the assumption that humans naturally generate effective motion to accomplish a given task in such a way as to minimize certain cost function(s). Note that the problem for manipulators is "motion planning," while for humans it is "motion prediction."

For convenience, we will use the term "motion generation" for both manipulators and humans. The details of each component of the optimization problem are described in the following sections.

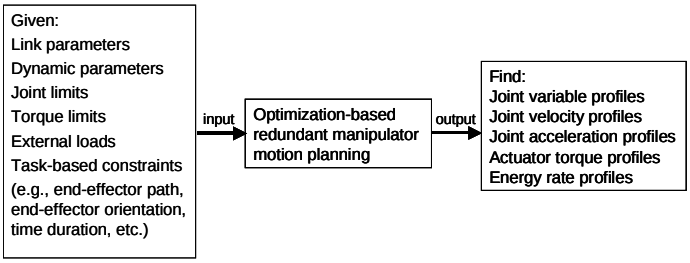


Figure 7. Problem of motion planning for redundant manipulators

For the numerical optimization algorithm, we use the sequential quadratic programming (SQP) method. The SQP uses quasi-Newton approximations to the Hessian of the augmented Lagrangian and obtains search directions from a sequence of quadratic programming subproblems. SQP methods have proved reliable and highly effective for solving constrained optimization problems with smooth nonlinear cost function and constraints. The details on the SQP method can be found in optimization texts.

4.1 Joint Variable Profiles Using B-spline Curves

Since the joint variables as functions of time are non-uniform curves, we use the B-spline curves [29], which have many beneficial properties such as continuity, differentiability, endpoint interpolations, local control, and convex hull. We use the recursive formula to represent the B-splines, such that its control points will be calculated as a result of the iterative numerical optimization algorithm. Let *nk* be the number of knots and $\mathbf{U} = \{u_0, \ldots, u_{nk-1}\}$ be a knot vector with a non-decreasing sequence of knots. The i^{th} B-spline basis function of *p*-degree (order $p + 1$), denoted by $N_{i,p}(u)$, is defined as

$$N_{i,0}(u) = \begin{cases} 1 & \text{if } u_i \le u < u_{i+1} \\ 0 & \text{otherwise} \end{cases};$$
$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u) \qquad (17)$$

To enforce endpoint interpolations, we choose a $(nk \times 1)$ non-periodic knot vector that has multiplicities at the start and the end as shown below.

$$\mathbf{U} = \{\underbrace{a, \ldots, a}_{p+1}, u_{p+1}, \ldots, u_{nk-p-2}, \underbrace{b, \ldots, b}_{p+1}\}$$

Then the p^{th}-degree B-spline curve for each joint can be written as

$$q_j(u) = \sum_{i=0}^{nc-1} N_{i,p}(u) P_{i,j} \quad (a \le u \le b;\ j = 1, \ldots, n) \qquad (18)$$

where *n* is the total DOF of the system, *nc* is the number of control points, and $\{P_{i,j}\}$ are the (*i*, *j*) components of the control points matrix $\mathbf{P}_{(nc \times n)}$. Here, the degree *p*, the number of the control points *nc*, and the number of knots *nk* are related by $nk = nc + p + 1$.

The degree and the multiplicity of the knots of the B-spline curve determine the continuity and differentiability. A smooth joint motion requires continuity in acceleration, which will in turn require the joint B-spline curve to be at least of degree-3, i.e., $p = 3$. Let the initial time $a = 0$ and the final time $b = t_f$, and a total of 11 distinct knots are used: 0, $0.1\,t_f$, $0.2\,t_f$, …, and t_f. Therefore, $nk = 17$ and $nc = 13$, i.e., each joint B-spline curve has 13 control points.

4.2 Energy Consumption

Energy has a unifying property, into which the dynamic as well as the kinematic characteristics of manipulator motion are incorporated. We use an approximate form of manipulator energy consumption as a cost function for our optimization problem. The actual formula representing the energy consumption for a manipulator varies depending on the specific design of the system, as well as the types of actuators. Therefore, unless the details of the specific machine information and physical characteristics are provided, it is not possible to obtain the exact formula for the energy consumption. For this reason, simplified forms of the general energy consumption are widely used in literature. Usually, the energy consumption is modeled to be proportional to the actuator torques. We use the squared norm of the actuator torque vector function as an approximate form of the energy consumption from time t_1 to t_2.

$$E = \|\boldsymbol{\tau}(t)\|^2 = \int_{t_1}^{t_2} \sum_{i=1}^{n} (\tau_i(t))^2 dt \qquad (19)$$

where the actuator torque vector $\boldsymbol{\tau}(t) = [\tau_1(t), \ldots, \tau_n(t)]^T$ is obtained from the equations of motion.

The use of energy consumption as a cost function implies several important points. First of all, minimum energy consumption indicates minimum fuel usage. Secondly, for smooth movement of each joint, the magnitude of the second derivatives of the joint curves needs to be minimized to avoid an abrupt change in the joint velocity. Although minimum jerk has been used in literature as a stronger criterion (e.g., [2]), the second derivatives of the joint variables in the energy cost function provide a natural way to ensure the smooth movement of each joint by reducing unnecessary fluctuations in the joint

curves. Finally, minimizing energy consumption implicitly indicates minimizing required actuator torques and joint stress.

For human motion prediction, the metabolic energy consumption [20] will be used as a cost function for the optimization problem.

$$E_{Metabolic} \approx \int_{t_1}^{t_2} \sum_{i=1}^{n} |\tau_i(t)\dot{q}_i(t)| dt + \int_{t_1}^{t_2} \sum_{i=1}^{n} h_m^i |\tau_i(t)| dt + \int_{t_1}^{t_2} \dot{B} dt \quad (20)$$

where h_m^i $(i = 1,...,n)$ are the coefficients of the generalized maintenance heat, and $\dot{B}$ is the basal metabolic rate. It had been shown that h_m^i is inversely proportional to the maximum torque limit of joint *i*. Therefore, for small joint velocities, the human motion of minimum energy (thus minimum weighted torques) implies that humans tend to use the stronger joints to accomplish a given task rather than the weaker ones. This means that the actuator torques are distributed so that the larger torques are exerted at the stronger joints and vice versa, which can be observed in real-world human tasks.

4.3 Constraints

The following is a list of basic constraints that are typically given from the manipulator design and the task requirements. Depending on the task definition and the environment, various other constraints can be imposed in addition.

(1) Joint variable limits: Each joint variable has bilateral constraints imposed in the form of

$$q_i^L \le q_i \le q_i^U \quad (i = 1,...,n) \quad (21)$$

where q_i^L and q_i^U are the lower and upper limits for each joint variable, respectively. These joint limits are usually given from the design of the manipulator.

(2) Actuator torque limits: The torque limit is usually a function of the joint velocity, which is represented as a torque-speed curve of each actuator. The torque-speed curves depend on the class and capacity of the actuators and are usually supplied by the manufacturer.

$$\tau_i^L(\dot{q}_i(t)) \le \tau_i \le \tau_i^U(\dot{q}_i(t)) \quad (i = 1,...,n) \quad (22)$$

where τ_i^L and τ_i^U are the lower and upper limits, respectively, for each actuator torque. Generally, τ_i^L is negative and τ_i^U is positive.

(3) Position and orientation constraints: In general, the configuration of a single rigid body in space is uniquely determined in terms of three independent position coordinates and three independent orientation angles. Thus, the configuration of a link of a manipulator system can be described uniquely by assigning its position and orientation. Depending on the task requirements, some of these six coordinates can be constrained, while the rest are left as free DOFs. For position constraints, the Cartesian coordinates of the point as a function of the joint variables are constrained. For orientation constraints, the direction of the unit vectors of the link local frame is constrained in terms of the global frame.

(4) Path constraints: Every manipulator motion generates an end-effector path along the time in Cartesian space. This end-effector path may be either constrained by task requirements or naturally unconstrained. Usually, the path is given as task requirement. For example, the end-effector paths of drawing a straight line or welding on a surface are pre-determined from the task requirements. Suppose the end-effector path for the task is assigned as a parametric curve in Cartesian space such as

$$\mathbf{path}(t) = \left[x_{path}(t), \quad y_{path}(t), \quad z_{path}(t) \right]^T . \quad (23)$$

To ensure that the end-effector point characterized by $\mathbf{x} = [x, y, z]^T$ as a function of joint variables stays on the path during the motion, the distance from the end-effector point to the desired path in the Cartesian space is enforced as a constraint.

$$\| \mathbf{x}(\mathbf{q}(t)) - \mathbf{path}(t) \| \le \varepsilon \quad (24)$$

where $0 \le t \le t_f$ and ε is a small positive number as a specified tolerance (e.g., 0.001).

4.4 Manipulator Power/Work Measures

A general form of the energy consumption includes various terms specified by the design and properties of the actuators. Thus, in practice, it is difficult to obtain the general form of realistic energy consumption of a manipulator in motion. However, given the actuator torques and the joint velocities, the power profiles along the duration can be obtained. The mechanical power at each joint is defined as the product of the actuator torque and the corresponding joint velocity. The total mechanical power of the *n*-DOF manipulator at a given time is the summation of the mechanical power of each joint. Then, the total mechanical work done during the task can be calculated as a time-integration of the total mechanical power.

$$\dot{W}_{mech} = \tau_1\dot{q}_1 + \tau_2\dot{q}_2 + ... + \tau_n\dot{q}_n = \sum_{i=1}^{n} \tau_i\dot{q}_i ;$$

$$W_{mech} = \int_{t_0}^{t_1} \dot{W}_{mech} dt = \int_{t_0}^{t_1} \left(\sum_{i=1}^{n} \tau_i\dot{q}_i \right) dt \quad (25)$$

Although the above mechanical power is obtained from the rigorous definition in physics, it is not a proper performance measure, since each term in the total mechanical power given in Equation (25) can be positive or negative, resulting in cancellations among several terms with opposite signs. To

avoid this kind of problem, we define the total absolute power and the total absolute work as follows:

$$\dot{W}_{abs} = |\tau_1\dot{q}_1| + |\tau_2\dot{q}_2| + \ldots + |\tau_n\dot{q}_n| = \sum_{i=1}^{n} |\tau_i\dot{q}_i| \; ;$$

$$W_{abs} = \int_{t_0}^{t_1} \dot{W}_{abs}\, dt = \int_{t_0}^{t_1} \left(\sum_{i=1}^{n} |\tau_i\dot{q}_i| \right) dt \qquad (26)$$

By taking the absolute value of each term before adding them, the mechanical power at each joint contributes to the total manipulator power without canceling each other out.

Another form of the power that prevents cancellations among opposite signs can be obtained by taking the norm of the mechanical power vector. We define the norm power and the total norm work as shown below.

$$\dot{W}_{norm} = \sqrt{(\tau_1\dot{q}_1)^2 + \ldots + (\tau_n\dot{q}_n)^2} = \sqrt{\sum_{i=1}^{n} (\tau_i\dot{q}_i)^2} \; ;$$

$$W_{norm} = \int_{t_0}^{t_1} \dot{W}_{norm}\, dt = \int_{t_0}^{t_1} \sqrt{\sum_{i=1}^{n} (\tau_i\dot{q}_i)^2}\, dt \qquad (27)$$

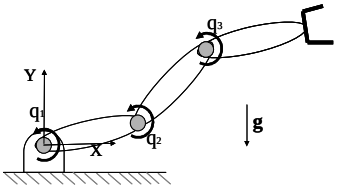
For each manipulator motion, the total mechanical, absolute, and norm power/work will be calculated as outputs to show different performance measures for a given task.

5 EXAMPLE RESULTS AND DISCUSSION

The proposed formulation will be demonstrated using a 3-DOF planar manipulator (motion planning) and the 30-DOF dual-arm human model (motion prediction). Although simple tasks are illustrated as examples in this presentation, our method can be easily applied to more complicated tasks.

5.1 Planning Load-Effective Motion of a 3-DOF Planar Manipulator

A 3R planar manipulator (Figure 8) is given where each link is modeled as a thin rod with 1 m of length and 10 kg of mass. For the equations of motion, we neglect the torques due to the stiffness and dissipative properties at the joints. The joint variable limits (radians) and the actuator torque limits (Nm) are chosen as shown below. Since the purpose of this example is to illustrate the proposed methodology, for simplicity we assume that the torque-velocity curves of the actuators are almost constant near the operation regions. However, implementing the actual torque-velocity curves as constraints is straightforward once the motor characteristics are provided.



$$-\pi \le q_1 \le \pi \; ; \; -8500 \le \tau_1 \le 8500$$
$$-\pi \le q_2 \le \pi \; ; \; -4300 \le \tau_2 \le 4300$$
$$-\pi \le q_3 \le \pi \; ; \; -1500 \le \tau_3 \le 1500$$

Figure 8. A 3-DOF planar manipulator

The task is defined in the X-Y plane as follows. Given (2.6, 0.866, 0) (m) and (1.5, 0.866, 0) (m) as the initial and the final global coordinates of the end-effector, respectively, the manipulator is required to pull (or drag) in -X direction along a straight line with external forces at the end-effector. The time duration is given as 2 seconds. However, depending on the task requirements or user input, different time durations can be used. This task represents simple manipulations such as pulling an object or opening a door.

Motion (A) with small load. First, the motion is generated for the 1-N pulling force using the proposed optimization method. The snapshots of generated motions and the joint profiles are shown in Figures 9 and 10, where each joint moves smoothly toward the final position. For convenience, we call these joint profiles "Motion (A)." At each time instant, the required actuator torques and power profiles are calculated (Figure 11). The results show that the actuator torques strictly satisfy the torque limits. The positive torque values of joint 2 are used to balance the gravity. It is shown that the values of the link weights dominate the motion and the effect of the small pulling force is negligible. Motion (A) is the energy-efficient motion of pulling with 1-N force.

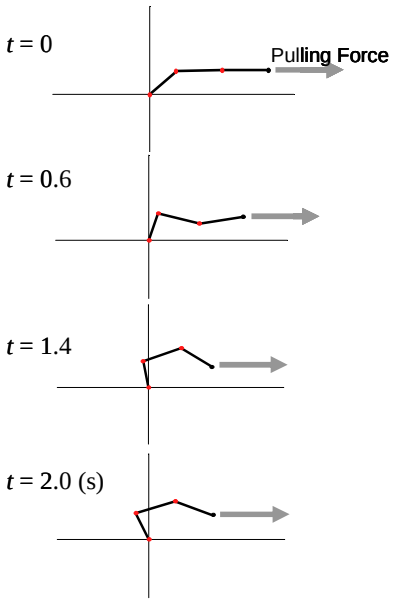


Figure 9. Generated pulling motion – Motion (A).

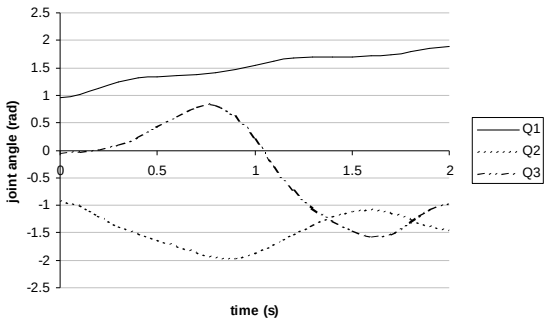


Figure 10. Generated joint profiles for Motion (A).

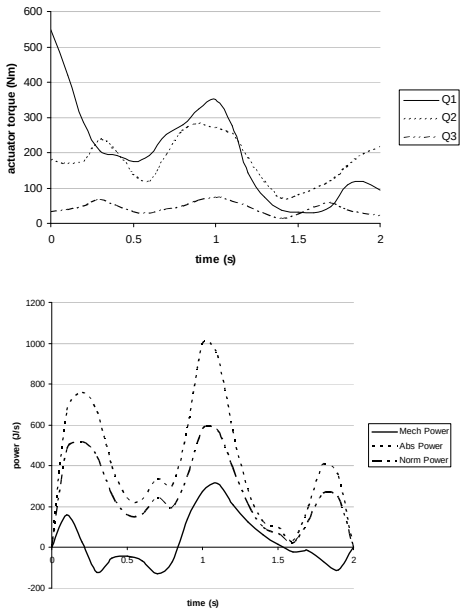


Figure 11. Required actuator torques and power profiles for Motion (A) with small load.

The total mechanical, absolute, and norm work values for our examples are listed in Table I for comparison. In each case, the average work rate can be obtained by dividing these work values by the total time duration (2 seconds in our example). Note that these work measures provide a single criterion that represents the effort input to the system that is required to execute the motion. As a brief validation, the mechanical work can also be calculated analytically from the fundamental definition given in elementary physics. It can be easily shown that our numerical results of mechanical work match the analytical value.

Motion (A) with large load. Next, we take the joint trajectories obtained for planned Motion (A) and would like to execute the same motion to perform a large-force pulling (10000 N) task. In other words, the configurations along the time remain the same, and the only difference is the increased external load. Note that the joint trajectories of Motion (A) were generated without considering the large pulling force. Can the large-force pulling be executed with joint trajectories of Motion (A)? To answer this question, we first solve the inverse dynamics problem with the given joint profiles and the increased external load. As results, the required actuator torques, power profiles, and work measures are calculated from the equations of motion (Figure 12 and Table I).

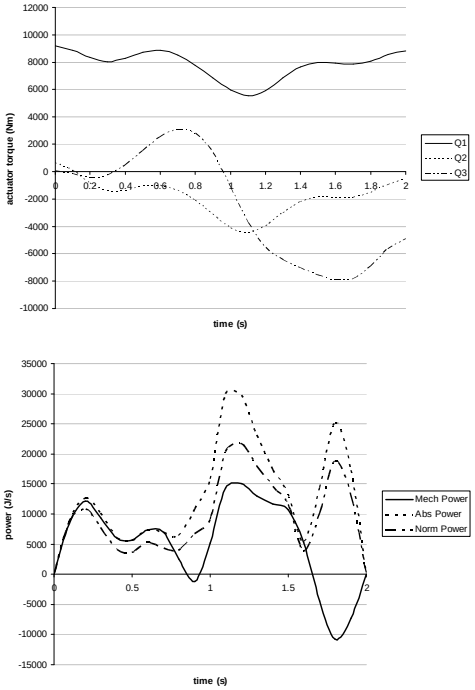


Figure 12. Required actuator torques and power profiles for Motion (A) with large load.

The required actuator torque values of all joints exceeded the torque limits at some time periods. Also the required powers and work measures are much larger than those of Motion (A) with small force. Thus it is seen that the planned Motion (A), in which the large external loads are not incorporated, is not executable for the 10000-N pulling task. In other words, Motion (A) is not effective in pulling with large force. This shows that, in general, the joint trajectories obtained without considering the external loads do not always guarantee the execution of the task under given actuator capacity limits.

Motion (B) with large load. To generate the load-effective motion for large-force pulling, the large external pulling force is incorporated into the algorithm via the equations of motion. Now the optimization problem is solved with 10000-N force (Figures 13-15). Let us call the resulting joint profiles "Motion (B)."

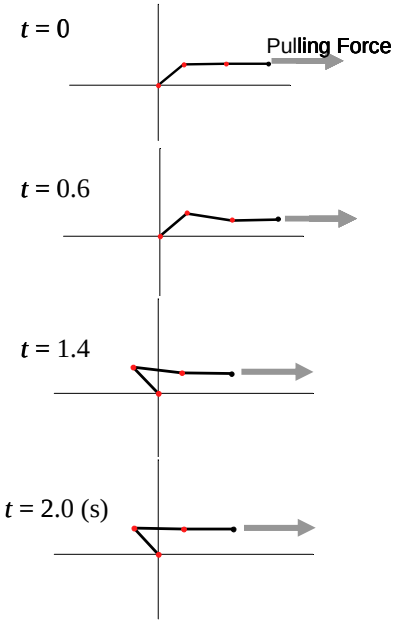


Figure 13. Generated pulling motion – Motion (B).

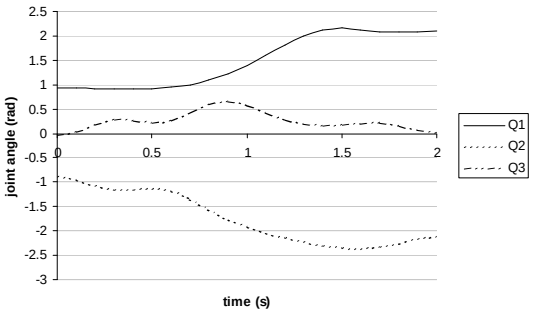


Figure 14. Generated joint profiles for Motion (B).

The required actuator torques for Motion (B) satisfy the torque limits and thus guarantee the successful execution of the task (controllable), i.e., Motion (B) is a load-effective motion for 10000-N force pulling. Note that the power profiles and the work measures are also smaller than those of Motion (A) with 10000-N force.

The successful execution of the large-force pulling task with Motion (B) can be explained by analyzing the manipulator configurations and the force equilibrium of the system free-body diagram at each time step. It is observed that the manipulator tries to maintain the alignment of links 2 and 3 with the line of application of the large pulling force (Figure 13); this is not notable in Motion (A). Since the magnitude and direction of the pulling force is constant, the actuator torque to sustain the pulling force at joint 1 is almost constant (except for the initial and final stages) due to the constant moment-arm length. The actuator torques at joints 2 and 3 can be roughly calculated as the product of the constant pulling force and the offset distance perpendicular from the line of the pulling force (of course, the inertia forces and gravity should be added to obtain exact torque values). Thus, in the case of the large-force pulling task where the effects of the link weights are relatively small, the actuator torques are mostly used to sustain the large pulling force. By positioning joints 2 and 3 as close to the line of force as possible, the actuator torque values for those joints are minimized. Note that a similar trend is observed in human arm motion when pulling with large force.

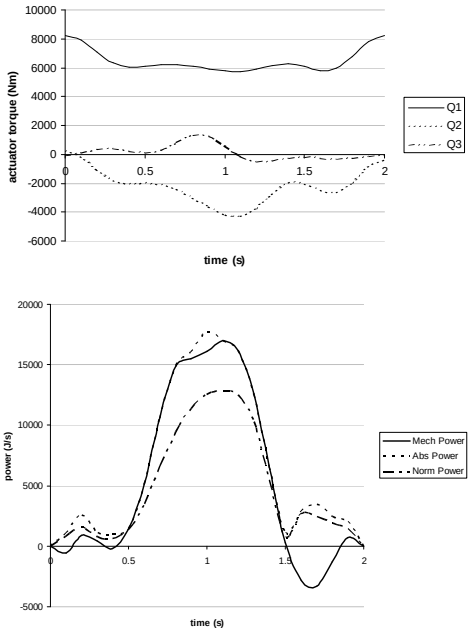


Figure 15. Required actuator torques and power profiles for Motion (B) with large load.

Table I. Various work measures for the 3-DOF examples

Work Measures (J)	Motion (A) with 1 N	Motion (A) with 10000 N	Motion (B) with 10000 N
Mechanical Work	62.77	11039.00	10990.55
Absolute Work	821.07	26645.20	13586.01
Norm Work	539.79	19649.42	9883.05

It can easily be shown from inverse dynamics that if Motion (B) is applied for 1-N pulling, the motion is still executable, but it is less energy-efficient. Also, it should be noted that our optimization-based method also determines the existence of the feasible motion. If there is no feasible region for the optimal motion problem, then this indicates that the task cannot be accomplished by any choice of manipulator configurations under given actuator capacity limits.

5.2 Prediction of Dual-Arm Lever-Pulling Motion

We consider the dual-arm motion of pulling a lever with constant load from a given initial position to a given final

position. The initial position of the lever is (0, 19, 70) (cm), and the final position of the lever is (0, 19, 40) (cm) in the global coordinate frame where the path of the lever between the initial and final positions is given as a straight line. The time duration for the task is given as 2 seconds. Two different load magnitudes are tested: 1 N and 800 N. To compare the motions of two extreme cases, the actuator torque limits are not enforced. Figures 16–19 show the predicted motions and calculated results of several notable joints for both cases. The total metabolic energy consumed for the 1-N lever-pulling is 301.21 Joules, and that for the 800-N lever-pulling is 1440.55 Joules. These values also represent how much effort is required to perform each task.

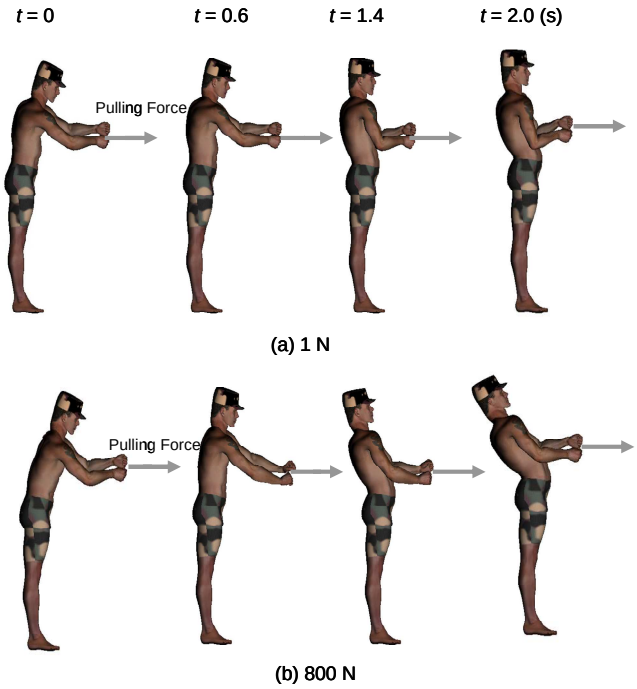
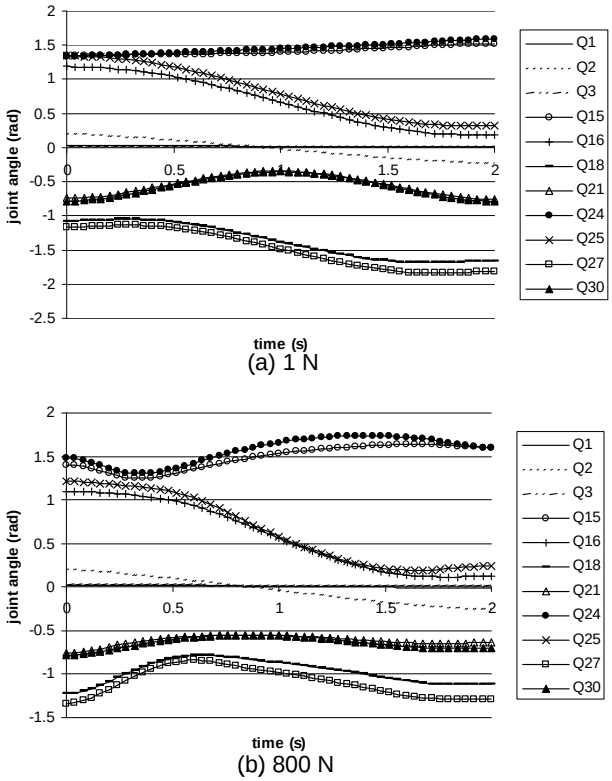


Figure 16. Predicted motion of dual-arm lever-pulling.

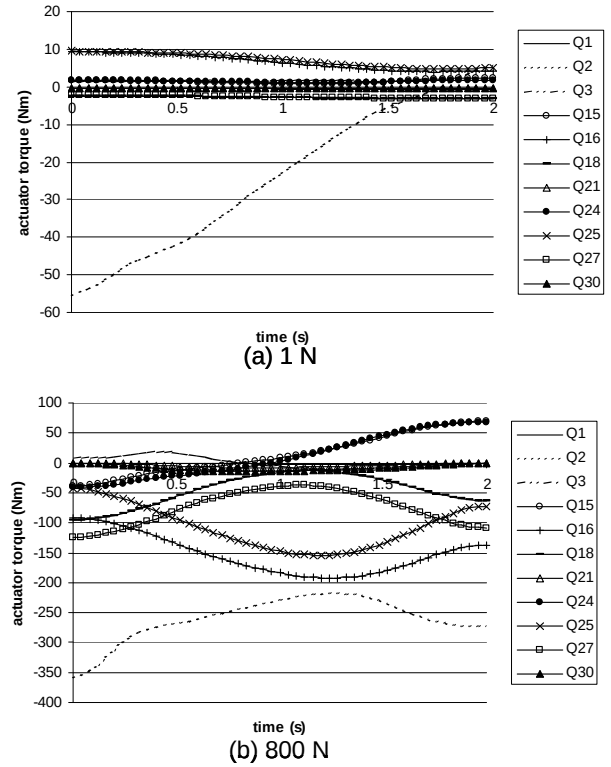
It is observed that the digital human generates different motions for different amounts of external load. For the lever load of 1 N, the digital human's torso is almost vertical, while for the larger load of 800 N, the torso is extended backward to use its own body weight to counter-balance the large pulling load at both hands. In Figure 18 (b), the large negative actuator torque values for torso extension (joint 2) and shoulders (joints 16 and 25) for the 800-N lever-pulling indicate the major contributions of these joints to the pulling motion. These large actuator torques are used to generate the torso motion while sustaining the large pulling force in the forward direction. In this way, the digital human can also straighten both its arms in the final posture to minimize the actuator torques at the wrists and elbows. This trend is similar to the previous load-effective motion results for the 3-DOF manipulator.



(a) 1 N

(b) 800 N

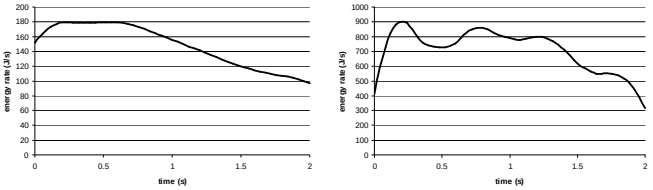
Figure 17. Predicted joint profiles for dual-arm lever-pulling.



(a) 1 N

(b) 800 N

Figure 18. Predicted actuator torque profiles for dual-arm lever-pulling.



(a) 1 N (b) 800 N

Figure 19. Predicted metabolic rate profiles for dual-arm lever-pulling.

It is often observed in the real world that, when pulling or dragging a heavy object, a human usually leans his or her body in the desired direction of pulling. In other words, to accomplish a given task, humans naturally generate the effective motion that minimizes the required actuator torques within the actuator capacities (torque limits). A similar argument can be made for the motion of pushing a heavy object, where it is frequently observed that a human leans his or her body in the direction of pushing.

6 CONCLUSIONS

In past motion planning problems, general external loads other than gravity were not considered extensively. In this article, a method for planning load-effective dynamic motions for redundant manipulators is proposed. A comprehensive form of Lagrange's equations of motion for open-loop chains (serial and tree-structured) is derived in which the general external loads, as well as the inertia and gravity terms, are incorporated. The constraint forces and moments from external objects can also be included in this way. The comprehensive equations of motion are then implemented into the constraints and/or the cost function of the optimal motion planning problem. The redundant manipulator motions generated by the optimization formulation guarantee the successful execution of generic tasks with external loads under given actuator torque limits. Our examples also showed that the motion generated without taking the general external loads into account may not be executable, especially when a large load is desired. The proposed method is also applicable for physics-based human motion prediction. A dual-arm task that shows different effective human motions for different amounts of external load was predicted. Overall, the proposed optimization-based method is broadly applicable for any type of robotic system, such as multiple-arm manipulators, mobile manipulators, and whole-body humanoid robots.

7 ACKNOWLEDGMENTS

This research is funded by the U.S. Army Tank Automotive Research, Development, and Engineering Center (TARDEC) and the U.S. Army Natick Soldier Systems Center.

8 REFERENCES

[1] Sciavicco, L. and Siciliano, B., "A solution algorithm to the inverse kinematic problem for redundant manipulators," *IEEE Journal of Robotics and Automation*, Vol. 4, n 4, pp. 403-410, Aug. 1988.

[2] Piazzi, A. and Visioli, A., "Global minimum-jerk trajectory planning of robot manipulators," *IEEE Transactions on Industrial Electronics*, Vol. 47, n 1, pp. 140-149, Feb. 2000.

[3] Suh, K. and Hollerbach, J., "Local versus global torque optimization of redundant manipulators," *Proceedings of IEEE International Conference on Robotics and Automation*, Vol. 4, pp. 619-624, Mar. 1987.

[4] Gompertz, R.S. and Yang, D.C.H., "Feasibility evaluation of dynamically linearized kinematically redundant planar manipulators," *Proceedings of IEEE International Conference on Robotics and Automation*, Vol. 1, pp. 178-182, Apr. 1988.

[5] Nakamura, Y., *Advanced Robotics: Redundancy and Optimization*, Addison-Wesley, 1991.

[6] Nedungadi, A. and Kazerouinian, K., "A local solution with global characteristics for the joint torque optimization of a redundant manipulator," *Journal of Robotic Systems*, Vol. 6, n. 5, pp. 631-654, 1989.

[7] Chung, Y.S., Griffis, M., and Duffy, J., "Unique joint displacement generation for redundant robotic systems," *ASME Robotics, Spatial Mechanisms, and Mechanical Systems*, Vol. 45, pp. 637-641, 1992.

[8] Chung, C.Y., Lee, B.H., Kim, M.S., and Lee, C.W., "Torque optimizing control with singularity-robustness for kinematically redundant robots," *Journal of Intelligent and Robotic Systems*, Vol. 28, n 3, pp. 231–258, July 2000.

[9] Singh, S.K. and Leu, M.C., "Manipulator motion planning in the presence of obstacles and dynamic constraints," *The International Journal of Robotics Research*, Vol. 10, n 2, pp. 171-187, 1991.

[10] Wang, C.-Y.E., Timoszyk, W.K., and Bobrow, J.E., "Payload maximization for open chained manipulators: finding weightlifting motions for a Puma 762 robot," *IEEE Transactions on Robotics and Automation*, Vol. 17, n 2, pp. 218-224, Apr. 2001.

[11] Saramago, S.F.P. and Ceccarelli, M., "An optimum robot path planning with payload constraints," *Robotica*, Vol. 20, n 4, pp. 395-404, 2002.

[12] Vukobratovic, M. and Kircanski, M., "A dynamic approach to nominal trajectory synthesis for redundant manipulators," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-14, pp. 580-586, July-Aug. 1984.

[13] Hirakawa, A.R. and Kawamura, A., "Trajectory generation for redundant manipulators under optimization of consumed electrical energy," *IEEE Industry Applications Conference, Thirty-First IAS Annual Meeting*, Vol. 3, pp. 1626-1632, Oct. 1996.

[14] Nokleby, S.B. and Podhorodeski, R.P., "Pose optimization of serial manipulators using knowledge of their velocity-degenerate (singular) configurations," *Journal of Robotic Systems*, Vol. 20, n 5, pp. 239–249, Apr 2003.
[15] Papadopoulos, E. and Gonthier, Y., "On manipulator posture planning for large force tasks," *Proceedings of IEEE International Conference on Robotics and Automation*, Vol. 1, pp. 126-131, May 1995.
[16] Papadopoulos, E. and Gonthier, Y., "A framework for large-force task planning of mobile and redundant manipulators," *Journal of Robotic Systems*, Vol. 16, n 3, pp. 151–162, Feb 1999.
[17] Jung, E.S., Kee, D., and Chung, M.K., "Upper body reach posture prediction for ergonomic evaluation models," *International Journal of Industrial Ergonomics*, Vol. 16, pp. 95-107, 1995.
[18] Hase, K. and Yamazaki, N., "Development of three-dimensional whole-body musculoskeletal model for various motion analyses," *JSME International Journal*, Series C, Vol. 40, n 1, pp. 25-32, Mar. 1997.
[19] Anderson, F.C. and Pandy, M.G., "Dynamic optimization of human walking," *Journal of Biomechanical Engineering*, Vol. 123, n 5, pp. 381-390, 2001.
[20] Kim, J.H., Abdel-Malek, K., Yang, J., and Marler, T., "Prediction and analysis of human motion dynamics performing various tasks," *The International Journal of Human Factors Modeling and Simulation*, Vol. 1, n 1, pp. 69-94, 2006.
[21] Kawasaki, H., Beniya, Y., and Kanzaki, K., "Minimum dynamics parameters of tree structure robot models," *Proceedings of IEEE International Conference on Industrial Electronics, Control and Instrumentation*, Kobe, Japan, Vol. 2, pp. 1100-1105, 1991.
[22] Nakamura, Y. and Yamane, K., "Dynamics computation of structure-varying kinematic chains and its application to human figures," *IEEE Transactions on Robotics and Automation*, Vol. 16, n 2, pp. 124-134, Apr. 2000.
[23] Bobrow, J.E., Martin, B., Sohl, G., Wang, E. C., Park, F.C., and Kim, J., "Optimal robot motions for physical criteria," *Journal of Robotic Systems*, Vol. 18, n 12, pp. 785–795, Dec. 2001.
[24] Featherstone, R. and Orin, D., "Robot dynamics: equations and algorithms," *Proceedings of IEEE International Conference on Robotics and Automation*, Vol. 1, pp. 826-834, 2000.
[25] Denavit, J. and Hartenberg, R.S., "A kinematic notation for lower-pair mechanisms based on matrices," *Journal of Applied Mechanics*, Vol. 77, pp. 215-221, 1955.
[26] Fu, K.S., Gonzalez, R.C., and Lee, C.S.G., *Robotics: Control, Sensing, Vision, and Intelligence*, McGraw-Hill, 1987.
[27] Craig, J.J., *Introduction to Robotics: Mechanics and Control*, 2nd ed., Addison-Wesley, 1989.
[28] Langhaar, H.L., *Energy Methods in Applied Mechanics*, Krieger Publishing Co., 1989.
[29] Anand, V.B., *Computer Graphics and Geometric Modeling for Engineers*, John Wiley and Sons, New York, 1993.