

Research Article

The Robotic Lumbar Spine: Dynamics and Feedback Linearization Control

Ernur Karadogan¹ and Robert L. Williams II²

¹ Mechanical Engineering Department, University of Texas-Pan American, Edinburg, TX 78539, USA

² Mechanical Engineering Department, Ohio University, Athens, OH 45701, USA

Correspondence should be addressed to Robert L. Williams II; williar4@ohio.edu

Received 19 June 2013; Accepted 2 August 2013

Academic Editor: Zhonghua Sun

Copyright © 2013 E. Karadogan and R. L. Williams II. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The robotic lumbar spine (RLS) is a 15 degree-of-freedom, fully cable-actuated robotic lumbar spine which can mimic *in vivo* human lumbar spine movements to provide better hands-on training for medical students. The design incorporates five active lumbar vertebrae and the sacrum, with dimensions of an average adult human spine. It is actuated by 20 cables connected to electric motors. Every vertebra is connected to the neighboring vertebrae by spherical joints. Medical schools can benefit from a tool, system, or method that will help instructors train students and assess their tactile proficiency throughout their education. The robotic lumbar spine has the potential to satisfy these needs in palpatory diagnosis. Medical students will be given the opportunity to examine their own patient that can be programmed with many dysfunctions related to the lumbar spine before they start their professional lives as doctors. The robotic lumbar spine can be used to teach and test medical students in their capacity to be able to recognize normal and abnormal movement patterns of the human lumbar spine under flexion-extension, lateral bending, and axial torsion. This paper presents the dynamics and nonlinear control of the RLS. A new approach to solve for positive and nonzero cable tensions that are also continuous in time is introduced.

1. Introduction

Teaching art of palpation to medical students is a challenging task. In institutions that teach palpatory diagnosis, it is taught by using voluntary human patients who are mostly palpated by the instructor for demonstrative purposes. Meanwhile, the students usually watch the process and get to palpate only their lab partners as “patients” who are, considering the general population of medical students, relatively young and healthy (many with limited dysfunctions). It is, however, very difficult to be able to find and demonstrate a different patient for every single dysfunction that the students are taught during the lectures or in the laboratories. Therefore, it is still hard to teach and learn palpatory diagnosis for different variations of dysfunctions. The lack of a means for evaluating the transfer of practical information from the instructor to the students is another drawback that the medical schools are facing today. There exists no assessment device for instructors to objectively evaluate progress and success of the students in real-life situations.

The need for a “gold standard” to objectively assess the palpation accuracy is apparent. The design of such a device has the potential of becoming a standardized means for training medical students since the repeatability of many dysfunctions would be possible. Repeatability is a main concern in real-life medical education situations, because the properties of human soft tissue (stiffness, tenderness, etc.) can alter when it is touched by the examiner. The tissue properties are not the same even between the beginning and end of an examination. A legitimate method of evaluating the students would be comparing the first diagnosis of the instructor with the diagnosis of the student. However, when the student takes over the patient, he/she tries to diagnose movement patterns and/or the tissue properties that have already been changed due to the stimulation of the instructor.

The role of simulation in medical education is rapidly increasing. The simulations to train nurses, veterinarians, and doctors (osteopathic and allopathic) have been and are still being developed due to their effectiveness and cost-reducing advantages. These simulations can be computer based or in

the form of mannequins that can simulate some functions of the real human body such as breathing and blood pressure. Computer-based haptic simulations require the utilization of a haptic interface to interact with the virtual objects inside a computer screen. That is clearly not the case when humans really interact with real objects. For instance, the VHB [1], the only simulation that is being used to improve palpatory skills of medical students, simulates somatic dysfunctions by increased stiffness of an area on the virtual back and the users “touch” the back with PHANToM haptic devices which only stimulate the proprioceptive receptors and introduces an extra layer of disturbance between the fingers and the computer-generated objects to be sensed. Therefore, a simulation system which allows the user to interact with a real object would be a better and more effective approach.

The robotic spine concept has been studied over the past years [2–4]. These studies built humanoid robots with a flexible spine which would enhance the human-like movements of the robots and increase the range of movement of the robot’s torso. These humanoid robots dealt with the movement of the whole spine, rather than the relative position and stiffness of a vertebra with respect to the adjacent ones. They sufficiently accomplished flexible spine movements with less than the total number of vertebrae in a human spine. However, no research has yet been completed on the subject of developing a robotic spine with anatomically accurate vertebrae geometry and movements for tactile medical education and/or proficiency assessment. In this paper, the dynamic model of a robotic lumbar spine is derived and used in designing a nonlinear controller. A new method to solve for positive and nonzero cable tensions that are also continuous in time is introduced. Simulations to test the controller for the RLS are presented. In the RLS, individual vertebra is controlled by four cables that are attached to four motors. In this case, a cable-actuated robot is practical due to the space limitations between vertebrae. The robot will be controlled by a joystick or autonomously by preprogramming. The user will interact by touching the posterior aspect of the lumbar spine that is covered with a skin-like material. The user will try to find the type and region of the dysfunction by comparing the movement patterns at different configurations of the robotic lumbar spine. In this current form, the RLS will be capable of training users in terms of healthy and dysfunctional movements of the lumbar spine. Addition of the capability to adjust (rotational) stiffness of the vertebrae associated with normal and abnormal rotational limits is also underway. This will enable users to train on anatomically normal (abnormal) movement patterns as well as feel normal (abnormal) joint stiffness associated with a healthy (dysfunctional) lumbar spine.

2. Construction of the Lumbar Spine Geometry

The geometry of the lumbar spine was constructed using dimensions of an average human spine based on published experimental data [5]. All parameters used for the reconstruction of the geometry except for the facet plane and facet plane angle (ϕ) have been previously used in the literature and measured to define the morphology of the vertebrae.

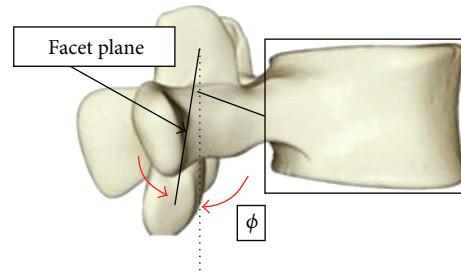


FIGURE 1: Facet plane and angle.

We, assuming sagittal symmetry, define a facet plane as the plane that connects the centers of the four facets (left/right, superior/inferior) of a vertebra. This plane (manufactured as a plate) will allow the attachment of posterior elements with various dimensions on the same vertebral body making the system modular. The facet plane angle is defined to be the angle between the facet plane and the posterior wall of the vertebral body. In modeling, a cylindrical shape is assumed for the vertebral bodies. Figure 1 shows the facet plane angle and the approximation of the vertebral bodies as cylinders. The constructed lumbar spine geometry is shown Figure 2. A detailed explanation on the construction of the lumbar spine geometry can be found in [6].

3. 3D Static Model of the Human Lumbar Spine

In order to design a device that mimics an average adult’s lumbar spine, it is necessary to have anatomically correct movement patterns of each lumbar vertebra. In this study, these movement patterns were acquired by using a three-dimensional static model of the human spine. The mathematical model includes five lumbar vertebrae and the sacrum, elastic elements that connect inferior facets of one vertebra to the superior facets of the lower one and torsion springs that represent the collective torque resisting effects of the intervertebral disc and ligaments. It has been shown with several studies [5, 7–9] that the significant motion of the vertebrae during the movement of the spine is the rotational motion. Therefore, a spherical joint is chosen to connect each vertebra to the lower one. The location of this joint is critical in order to provide anatomically correct motion for each vertebra during the movement of the entire lumbar spine. In this model, the spherical joints are located at the inferoposterior corners of the vertebral bodies since the experimental data used for validation is based on the findings from [5]. The complete details including derivation and validation of this model can be found in [10].

4. RLS Kinematics

The robotic spine, shown in Figure 3, was designed based on the study by [5] since it details how lumbar spine moves in prespecified loading conditions. In that study, the uppermost vertebrae of freshly frozen cadaveric human lumbar spines with no abnormalities were exposed to external pure

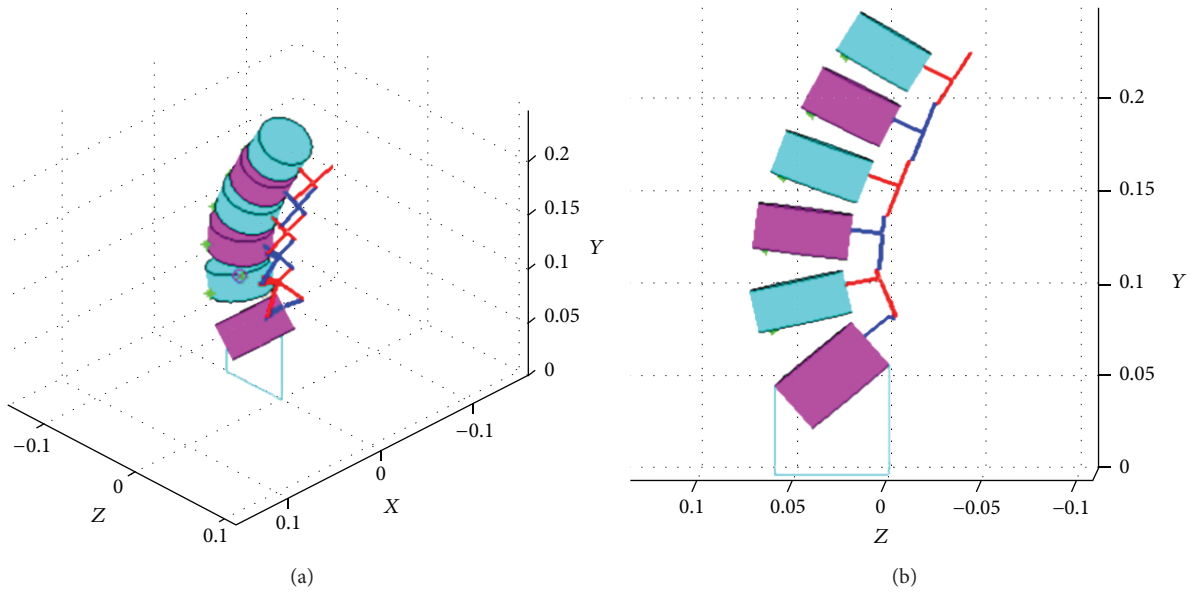


FIGURE 2: Three-dimensional geometry of the lumbar spine [6].

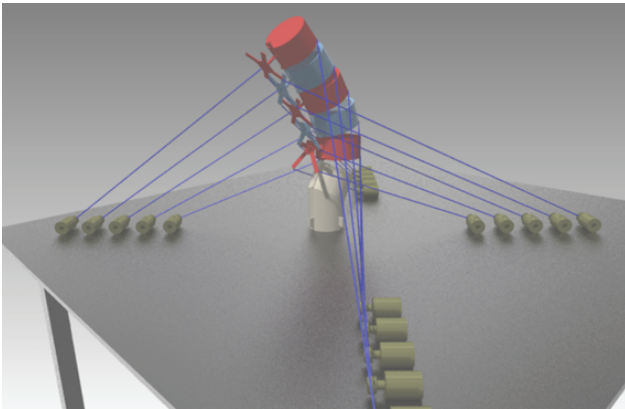


FIGURE 3: The robotic lumbar spine (RLS).

moments in order to induce motion, and both rotational and translational movements of each vertebra were recorded.

The RLS is actuated by 20 cables connected to electric motors. Every vertebra is connected to the neighboring vertebrae by spherical joints. The use of spherical joints is intentional since the rotational motion of the vertebrae is more prominent as compared to their translational motion. The location of the spherical joint for each vertebra is at the inferoposterior corner in the mid-sagittal plane of the vertebral body. These locations correspond to the origin of the coordinate frames with respect to which the angles of rotation were recorded in [5]. As discussed previously, the facet plane in Figure 1 was designed to be used as the base on which posterior elements with various dimensions can be attached.

The cable connection points on the ground are at the corners of five trapezoids. The innermost trapezoid that includes the connection points for the fifth lumbar vertebra

(L5) has posterior base length of 0.4 m, anterior base length of 0.2 m, and height of 0.15 m. The remaining four trapezoids are constructed with increasing the height of the adjacent (inner) one by 0.05 m anteriorly and 0.05 m posteriorly. This placement of the cable connections on the ground prevented cable interference during the simulations for six motion types: flexion/extension, right/left bending, and right/left axial torque [6].

5. RLS Dynamics and Control

5.1. Dynamic Model. In this section, we will develop the dynamic equations for the robotic lumbar spine. The dynamic equations can be stated in the following general format:

$$\mathbf{M}(q)\ddot{q} + \mathbf{V}(q, \dot{q}) + \mathbf{G}(q) = \boldsymbol{\tau}, \quad (1)$$

where q is the vector of generalized coordinates (joint variables), \dot{q} is the vector of generalized velocities, \mathbf{M} is the inertia matrix as a function of q , \mathbf{V} is the Coriolis/centripetal term as a function of q and \dot{q} , \mathbf{G} is the gravity term as a function of q and $\boldsymbol{\tau}$ is the vector of generalized forces that are found by using the forces applied by the cables. Note that, for systems the links of which are actuated at the joints, $\boldsymbol{\tau}$ is independent of q when they are defined to be the joint variables (angle of rotation for revolute and distance for prismatic joints). However, as shown later in the text, $\boldsymbol{\tau}$ for the robotic lumbar spine are functions of the generalized coordinates as well. The friction is neglected during the dynamic modeling stage.

We will use the energy-based Lagrange equation to derive the equations of motion for the robotic spine. The Lagrange formulation does not require the knowledge of the constraint forces when all of the constraints in a system are *holonomic* [11]. In Newton-Euler formulation, however, the constraint forces between adjacent links must be included as variables. The robotic lumbar spine is composed of only rigid bodies

that are connected to each other via spherical joints. That is, all constraints in the system are *holonomic* constraints. The Lagrange equation can be stated as

$$\frac{d}{dt} \left(\frac{dL}{d\dot{q}} \right) - \frac{dL}{dq} = \tau, \quad (2)$$

where L is the Lagrangian and defined as the difference between the kinetic and potential energy

$$L(q, \dot{q}) = K(q, \dot{q}) - U(q). \quad (3)$$

Since the potential energy is not a function of the generalized velocities, the Lagrange equation for the i th generalized coordinate can be written as

$$\frac{d}{dt} \left(\frac{\partial K}{\partial \dot{q}_i} \right) - \frac{\partial K}{\partial q_i} + \frac{\partial U}{\partial q_i} = \tau_i. \quad (4)$$

By using the chain rule, the above equation can also be written as

$$\sum_{j=1}^n \left(\frac{d(\partial K / \partial \dot{q}_i)}{dq_j} \dot{q}_j + \frac{d(\partial K / \partial \dot{q}_i)}{d\dot{q}_j} \ddot{q}_j \right) - \frac{\partial K}{\partial q_i} + \frac{\partial U}{\partial q_i} = \tau_i, \quad (5)$$

where n is the total number of generalized coordinates.

Adapting to the general format of the dynamic equations (1), $\mathbf{M}(q)$, $\mathbf{V}(q, \dot{q})$ and $\mathbf{G}(q)$ can be deduced from (5) as:

$$\begin{aligned} \mathbf{M}(q) &= \begin{bmatrix} m(q)_{11} & \cdots & m(q)_{1j} \\ \vdots & \ddots & \vdots \\ m(q)_{i1} & \cdots & m(q)_{ij} \end{bmatrix}, \\ \mathbf{V}(q, \dot{q}) &= \begin{bmatrix} v(q, \dot{q})_1 \\ \vdots \\ v(q, \dot{q})_i \end{bmatrix}, \\ \mathbf{G}(q) &= \begin{bmatrix} g(q)_1 \\ \vdots \\ g(q)_i \end{bmatrix}, \end{aligned} \quad (6)$$

where

$$\begin{aligned} m(q)_{ij} &= \frac{d(\partial K / \partial \dot{q}_i)}{d\dot{q}_j}, \\ v(q, \dot{q})_i &= \sum_{j=1}^n \left(\frac{d(\partial K / \partial \dot{q}_i)}{dq_j} \dot{q}_j \right) - \frac{\partial K}{\partial q_i}, \\ g_j(q) &= \frac{\partial U}{\partial q_j} = \frac{\partial(-m_i \mathbf{g}^T \mathbf{B}^T \mathbf{P}_{CGi})}{\partial q_j} \\ &= \sum_{i=1}^{N_v} -m_i \mathbf{g}^T \frac{\partial({}^B T)}{\partial q_j} {}^i \mathbf{P}_{CGi}, \end{aligned} \quad (7)$$

where m_i is the mass, ${}^B \mathbf{P}_{CGi}$ is the augmented vector involving center of gravity coordinates of the i th vertebra, $\mathbf{g} = [g_x \ g_y \ g_z \ 0]^T = [0 \ -9.806 \ 0 \ 0]^T$ is the augmented (a zero is added as the last element) gravitational acceleration since y -axis is directed upward and $[{}^B T] =$

$[{}^B T][{}^1 T] \cdots [{}^{i-1} T]$ is the 4×4 homogenous transformation matrix that represents i th vertebra coordinate system with respect to the base frame $\{B\}$. The transformation matrix of a frame with respect to the neighboring one is expressed as:

$$[{}^{i+1} T]_{4 \times 4} = \begin{bmatrix} [{}^{i+1} R]_{3 \times 3} & \{ {}^i \mathbf{P}_{(i+1)ORG} \}_{3 \times 1} \\ 0 \ 0 \ 0 & 1 \end{bmatrix}. \quad (8)$$

Total kinetic energy of the RLS is:

$$K = \sum_1^{N_v} {}^i K_i, \quad (9)$$

where N_v is the total number of vertebrae and ${}^i K_i$ is the kinetic energy of the i th vertebra expressed in the local vertebral frame and defined as

$${}^i K_i = \frac{1}{2} m {}^i \mathbf{V}_{G_i} \cdot {}^i \mathbf{V}_{G_i} + \frac{1}{2} {}^i \boldsymbol{\omega}_i \cdot {}^i \mathbf{H}_{G_i}, \quad (10)$$

where ${}^i \mathbf{V}_{G_i}$ is the linear velocity of the center of gravity, ${}^i \boldsymbol{\omega}_i$ is the angular velocity, ${}^i \mathbf{H}_{G_i} = {}^i \mathbf{I}_{G_i} {}^i \boldsymbol{\omega}_i$ is the angular momentum of the i th vertebra with respect to its local frame, and ${}^i \mathbf{I}_{G_i}$ is the inertia tensor.

The right hand side of (2) is composed of the generalized cable forces and the partial derivative of the potential energy with respect to joint variables (q) which is actually the gravity term, $\mathbf{G}(q)$, in the Lagrange equation. Note that $\boldsymbol{\tau}$ is the vector of generalized cable forces. The resulting generalized forces must be calculated for proper use of the Lagrange equation. The k th generalized force for the robotic lumbar spine can be written as

$$\begin{aligned} \boldsymbol{\tau}_k &= \sum_{i=1}^{N_v} \sum_{j=1}^{N_c} \mathbf{t}_{ij}^T \frac{\partial({}^B T)}{\partial q_k} {}^i \mathbf{P}_{ij} \\ &= \sum_{i=1}^{N_v} \sum_{j=1}^{N_c} \mathbf{t}_{ij}^T \frac{\partial({}^B T)}{\partial q_k} {}^i \mathbf{P}_{ij} \\ &= \sum_{i=1}^{N_v} \sum_{j=1}^{N_c} \mathbf{t}_{ij} \hat{\mathbf{L}}_{ij}^T \frac{\partial({}^B T)}{\partial q_k} {}^i \mathbf{P}_{ij} \\ &= \mathbf{A}(q)_{15 \times 20} \{\mathbf{t}\}_{20 \times 1}, \end{aligned} \quad (11)$$

where ${}^i \mathbf{P}_{ij}$ is the augmented position vector from the origin of the local coordinate frame of the i th vertebra to the connection point of the j th cable in $\{i\}$, $\hat{\mathbf{L}}_{ij}$ is the unit vector in the corresponding cable direction, and $[{}^B T]$ is the previously defined 4×4 homogenous transformation matrix.

5.2. Cable Tension Optimization. One of the challenges of designing a cable-actuated robot is the fact that the cables must be in tension (positive) at all times during the operation of a task. The robots with rigid links that are actuated with motors are not subject to this limitation. The RLS, being a fully cable-actuated robot, needs to be supplied with positive

cable tensions. We start with the previously derived dynamics equations (independent variables are not shown for clarity):

$$\mathbf{M}_{(m \times m)} \ddot{\mathbf{q}}_{(m \times 1)} + \mathbf{V}_{(m \times 1)} + \mathbf{G}_{(m \times 1)} = \mathbf{A}_{(m \times n)} \mathbf{t}_{(n \times 1)}, \quad (12)$$

where m is the number of degrees of freedom (=15) and n is the number of cables (=20). In order to solve for positive cable tensions we introduce an intermediate variable $\boldsymbol{\tau}_v$, which is the virtual input [12] to the system and defined as

$$\boldsymbol{\tau}_{v(n \times 1)} = \mathbf{M} \ddot{\mathbf{q}} + \mathbf{V} + \mathbf{G}. \quad (13)$$

Therefore, the dynamics equation can be written as:

$$\boldsymbol{\tau}_v = \mathbf{A} \mathbf{t}. \quad (14)$$

Equation (14) can be solved by:

$$\mathbf{t} = \mathbf{A}^+ \boldsymbol{\tau}_v + (\mathbf{I}_m - \mathbf{A}^+ \mathbf{A}) \mathbf{z}, \quad (15)$$

where $\mathbf{A}^+ = \mathbf{A}^T (\mathbf{A} \mathbf{A}^T)^{-1}$ is the Moore-Penrose pseudo-inverse of \mathbf{A} , \mathbf{I}_m is the $(m \times m)$ identity matrix, the first term on the right hand side is the particular solution and the second term is the homogenous solution which maps \mathbf{z} ($m \times 1$ vector) to the null space of \mathbf{A} . The homogenous solution can take any value making the solution nonunique. This property can be utilized to search for a solution that will generate positive cable tensions that are needed to control the RLS. On the other hand, when the homogenous solution is zero, the tensions are calculated in the least-squares sense which does not guarantee a solution that will satisfy the positive cable tensions criterion. It is also imperative to obtain positive and nonzero cable tensions in order to be able to keep the robot under control during a task. Equation (15) can also be written as [13]:

$$\mathbf{t} = \mathbf{A}^+ \boldsymbol{\tau}_v + N(\mathbf{A}) \boldsymbol{\sigma}, \quad (16)$$

where $N(\mathbf{A})$ is the $(n \times n - m)$ kernel matrix of \mathbf{A} and $\boldsymbol{\sigma}_{(n-m \times 1)} = \{\sigma_1, \sigma_2, \dots, \sigma_{n-m}\}^T$ is an arbitrary vector.

An optimization procedure can be employed with a proper objective function to solve (16) with positive and nonzero cable tensions. One of the optimization procedures is by using linear programming, which is formulated as [14]

$$\begin{aligned} \min_{\boldsymbol{\sigma}} \quad & \mathbf{f}^T \boldsymbol{\sigma}, \\ \text{such that} \quad & \mathbf{t} = \mathbf{A}^+ \boldsymbol{\tau}_v + N(\mathbf{A}) \boldsymbol{\sigma}, \\ & -N(\mathbf{A}) \boldsymbol{\sigma} \leq \mathbf{A}^+ \boldsymbol{\tau}_v - \mathbf{b}, \\ & \boldsymbol{\sigma}_l \leq \boldsymbol{\sigma} \leq \boldsymbol{\sigma}_u, \end{aligned} \quad (17)$$

where \mathbf{f} is the $(n - m \times 1)$ linear objective function vector, \mathbf{b} is the $(n \times 1)$ vector that holds minimum allowed positive tensions (lower boundary for \mathbf{t}), and $\boldsymbol{\sigma}_l$ and $\boldsymbol{\sigma}_u$ are, respectively the lower and upper boundaries for the arbitrary $\boldsymbol{\sigma}$ vector in (16). This optimization procedure, when converges to a minimum solution, produces point-wise feasible positive cable tensions that are equal or higher than the limits specified in \mathbf{b} . However, these feasible cable tensions are not

guaranteed to be continuous in time during the task. This discontinuity is not desirable since it may cause instability during real-time control of the robot. Therefore, we introduce a new optimization scheme that will result in cable tensions that are both point-wise feasible and continuous in time. The proposed optimization scheme makes use of the previous solution to be able to choose the current solution to be as close to it as possible. This scheme which minimizes the norm of the difference between the previous and current solution can be formulated as follows:

$$\begin{aligned} \min_{\boldsymbol{\sigma}_i} \quad & \|\mathbf{t}_i - \mathbf{t}_{i-1}\|, \\ \text{such that} \quad & \mathbf{t}_i = \mathbf{A}_i^+ \boldsymbol{\tau}_{v_i} + N(\mathbf{A}_i) \boldsymbol{\sigma}_i, \\ & \mathbf{t}_{i-1} = \mathbf{A}_{i-1}^+ \boldsymbol{\tau}_{v_{i-1}} + N(\mathbf{A}_{i-1}) \boldsymbol{\sigma}_{i-1}, \\ & \mathbf{t}_0 = \mathbf{b} \\ & -N(\mathbf{A}_i) \boldsymbol{\sigma} \leq \mathbf{A}_i^+ \boldsymbol{\tau}_{v_i} - \mathbf{b}, \\ & \boldsymbol{\sigma}_l \leq \boldsymbol{\sigma}_i \leq \boldsymbol{\sigma}_u. \end{aligned} \quad (18)$$

In order to minimize this constrained nonlinear multivariable objective function, a numerical method can be applied. In this study, the built-in MATLAB (The MathWorks, Inc.) function *fmincon()* is used for that purpose. The effect of using the previous solution on the acquisition of positive cable tensions with the above formulation is discussed after the control problem is addressed.

5.3. Trajectory Control with Feedback Linearization. In this section, we solve the control problem for the RLS by using feedback linearization technique. Feedback linearization control, also known as computed-torque control, aims to cancel the nonlinearities of a system and reduce it to a linear system to be controlled by a linear servo law. Decomposing the controller design into model-based and servo-based portions helps solve the control problem in a more systematic way. Model-based portion contains a model of the nonlinearity and includes system parameters. Servo-based portion includes only the control law and is independent of the model-based portion and, therefore, system parameters [15]. The dynamics equation for the RLS is

$$\mathbf{M}(q) \ddot{\mathbf{q}} + \mathbf{V}(q, \dot{q}) + \mathbf{G}(q) = \boldsymbol{\tau}_v, \quad (19)$$

where $\boldsymbol{\tau}_v = \mathbf{A} \mathbf{t}$ is the virtual input to the system which was previously introduced as an intermediate variable. This virtual input is utilized to be able to find positive and nonzero cable tensions. The model-based portion of the controller is defined as

$$\boldsymbol{\tau}_v = \boldsymbol{\alpha} \boldsymbol{\tau}'_v + \boldsymbol{\beta}, \quad (20)$$

where

$$\boldsymbol{\alpha} = \mathbf{M}(q), \quad (21)$$

$$\boldsymbol{\beta} = \mathbf{V}(q, \dot{q}) + \mathbf{G}(q).$$

The servo-based portion that includes a proportional-derivative control law is

$$\boldsymbol{\tau}'_v = \ddot{\mathbf{q}}_d + \mathbf{K}_p \mathbf{e} + \mathbf{K}_d \dot{\mathbf{e}}, \quad (22)$$

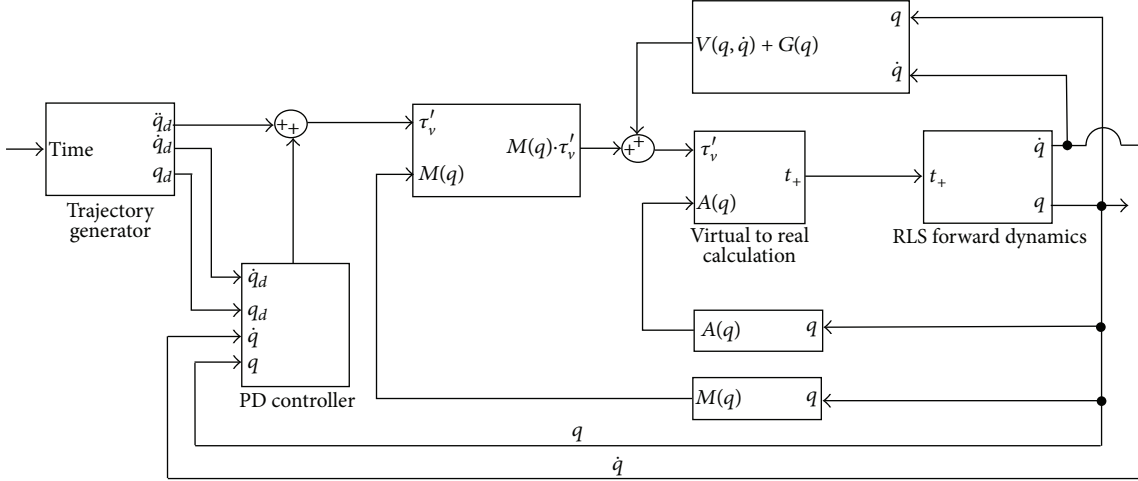


FIGURE 4: RLS controller schematic.

where $\ddot{\mathbf{q}}_d$ is the desired accelerations, \mathbf{K}_p and \mathbf{K}_d are, respectively, proportional and derivative control gain matrices. The control gains are both 15×15 and diagonal matrices which implies that the PD control law is implemented independently for each degree of freedom (i.e., angle of rotation). $\mathbf{e} = \mathbf{q}_d - \mathbf{q}$ is the servo error between desired and actual trajectories. The error dynamics of the proposed control law can be found by first plugging (22) into (20) and the resulting equation into the dynamics equation:

$$\ddot{\mathbf{q}} = \ddot{\mathbf{q}}_d + \mathbf{K}_p \mathbf{e} + \mathbf{K}_d \dot{\mathbf{e}}. \quad (23)$$

Noting that $\dot{\mathbf{e}} = \dot{\mathbf{q}}_d - \dot{\mathbf{q}}$ above equation written in error space becomes

$$\ddot{\mathbf{e}} + \mathbf{K}_d \dot{\mathbf{e}} + \mathbf{K}_p \mathbf{e} = 0. \quad (24)$$

The equation above is a second-order differential equation and the coefficients now can be chosen to shape the dynamic response of the system. It should also be noted that the left-hand side of the equation must be a Hurwitz polynomial to provide a stable closed-loop response.

Controller architecture for the RLS is shown in Figure 4. It is composed of a trajectory generator, PD controller, virtual to real calculation, and the forward dynamics blocks. Trajectory generator provides the desired angles (\mathbf{q}_d) at every time step based on a quintic polynomial in order to control first and second derivatives ($\dot{\mathbf{q}}_d$, $\ddot{\mathbf{q}}_d$) of the desired angles at the beginning and end of a path segment or trajectory. These derivatives are generally set to be zero to be able to obtain smooth movement of the robot. PD controller, as discussed previously, is needed to control the robot to follow the trajectory with a diminishing error between actual and desired angles of rotation. Controller gains can be chosen to obtain desired dynamic response, and they affect the error dynamics, that is, how fast the robot can recover from an error at any given time during the task. Virtual to real calculation is necessary in order to acquire positive nonzero cable tensions (\mathbf{t}_+).

TABLE 1: Desired angles of rotation for right bending [5].

Motion segment	α ($^\circ$) [†]	β ($^\circ$) [†]	γ ($^\circ$) [†]
L5-S1	0.50	1.00	2.60
L4-L5	1.00	1.00	3.00
L3-L4	0.75	0.75	3.10
L2-L3	0.75	0.50	3.50
L1-L2	0.25	0.00	2.75

[†] α , β and γ are the angles of rotations about x -, y -, and z -axes, respectively.

The inner workings of this block were detailed previously in the section that describes the cable tension optimization.

5.4. Simulation Results. The simulations were run for six different motion types (flexion/extension, right/left lateral bendings and right/left torsions). Due to space considerations, however, the results for only (right) lateral bending motion are presented. The results for remaining motion types can be found in [10]. Lateral bending is one of the most involved motion types in terms of the existence of coupled movements. Coupled movement of vertebrae occurs when the motion to the lumbar spine is induced in one specific plane (in frontal plane for lateral bending) which causes vertebrae to move in more than one plane. The desired angles of rotations (Table 1) for the RLS are acquired from experimental data [5] which were obtained after applying 2.5 Nm pure moment about z -axis of the freshly frozen lumbosacral spine specimens. The masses of the vertebrae from L5 to L1 are 0.0125, 0.0132, 0.0123, 0.0113 and 0.0100 kg, respectively. The trajectory generator starts at 0.1 sec, and simulation is run for a total of 1.5 sec. The control gains used are 5 and 6 for \mathbf{K}_p and \mathbf{K}_d , respectively.

Figure 5 shows the commanded (desired) and actual paths followed by the RLS. The corresponding tracking errors are shown in Figure 6.

In order to test the effectiveness of the new method for solving positive continuous cable tensions, the cable

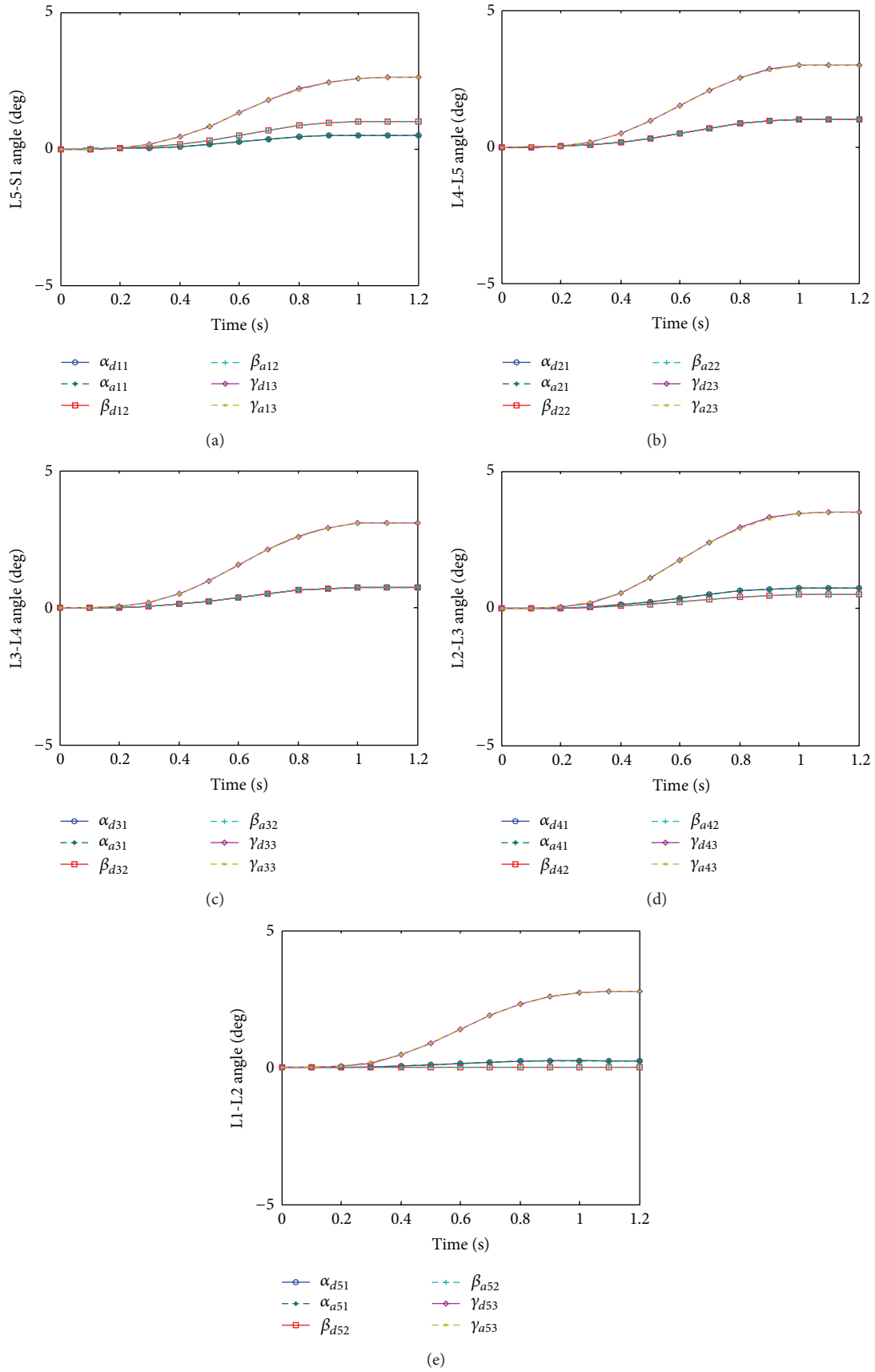


FIGURE 5: Desired and actual angles of rotations (right bending, $\alpha_d, \beta_d, \gamma_d$: desired, $\alpha_a, \beta_a, \gamma_a$: actual angles of rotation about x-, y- and z-axes, resp.).

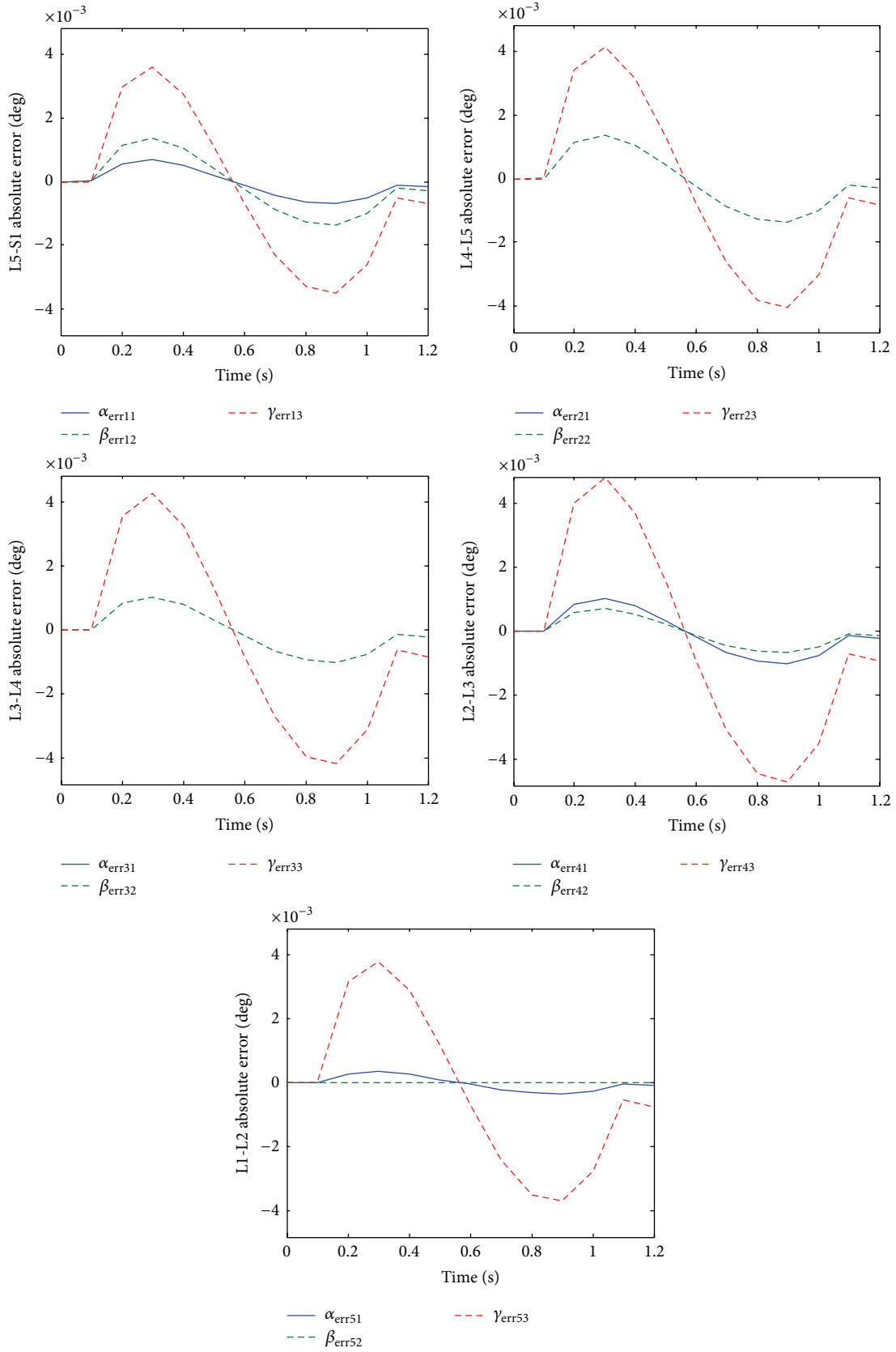


FIGURE 6: Tracking errors (right bending, $\alpha_{err} = \alpha_d - \alpha_a$, $\beta_{err} = \beta_d - \beta_a$, and $\gamma_{err} = \gamma_d - \gamma_a$).

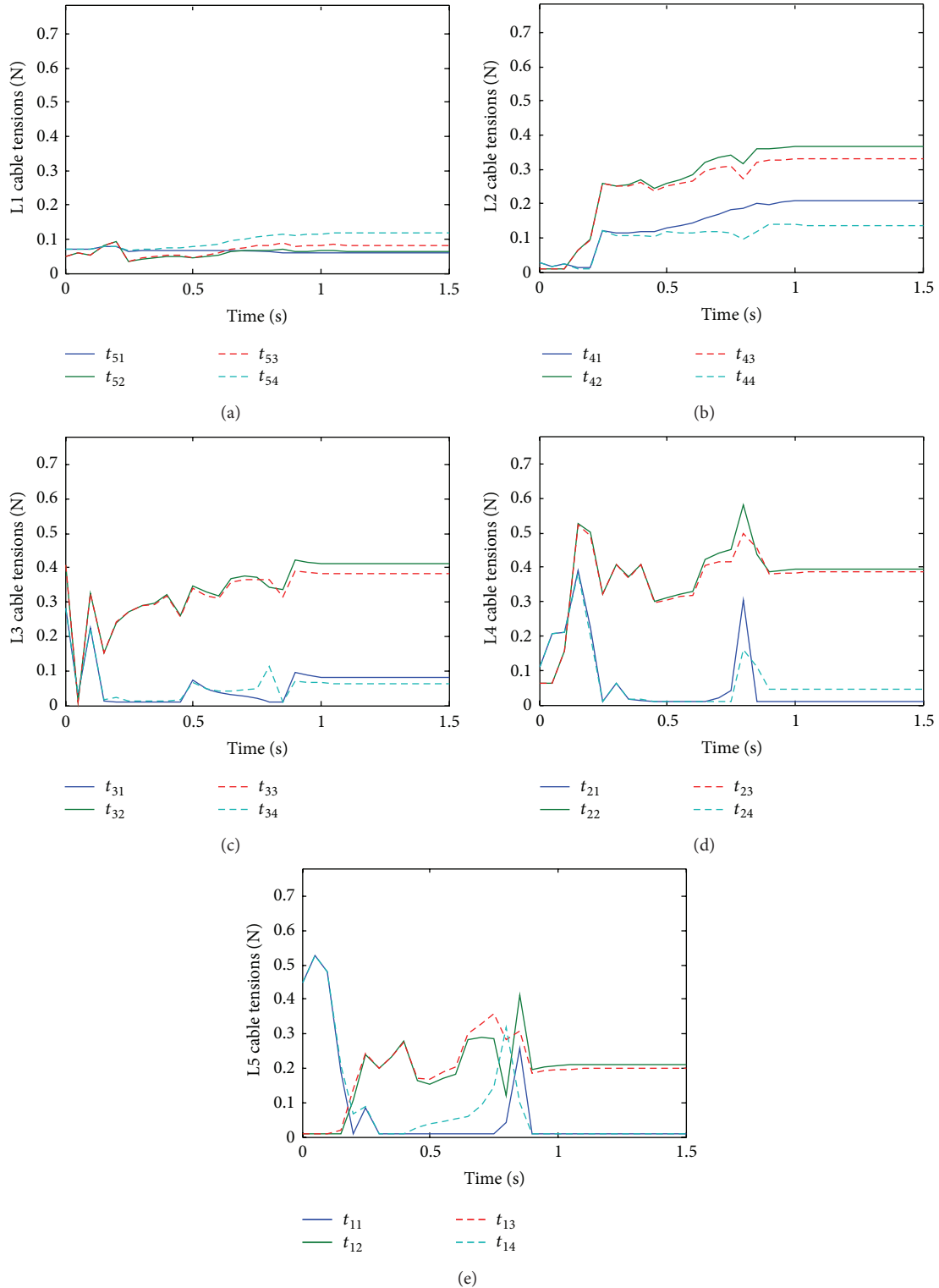


FIGURE 7: Cable tensions solved without continuity algorithm (right bending).

tensions are solved with and without implementing the proposed continuity algorithm. Figure 7 shows the cable tensions without the continuity algorithm. Even though all tensions are solved to be positive discontinuity is apparent.

Figure 8 shows the results of the simulation with the same parameters before but with the continuity algorithm. It is seen that the solved tensions are all positive and continuous in time.

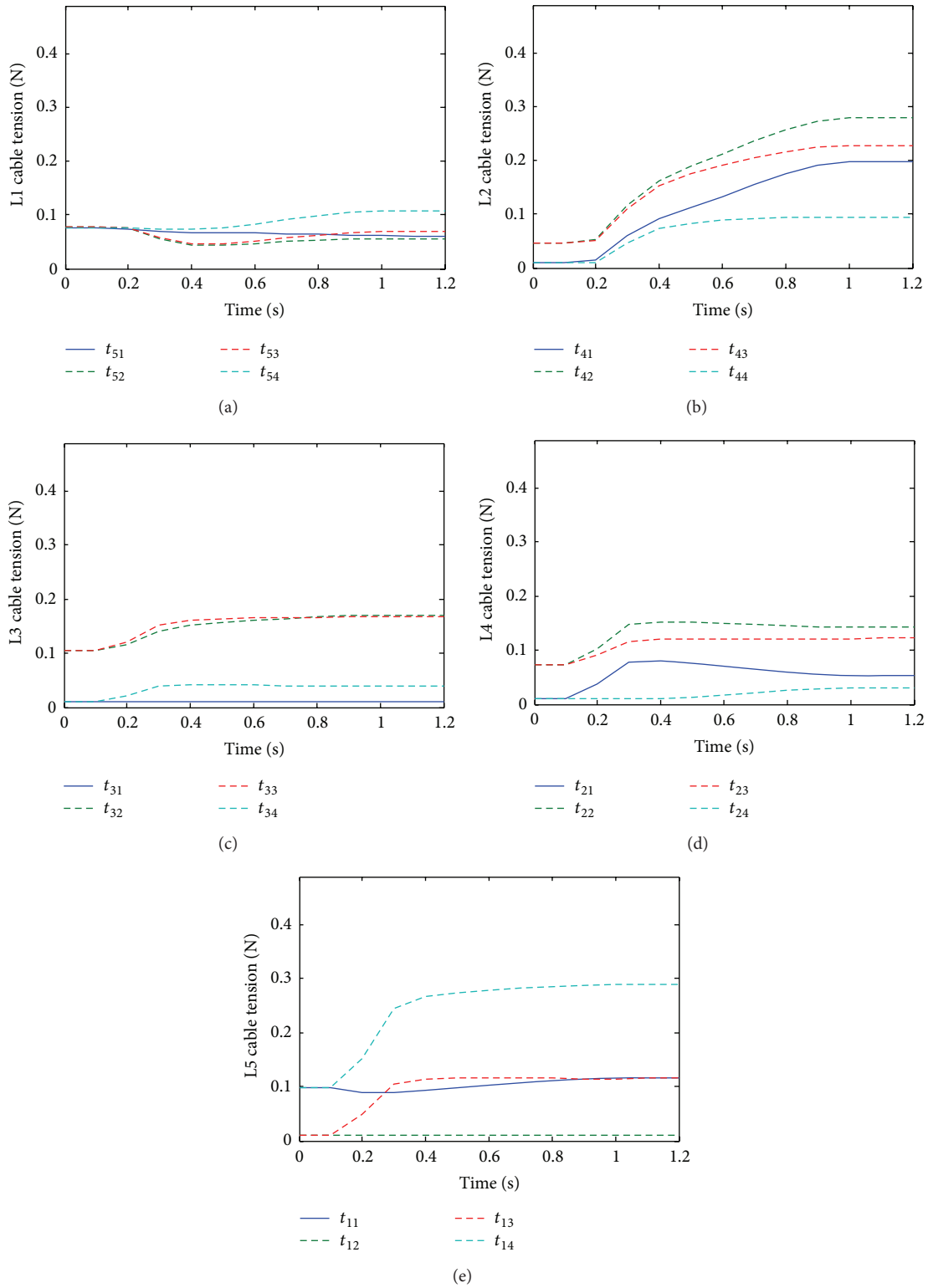


FIGURE 8: Cable tensions solved with continuity algorithm (right bending).

6. Discussion

Conceptually, the RLS was designed to support some apparent needs that the instructors and the students of institutions that teach palpatory diagnosis are currently facing. These needs can be collected under three main items:

- (1) limited variation of dysfunctions that can be practiced in a lab environment,
- (2) repeatability issues due to the inherent characteristics of tissues to change properties due to repetitive manual manipulation,
- (3) lack of an objective assessment tool to evaluate the transfer of practical knowledge from the instructor to the students.

With the RLS, there would be virtually no limit to the abnormal movement patterns to practice. These abnormal movement patterns could be programmed easily if the data are readily available, that is, if experimental data or accurate models exist. If no data are available, experience of professional experts may be utilized to generate the required data for abnormal movement patterns by trial and error until a general consensus among the experts is reached. The RLS, as any other robot, would be repeatable (to a certain degree that needs to be calculated and validated experimentally) by configuring itself correctly according to the user's input from the joystick/haptic device. As mentioned previously, there exists no assessment device for instructors to objectively evaluate progress and success of the students in real-life situations. By means of the RLS, all students can be objectively tested on identifying the normal/abnormal movement patterns of the lumbar spine. Since the RLS is also repeatable, any number of students may be tested for the same or different dysfunctions as needed.

The equations of motion were complex and highly nonlinear. This is expected due to the number of degrees of freedom considered (15 DOFs) and the actuation redundancy. Note that, for systems the links of which are actuated at the joints, τ in (1) is independent of the generalized coordinates (q) when they are defined to be the joint variables (angle of rotation for revolute and distance for prismatic joints). However, as shown in the text, τ for the robotic lumbar spine is a function of the generalized coordinates since the actuation is not performed at the spherical joints. This adds to the complexity of the equations.

The simulations for the control of the RLS showed that the tracking errors were less than 0.005 degrees for all degrees of freedom during the entire range of motion which implied that the designed controller performed as expected. The results of the simulations also showed that the new method proposed to solve for positive cable tensions was very effective in eliminating the spikes in the cable tensions. The results for all motion types, when the effect of the continuity algorithm was tested, were very similar to the results of the right lateral bending as presented here.

The RLS was designed to change configuration by a force-feedback joystick or an affordable haptic device (such as Falcon from Novint Technologies Inc.). By moving the joystick, the angles of rotations will be commanded to the RLS,

therefore representing a normal lumbar spine movement. A static model of the human lumbar spine was derived to obtain these normal movement patterns of the lumbar spine for six different types of motion. It is also planned that some abnormalities consistent with known dysfunctional movement patterns (vertebral fusion, rotational resistance of vertebra about an axis, etc.) could be mimicked based on these normal movement patterns.

7. Conclusion

The dynamic model and nonlinear control of a 15-degree-of-freedom, cable-actuated robotic lumbar spine (RLS) were presented. A new method was proposed that enables the solution of positive and continuous cable tensions for cable-actuated robots. The simulation results confirmed that the tracking errors during the simulated motion were small and the proposed continuity algorithm proved to be very effective in obtaining positive cable tensions that are also continuous in time.

References

- [1] J. N. Howell, R. R. Conatser, R. L. Williams, J. M. Burns, and D. C. Eland, "The virtual haptic back: a simulation for training in palpatory diagnosis," *BMC Medical Education*, vol. 8, article 14, 2008.
- [2] I. Mizuuchi, M. Inaba, and H. Inoue, "A flexible spine human-form robot—development and control of the posture of the spine," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2099–2104, November 2001.
- [3] I. Mizuuchi, R. Tajima, T. Yoshikai et al., "The design and control of the flexible spine of a fully tendon-driven humanoid "Kenta"," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2527–2532, October 2002.
- [4] L. Roos, F. Guenter, A. Guignard, and A. G. Billard, "Design of a biomimetic spine for the humanoid robot Robota," in *Proceedings of the 1st IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechanics, 2006 (BioRob '06)*, pp. 329–334, February 2006.
- [5] M. M. Panjabi, T. R. Oxland, I. Yamamoto, and J. J. Crisco, "Mechanical behavior of the human lumbar and lumbosacral spine as shown by three-dimensional load-displacement curves," *Journal of Bone and Joint Surgery A*, vol. 76, no. 3, pp. 413–424, 1994.
- [6] E. Karadogan and R. L. Williams II, "A cable-actuated robotic lumbar spine for palpatory training of medical students," in *Proceedings of the ASME International Design Technical Conferences, 34th Mechanisms and Robotics Conference*, pp. 309–318, Quebec, Canada, August 2010.
- [7] J. A. Sullivan, A. H. Soni, and A. G. Patwardhan, "Kinematic analysis of intervertebral motion of the human lumbar spine," Final Report, OREF, 1979.
- [8] A. H. Soni, J. A. Sullivan Jr., A. G. Patwardhan, M. R. Gudavalli, and J. Chitwood, "Kinematic analysis and simulation of vertebral motion under static load. Part I: kinematic," *Journal of Biomechanical Engineering*, vol. 102, no. 2, pp. 105–111, 1982.
- [9] M. Sharma, N. A. Langrana, and J. Rodriguez, "Modeling of facet articulation as a nonlinear moving contact problem: sensitivity study on lumbar facet response," *Journal of Biomechanical Engineering*, vol. 120, no. 1, pp. 118–125, 1998.

- [10] E. Karadogan, *A cable-actuated robotic lumbar spine as the haptic interface for palpatory training of medical students [Ph.D. dissertation]*, Ohio University, Athens, Ohio, USA, 2011.
- [11] M. W. Spong, S. Hutchinson, and M. Vidyasagar, *Robot Dynamics and Control*, John Wiley & Sons, 2004.
- [12] R. L. Williams II, P. Gallina, and J. Vadia, "Planar translational cable-direct-driven robots," *Journal of Robotic Systems*, vol. 20, no. 3, pp. 107–120, 2003.
- [13] Y. Shen, H. Osumi, and T. Arai, "Manipulability measures for multi-wire driven parallel mechanisms," in *Proceedings of the IEEE International Conference on Industrial Technology*, pp. 550–554, 1994.
- [14] K. Yu, L.-F. Lee, and V. N. Krovi, "Simultaneous trajectory tracking and stiffness control of cable actuated parallel manipulator," in *Proceedings of the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (IDETC/CIE '09)*, pp. 55–63, September 2009.
- [15] J. Craig, *Introduction to Robotics: Mechanics and Control*, Addison-Wesley, 1989.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

