Robotica (2013) volume 31, pp. 611–621. © Cambridge University Press 2012
doi:10.1017/S026357471200063X

Optimal spring balancing of robot manipulators in point-to-point motion

A. Nikoobin†*, M. Moradi‡ and A. Esmaili§

†*Department of Mechanical Engineering, Tarbiat Modares University, Tehran, Iran*
‡*Department of Mechanical Engineering, Amirkabir University of Technology, Tehran, Iran*
§*Department of Computer & Electrical Engineering, Semnan University, Semnan, Iran*

(Accepted September 28, 2012. First published online: November 2, 2012)

SUMMARY
The balancing of robotic systems is an important issue, because it allows significant reduction of torques. However, the literature review shows that the balancing of robotic systems is performed without considering the traveling trajectory. Although in static balancing the gravity effects on the actuators are removed, and in complete balancing the Coriolis, centripetal, gravitational, and cross-inertia terms are eliminated, but it does not mean that the required torque to move the manipulator from one point to another point is minimum. In this paper, "optimal spring balancing" is presented for open-chain robotic system based on indirect solution of open-loop optimal control problem. Indeed, optimal spring balancing is an optimal trajectory planning problem in which states, controls, and all the unknown parameters associated with the springs must be determined simultaneously to minimize the given performance index for a predefined point-to-point task. For this purpose, on the basis of the fundamental theorem of calculus of variations, the necessary conditions for optimality are derived that lead to the optimality conditions associated with Pontryagin's minimum principle and an additional condition associated with the constant parameters. The obtained optimality conditions are developed for a two-link manipulator in detail. Finally, the efficiency of the suggested approach is illustrated by simulation for a two-link manipulator and a PUMA-like robot. The obtained results show that the proposed method has dominant superiority over the previous methods such as static balancing or complete balancing.

KEYWORDS: Optimal balancing; Robot manipulators; Trajectory planning; Optimal control; Pontryagin minimum principle; Springs.

1. Introduction

Achieving the optimal performance of robot manipulators in repeating tasks has attracted lot of attention over the recent years. In an optimal task, minimum consumed energy, minimum torque, or minimum time can be considered. Often the used manipulators in an assembly or manufacturing line are fixed so for a new product, the end effectors of the manipulators and their predefined trajectories can be changed. Since changing the robot and its structure is a hard task or often impossible, besides the trajectory planning, one efficient way to increase the robot performance is balancing. Balancing introduces some simple modifications in the architecture of the original mechanism, which actually simplifies its dynamic model and, as a result, its control as well. Besides control simplification, balancing can also provide reduction of driving torques. Basically, balancing can be categorized into two types: active and passive balancing. In active balancing, an external electric, pneumatic, or hydraulic force is applied to the system,[1] while in passive balancing, compensation inertias[2,3] or springs[4,5] are used. Since the additional actuators are not required in passive balancing, it is more economical and simpler than the active one.

Two methods studied in literature for passive balancing are using counter-weights and using springs. The balancing by masses is due to added counter-weights or due to link's mass redistribution. Counter-weight balancing is simple and has some advantages, but it increases the inertia of the manipulators. In case of balancing by springs, changes in the mass and inertia parameters of the robot mechanism are insignificant because the weight of the springs is very less in comparison with link weight.[4] Unlike the counter-weight balancing, which is straightforward and almost simple, spring balancing can be performed in different forms[5] as: balancing by springs jointed directly with links,[6] balancing by using a cable and pulley arrangement,[7] balancing by using an auxiliary linkage,[8] balancing by using a cam mechanism,[9] and balancing by using gear train.[10] Arakelian *et al*. [5] have reviewed different types of spring balancing mechanisms. Ulrich *et al*.[11] presented a cable–pulley-spring compensation method for illuminating the influence of gravity for one-link manipulator. Kolarski *et al*. [12] compared the dynamical behavior of the unbalanced spring and counter-weight-balanced PUMA robot configuration. Agrawal *et al*. [8, 13, 14] presented spring balancing for two-degree-of-freedom (DOF) spatial manipulator and three-DOF spatial manipulator. They described the theory of gravity-balanced spatial robotic manipulators through a hybrid strategy that uses springs in addition to identification of the center of mass using auxiliary parallelograms. Herder *et al*. [15] have used the storage spring concept for spring-to-spring balancing as energy-free adjustment method in gravity equilibrators.

On the other hand, passive balancing approaches can be classified into four types: static balancing,[12,16]

* Corresponding author. E-mail: anikoobin@iust.ac.ir

dynamic balancing,[17] complete balancing,[18,3] and optimal balancing.[19–21] A machine is said to be static-balanced if its potential energy is constant for all possible configurations.[12] Dynamic balancing has one step more than static balancing, and it is reducing the reaction forces and moments on the base and among actuators, for all situations. Thus, the dynamic-balanced robot will lightly transfer some reactions to its adjacent actuators and environments when it is operated.[17] Complete balancing is the third type of balancing technique that brings some modifications to unbalanced mechanisms in such a way as to obtain static balancing and complete decoupling of dynamic equations.[18] All of these methods are applied without considering the trajectory of robot, whereas in optimal balancing, this aspect is also considered. Saravanan *et al*. performed optimum balancing for an industrial robot, while a rectangular path must be tracked.[19] Ravichandran *et al*. by considering a nonlinear proportional-derivative (PD) controller for a two-DOF robot manipulator applied optimal balancing in order to determine the controller parameters and counter-weight values.[20] Optimal balancing problem has been solved for counter-weight-balanced robot manipulators in point-to-point motion by Nikoobin *et al*. [21].

In all the above-mentioned balancing methods, it is assumed that the structure and parameter of the robot are predefined and only balancing elements such as spring or mass are added to the main mechanism in order to achieve the desired performance. While for designing a new mechanism, there are other methods to attain the given objective function. Gosselin *et al*. based on the conditions of dynamic balancing, without using the separate counter-rotations, determined the proper architecture and parameters for a planar four-bar linkage.[22] The Design for control (DFC) approach in which the structure and parameter of both the machine body and the control algorithm are designed to fulfill the specific task has been presented for a five-bar close-chain robot by Cheng *et al*. [23, 24]. To this end, a PD plus robust term control algorithm in the DFC approach has been proposed to obtain the desired performance in terms of change of tasks.[23]

Generally speaking, optimal balancing is the trajectory planning problem with some unknown parameters. Optimal trajectory planning of manipulators is based on optimizing the objective function while the dynamic equations of motion as well as bounds on joint positions, velocities, and torques must be taken into account.[25] This generic optimal control problem is so complicated that it can only be solved by a computer.[26] At this point, two strategies are well-known for path optimization: indirect methods[27] and direct methods.[28] These techniques are used in many articles and they have own benefits and weaknesses.[25–31]

Nikoobin and Moradi have presented optimal balancing for counter-weight-balanced robot manipulators based on an indirect solution of the optimal control problem.[21] In fact, an optimal trajectory planning problem is outlined in which the states, the controls, and the values of counter-weights are determined simultaneously in order to minimize the given performance index for a predefined point-to-point task. Although the optimal counter-weight balancing method in comparison with previous methods such as unbalancing,[29] static balancing,[12] or complete balancing[3] has significant superiorities to optimize the given performance index, it suffers from increasing inertia. In order to overcome this drawback in this paper, optimal balancing is developed for the spring-balanced robot manipulators and it is shown that spring balancing is more practical and efficient than counter-weight balancing.

The paper is organized as follows: general formulation of optimal balancing and static balancing is presented in Sections 2 and 3, respectively. Then, in Section 4, using the obtained general formulation, modeling, and optimality conditions are derived for a two-link manipulator in detail. Finally, in order to verify the method, simulation results for a two-link manipulator and a PUMA-like robot are presented in Section 5.

2. Optimal Balancing Approach

Optimal balancing is simultaneous achievement of unknown parameters and trajectory of system using optimal control. The optimal control problem for a dynamic system involving parameters can be stated as follows:[21,30]

Find the parameter vector $\mathbf{b}$ and continuous admissible control history $\mathbf{u} = [t_0,\ t_f] \rightarrow \Omega \subseteq \mathbb{R}^m$ generating the corresponding state trajectory, $\mathbf{x} = [t_0,\ t_f] \rightarrow \mathbb{R}^n$, which minimizes the objective function

$$J = \phi(\mathbf{x}_f, \mathbf{b}) + \int_{t_0}^{t_f} L(\mathbf{x}, \mathbf{u}, \mathbf{b})\, dt, \tag{1}$$

subject to the system dynamics

$$\mathbf{M}(\mathbf{q}, \mathbf{b})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{b}) + \mathbf{G}(\mathbf{q}, \mathbf{b}) = \mathbf{u}, \tag{2}$$

where $\mathbf{M} \in \mathbb{R}^{n\times n}$ is the mass matrix, $\mathbf{C} \in \mathbb{R}^n$ is the coupling matrix, $\mathbf{G} \in \mathbb{R}^n$ is a gravity-dependent term, $\mathbf{q} \in \mathbb{R}^n$ is the position vector of manipulator, and $\mathbf{b} \in \mathbb{R}^r$ is the parameter vector. Generally, vector $\mathbf{b}$ contains all the unknown constant parameters in which their optimal value must be obtained during the problem solution. $\mathbf{u} \in \mathbb{R}^m$ is the control vector, Ω is an acceptable control region in $\mathbb{R}^m$, t_0 and t_f are initial and final time, and $\mathbf{x}_f$ is the predefined final state. ϕ and L are scalar continuously differentiable functions in which ϕ is the final state penalty term and L is the integrand of the cost function. ϕ and L can be selected to obtain different optimal control problems such as minimum time, terminal control, minimum effort, tracking problem, or regulator problem.[32] By defining the continuous state vector as

$$\mathbf{x} = \begin{bmatrix}\mathbf{x}_1^T & \mathbf{x}_2^T\end{bmatrix}^T = \begin{bmatrix}\mathbf{q}^T & \dot{\mathbf{q}}^T\end{bmatrix}^T, \tag{3}$$

the dynamic Eq. (2) can be written in state space form as

$$\mathbf{f} = \begin{bmatrix}\mathbf{f}_1 \\ \mathbf{f}_2\end{bmatrix} = \begin{bmatrix}\mathbf{x}_2 \\ \mathbf{M}^{-1}(\mathbf{x}_1, \mathbf{b})[\mathbf{u} - \mathbf{C}(\mathbf{x}_1, \mathbf{x}_2, \mathbf{b}) - \mathbf{G}(\mathbf{x}_1, \mathbf{b})]\end{bmatrix}, \tag{4}$$

where $\mathbf{f}$ is piecewise continuous in the variables $\mathbf{u}$ and t, and is continuously differentiable with respect to $\mathbf{x}$. So, the system dynamics can be written finally as

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{b}) \tag{5}$$

https://doi.org/10.1017/S026357471200063X Published online by Cambridge University Press

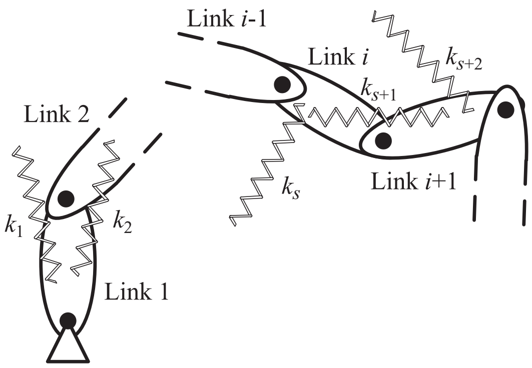

Fig. 1. The general representation of open-chain robot manipulator including springs.

with the given initial condition

$$\mathbf{x}(t_0) = \mathbf{x}_0 \tag{6}$$

and the prescribed final conditions

$$\mathbf{x}(t_f) = \mathbf{x}_f, \tag{7}$$

where $\mathbf{x} \in \mathbb{R}^n$ is the state vector and t_0 is initial time. The constant state vector can be appended for considering the parameters as $\dot{\mathbf{b}} = \mathbf{0}$. Then, by defining the Hamiltonian function as

$$H = L + \boldsymbol{\lambda}^T \mathbf{f} + \boldsymbol{\mu}^T \mathbf{b}, \tag{8}$$

optimality conditions to minimize the objective function Eq. (4), subject to dynamic Eq. (5), and boundary condition Eqs. (6) and (7) can be derived as follows:

$$\text{ODEs}: \ \dot{\mathbf{x}} = \nabla_\lambda H, \ \dot{\boldsymbol{\lambda}} = -\nabla_x H, \ \dot{\boldsymbol{\mu}} = -\nabla_{\mathbf{b}} H,$$

$$\text{Algebraic Equations: } \nabla_u H = 0,$$

$$\text{BCs}: \mathbf{x}(t_0) = \mathbf{x}_0, \boldsymbol{\mu}(0) = \mathbf{0}, \mathbf{x}(t_f) = \mathbf{x}_f, \boldsymbol{\mu}(t_f) = \phi_{\mathbf{b}}. \tag{9}$$

Derivation of the aforesaid optimality conditions are given in ref. [21] in detail. With the realization that the parameters behave like states, for the optimal control $\mathbf{u}^*$ the Legendre–Clebsch condition,

$$\nabla^2 H(\mathbf{x}, \mathbf{u}^*, \boldsymbol{\lambda}, \mathbf{b}) \geq 0, \tag{10}$$

must be satisfied.

3. Static Balancing of Robot Manipulators

3.1. General formulation

The general spring balancing schematic is presented in Fig. 1 for open-chain robots, and it has been used for structure in static and optimal balancing. In this figure, k_i denotes the spring between bodies whereas the ground body is inertial and considered as the 0th body.

The existence of spring in the manipulator will change the potential energy of the manipulator. For statically balanced robotic systems, the weight of the links does not exert any force at the actuators for any configuration. In other words, it removes the gravitational effects in mechanical systems.[2] Another appropriate and practical meaning of this concept can be stated to be the constant potential energy of the manipulator. This can be applied by establishing the additional mechanical elements into the system, such as counter-weights or springs to make potential energy constant. The use of counter-weights has some advantages along with disadvantages that serve to limit its usefulness. For instance, it is undesirable to provide an extra mass on robot where minimum weight is an important criterion. Also, adding the counter-weights increases the moment of inertia of the manipulator. So, applying the springs instead of counter-weights affords more convenience. Here, the general theory for static balancing of manipulator-based energy method is developed. The static balancing using energy index can be stated as

$$\frac{\partial P}{\partial q_i} = 0, \quad i = 1, \ldots, n, \tag{11}$$

where P is the total potential function of the manipulator and q_i is the position of link i as generalized coordination of the system. Consequently, for systems consisting of springs, the unknown parameters can be found by using these n equations. In some cases, these equations have no solution and this means such systems are not completely balanceable. For the general system shown in Fig. 1, the potential energy can be computed as

$$\begin{aligned} P &= \sum_{j=1}^{n_m} P_{jG} + \sum_{j=1}^{n_k} P_{jE} \\ &= \sum_{j=1}^{n_m} \left(\frac{1}{2} k_j (l'_j - l_j)^2 \right) + \sum_{j=1}^{n_k} (m_j g h_j), \end{aligned} \tag{12}$$

where P_G is the gravitational potential function and P_E is the elastic potential of the system. l denotes initial length of the spring, l' denotes deflected length of the spring, m denotes mass of the link, h denotes the height of center of gravity for the link, k denotes spring stiffness, g denotes gravitational acceleration, n_k denotes number of springs, and n_m denotes number of masses in the system. Substituting Eq. (12) into Eq. (11) yields

$$\sum_{j=1}^{n_m} \left[k_j (l'_j - l_j) \frac{\partial l'_j}{\partial q_i} \right] + \sum_{j=1}^{n_k} \left(m_j g \frac{\partial h_j}{\partial q_i} \right) = 0$$
$$i = 1, \ldots, n, \tag{13}$$

where represents n nonlinear equations with n unknown parameters, which depend on the choice of springs structure, which may have one, many, or no solution. If there is a solution for Eq. (13), n unknown parameters are obtained, which eliminates the gravitational forces as

$$\mathbf{G}(\mathbf{q}) = 0. \tag{14}$$

So, the system dynamic described in Eq. (2) reduces to

$$\hat{\mathbf{u}} = \hat{\mathbf{M}} \ddot{\mathbf{q}} + \hat{\mathbf{C}}(\mathbf{q}, \dot{\mathbf{q}}), \tag{15}$$

https://doi.org/10.1017/S026357471200063X Published online by Cambridge University Press

where $\hat{\mathbf{M}}$ is the inertia matrix and $\hat{\mathbf{C}}$ is the vector of centripetal and Coriolis forces of the static balanced manipulator.

3.2. Static balancing in term of optimal balancing
In order to show the relation between static and optimal balancing, the scaled time $\tau \in [0\ 1]$ is defined to represent the time as

$$t = t_f \tau. \tag{16}$$

Using this time-scaling, the derivatives of position vector become

$$\dot{\mathbf{q}} = \frac{d\mathbf{q}}{dt} = \frac{d\mathbf{q}}{t_f d\tau} = \frac{\mathbf{q}'}{t_f}, \quad \ddot{\mathbf{q}} = \frac{d^2\mathbf{q}}{dt^2} = \frac{\mathbf{q}''}{t_f^2}, \tag{17}$$

so Eq. (2) can be rewritten as follows:

$$\mathbf{M}(\mathbf{q}, \mathbf{b})\frac{\mathbf{q}''}{t_f^2} + \mathbf{C}\left(\mathbf{q}, \frac{\mathbf{q}'}{t_f}, \mathbf{b}\right)\frac{\mathbf{q}''}{t_f} + \mathbf{G}(\mathbf{q}, \mathbf{b}) = \mathbf{u}. \tag{18}$$

The effort-optimal pay-off functional is now selected as

$$J = \int_0^1 \|\mathbf{u}\|^2 \, d\tau. \tag{19}$$

By substituting Eq. (18) into Eq. (19), one can write

$$J = \int_0^1 \left\| \mathbf{M}(\mathbf{q}, \mathbf{b})\frac{\mathbf{q}''}{t_f^2} + \mathbf{C}\left(\mathbf{q}, \frac{\mathbf{q}'}{t_f}, \mathbf{b}\right)\frac{\mathbf{q}''}{t_f} + \mathbf{G}(\mathbf{q}, \mathbf{b}) \right\|^2 d\tau. \tag{20}$$

By approaching $t_f \to \infty$, since the mass matrix and coupling/Coriolis are norm bounded matrices thus, all terms in Eq. (20) will vanish except the gravitational one. Thus,

$$\lim_{t_f \to \infty} J = \int_0^1 \|\mathbf{G}(\mathbf{q}_s, \mathbf{b}_s)\|^2 \, d\tau, \tag{21}$$

where $\mathbf{q}_s$ denotes the optimal static trajectory and $\mathbf{b}_s$ the static-balanced vector. The minimum solution of this function is defined as static balancing of the robotic manipulator in terms of optimal balancing.

As usual, static balancing is considered as solution of the vector equation

$$\mathbf{G}(\mathbf{q}_s, \mathbf{b}_s) \equiv \mathbf{0}. \tag{22}$$

It is obvious that this leads to global minimum of Eq. (21). Thus, by approaching the final time to infinity, optimal balancing leads to static balancing.

4. Static and Optimal Balancing of Two-Link Manipulator

4.1. Modeling of two-link manipulator
Here, three different conditions are considered: unbalanced, statically balanced, and optimally balanced manipulator. Dynamic equations of all these cases can be presented in the general form. Using the structure presented in Fig. 2 introduced by Agrawal,[14] a two-link manipulator can be statically balanced.

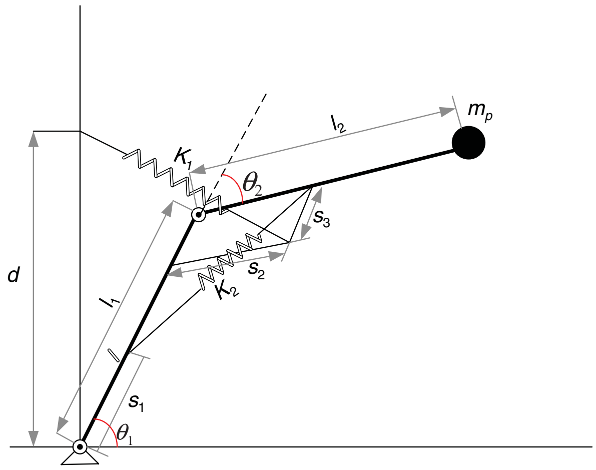

Fig. 2. (Colour online) Two-link manipulator with balancing springs.

In deriving the dynamic equations, the zero-free-length springs are supposed. For zero-free-length springs, the potential energy using Eq. (12) can be written as follows:

$$\begin{aligned} P &= -m_1 g r_{g1} \sin\theta_1 - m_2 g r_{g2} \sin(\theta_1 + \theta_2) \\ &\quad + m_{s1} g \left[\left(l_1 - \frac{1}{2}s_1\right)\sin\theta_1 + \frac{1}{2}s_2 \cos(\theta_1 + \theta_2)\right] \\ &\quad - m_{s2} g \left[s_1 \sin\theta_1 + s_2 \sin(\theta_1 + \theta_2)\right] + m_p[l_1 \sin\theta_1 \\ &\quad + l_2 \sin(\theta_1 + \theta_2)] + \frac{1}{2}k_1 x_1^2 + \frac{1}{2}k_2 x_2^2, \end{aligned} \tag{23}$$

where m_{s1} and m_{s2} are mass of the fractional mechanism, m_1 and m_2 are mass of links, m_p is the payload mass, l_1 and l_2 are length of links, r_{gi} denotes the distance from joint i to the center of mass of the link i, k_1 and k_2 are stiffness of springs, x_{01} and x_{02} are initial length of springs, and d, s_1, s_2, and s_3 are the connecting position of springs as shown in Fig. 2. x_1 and x_2 are instantaneous length of springs, which are functions of θ_1 and θ_2 as follows:

$$\begin{aligned} x_1^2 &= 2(l_1 - s_3)(s_2 \cos\theta_2 - d \sin\theta_1) - 2ds_2 \sin(\theta_1 + \theta_2) \\ &\quad + d^2 + s_2^2 + (l_1 - s_3)^2, \\ x_2^2 &= s_1^2 + (l_1 - s_2)^2 + 2s_2(l_1 - s_1)\cos\theta_2. \end{aligned} \tag{24}$$

For convenience, the parameters α and β are defined as follows:

$$\alpha = m_2 r_{g2} + l_2 m_p, \quad \beta = m_1 r_{g1} + (m_2 + m_p) l_1. \tag{25}$$

The dynamic equations for such a general two-link manipulator can be described as follows:

$$\begin{bmatrix} M_{11} & M_{12} \\ M_{12} & M_{22} \end{bmatrix}\begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} + \begin{bmatrix} G_1 \\ G_2 \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix}, \tag{26}$$

where

https://doi.org/10.1017/S026357471200063X Published online by Cambridge University Press

$$
\begin{aligned}
M_{22} &= m_2 r_{g2}^2 + m_p l_2^2 + I_2,\\
M_{11} &= m_1 r_{g1}^2 + (m_2+m_p) l_1^2 + I_1 + 2 l_1 \alpha \cos\theta_2 + M_{22},\\
M_{12} &= l_1 \alpha \cos\theta_2 + M_{22},\\
C_1 &= -l_1 \alpha \sin\theta_2 (2\dot\theta_1 + \dot\theta_2)\dot\theta_2,\\
C_2 &= l_1 \alpha \dot\theta_1^2 \sin\theta_2,\\
G_1 &= \cos\theta_1 (k_1 d (s_3 - l_1) + \beta g)\\
&\quad + \cos(\theta_1+\theta_2)(-k_1 d s_2 + \alpha g),\\
G_2 &= (s_2(k_2(s_1 - l_1) - k_1(l_1 + s_3)))\sin\theta_2\\
&\quad + (\alpha g - k_1 d s_2)\cos(\theta_1+\theta_2),
\end{aligned} \tag{27}
$$

where I_i denotes the mass moment of inertia of links.

4.2. *Static balancing of two-link manipulator*

In an unbalanced case, the spring parameters of the manipulator are zero ($k_1 = k_2 = 0$), which is called normal case in this paper. Now, in order to achieve the static balancing, s_1, d, and s_3 must be obtained in such a way that the gravity effects in Eq. (27) vanish. Here, k_1, k_2, and s_2 are supposed to be known parameters. Static balancing implies that $G_1 = G_2 = 0$, so by defining the spring parameters as follows:

$$
d = \frac{g\alpha}{k_1}, \quad s_1 = l_1 + \frac{k_1}{k_2}\frac{\beta}{\alpha} s_2, \quad s_3 = l_1 - \frac{\beta}{\alpha} s_2, \tag{28}
$$

static balancing is applied and then, the dynamic parameters in Eq. (27) become

$$
\begin{aligned}
M_{22} &= m_2 r_{g2}^2 + m_p l_2^2 + I_2,\\
M_{11} &= m_1 r_{g1}^2 + (m_2+m_p) l_1^2 + I_1 + 2 l_1 \alpha \cos\theta_2 + M_{22},\\
M_{12} &= l_1 \alpha \cos\theta_2 + M_{22},\\
C_1 &= -l_1 \alpha \sin\theta_2 (2\dot\theta_1 + \dot\theta_2)\dot\theta_2,\\
C_2 &= l_1 \alpha \dot\theta_1^2 \sin\theta_2,\\
G_1 &= G_2 = 0.
\end{aligned} \tag{29}
$$

4.3. *Optimality conditions for normal and static-balanced case*

In this section, using the general formulation mentioned in Section 2, optimality conditions are derived for the considered two-link manipulator at unbalanced and static-balanced cases. For the unbalanced case, all the parameters associated with the springs are supposed to be zero, and finding the optimal trajectory between the two given points of the end-effector is considered. For the static-balanced case, at first, unknown parameters are obtained using Eq. (28), then the optimal trajectory for the given performance index will be achieved. Consequently, in both unbalanced and statically balanced cases, the unknown parameters do not appear in the trajectory optimization procedure. The initial position of the end-effector in XY plane at $t = 0$ is $P_0 = (x_0, y_0)$ and the final position at $t = t_f$ is $P_f = (x_f, y_f)$. The initial and final velocities are considered to be zero. So, by solving the inverse kinematic equations, one can write the boundary conditions as follows:

$$
\begin{aligned}
&\theta_1(0) = \theta_{10},\ \theta_2(0) = \theta_{20}, \theta_1(t_f) = \theta_{1f},\ \theta_2(t_f) = \theta_{2f}\\
&\dot\theta_1(0) = \dot\theta_2(0) = \dot\theta_1(t_f) = \dot\theta_2(t_f) = 0.
\end{aligned} \tag{30}
$$

At the first step, using Eq. (3) and by defining the continuous state vector as follows:

$$
X_1 = \begin{bmatrix}\theta_1(t)\\ \theta_2(t)\end{bmatrix} = \begin{bmatrix}X_1(t)\\ X_2(t)\end{bmatrix}, \quad X_2 = \begin{bmatrix}\dot\theta_1(t)\\ \dot\theta_2(t)\end{bmatrix} = \begin{bmatrix}X_3(t)\\ X_4(t)\end{bmatrix}, \tag{31}
$$

the state space form of Eq. (26), using Eq. (4) becomes

$$
\begin{aligned}
\dot x_1 &= x_3,\\
\dot x_2 &= x_4,\\
\dot x_3 &= \frac{\begin{array}{c}M_{22}(X_1)(\tau_1 - C_1(X_1, X_2) - G_1(X_1))\\ -M_{12}(X_1)(\tau_2 - C_2(X_1, X_2) - G_2(X_1))\end{array}}{M_{11}(X_1)M_{22}(X_1) - M_{12}(X_1)M_{21}(X_1)},\\
\dot x_4 &= \frac{\begin{array}{c}M_{11}(X_1)(\tau_2 - C_2(X_1, X_2) - G_2(X_1))\\ -M_{21}(X_1)(\tau_1 - C_1(X_1, X_2) - G_1(X_1))\end{array}}{M_{11}(X_1)M_{22}(X_1) - M_{12}(X_1)M_{21}(X_1)},
\end{aligned} \tag{32}
$$

where M_{ij}, C_i, and G_i $(i, j = 1, 2)$ are substituted from Eqs. (27) and (29) for the normal case and static-balanced case, respectively. For the unbalanced case, unknown parameters are considered to be zero in Eq. (27).

Now, according to Eq. (8), by considering the performance index as minimum control effort defined as follows:

$$
J = \int_{t_0}^{t_f} \left(\tau_1^2 + \tau_2^2\right) dt \tag{33}
$$

and the costate vector as $\lambda = [\,x_5\ x_6\ x_7\ x_8\,]$, the Hamiltonian function becomes

$$
H = \tau_1^2 + \tau_2^2 + x_5\dot x_1 + x_6\dot x_2 + x_7\dot x_3 + x_8\dot x_4, \tag{34}
$$

where $\dot x_i$, $i = 1, \ldots, 4$ can be substituted from Eq. (32). Then, by substituting Eq. (32) into Eq. (34), and differentiating the Hamiltonian function with respect to the states, according to Eq. (9), the costate equations are obtained as follows:

$$
\begin{aligned}
\dot x_5 &= -\frac{\partial H}{\partial x_1} = -\frac{\partial}{\partial x_1}\\
&\times\left(\frac{(-C_1 - G_1)(x_7 M_{22} - x_8 M_{12}) + (-C_2 - G_2)(-x_7 M_{12} + x_8 M_{11})}{M_{11}M_{22} - M_{12}M_{21}}\right),\\
\dot x_6 &= -\frac{\partial H}{\partial x_2} = -\frac{\partial}{\partial x_2}\\
&\times\left(\frac{(-C_1 - G_1)(x_7 M_{22} - x_8 M_{12}) + (-C_2 - G_2)(-x_7 M_{12} + x_8 M_{11})}{M_{11}M_{22} - M_{12}M_{21}}\right),\\
\dot x_7 &= -\frac{\partial H}{\partial x_3} = -x_5 - \frac{\partial}{\partial x_3}\\
&\times\left(\frac{(-C_1)(x_7 M_{22} - x_8 M_{12}) + (-C_2)(-x_7 M_{12} + x_8 M_{11})}{M_{11}M_{22} - M_{12}M_{21}}\right),
\end{aligned}
$$

https://doi.org/10.1017/S026357471200063X Published online by Cambridge University Press

$$\dot{x}_8 = -\frac{\partial H}{\partial x_4} = -x_6 - \frac{\partial}{\partial x_4} \times \left(\frac{(-C_1)(x_7 M_{22} - x_8 M_{12}) + (-C_2)(-x_7 M_{12} + x_8 M_{11})}{M_{11}M_{22} - M_{12}M_{21}} \right). \quad (35)$$

After that, the control values can be obtained by solving the following equations:

$$\frac{\partial H}{\partial \tau_1} = 0, \quad \frac{\partial H}{\partial \tau_2} = 0. \quad (36)$$

So, by substituting the Hamiltonian function from Eq. (34) into Eq. (36), the optimal control laws become

$$\tau_1 = \frac{0.5}{M_{11}M_{22} - M_{12}M_{21}}(-X_7 M_{22} + X_8 M_{21}),$$
$$\tau_2 = \frac{0.5}{M_{11}M_{22} - M_{12}M_{21}}(X_7 M_{12} + X_8 M_{11}). \quad (37)$$

Finally, by substituting Eq. (37) into Eqs. (32) and (35), eight nonlinear ordinary differential equations will be obtained and this combined with the eight boundary conditions given in Eq. (30), lead to a two-point boundary value problem. This problem can be solved using the bvp4c command in MATLAB®.

4.4. Optimal spring balancing of two-link manipulator

Unlike the static-balanced case in which the unknown parameters are dependent on manipulator parameters as in Eq. (28), in the optimal-balanced case the values of unknowns are dependent on dynamic equations, performance index, and boundary conditions according to Eq. (9). Therefore, the unknown parameters and optimal trajectory are obtained simultaneously in such a way that the given performance index is minimized. In this case, the optimal control problem involving parameters, with its optimality conditions given in Eq. (9), must be considered. All of the dynamic equations, costate equations, and optimal control law are the same as unbalanced case obtained in the last section. For convenience, the optimization process selection of unknown parameters is divided into two steps. At the first step, k_1, k_2, and s_2 are considered to be known parameters and the optimal value of s_1, d, and s_3 are obtained, on the other hand the parameter vector in Eq. (2) is considered to be $\mathbf{b} = [s_1 \ d \ s_3]^T$. At the second step, the obtained values for s_1, d, and s_3 at the first step are rounded, and unknown parameters vector is considered to be $\mathbf{b} = [k_1 \ k_2]^T$. In the first step, by defining the three new state variables as x_9, x_{10}, and x_{11}, the optimality conditions associated with the parameters become

$$\dot{x}_9 = -\frac{\partial H}{\partial s_1}, \quad \dot{x}_{10} = -\frac{\partial H}{\partial d}, \quad \dot{x}_{11} = -\frac{\partial H}{\partial s_3}, \quad (38)$$

where according to Eq. (9), the associated boundary conditions become

$$x_{9,10,11}(0) = x_{9,10,11}(t_f) = 0. \quad (39)$$

Table I. Parameters of two-link manipulator[21].

Parameters	Values	Unit
Mass	$m_1 = m_2 = 1$	kg
Payload mass	$m_p = 2$	kg
Length of links	$L_1 = L_2 = 1$	m
Moment of inertia	$I_1 = I_2 = 1/12$	kg m^2
Length of adjacent links	$r_1 = r_2 = 0.5$	m
Length of parallelogram side	$s_2 = 0.5$	m

For the second step, by defining two new state variables as x_9 and x_{10}, one can write the optimality conditions as

$$\dot{x}_9 = -\frac{\partial H}{\partial k_1}, \quad \dot{x}_{10} = -\frac{\partial H}{\partial k_2}, \quad (40)$$

where according to Eq. (9), the associated boundary conditions become

$$x_{9,10}(0) = x_{9,10}(t_f) = 0. \quad (41)$$

At last, by substituting Eq. (37) into Eqs. (32), (35), and (38), 11 nonlinear ordinary differential equations in terms of the state $[x_1 \ x_2 \ x_3 \ x_4]$, costate $[x_5 \ x_6 \ x_7 \ x_8]$, new states $[x_9 \ x_{10} \ x_{11}]$, and unknown parameters (s_1, d, s_3) will be achieved. These 11 equations with 14 boundary conditions given in Eqs. (30) and (39) construct a two-point boundary value problem solving which all the states and unknown parameters can be obtained.

5. Simulation Results

5.1. Simulation results for a two-link manipulator

A two-link manipulator at vertical plan is considered as shown in Fig. 2. All required parameters of the robot manipulator are given in Table I. In this simulation, the five different methods are as follows: normal case means the unbalanced form of the manipulator, counter-weight-static balanced means static-balanced manipulator using mass, counter-weight-optimal means optimally balanced manipulator using mass,[21] zero-free-length spring-static means static-balanced with spring, and zero-free-length spring-optimal means optimally balanced manipulator using spring.

The initial position of the end-effector in the XZ plan at $t = 0$ is $p_0 = (1, 0)$ m and the final position at $t = 1$ s is $p_f = (0, 1.73)$ m. The initial and final velocities are zero. From the inverse kinematic equations, the boundary condition can be expressed as

$$\theta_1(0) = 60^\circ, \ \theta_2(0) = 120^\circ, \ \theta_1(t_f) = 120^\circ, \ \theta_2(t_f) = -60^\circ,$$
$$\dot{\theta}_1(0) = \dot{\theta}_2(0) = \dot{\theta}_1(t_f) = \dot{\theta}_2(t_f) = 0. \quad (42)$$

The results of simulations for the normal case, counter-weight-static-balanced case, and counter-weight-optimal-balanced case are the same as reported in ref. [21]. For spring-static-balanced case, at first, the values of parameters are obtained using Eq. (28). Then, the corresponding boundary value problem derived in Section 4.3 is solved to

https://doi.org/10.1017/S026357471200063X Published online by Cambridge University Press

Table II. Manipulator parameters for static and optimal-balanced cases.

Parameter	Static	Optimal
Ground joint of spring, d (m)	0.4905	1.3
Length of parallelogram sides, s_1, s_2, s_3 (m)	1.7, 0.5, 0.3	0.4, 0.5, 0.2
First spring stiffness, k_1 (N/m)	100	82.02
Second spring's stiffness, k_2 (N/m)	100	0

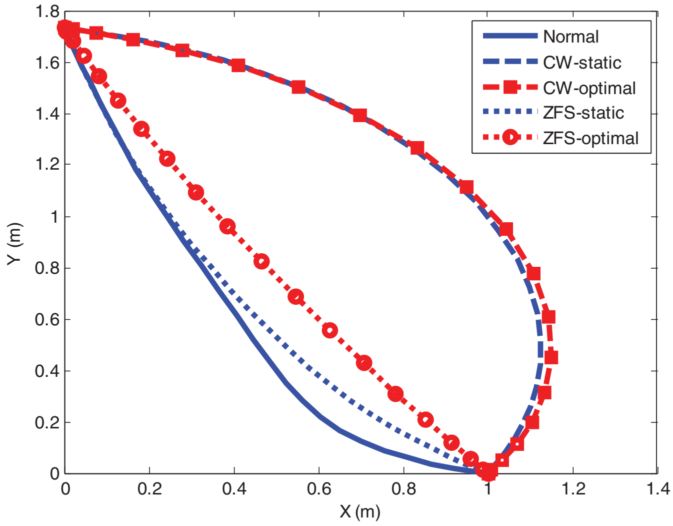

Fig. 3. (Colour online) Optimal trajectories for different cases.

obtain the states and controls. For spring-optimal-balanced case, the corresponding boundary value problem derived in Section 4.4 is solved to obtain states, controls, and unknown parameters. The manipulator parameters for static- and optimal-balanced cases are given in Table II. In the optimal case, since the second spring's stiffness k_2 is zero, the value of s_1 is unimportant and it probably eliminate the second spring in practice.

The obtained optimal trajectories between the initial and final points for the five cases are shown in Fig. 3. Figure 4 shows the obtained torque of motors. The angular position

Table III. Comparison of performance indexes.

Case	Pay-off (N m^2/s)	Reduction (times)	Amplification (times)
Normal	1090	1	1
Counter-weight-static-balanced	5770	–	5.29
Counter-weight-optimal-balanced	564	1.93	–
Spring-static-balanced	361	3.02	–
Spring-optimal-balanced	52	20.96	–

Table IV. Denavit–Hartenberg parameters for a PUMA-like robot[31].

Link	θ_i (rad)	α (rad)	a_i (m)	d_i (m)
1	q_1	$\pi/2$	0	0.4
2	q_2	0	0.5	0
3	q_3	0	0.5	0

and angular velocity of links are illustrated in Fig. 5. The second column of Table III shows the values of performance index defined in Eq. (33) for five considered cases. The third and forth columns represent the improvement relative to the normal case in state of amplification or reduction. As reported in ref. [21] and can be shown in Table III, the performance index for counter-weight-optimal-balanced case is less than the normal case and counter-weight-static-balanced case. While the performance index for spring-optimal-balanced case is less than all other cases. In the following figures, readers should notice optimal balancing decrement of input torques, and its effect on the trajectory (path and velocity profile of joints).

5.2. *Simulation results for PUMA-like robot*

A spatial three-jointed PUMA robot is considered as shown in Fig. 6. All of the manipulator parameters are the same used in ref. [31]. Denavit–Hartenberg (DH) parameters and links parameters are given in Tables IV and V, respectively. In this robot, the first spring is connected between the base

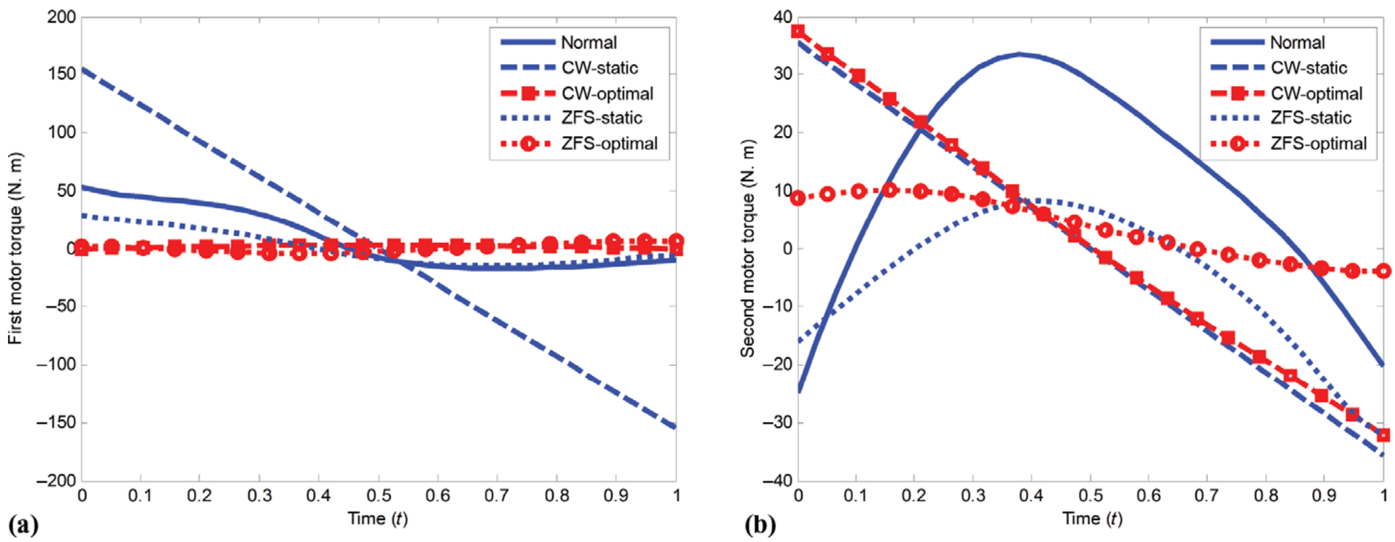

Fig. 4. (Colour online) Input torques of motors 1 and 2 (effect of optimal spring and mass balancing).

https://doi.org/10.1017/S026357471200063X Published online by Cambridge University Press

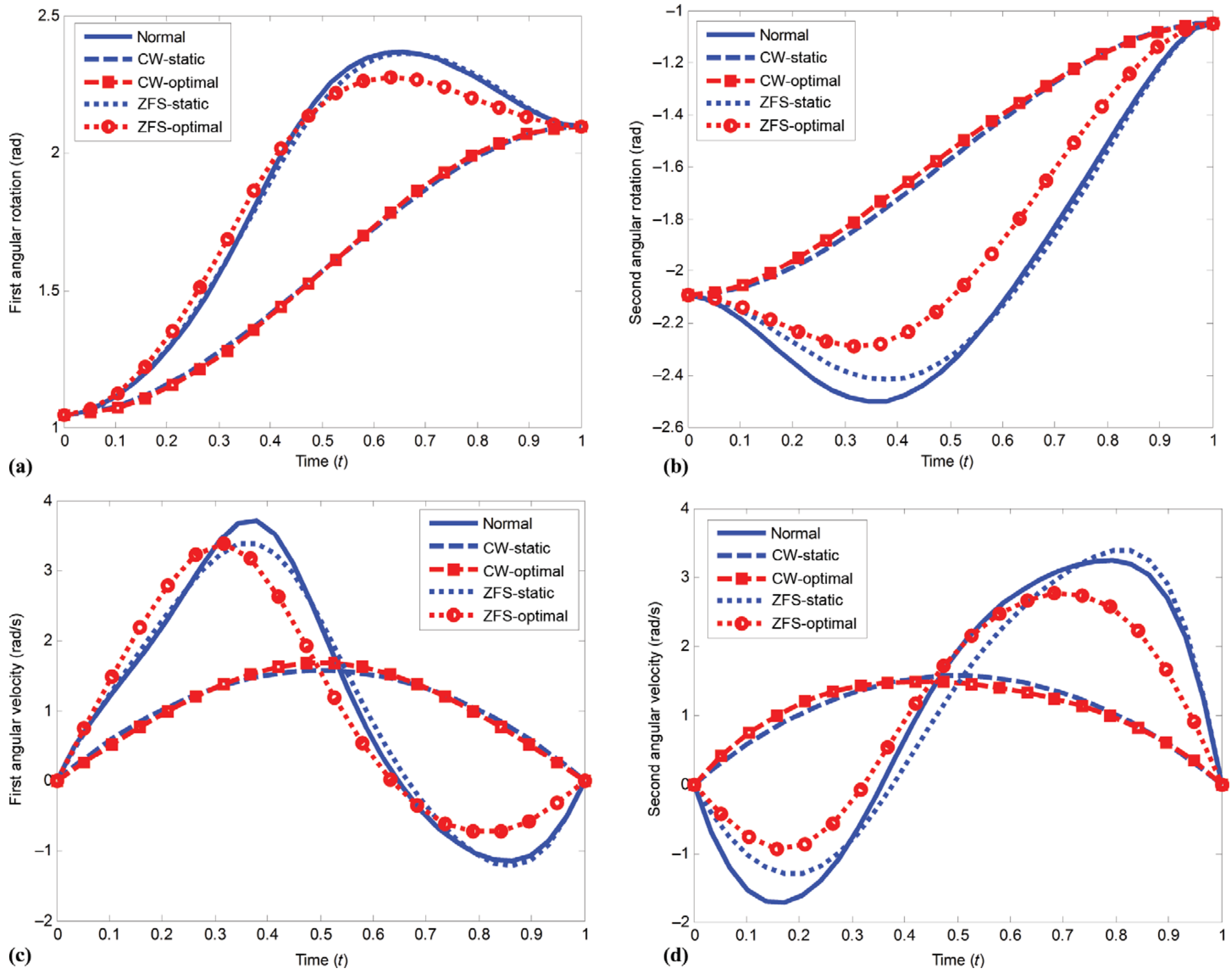

Fig. 5. (Colour online) Angular position and angular velocity of links (optimal balancing effect on velocity).

Table V. Link parameters and inertia properties[31].

Link	Length (m)	Mass (kg)	Moment of inertia (kg m^2)			Link center of mass (m)
1	0.4	12	0	0	0	0
			0	0.2	0	−0.2
			0	0	0	0
2	0.5	10	0	0	0	−0.25
			0	0.2	0	0
			0	0	0.2	0
3	0.5	5	0	0	0	−0.25
			0	0.1	0	0
			0	0	0.1	0

and the parallel fractional mechanism. The second spring is connected between the second and third link as shown in Fig. 6.

For obtaining the dynamic equations, the Lagrangian formulation is used. Total Lagrangian for this robot can be written as follows:

$$L_t = L + L_{sp}, \tag{43}$$

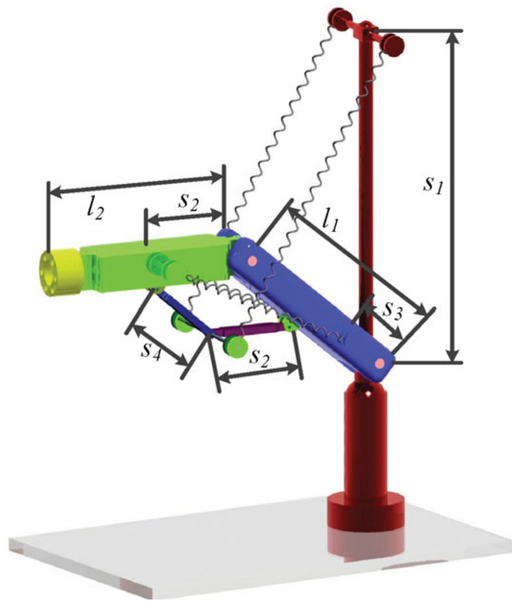

Fig. 6. (Colour online) PUMA-like robot with additional springs and parallelogram mechanism.

https://doi.org/10.1017/S026357471200063X Published online by Cambridge University Press

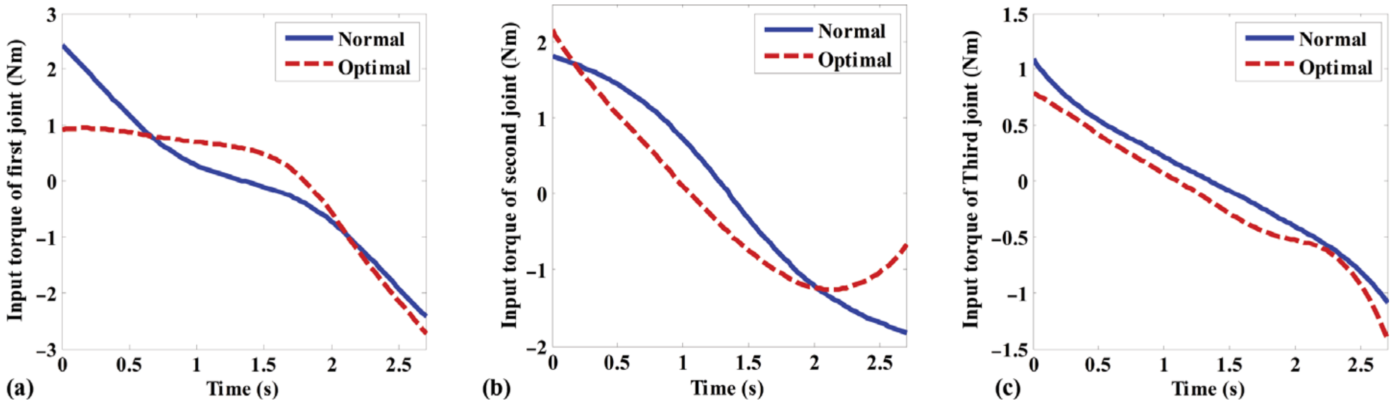

Fig. 7. (Colour online) Input torques of motors.

where L_t is the total Lagrangian, L is the Lagrangian of robot, and L_{sp} is additional Lagrangian due to springs. The additional Lagrangian can be stated as

$$L_{sp} = K - U = 0 - U = -\sum_{i=1}^{2} \frac{1}{2} k_i x_i^2, \tag{44}$$

where k_i is the stiffness of the springs and x_i is the deformed length of springs. After deriving the dynamic equations for this robot, using Eq. (9), the optimality condition can be obtained in the same way as presented for two-link manipulator. The boundary conditions are considered as follows:

$$\begin{aligned}
&\theta_1(0) = 17^\circ,\ \theta_1(t_f) = 29.22^\circ,\ \theta_2(0) = 29^\circ,\ \theta_2(t_f) = -24^\circ \\
&\theta_3(0) = 11.45^\circ,\ \theta_3(t_f) = 32.23^\circ, \\
&\dot{\theta}_1(0) = \dot{\theta}_1(t_f) = \dot{\theta}_2(0) = \dot{\theta}_2(t_f) = \dot{\theta}_3(0) = \dot{\theta}_3(t_f) = 0.
\end{aligned} \tag{45}$$

For this robot, simulations are performed for two cases: normal case and optimal-balanced case. For the normal case, all the parameters dealing with the springs are considered to be zero. For the optimal-balanced case, at first, the stiffness of springs, k_1 and k_2 are considered to be known and the values of distance between joints and spring connection points are

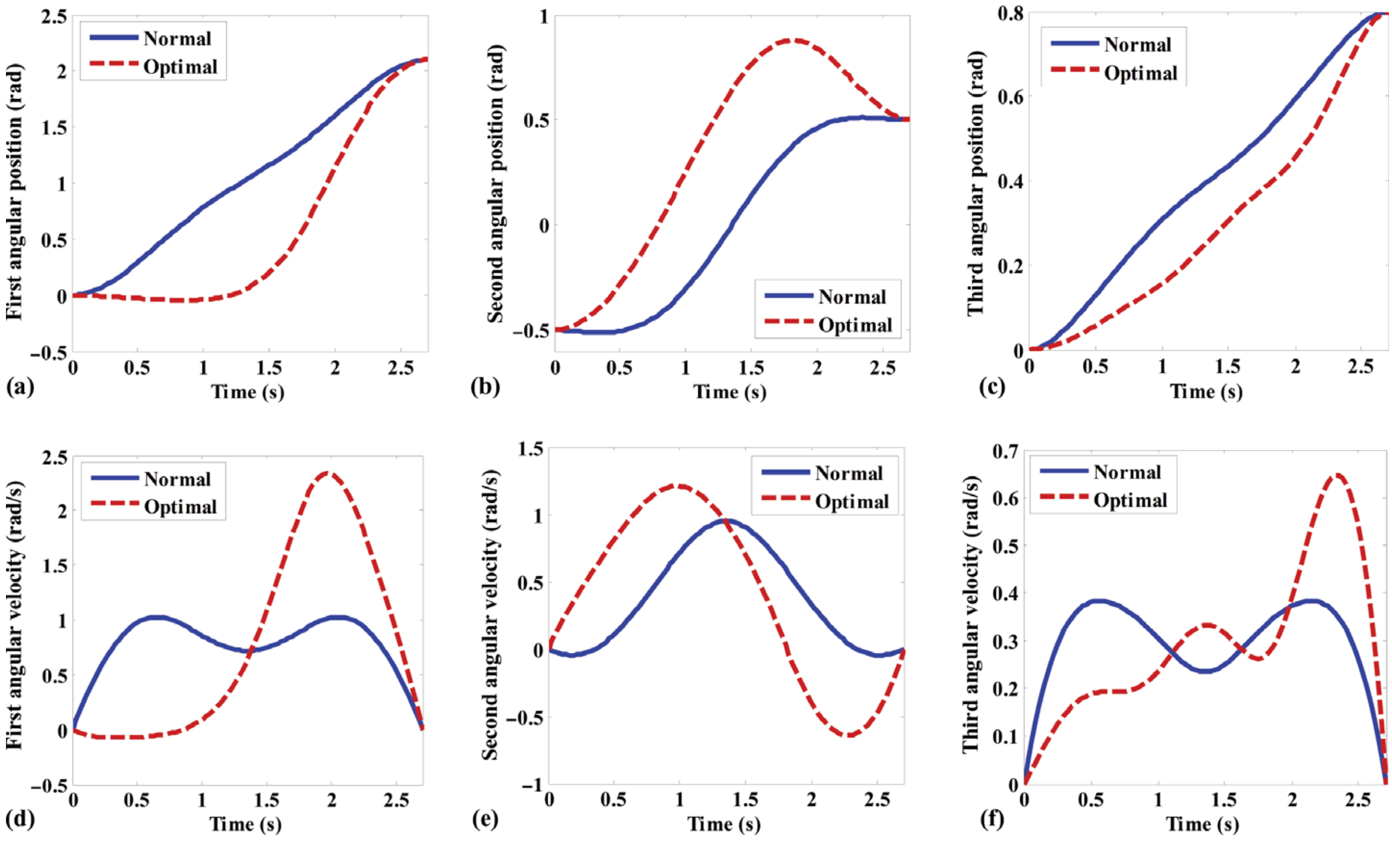

Fig. 8. (Colour online) Angular position and velocity of links.

https://doi.org/10.1017/S026357471200063X Published online by Cambridge University Press

Table VI. Optimal values of parameters for PUMA-like robot.

Parameter	Value (Unit)
First spring stiffness, k_1	1.23 N/m
Second spring stiffness, k_2	0 N/m
Ground joint of spring, s_1	1.3 m
Length of second spring application point, s_2	0.186 m

Fig. 9. (Colour online) Optimal trajectories for normal cases.

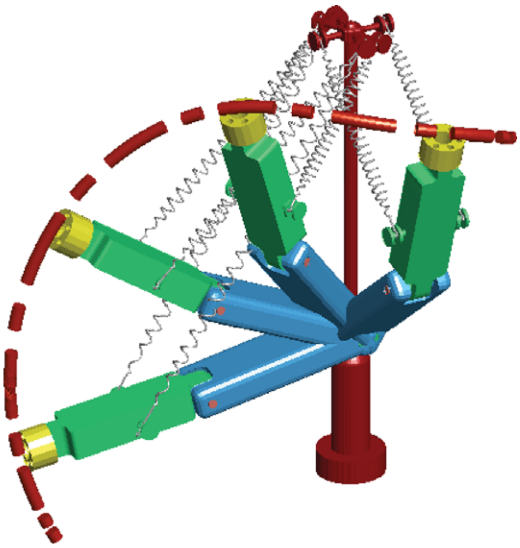

Fig. 10. (Colour online) Optimal trajectories for optimal cases.

determined. In the next step, by considering the rounded position values, the optimal value of stiffness is obtained. Optimal values of parameters are listed in Table VI.

The obtained optimal controls are shown in Fig. 7. The angular position and angular velocity of links are illustrated in Fig. 8. The optimal trajectories for normal and optimal-balanced cases are given in Figs. 9 and 10, respectively. Performance index for normal case is found to be 8.81 N m^2 and for spring-optimal-balanced case is found to be 7.08 N m^2 and that that this value is 20% less than the normal case.

6. Conclusion

In this paper, the formulation of optimal spring balancing for robotic system in point-to-point motion, based on the indirect solution of optimal control problem, is presented. After deriving the dynamic equation using the Lagrange formulation, optimality conditions are derived using the Pontryagin's minimum principle as well as an additional condition associated with the constant parameters. The obtained equations lead to a standard form of a two-point boundary value problem, which should be solved.

The efficiency of the proposed method is investigated through computer simulations for a two-link manipulator. The obtained results show that although the performance index for the static-balanced manipulator has been reduced 1.93 times (66.8%) with respect to unbalanced case, by applying the proposed method this reduction reaches to 21 times (95%). It is also shown that the performance index for spring balancing is very less than the performance index for the counter-weight balancing reported in ref. [21]. This result is expected, because in counter-weight balancing the moment of inertia increases due to the added masses. Finally, simulation is performed for a PUMA-like robot and the capability of the method to solve the complicated problem is shown. For this case study, performance index for optimal-balanced case is obtained which is 20% less than the unbalanced case. In the future work, by building the prototype, the efficiency of spring balancing method versus the other balancing approach will be shown experimentally.

References

1. J. Park, J. Haan and F. C. Park, "Convex optimization algorithms for active balancing of humanoid robots," *IEEE Trans. Robot.* **23**(4), 817–822 (2007).
2. W. K. Chung and H. S. Cho, "On the dynamic characteristics of a balanced PUMA-760 Robot," *IEEE Trans. Ind. Electron.* **35**(2), 222–230 (1988).
3. T. A. H. Coelho, L. Yong and V. F. A. Alves, "Decoupling of dynamic equations by means of adaptive balancing of 2-dof open-loop mechanism," *Mech. Mach. Theory* **39**, 871–881 (2004).
4. J. L. Pons, R. Ceres and A. R. Jimenez, "Quasi exact linear spring counter gravity system for robotic manipulators," *Mech. Mach. Theory* **33**(1–2), 59–70 (1998).
5. V. Arakelian and S. Ghazaryan, "Improvement of balancing accuracy of robotic systems: Application to leg orthosis for rehabilitation devices," *Mech. Mach. Theory* **43**, 565–575 (2008).
6. D. A. Streit and B. J. Gilmore, "Perfect spring equilibrators for rotatable bodies," *ASME Trans. J. Mech. Transm. Autom. Des.* 111(4), 451–458 (1989).
7. N. Vrijlandt and J. L. Herder, "Seating unit for supporting a body or part of a body," *Patent*, NL1018178, Dec 3, (2002).
8. S. K. Agrawal and A. Fattah, "Gravity-balancing of spatial robotic manipulators," *Mech. Mach. Theory* **39**, 1331–1344 (2004).
9. A. T. Kondrin, L. N. Petrov and N. F. Polishchuk, "Pivoted arm balancing mechanism," *Patent*, SU1596154, Sep 30, (1990).
10. Y. F. Gvozdev, "Manipulator," *Patent*, SU1777993, Nov 30, (1992).
11. N. Ulrich and V. Kumar, "Passive Mechanical Gravity Compensation for Robot Manipulators," *Proceedings of the International Conference on Robotics and Automation*, Sacramento, California (Apr 1991) vol. 2, pp. 1536–1541.

https://doi.org/10.1017/S026357471200063X Published online by Cambridge University Press

12. M. Kolarski, M. Vukobratovic and B. Borovac, "Dynamic analysis of balanced robot mechanisms," *Mech. Mach. Theory* **29**(3), 427–454 (1994).
13. S. K. Banala, S. K. Agrawal, A. Fattah, V. Krishnamoorthy, W. L. Hsu, J. Scholz and K. Rudolph, "Gravity-balancing leg orthosis and its performance evaluation," *IEEE Trans. Robot.* **22**(6), 1228–1239 (2006).
14. A. Fattah and S. K. Agrawal, "Gravity-Balancing of Classes of Industrial Robots," *Proceedings of the IEEE International Conference on Robotics and Automation*, Orlando, Florida (May 2006) pp. 2872–2877.
15. R. Barents, M. Schenk, W. D. van Dorsser, B. M. Wisse and J. L. Herder, "Spring-To-Spring Balancing As Energy-free Adjustment Method In Gravity Equilibrator," *33rd Mechanism and Robotics Conference, Parts A and B*, San Diego, California (Aug–Sep 2009) vol. 7, pp. 689–700.
16. I. Simionescu and L. Ciupitu, "The static balancing of the industrial arms. Part I: Discrete balancing," *Mech. Mach. Theory* **35**, 1287–1298 (2000).
17. I. S. Kochev, "General theory of complete shaking moment balancing of planar linkages: A critical review," *Mech. Mach. Theory* **35**, 1501–1514 (2000).
18. M. Moradi, A. Nikoobin and S. Azadi, "Adaptive decoupling for open chain planar robots," *Iranica Sci.* **17**(5B), 376–386 (2010).
19. R. Saravanan, S. Ramabalan and P. D. Babu, "Optimum static balancing of an industrial robot mechanism," *Eng. Appl. Artif. Intell.* **21**(6), 824–834 (2008).
20. T. Ravichandran, D. Wang and G. Heppler, "Simultaneous plant-controller design optimization of a two-link planar manipulator," *Mechatronics* **16**, 233–242 (2006).
21. A. Nikoobin and M. Moradi, "Optimal balancing of robot manipulators in point-to-point motions," *Robotica* **29**(2), 233–244 (2011).
22. C. M. Gosselin, F. Vollmer, G. Cote and Y. Wu, "Synthesis and design of reactionless three-degree-of-freedom parallel mechanisms," *IEEE Trans. Robot. Autom.* **20**(2), 191–199 (2004).
23. L. Cheng, Y. Lin, Z. G. Hou, M. Tan, J. Huang and W. J. Zhang, "Integrated design of machine body and control algorithm for improving the robustness of a closed-chain five-bar machine," *IEEE/ASME Trans. Mechatronics* **17**(3), 587–591 (2012).
24. L. Cheng, Z. G. Hou, M. Tan and W. J. Zhang, "Tracking control of a closed-chain five-bar robot with two degrees of freedom by integration of approximation-based approach and mechanical design," *IEEE Trans. Syst. Man Cybern. Part B: Cybernetics* **42**(5), 1470–1479 (2012).
25. T. Chettibi, H. E. Lehtihet, M. Haddad and S. Hanchi, "Minimum cost trajectory planning for industrial robots," *Eur. J. Mech. Part A :Solids* **23**(4), 703–715 (2004).
26. D. G. Hull, "Conversion of optimal control problems into parameter optimization problems," *J. Guid. Control Dyn.* **20**(1), 57–60 (1997).
27. R. Callies and P. Rentrop, "Optimal control of rigid-link manipulators by indirect methods," *GAMM-Mitteilungen* **31**(1), 27–58 (2008).
28. J. T. Betts, "Survey of numerical methods for trajectory optimization," *J. Guid. Control Dyn.* **21**(2), 193–207 (1998).
29. M. H. Korayem and A. Nikoobin, "Formulation and numerical solution of robot manipulators in point-to-point motion with maximum load carrying capacity," *Scientia Iranica J.* **16**(1), 101–109 (2009).
30. D. G. Hull, "Sufficiency for optimal control problems involving parameters," *J. Optim. Theory Appl.* **97**(3), 579–590 (1998).
31. M. H. Korayem, A. Nikoobin and V. Azimirad, "Maximum load carrying capacity of mobile manipulators: Optimal control approach," *Robotica* **27**, 147–159 (2009).
32. D. E. Kirk, *Optimal Control Theory: An Introduction* (Prentice-Hall, Englewood Cliffs, NJ, 1970).