Robotica (2011) volume 29, pp. 233–244. © Cambridge University Press 2010
doi:10.1017/S0263574710000093

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

A. Nikoobin†* and M. Moradi‡

† *Department of Mechanical Engineering, Semnan University, Semnan, Iran*
‡*Department of Computer & Electrical Engineering, Semnan University, Semnan, Iran*

(Received in Final Form: February 19, 2010. First published online: March 24, 2010)

SUMMARY

In this paper, a new balancing approach called "*optimal balancing*" is presented for open-chain robot manipulators based on open-loop optimal control. In fact, an optimal trajectory planning problem is outlined in which states, controls and the values of counterweights must be determined simultaneously in order to minimize the given performance index for a predefined point-to-point task. Optimal balancing method can be propounded beside the other methods such as unbalancing, static balancing and adaptive balancing, with this superiority that the objective criterion value obtained of proposed method is very lower than the objective criterion value obtained of other methods. For this purpose, the optimal control problem is extended to the case where the performance index, the differential constraints and the prescribed final conditions contain parameters. Using the fundamental theorem of calculus of variations, the necessary conditions for optimality are derived which lead to the optimality conditions associated with the Pontryagin's minimum principle and an additional condition associated with the constant parameters. By developing the obtained optimality conditions for the two-link manipulator, a two-point boundary value problem is achieved which can be solved with bvp4c command in MATLAB®. The obtained results show that optimal balancing in comparison with the previous methods can reduce the performance index significantly. This method can be easily applied to the more complicated manipulator such as a three degrees of freedom articulated manipulator.

KEYWORDS: Robot manipulators; Trajectory planning; Optimal balancing; Pontryagin minimum principle; Point-to-point motion; Counterweights.

1. Introduction

For years, industrial robots have been used in place of human workers in several situations in cause of their magnificent rapidity, wonderful exactness, and unbelievable cost-effectiveness in boring and tiresome tasks. But these powerful machines are strongly dependent on some preliminary actions, such as calibration or motion planning, to achieve specified tasks. Frequently, many different ways are possible to perform the same task. This freedom of choice can be lead to optimize a given performance criterion called optimal trajectory planning. Among the tasks which robot manipulators are devoted to, a distinction can be made in regards to the desired motion nature. Depending on the robot task, it might be necessary to specify the end-effector trajectory in the work space. Gluing, arc welding and laser cutting operations are the tasks in which planner defines optimal tracking modalities of the predefined path. On the other hand, in point to point motions the end effector is free to move between two extremal positions. Pick-and-place operations and point to point welding are the tasks in which planner tries to define the optimal trajectory and the corresponding controls.[1]

Generally, 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. This generic optimal control problem is so complicated which can only be solved by computer. Therefore, both "Discretization" due to main characteristic of computer, and "Optimization" due to optimal control problem are necessary to obtain an optimal path, but their execution order may be changed. Here, two strategies can be distinguished for path optimization: direct methods and indirect methods. These techniques are used in enormous articles and they have own benefits and drawbacks.[2–4]

In industrial application, optimal control problems for manipulators are usually solved by so-called direct methods characterized by "first discretize, then optimize" strategy. This method transfers the motion planning problem from an optimal control problem into a parameter optimization one. It have done by means of discretization of dynamic variables including states and controls.[1,2,5] These steps are considerable in direct method. In the first step, the control variables—and often the state variables too—are discretized and the optimal control problem is copy out into a large, constrained and often sparse nonlinear programming problem. In the second step, the resulting nonlinear programming problem is treated by standard algorithms like sequential quadratic programming methods,[6] interior point methods,[7] evolutionary,[8,9] or classical stochastic techniques[1] to obtain optimal values of the parameters. In some of the previous works dealing with path planning of robot manipulators the direct methods are employed, and often the Spline or polynomial functions are used as the motion profiles.[1,5,10] A weak point of this method is that it leads to an approximate optimal solution. Furthermore, these techniques suffer from numerical explosion when

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

https://doi.org/10.1017/S0263574710000093 Published online by Cambridge University Press

treating high-dimension problems. They are exhaustively time consuming and quite inefficient due to the large number of parameters involved, especially for systems with a large number of degrees of freedom (dof).[11]

On the other hand, indirect methods are based on Pontryagin minimum principle (PMP) which solves the optimal control problem exactly.[12] The strategy of indirect method can be characterized by "first optimize, then discretize." Based on PMP, the problem of optimal control is first transformed into multipoint boundary value problem. In the following step, this problem is discretized to achieve the numerical solution using the method such as shooting, relaxation or collocation techniques. Indirect method was first used to solve the minimum time motion problems along the specified paths. Then, it was extended to handle free motions as well.[13] PMP is also used to treat directly the optimal dynamic motion planning problem.[14] This method is widely used as a powerful and efficient tool in analyzing the nonlinear systems and path planning of different types of systems.[15–18] Fotouhi *et al.* used indirect method to obtain the time optimal trajectory of a two-link manipulator,[15] Furuno *et al.* used this technique for trajectory planning of mobile manipulators,[19] and Nikoobin *et al.* applied this method for trajectory planning of flexible joint,[20] flexible link,[21] and redundant manipulators.[22] In ref. [4] a survey on the treatment of optimal control problems of rigid-link manipulators by indirect methods is presented.

Besides the trajectory planning in which controls and states are obtained to minimize the desired performance objective, one efficient way to increase the robot performance is balancing.[23] Balancing introduces some 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. Balancing can be categorized into two types, namely, active and passive balancing. In active balancing an external electric, pneumatic or hydraulic force is applied to the system,[24] while in passive balancing compensation inertias[25,26] or springs[28,29] are used. Since additional actuators in passive balancing are not required, it is more efficient and simpler than active one. All the works based on passive balancing can be classified into either static or dynamic balancing. Static balancing means that the weight of the links does not produce any force at actuators for any configuration of the manipulator. When a robot is statically balanced, its potential energy is constant for all possible configurations.[25,29,30] In dynamic balancing, some modifications to the original kinematic chain of unbalanced mechanisms are applied in such a way to obtain static balancing and complete decoupling of dynamic equations.[26] Thus Coriolis, centripetal, gravitational, and cross-inertia terms are eliminated.

In this paper, a new method based on indirect solution of optimal control problem is presented to determine the optimal trajectory of robot manipulator in point to point motion with considering the counterweights as optimization parameters. In the proposed method called optimal balancing, states, controls, and the values of counterweights are obtained simultaneously, in such a way that the given performance objective is minimized. The major contribution of this article can be considered in two fields: path planning and balancing. None of the previous work dealing with indirect solution of path planning problem[4,14–22] has considered constant parameters in optimization procedure. On the other hand, a new balancing method is introduced which in comparison with previous methods such as static balancing[25,29,30] or dynamic balancing[26] has significant superiorities to optimize the given performance index.

The paper is organized as follow: optimality conditions for the optimal control problem including the parameter are derived in general form in Section 2. In Section 3 traditional balancing methods including static and adaptive balancing methods are described. Then using the general formulation given in Section 2 the formulation of optimal balancing is derived for n-dof manipulators in Section 4. In order to clarify the proposed method, problem formulation is derived for a two-link manipulator in details in Section 5. For this purpose, dynamic equation of two-link manipulator for unbalanced, static balanced and adaptive balanced ceases are derived. Then by considering the performance index and defining the costate variables, optimality conditions are obtained for different balanced cases. Finally in order to verify the method, simulation results for a two-link manipulator and a three-link manipulator are presented in Section 6.

2. Formulation of Optimal Control Including Parameters

Optimal trajectory planning of robot manipulators is an optimization problem which can be stated in the form of the optimal control problem.[1] Since, the point to point motion is considered in this paper, the initial state, final state and time interval are predefined. So, according to ref. [31], the optimal control problem for a dynamic system involving parameters can be stated as follows: Find the continuous admissible control history $u : [t_0, t_f] \rightarrow \Omega \subseteq R^m$ and parameter b generating the corresponding state trajectory $x : [t_0, t_f] \rightarrow R^n$ which minimize the cost function

$$J = \phi(x_f, b) + \int_{t_0}^{t_f} L(x, u, b)\mathrm{d}t, \tag{1}$$

subject to the system dynamics

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}, b), \tag{2}$$

the given initial condition

$$x(t_0) = x_0, \tag{3}$$

and the prescribed final conditions

$$\mathbf{x}(t_f) = x_f. \tag{4}$$

Here, $x \in R^n$ is the state vector, $u \in R^m$ is the control vector, $b \in R^r$ is the parameter vector, Ω is an acceptable region in R^m, t_0, and t_f are initial and final time, $\mathbf{x}_0$ and $\mathbf{x}_f$ are predefined initial and final state, respectively. ϕ and L are scalar continuously differentiable functions in which ϕ is the

https://doi.org/10.1017/S0263574710000093 Published online by Cambridge University Press

final state penalty term and L is the integrand of the cost function.

Adjoining the constraints to the performance index by Lagrange multipliers λ leads to the augmented performance index

$$J' = \phi(x_f, b) + \int_{t_0}^{t_f} \left[H(x, u, \lambda) - \lambda^T \dot{x}\right] \mathrm{d}t, \tag{5}$$

where the variational Hamiltonian is defined as

$$H = L + \lambda^T f. \tag{6}$$

According to the fundamental theorem of calculus of variations,[12] if x^* is an extremal of functional J', the variation of J' must vanish on x^*, that is $\delta J'(x^*, \delta x) = 0$ for all δx. Then, taking the variation of J' and performing the integration by parts gives

$$\begin{aligned}\delta J' = &\left(\phi_{xf} - \lambda_f^T\right)\delta x_f + \left(\phi_b + \int_{t_0}^{t_f} H_b \mathrm{d}t\right) + \lambda_0 \delta x_0 \\ &+ \int_{t_0}^{t_f} \left[\left(H_x + \dot{\lambda}^T\right)\delta x + H_u \delta u + \left(f^T - \dot{x}^T\right)\delta\lambda\right] \mathrm{d}t \\ &= 0.\end{aligned} \tag{7}$$

Because initial and final states are predefined, so δx_0 and δx_f are zero. Also from Eq. (2) we have $f^T - \dot{x}^T = 0$. Since the parameter b and control u are independent, consider the class of admissible comparison paths where there is no change in the parameter ($\delta b = 0$). The problem reduces to the standard fixed final time problem for which

$$\dot{\lambda} = -H_x^T, \quad H_u = 0, \tag{8}$$

then, for admissible comparison paths where $\delta b \neq 0$, the first variation yields the condition

$$\phi_b + \int_{t_0}^{t_f} H_b \mathrm{d}t = 0. \tag{9}$$

Now by introducing the function $\mu(t)$ defined by

$$\dot{\mu} = -H_b^T, \quad \mu_0 = 0. \tag{10}$$

Equation (9) becomes

$$\mu_f = \phi_b^T. \tag{11}$$

Consequently the complete set of equations defining the extremal path is given by

$$\begin{aligned} &\dot{x} = f(x, u, b), \quad \dot{\lambda} = -H_x^T, \\ &H_u = 0, \quad \dot{\mu} = -H_b^T, \\ &x(t_0) = x_0, \quad \mu(0) = 0, \\ &x(t_f) = x_f, \quad \mu(t_f) = \phi_b. \end{aligned} \tag{12}$$

Also with the realization that the parameters behave like states, the Weiestrass condition,

$$H(x, u^*, \lambda, b) \geq H(x, u, \lambda, b) \quad \text{for all } u^*, \tag{13}$$

must be satisfied as must the Legendre–Clebsch condition,

$$H_{uu}(x, u, \lambda, b) \geq 0. \tag{14}$$

As it can be seen, eliminating the parameter b in Eqs. (12)–(14) leads to the well known optimality conditions obtained from Pontryagin's minimum principle.[12]

3. Counterweights Balancing

Besides the different methods of balancing, counterweights balancing is a simple and general method of passive balancing. The traditional counterweights balancing can be categorized into two basic types: static balancing and adaptive balancing.

3.1. Static balancing

Static balancing is defined as that the weight of the links does not produce any force at actuators for any configuration of the manipulator, in the other word releasing the gravitational effects in mechanical systems.[25] Static balancing is typically achieved by pioneering additional mechanical elements to the system, such as counterweights or elastic elements (springs). The use of counterweights is more perceptive and has been applied to the design of planar linkages for a long time. This method has advantages along with disadvantages that serve to limit its usefulness. For example, in space robots this is undesirable to provide an extra mass on robot where minimum weight is an important criterion. Also, adding the balancing masses increases the moment of inertia of the manipulator. In contrast, the required torque to remain the manipulator at rest decreases considerably.

The dynamical model of a robot manipulator with n-dof is described in Lagrangian formulation as:

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

where $u \in R^n$ is the vector of input torques, q, $\dot{q}$, and $\ddot{q}$ are vectors of joint angular positions, velocities and accelerations, respectively. $M(q) \in R^{n\times n}$ is inertia matrix, $C(q, \dot{q}) \in R^n$ represents the centripetal and Coriolis forces and $G(q) \in R^n$ describes the gravity effects. Because of the main property of static balancing that is eliminating the gravitational effect of system, one can write

$$\mathbf{G}(\mathbf{q}) = 0. \tag{16}$$

By this way, equation of motion described in Eq. (15) reduce to

$$\tau = \hat{\mathbf{M}}\ddot{\mathbf{q}} + \hat{\mathbf{C}}(\mathbf{q}, \dot{\mathbf{q}}), \tag{17}$$

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

3.2. Adaptive balancing

The main idea of adaptive balancing is to achieve the complete decoupling of dynamic equations.[26] Thus

https://doi.org/10.1017/S0263574710000093 Published online by Cambridge University Press

Coriolis, centripetal, gravitational, and cross-inertia terms are eliminated and the Eq. (15) decrease to

$$\tau = \hat{\mathbf{M}}\ddot{\mathbf{q}}, \tag{18}$$

where $\hat{\mathbf{M}}$ is a diagonal matrix represent the inertia terms. In ref. [26], an adaptive balancing method is proposed for a 2-dof open-chain manipulator in which whole nonlinear effects are removed by adding two rotary masses. Therefore dynamic equations are simplified more than static balancing and each actuator can be controlled independently, which really simplifies the control of the system. Also according to the results given in ref. [26], for a given path of the end effector, the adaptively balanced mechanism presented the lowest driving torques among other kinematically identical mechanisms but unbalanced or statically balanced. But, applying this method for a manipulator with more than 2-dof is very complicated.

4. Formulation of optimal balancing

In optimal balancing approach proposed in this paper, the values of counterweights, trajectory of manipulator and applied torque (force) in each joint must be obtained simultaneously in such a way that a given performance index is minimized. For this purpose counterweights are supposed as unknown variables, and then dynamic equations for robot manipulator are derived. The necessary number of counterweight for balancing the n dof robot manipulator is assumed to be r, argued before by others, e.g. Kamenski.[27] By considering the counterweights vector denoted by m_{ci}, Eq. (15) becomes

$$M(q, m_c)\ddot{q} + C(q, \dot{q}, m_c) + G(q, m_c) = u. \tag{19}$$

Now, optimal balancing problem can be solved using the formulation presented in Section 2.2. So, by defining the continuous state vector as

$$x = [\, x_1 \quad x_2 \,]^T = [\, q \quad \dot{q} \,]^T, \tag{20}$$

dynamic equation (15) can be rewritten in state space form as

$$f = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ M^{-1}(x_1, m_c)[u - C(x_1, x_2, m_c) - G(x_1, m_c)] \end{bmatrix}. \tag{21}$$

where f is continuous in the variables x, u and t, and is continuously differentiable with respect to x. Then by considering the performance index (1), Hamiltonian can be written using Eq. (6) as follow

$$H_{mc} = L(x, u, m_c) + \psi^T(t) f(x, u, m_c), \tag{22}$$

where $\psi = [\, \psi_1 \; \psi_2 \,]$ denotes costate vector. Substituting for $f(x, u, m_c)$ from Eq. (21) into Eq. (22) gives

$$H_{mc} = L(x, u, m_c) + \psi_1^T(t)x_2 + \psi_2^T(t)[M^{-1}(x_1, m_c) \times [u - C(x_1, x_2, m_c) - G(x_1, m_c)]]. \tag{23}$$

Now, by substituting the obtained Hamiltonian function into Eq. (12) the optimality conditions becomes

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= M^{-1}(x_1, m_c)[u - C(x_1, x_2, m_c) - G(x_1, m_c), \end{aligned} \tag{24}$$

$$\dot{\psi}_1 = -\left[\frac{\partial L}{\partial x_1} + \psi_2^T(t) \times \frac{\partial [M^{-1}(x_1, m_c)[u - C(x_1, x_2, m_c) - G(x_1, m_c)]]}{\partial x_1}\right], \tag{25}$$

$$\dot{\psi}_2 = -\left[\frac{\partial L}{\partial x_2} + \psi_1^T(t) + \psi_2^T(t)\frac{\partial [-C(x_1, x_2, m_c)]}{\partial x_2}\right],$$

$$\dot{\mu} = -\left[\frac{\partial L}{\partial m_c} + \psi_2^T(t) \times \frac{\partial [M^{-1}(x_1, m_c)[u - C(x_1, x_2, m_c) - G(x_1, m_c)]]}{\partial m_c}\right], \tag{26}$$

$$\frac{\partial L}{\partial u} + \psi_2^T(t)M^{-1}(x_1, m_c) = 0, \tag{27}$$

$$\begin{aligned} x(t_0) &= x_0, \; \mu(0) = 0, \\ x(t_f) &= x_f, \; \mu(t_f) = 0. \end{aligned} \tag{28}$$

where for a n-dof robot manipulator, Eq. (24) represents $2n$ equations dealing with states, Eq. (25) represents $2n$ equations dealing with costates, Eq. (26) represents r equations dealing with and r auxiliary states μ, Eq. (27) represents n equations dealing with controls and Eq. (28) represent $4n + 2r$ boundary conditions. By substituting the control value u obtained of Eq. (27) into Eqs. (24), (25), and (26), a set of $4n + r$ ordinary differential equations is established which beside the $4n + 2r$ boundary value conditions given in Eq. (28) forms a two point boundary value problem. Finally the derived TPBVP is solved to obtain $2n$ state, $2n$ costate, r counterweight value, and r auxiliary state μ.

5. Balancing for Two-Link Manipulator

Two-link planar manipulator is an illustrative and important robot used in many applications and simulation studies. In this section, after deriving the dynamic equations, the above mentioned optimality conditions are derived for 2-dof planar robot at point to point motion in details. Here four different

https://doi.org/10.1017/S0263574710000093 Published online by Cambridge University Press

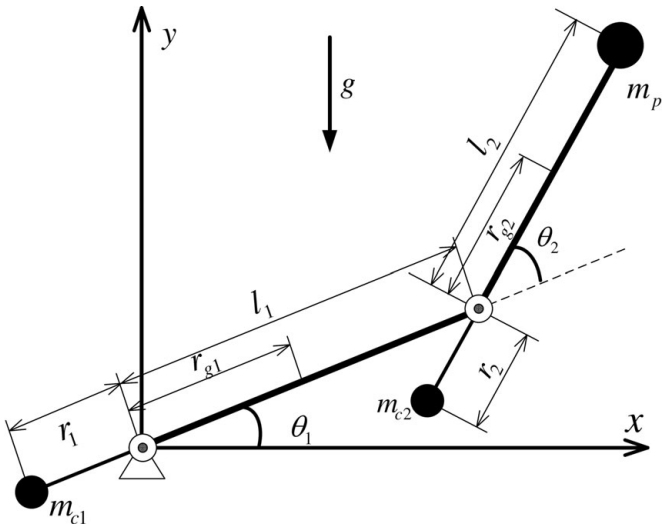


Fig. 1. Schematic diagram of two-link manipulator with counterweights.

conditions are considered: unbalanced, statically balanced, adaptively balanced and optimally balanced manipulator.

5.1. Dynamic equations of static balanced two-link manipulator

In Fig. 1 the two-link manipulator including the counterweights is shown. As it can be seen, two compensation masses m_{c1} and m_{c2} are attached to links 1 and 2, respectively.

The dynamic equations for such general two-link manipulator with supposing of arbitrary value of counterweights can be described as follow:

$$\begin{bmatrix} M_{11} & M_{12} \\ M_{12} & M_{22} \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} + \begin{bmatrix} G_1 \\ G_2 \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix}, \quad (29)$$

where

$$\begin{aligned}
M_{11} &= m_1 r_{g1}^2 + m_2\left(l_1^2 + 2l_1 r_{g2}\cos\theta_2 + r_{g2}^2\right) + m_p\left(l_1^2 + 2l_1\right. \\
&\quad \left. \times l_2\cos\theta_2 + l_2^2\right) + m_{c1} r_1^2 + m_{c2}\left(l_1^2 - 2l_1 l_2\cos\theta_2\right. \\
&\quad \left. + r_2^2\right) + I_1 + I_2, \\
M_{12} &= m_2\left(l_1 r_{g2}\cos\theta_2 + r_{g2}^2\right) + m_p\left(l_1 l_2\cos\theta_2 + l_2^2\right) + m_{c2} \\
&\quad \times\left(-l_1 r_2\cos\theta_2 + r_2^2\right) + I_2, \\
M_{22} &= m_2 r_{g2}^2 + m_p l_2^2 + m_{c2} r_2^2 + I_2, \\
C_1 &= l_1\sin\theta_2\left(-m_2 r_{g2} - m_p l_2 + r_2 m_{c2}\right)\left(2\dot{\theta}_1\dot{\theta}_2 + \dot{\theta}_2^2\right), \\
C_2 &= l_1\sin\theta_2\left(m_{c2} r_2 - m_2 r_{g2} - m_p l_2\right)\dot{\theta}_1^2, \\
G_1 &= \left(m_1 r_{g1} + m_2 l_1 + m_p l_1 - m_{c1} r_1 + m_{c2} l_1\right) g\cos\theta_1 \\
&\quad + \left(m_2 r_{g2} + m_p l_2 - m_{c2} r_2\right) g\cos\left(\theta_1 + \theta_2\right), \\
G_2 &= \left(m_{c2} r_2 - m_2 r_{g2} - m_p l_2\right) g\cos\left(\theta_1 + \theta_2\right).
\end{aligned} \quad (30)$$

For link i ($i = 1, 2$), m_i denotes the mass, l_i denotes the length, I_i denotes the mass moment of inertia, m_p denotes the payload mass, r_{gi} denotes the distance from joint i to the center of mass of the link i, m_{ci} denotes the counterweights attached to link i and r_i denotes the distance from the joint to the counterweights.

In unbalanced case, counterweights of manipulator are zero ($m_{c1} = m_{c2} = 0$) which called normal case in this article. Now, in order to achieve the static balancing, m_{c1} and m_{c2} must be obtained in such a way that the gravity effects in Eq. (30) vanish. It implies that $G_1 = G_2 = 0$, so by defining the counterweights as follow:

$$\begin{aligned}
m_{c2} &= (m_2 r_{g2} + m_p l_2)/r_2, \\
m_{c1} &= (m_1 r_{g1} + m_2 l_1 + m_p l_1 + m_{c2} l_1)/r_1,
\end{aligned} \quad (31)$$

the dynamic parameters in Eq. (29) are changed to

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} \\ M_{12} & M_{22} \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix}, \quad (32)$$

where the value of inertia matrix is

$$\begin{aligned}
M_{11} &= m_1\left(r_{g1}^2 + r_g r_1\right) + m_2\left(l_1^2 + r_{g2}^2 + l_1 r_{g2}\frac{r_1 + l_1}{r_2} + l_1 r_1\right. \\
&\quad \left. + r_{g2} r_2\right) + m_p\left(l_1^2 + l_2^2 + l_1 r_1 + l_2 r_2 + l_1 l_2\frac{r_1 + l_1}{r_2}\right) + I_1, \\
M_{12} &= M_{21} = m_2 r_{g2}^2 + m_p l_2^2 + m_2 r_{g2} r_2 + m_p l_2 r_2, \\
M_{22} &= m_2 r_{g2}^2 + m_p l_2^2 + m_2 r_{g2} r_2 + m_p l_2 r_2 + I_2.
\end{aligned}$$

5.2. Dynamic equations of adaptive balanced two-link manipulator

The method of adaptive balancing is based on adding fix and rotary weights to obtain static balancing and complete decoupling of dynamic equations, thus eliminating Coriolis, centripetal, gravitational, and cross-inertia terms.[26] For adaptive balancing of the two-link manipulator, not only the gravity effects G_1 and G_2 must be released using the counterweights m_{c1} and m_{c2}, but also the cross-inertia term M_{12} must be vanished by adding the rotary weight m_b as shown in Fig. 2. So by defining m_{c1}, m_{c2}, and m_b as follow:

$$\begin{aligned}
m_{c2} &= (m_2 r_{g2} + m_p l_2)/r_2, \\
m_{c1} &= (m_p l_1 + m_1 r_{g1} + m_2 l_1 + m_{c2} l_1)/r_1, \\
m_b &= (m_2 r_{g2}(r_{g2} + r_2) + m_p l_2(l_2 + r_2) + I_2)/\left(r_b^2 r_T\right),
\end{aligned} \quad (33)$$

one can write the dynamic parameters in Eq. (29) as

$$\begin{aligned}
M_{11} &= m_1 r_{g1}(r_1 + r_{g1}) + m_2\left(\frac{l_1 r_{g2}}{r_2}(r_1 + l_1) + l_1(l_1 + r_1)\right. \\
&\quad \left. + 2r_{g2}(r_{g2} + r_2)\right) + m_p\left(\frac{l_1 l_2}{r_2}(r_1 + l_1) + l_1(l_1 + r_1)\right. \\
&\quad \left. + 2l_2(l_2 + r_2)\right) + I_1,
\end{aligned}$$

https://doi.org/10.1017/S0263574710000093 Published online by Cambridge University Press

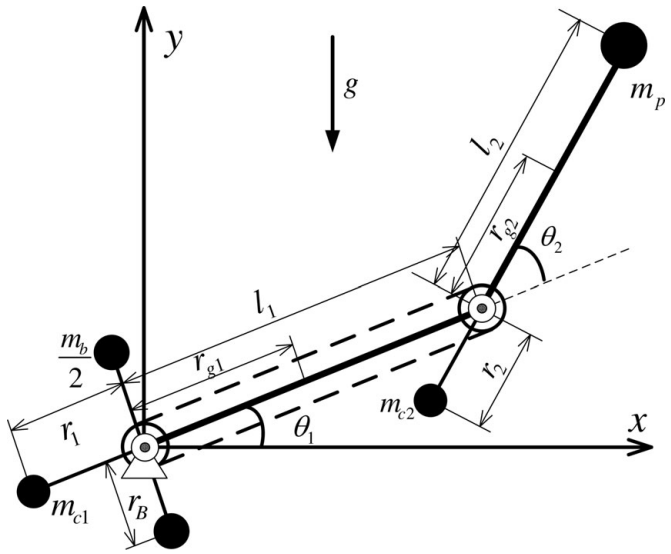


Fig. 2. Adaptively balanced manipulator.

corresponding boundary conditions are not required. The initial position of the end-effector in XY plan 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 velocity is considered to be zero. So by solving the inverse kinematic equations, one can write the boundary conditions as follow:

$$\begin{aligned}&\theta_1(0) = \theta_{10},\ \theta_2(0) = \theta_{20},\ \theta_1(t_f) = \theta_{1f},\ \theta_2(t_f) = \theta_{2f}\\ &\dot\theta_1(0) = \dot\theta_2(0) = \dot\theta_1(t_f) = \dot\theta_2(t_f) = 0.\end{aligned} \tag{35}$$

At the first step, using Eq. (20) by defining the continuous state vector as follow

$$X_1 = \begin{bmatrix}\theta_1(t)\\ \theta_2(t)\end{bmatrix} = \begin{bmatrix}x_1(t)\\ x_2(t)\end{bmatrix}, \quad X_2 = \begin{bmatrix}\dot\theta_1(t)\\ \dot\theta_2(t)\end{bmatrix} = \begin{bmatrix}x_3(t)\\ x_4(t)\end{bmatrix}, \tag{36}$$

the state space form of the dynamical Eq. (29), using Eq. (21) becomes

$$\begin{aligned}\dot x_1 &= x_3\\ \dot x_2 &= x_4\\ \dot x_3 &= \frac{M_{22}(X_1)(\tau_1 - C_1(X_1, X_2) - G_1(X_1)) - M_{12}(X_1)(\tau_2 - C_2(X_1, X_2) - G_2(X_1))}{M_{11}(X_1)M_{22}(X_1) - M_{12}(X_1)M_{21}(X_1)}\\ \dot x_4 &= \frac{M_{11}(X_1)(\tau_2 - C_2(X_1, X_2) - G_2(X_1)) - M_{21}(X_1)(\tau_1 - C_1(X_1, X_2) - G_1(X_1))}{M_{11}(X_1)M_{22}(X_1) - M_{12}(X_1)M_{21}(X_1)},\end{aligned} \tag{37}$$

$$\begin{aligned}M_{22} &= I_2 + 2m_2 r_{g2}(r_{g2} + r_2) + 2m_p l_2(l_2 + r_2) = 2m_b r_b^2 r_T,\\ M_{12} &= M_{21} = 0,\\ C_1 &= C_2 = G_1 = G_2 = 0,\end{aligned} \tag{34}$$

where r_b is the position of both compensation masses $m_b/2$ with respect to the first joint as shown in Fig. 2; and r_T is the transmission ratio.

5.3. Optimality conditions for two-link manipulator without parameter

In this section, using the general formulation mentioned in Section 4, optimality conditions are derived for the considered two-link manipulator at unbalanced, static balanced and adaptive balanced conditions. For the unbalanced case, finding the optimal trajectory between the two given points of the end-effector is considered. For the static and adaptive balanced cases, at first using Eqs. (31) and (33) counterweights and rotary weight are obtained, then the optimal trajectory for the given performance index is obtained. So the unknown constant parameters do not appear in trajectory optimization procedure. The formulation presented in Section 4 is for the path planning problem including parameters. So for optimal path planning the unbalanced, static balanced and adaptive balanced cases which constant parameter is not appeared, Eq. (26) and its where M_{ij}, C_i and G_i $(i, j = 1, 2)$ are substituted from Eqs. (32) and (34) for static balancing and adaptive balancing, respectively. For unbalanced case m_{c1} and m_{c2} are considered to be zero in Eq. (30). Now according to Eqs. (1) and (6), by considering the performance index as minimum control effort which is defined as follow:

$$J = \int_{t_0}^{t_f} \left(\tau_1^2 + \tau_2^2\right) \mathrm{d}t, \tag{38}$$

and the costate vector as $\psi_1 = [\,x_5\ x_6\,]^T$ and $\psi_2 = [\,x_7\ x_8\,]^T$, using Eq. (23) the Hamiltonian function becomes

$$H = \left(\tau_1^2 + \tau_2^2\right) + x_5\dot x_1 + x_6\dot x_2 + x_7\dot x_3 + x_8\dot x_4, \tag{39}$$

where $\dot x_i,\ i = 1, \ldots, 4$ can be substituted from Eq. (37).

Then by substituting Eq. (37) into Eq. (39), and differentiating the Hamiltonian function with respect to the states, according to Eq. (25), the costate equations are obtained as follows

$$\begin{aligned}\dot x_5 &= -\frac{\partial}{\partial x_1}\left(\frac{(-C_1-G_1)(x_7M_{22}-x_8M_{12})+(-C_2-G_2)(-x_7M_{12}+x_8M_{11})}{M_{11}M_{22}-M_{12}M_{21}}\right),\\ \dot x_6 &= -\frac{\partial}{\partial x_2}\left(\frac{(-C_1-G_1)(x_7M_{22}-x_8M_{12})+(-C_2-G_2)(-x_7M_{12}+x_8M_{11})}{M_{11}M_{22}-M_{12}M_{21}}\right),\\ \dot x_7 &= -x_5 - \frac{\partial}{\partial x_3}\left(\frac{(-C_1)(x_7M_{22}-x_8M_{12})+(-C_2)(-x_7M_{12}+x_8M_{11})}{M_{11}M_{22}-M_{12}M_{21}}\right),\\ \dot x_7 &= -x_6 - \frac{\partial}{\partial x_4}\left(\frac{(-C_1)(x_7M_{22}-x_8M_{12})+(-C_2)(-x_7M_{12}+x_8M_{11})}{M_{11}M_{22}-M_{12}M_{21}}\right).\end{aligned} \tag{40}$$

https://doi.org/10.1017/S0263574710000093 Published online by Cambridge University Press

After that, using Eq. (27) the control values can be obtained by solving the following equations as

$$\frac{\partial H}{\partial \tau_1} = 0, \quad \frac{\partial H}{\partial \tau_2} = 0. \tag{41}$$

So by substituting the Hamiltonian Function from Eq. (39) into Eq. (41), and using Eq. (27), the optimal control laws become

$$\begin{aligned} \tau_1 &= \frac{0.5}{M_{11}M_{22} - M_{12}M_{21}} \left(-x_7 M_{22} + x_8 M_{21}\right), \\ \tau_2 &= \frac{0.5}{M_{11}M_{22} - M_{12}M_{21}} \left(x_7 M_{12} - x_8 M_{11}\right). \end{aligned} \tag{42}$$

Finally by substituting Eq. (42) into Eqs. (37) and (40), eight nonlinear ordinary differential equations will be obtained which with eight boundary conditions given in Eq. (35), construct a two point boundary value problem. This problem can be solved using the bvp4c command in MATLAB®.

5.4. Optimal balancing of two-link manipulator

Unlike the static and adaptive balanced cases in which the counterweights are depended on manipulator parameters as shown by Eqs. (31) and (33), in optimal balanced case the values of counterweights are depended on dynamic equations, performance index and boundary conditions according to Eqs. (24)–(28). Therefore, the counterweights and optimal trajectory are obtained simultaneously in such a way that the given performance index is minimized. In this case, optimal control problem involving parameters which its optimality conditions are given in Eqs. (24)–(28) must be considered. Dynamic equations, costate equations and optimal control law are the same as obtained in Eqs. (37), (40), and (42), respectively. The dynamic parameters M_{ij}, C_i, and G_i $(i, j = 1, 2)$ in Eqs. (37), (40), and (42), can be substituted form Eq. (30). Here, in all equations m_{c1} and m_{c2} are considered to be unknown parameters.

Now using Eqs. (26), by defining the two new state variables x_9 and x_{10}, the optimality conditions associated with the parameters are given by

$$\dot{x}_9 = -\frac{\partial H}{\partial m_{c1}}, \; \dot{x}_{10} = -\frac{\partial H}{\partial m_{c2}}, \tag{43}$$

where according to Eq. (28) the associated boundary conditions become

$$x_9(0) = x_{10}(0) = x_9(t_f) = x_{10}(t_f) = 0. \tag{44}$$

At last, by substituting Eq. (42) into Eqs. (37), (40), and (43), 10 nonlinear ordinary differential equations with respect to state $[x_1\ x_2\ x_3\ x_4]$, costate $[x_5\ x_6\ x_7\ x_8]$, new states $[x_9\ x_{10}]$, and unknown parameters $[m_{c1}\ m_{c2}]$ will be achieved. These ten equations with twelve boundary conditions given in Eqs. (35) and (44), construct a two point boundary value problem which by solving it using bvp4c command in MATLAB, all the states and unknown parameters can be obtained.

Table I. Simulation robot parameters.

Parameters	Values	Unit
Mass	$m_1 = m_2 = 1.0$	kg
Payload mass	$m_p = 2.0$	kg
Length of links	$L_1 = L_2 = 1.0$	m
Moment of inertia	$I_1 = I_2 = \frac{1}{12}$	kg.m^2
Length of adjacent links	$r_1 = r_2 = 0.5$	m

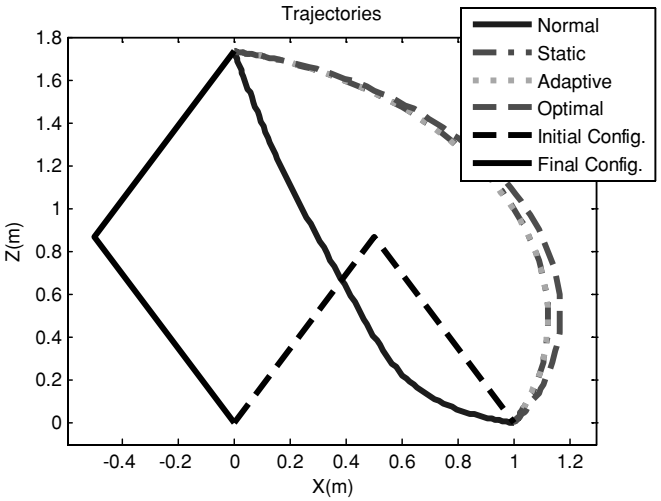


Fig. 3. Optimal trajectories for two-link manipulator.

6. Simulation Results

6.1. Simulation results for two-link manipulator

A two-link manipulator at vertical plan is considered as shown in Fig. 1. All required parameters of the robot manipulator are given in Table I.

The initial position of the end-effector in 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

$$\begin{aligned} &\theta_1(0) = 60^\circ,\ \theta_2(0) = 120^\circ,\ \theta_1(t_f) = 120^\circ,\ \theta_2(t_f) = -60^\circ \\ &\dot{\theta}_1(0) = \dot{\theta}_2(0) = \dot{\theta}_1(t_f) = \dot{\theta}_2(t_f) = 0. \end{aligned} \tag{45}$$

Using Eq. (31), the counterweights of static balanced manipulator are calculated to be: $m_{c1} = 17\,\text{kg}$, $m_{c2} = 5\,\text{kg}$, and using Eq. (33) the counterweights of adaptive balanced manipulator are calculated to be: $m_{c1} = 17\,\text{kg}$, $m_{c2} = 5\,\text{kg}$, $m_b = 5\,\text{kg}$. For unbalanced case the counterweights are considered to be zero.

By solving the two point boundary value problem obtained in Sections 5.3 and 5.4, the minimum control effort trajectories of the end effector for unbalanced, static, adaptive and optimal balanced cases are plotted in Fig. 3. For optimal case the counterweights are found to be $m_{c1} = 85\,\text{kg}$, $m_{c2} = 3\,\text{kg}$. As it can be seen the static and adaptive trajectories are very close together. It may in cause of low effectiveness of decoupling moments in static balanced system. The angular position and angular velocity of joints are illustrated in Figs. 4 and 5, respectively.

https://doi.org/10.1017/S0263574710000093 Published online by Cambridge University Press

Table II. The cost function values for different cases, $t_f = 1$s.

Balancing	Performance index J(Nm)^2
Normal	1093.39
Static	5854.67
Adaptive	5766.58
Optimal	564.22

Figure 6 shows input torques for the normal, static, adaptive and optimal balanced cases. As it can be seen the applied torque for optimal balanced case is lower than the other cases. In order to clarify this subject, the values of performance index for different cases obtained of Eq. (38), are shown in Table II. As it can be seen, performance index for optimal balanced case is smaller than the other ones significantly.

As mentioned before, the values of counterweights for static and adaptive balancing are constant values at every boundary conditions and time interval t_f. But the values of counterweights for optimal balanced case are depended on boundary conditions and time duration. The optimal balancing trajectory is a trajectory between unbalanced and static balanced one. This conception is effected by final time and it can be mentioned as: "if time interval t_f approaches to infinity, optimal counterweights approaches to static ones." In order to show this aspect, for the boundary conditions given in Eq. (45), the values of counterweights m_{c1} and m_{c2} for static and optimal cases are illustrated with respect to the time duration t_f in Fig. 7. As shown in this graph by increasing the final time t_f, the optimal counterweights converge to static ones. In Fig. 8, the obtained performance index values for normal, static, adaptive and optimal cases with respect to time duration t_f is shown. As it can be seen, the performance index values for static and adaptive cases are very close to each other. So it can be resulted that decoupling forces are small and have very little effect on optimal trajectory. In Fig. 9 changing of performance index with respect to counterweight values are shown. In this figure, it is possible to see that the effect of counterweight alternation has very

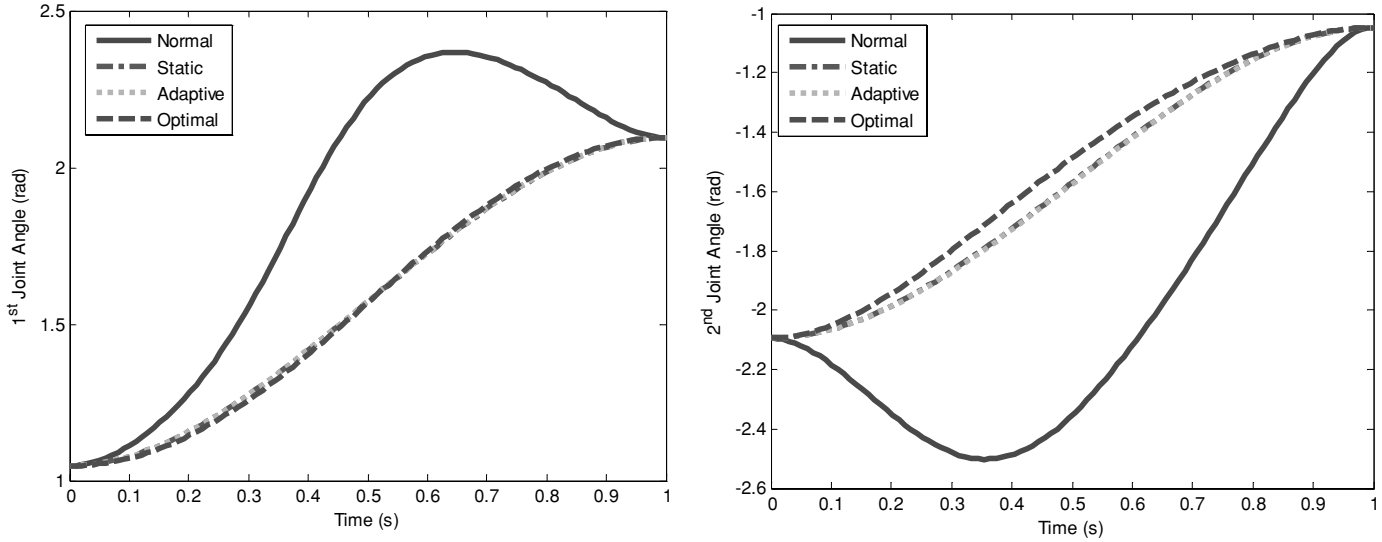


Fig. 4. Angular position of joints.

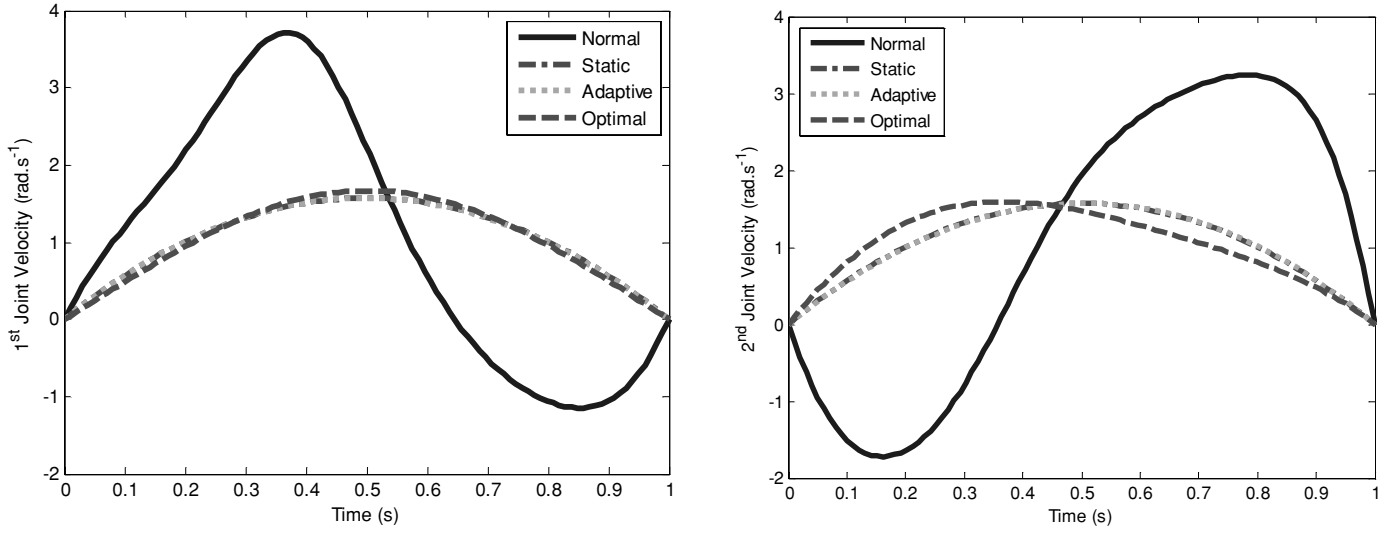


Fig. 5. Angular velocity of joints.

https://doi.org/10.1017/S0263574710000093 Published online by Cambridge University Press

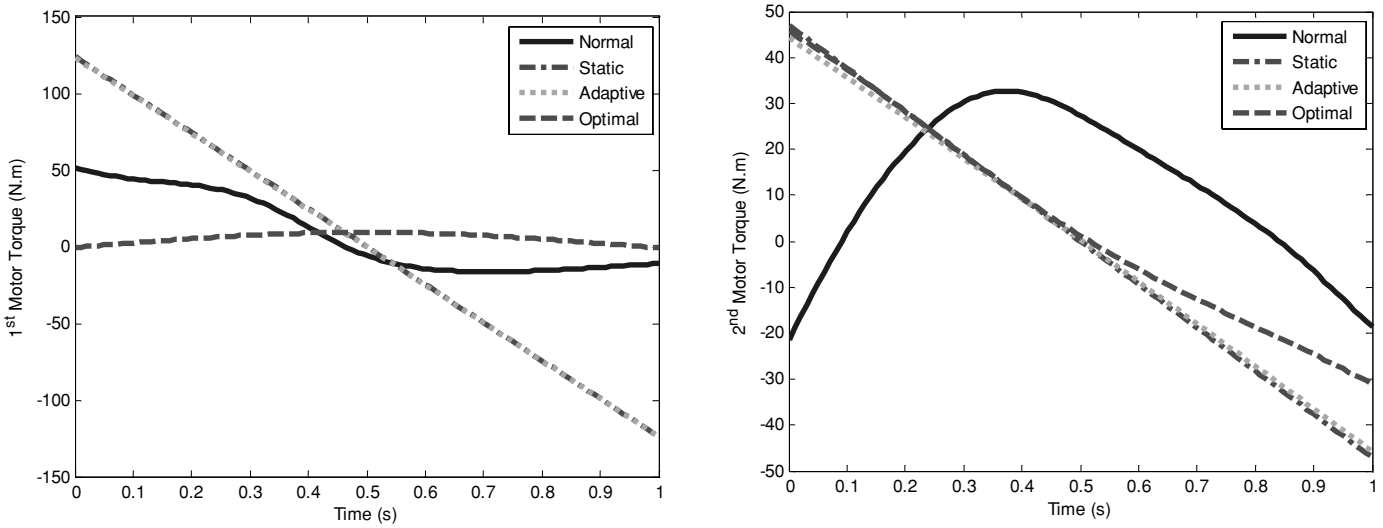


Fig. 6. Torques of joints.

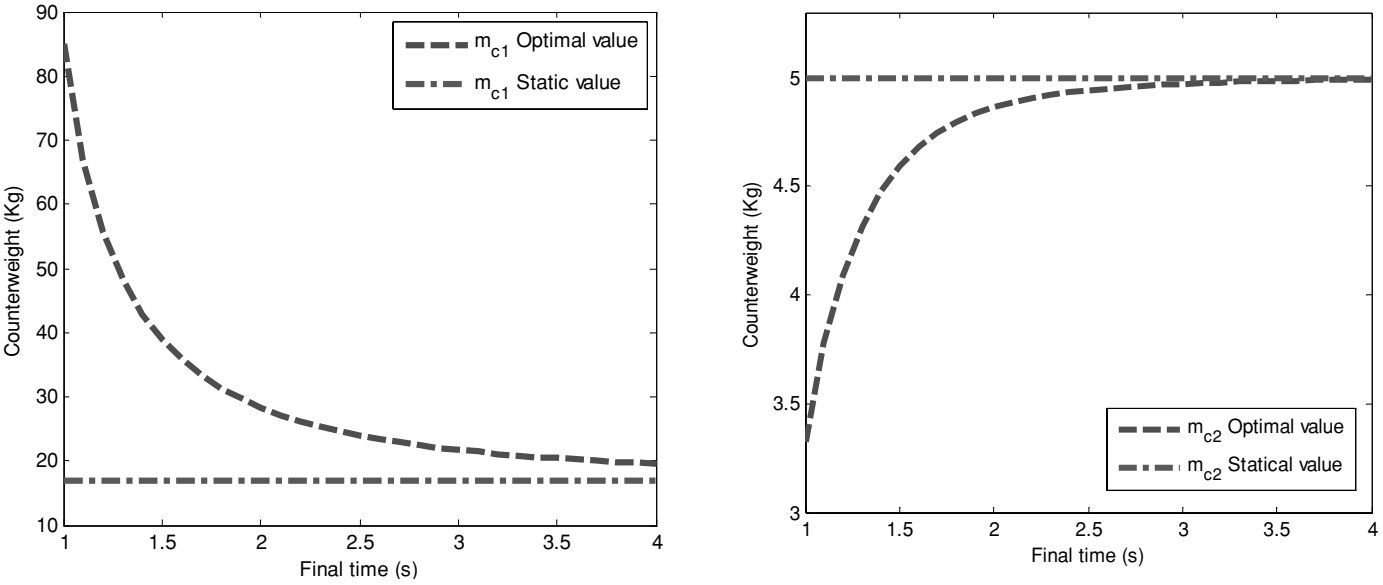


Fig. 7. Static and optimal counterweights with respect to time duration t_f.

effect on performance index. Also, as it can be shown in Fig. 9 the performance index for optimal case is always lower than the other ones considerably.

6.2. *Simulation results for spatial three-link manipulator*

As the second case study, the proposed method is applied on the three-link manipulator with rotational joints, described by Herman.[32] The schematic view of this manipulator is shown in Fig. 10, and its parameters are given in Table III. The initial conditions of the manipulator are $\theta_1(0) = 0$, $\theta_2(0) = \pi/6$, $\theta_3(0) = -\pi/2$, and must be reached to final conditions $\theta_1(t_f) = \pi/3$, $\theta_2(t_f) = 5\pi/12$, $\theta_3(t_f) = -\pi/3$ at $t_f = 1s$. Coincident task is a motion of gripper from $p_0(0, 0.793, 0.227)m$ to $p_f(-0.347, 0.200, 0.825)m$. The initial and final velocities are zero.

Now by considering the performance index as $\int_0^{t_f} (\tau_1^2 + \tau_2^2 + \tau_3^2)\mathrm{d}t$, the optimal trajectory for normal, static and optimal balanced cases, between the two given

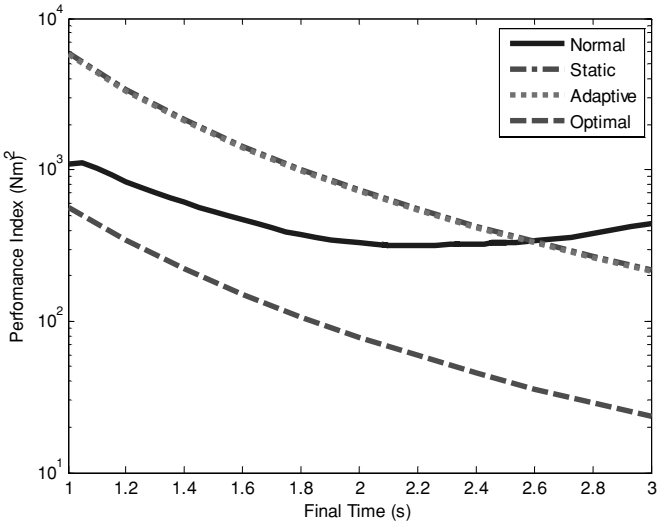


Fig. 8. Performance index due to final time t_f.

https://doi.org/10.1017/S0263574710000093 Published online by Cambridge University Press

Table III. Parameters of the three-link manipulator.

Parameter	Unit	Link 1	Link 2	Link 3
Mass (m_i)	kg	42	21	12
Mass moment of inertia (J)	kg.m^2	0 0 0 0 0 0 0 0 1.5	0.2 0 0 0 0.7 0 0 0 1	0.3 0 0 0 0.4 0 0 0 0.5
Center of mass	m	[0,0,0]	[0.4,0,0]	[0.2,0,0]
Lengths (l_i)	m	0	0.8	0.4

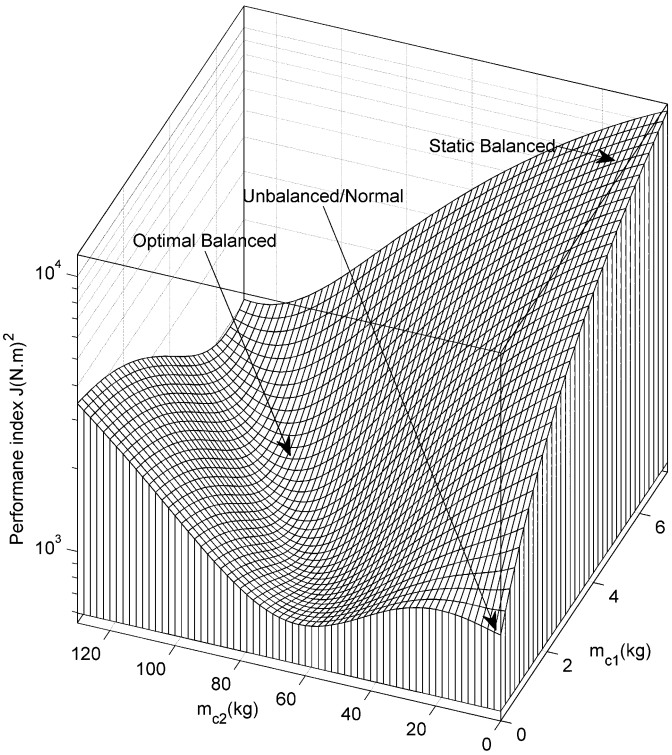


Fig. 9. Performance index with respect to counterweights.

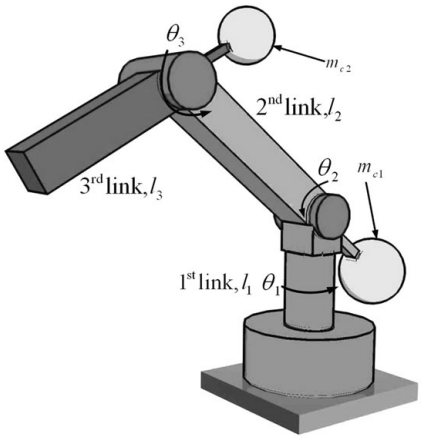


Fig. 10. Schematic view of three-link manipulator.

points are obtained. For this manipulator, the values of counterweights for static balancing are calculated to be: $m_{c1} = 57\,\text{kg}$, $m_{c2} = 6\,\text{kg}$, and for optimal balancing are calculated to be: $m_{c1} = 53.7\,\text{kg}$, $m_{c2} = 0$. After simulation in MATLAB©, the optimal trajectories for different cases

Table IV. The performance index values for different cases.

Balancing	Performance index J(Nm)^2
Normal	13,981.64
Static	5397.84
Optimal	1476.73

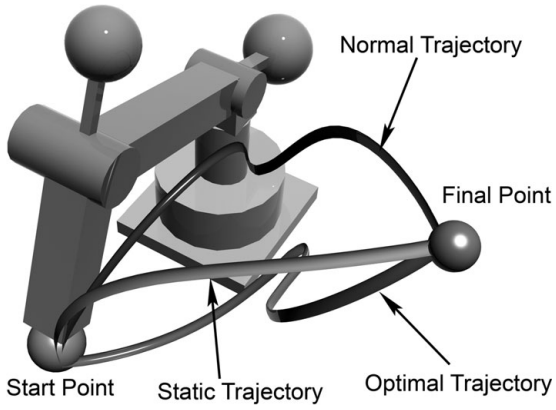


Fig. 11. Optimal trajectories for three-link manipulator.

are shown in Fig. 11 by using of 3D modeling program. The input torques for normal, static and optimal balanced cases are shown in Fig. 12, and the values of performance index for different cases are given in Table IV. As it can be seen, performance index for optimal balanced case is very smaller than the other ones.

7. Conclusion and Future Works

One important contribution of the optimal balancing is that states, controls, and the values of counterweights are determined simultaneously in order to minimize the given performance objective, by solving the equations obtained of optimality conditions. In previous balancing approaches, trajectory of manipulator is not considered, so some energy is wasted. Moreover, in previous trajectory planning method, values of counterweights are not included. In this article, these two areas of robotic are combined, and a new area for balancing and trajectory planning is suggested which results in the best possible response. Although, in static balancing the gravity effects on the actuators are released, and in adaptive balancing the Coriolis, centripetal, gravitational, and cross-inertia terms are eliminated, but it does not mean that the required torque or other criteria to move the manipulator from one point to another point is minimum

https://doi.org/10.1017/S0263574710000093 Published online by Cambridge University Press

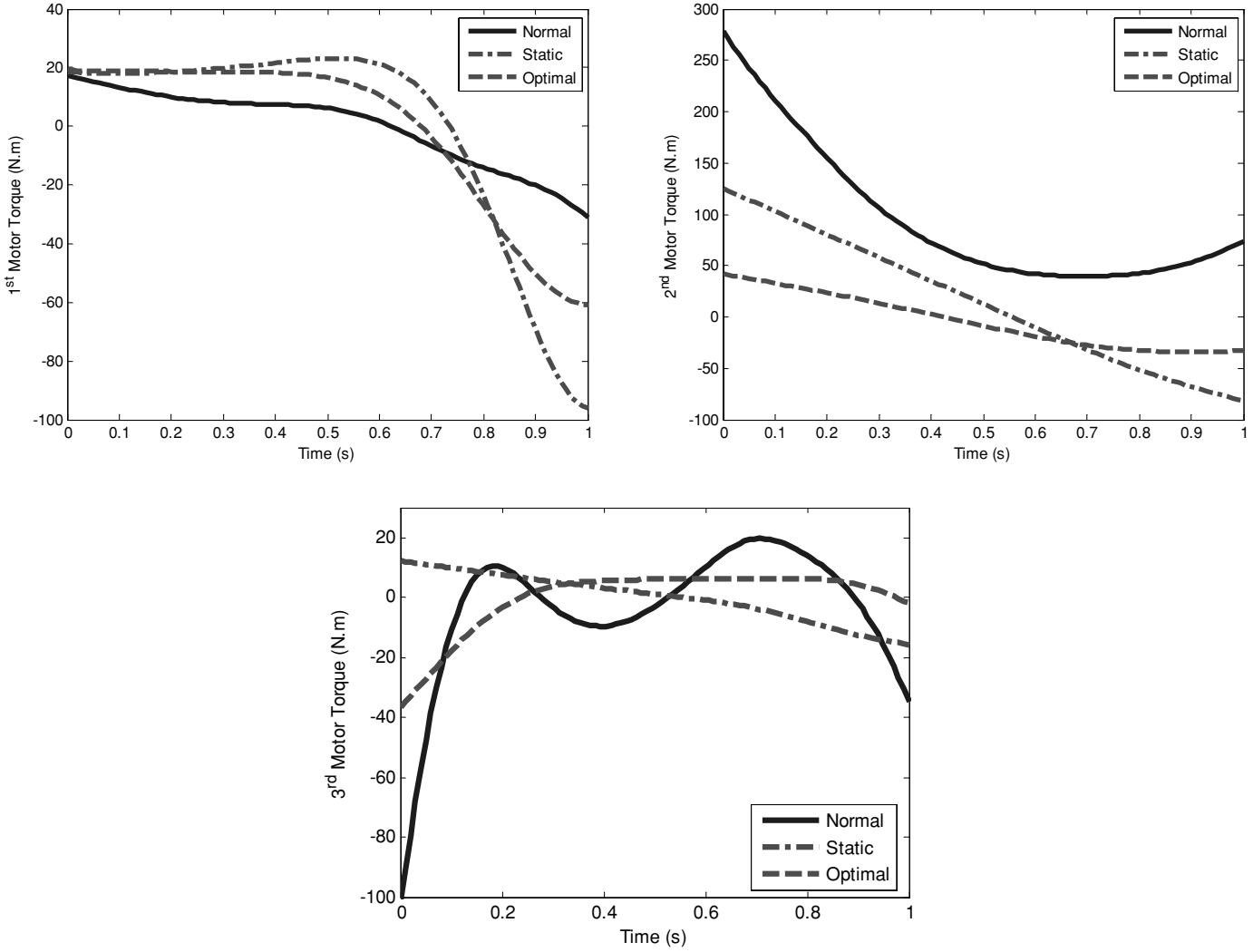


Fig. 12. Torque of joints for different cases.

essentially. In the other word, gravity, Coriolis and centripetal forces are not bad things basically, if they can be employed to decrease the given performance index. The efficiency of the proposed method is investigated through computer simulations by considering a two-link manipulator. The obtained results show that the performance index can be reduced significantly if the values of counterweights are chosen properly. Also it is illustrated that the optimal counterweights are depended on traveling time, in such a way that the optimal counterweights converge to static counterweights by increasing the traveling time. Finally, simulation is performed for a three-link manipulator and the capability of the method to solve the complicated problem is shown. For this case study, performance index for static balancing is obtained to be 5398, while for optimal balancing is found to be 1477. This major reduction in applying torques is occurred by choosing the counterweights as $m_{c1} = 53.7\,\text{kg}$, $m_{c2} = 0$ instead of $m_{c1} = 57\,\text{kg}$, $m_{c2} = 6\,\text{kg}$.

Solving the optimal balancing problem by considering the constraints and obstacles in the work space are future works in this field. In this paper, balancing is done using counterweight, solving the optimal balancing based spring is another field in future. Also it is possible to solve the time-optimal problem by combination of optimal balancing.

References

1. T. Chettibi, H. E. Lehtihet, M. Haddad and S. Hanchi, "Minimum cost trajectory planning for industrial robots," *Eur. J. Mech.* **23**(4), 703–715 (2004).
2. D. G. Hull, "Conversion of optimal control problems into parameter optimization problems," *J. Guid. Control Dyn.* **20**(1), 57–60 (1997).
3. J. T. Betts, "Survey of numerical methods for trajectory optimization," *J. Guid. Control Dyn.* **21**(2), 193–207 (1998).
4. R. Callies and P. Rentrop, "Optimal control of rigid-link manipulators by indirect methods," *GAMM-Mitteilungen* **31**(1), 27–58 (2008).
5. Y. C. Chen, "Solving robot trajectory planning problems with uniform cubic B-splines," *Optim. Control Appl. Methods* **12**(4), 247–262 (1991).
6. P. E. Gill, W. Murray and A. Saunders, "Large-Scale SQP Methods and Their Application in Trajectory Optimization," **In**: *Computational Optimal Control* (Birkhauser Verlag, Basel, Switzerland, 1994) pp. 29–42.
7. A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale

https://doi.org/10.1017/S0263574710000093 Published online by Cambridge University Press

nonlinear programming," *Math. Program.* **106**(1), 25–57 (2006).
8. X-S. Ge and L-Q. Chen, "Optimal motion planning for nonholonomic systems using genetic algorithm with wavelet approximation," *Appl. Math. Comput*. **180**, 76–85 (2006).
9. T. Ravichandran, D. Wang and G. Heppler, "Simultaneous plant-controller design optimization of a two-link planar manipulator," *Mechatronics* **16**(3), 233–242 (2006).
10. R. Saravanan, S. Ramabalan and C. Balamurugan, "Multiobjective trajectory planner for industrial robots with payload constraints," *Robotica* **26**(6), 753–765 (2008).
11. J. Arora, *Introduction to Optimum Design* (Elsevier Academic Press, San Diego, CA, 2004).
12. D. E. Kirk, *Optimal Control Theory, an Introduction* (Prentice-Hall, Englewood Cliffs, NJ, 1970).
13. Z. Shiller and S. Dubowsky, "Robot path planning with obstacles, actuators, gripper and payload constraints," *Int. J. Robot. Res.* **8**(6), 3–18 (1986).
14. Z. Shiller, "Time-energy optimal control of articulated systems with geometric path constraints," *IEEE Int. Conf. Robot. Autom.* **4**, 2680–2685 (1994).
15. R. Fotouhi and W. Szyszkowski, "An algorithm for time optimal control problems," *J. Guid. Control Dyn*. **120**, 414–418 (1998).
16. G. Bessonnet and S. Chessé, "Optimal dynamics of actuated kinematic chains. Part 2. Problem statements and computational aspects," *Eur. J. Mech.* **24**, 472–490 (2005).
17. E. Bertolazzi, F. Biral and M. Da Lio, "Symbolic–numeric indirect method for solving optimal control problems for large multibody systems," *Multibody Syst. Dyn.* **13**(2), 233–252, (2005).
18. M. R. Sentinella and L. Casalino, "Genetic Algorithm and Indirect Method Coupling for Low-Thrust Trajectory Optimization," *42nd AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*, California (July 9–12, 2006).
19. S. Furuno, M. Yamamoto and A. Mohri, "Trajectory planning of mobile manipulator with stability considerations," *IEEE Int. Conf. Robot. Autom.* **3**, 3403–3408 (2003).
20. M. H. Korayem and A. Nikoobin, "Maximum payload for flexible joint manipulators in point-to-point task using optimal control approach," *Int. J. Adv. Manuf. Technol.* **38** (9–10), 1045–1060 (2008).
21. M. H. Korayem, A. Nikoobin and V. Azimirad, "Trajectory optimization of flexible link manipulators in point-to-point motion," *Robot. J.* **27**(6), 825–840 (2009).
22. M. H. Korayem and A. Nikoobin, "Maximum-payload path planning for redundant manipulator using indirect solution of optimal control problem," *Int. J. Adv. Manuf. Technol.* **44**(7–8), 725–736 (2009).
23. E. I. Rivin, *Mechanical Design of Robots* (McGraw-Hill, New York, 1988).
24. 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).
25. 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).
26. 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).
27. V. A. Kamenskii, "On the problem of the number of counterweights in the balancing of plane linkages," *J. Mech.* **3**, 323–333 (1968).
28. 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).
29. 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).
30. B. G. Santangelo and R. Sinatra, "Static balancing of a six-degree-of freedom parallel mechanism with six two-link revolute legs," *Int. J. Robot. Autom.* **20**(4), 206–216 (2005).
31. D. G. Hull, "Sufficiency for optimal control problems involving parameters," *J. Optim. Theory Appl.* **97**(3), 579–590 (1998).
32. P. Herman, "Dynamical couplings reduction for rigid manipulators using generalized velocity components," *Mech. Res. Commun.* **35**, 553–561 (2008).

https://doi.org/10.1017/S0263574710000093 Published online by Cambridge University Press