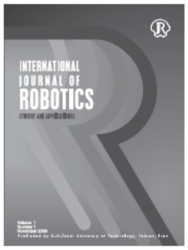
International Journal of Robotics, Vol. 4, No. 3, (2015) E. Sadraei, M. M. Moghaddam, 66-74



INTERNATIONAL JOURNAL OF ROBOTICS
(THEORY AND APPLICATIONS)
E-mail: IJR@kntu.ac.ir

On a Moving Base Robotic Manipulator Dynamics

E. Sadraei[a], M. M. Moghaddam[b, *]

[a] Faculty of Mechanical Engineering, Tarbiat Modares University, Tehran, Iran, P.O. Box, 14115-111
[b] Faculty of Mechanical Engineering, Tarbiat Modares University, Tehran, Iran, P.O. Box, 14115-111

ARTICLE INFO

Article history:

Received: June 10, 2015.
Received in revised form: September 19, 2015.
Accepted: October 2, 2015.

Keywords:

Robot manipulator
Manipulator dynamic
Base motion

ABSTRACT

There are many occasions where the base of a robotic manipulator is attached to a moving platform, such as on a moving ship, terrain or space shuttle. In this paper a dynamic model of a robotic manipulator mounted on a moving base is derived using both Newton-Euler and Lagrange-Euler methods. The presented models are simulated for a Mitsubishi PA10-6CE robotic manipulator characteristics mounted on a ship platform that is moving on ocean and the results are verified through both methods. In this simulation it is assumed that the inertia of the base of the robot is large enough and is not affected by the manipulator motion. However, the motion of the ship directly influences the dynamics of the manipulator in movements. Results and computation time of the two methods are compared and it is shown that the Newton-Euler method needs less computation time than the Lagrange method.

1. Introduction

One of the issues in robotics is where the base of a manipulator is not fixed. For example, there is a robot manipulator placed on a ship moving through the ocean or train on rails. In all of above cases motion of the base of the manipulator affects robots kinematic and dynamic, and subsequently causes deviation of the trajectory of robot arms and end effector from our desired trajectories. There have been many researches on the effects of base motion such as manipulator placed on underwater vehicles, space vehicles and manipulators on ships. In 1986 J. Joshi and A. Desrochers worked on a 1-DOF robot manipulator which was placed on a base with 3-DOF platform. They assumed that the base of the robot moves randomly and utilized a PID controller to control their linearized system [1]. Jamisola has provided a method called ‘dominant inertia’. In this method each joint's linear friction compensation and oscillatory control are used to obtain the lumped inertias. Moreover, models for static, coulomb and viscous frictions were obtained [2]. In 2007 identification of the dynamics of the 7-DOF PA10 with Stribeck friction and transmission compliance model was performed [3]. A 2 DOF base motion on tank with a linear model was used in trajectory planning and SISO and MIMO controller were designed for it by A.B. Tanner in 1987 [4]. In 1999 F. M. Carter and D.B. Cherchas derived the dynamic model of a robotic manipulator on a mobile base with two and three DOF of motions. They used a PD controller for the system and no compensation was considered in the dynamic of the manipulator arm [5].

A robot manipulator installed on a ship deck was considered in 2004 [6, 7]. The dynamics of the 6-DOF robot manipulator with 6-DOF base motion and different size of ships and height of waves on ship's

* Corresponding author address: Tarbiat Modares University, Jalal Al Ahmad Ave, Tehran, Iran
Tel.: +982182883358; fax: +982188005040, E-mail address: m.moghadam@modares.ac.ir

motion were considered. Also effects of sea-states and wave height on the ship motion are presented for different ship sizes. In other researches, modeling and motion planning for mechanisms on a non-inertial base with known trajectory is used for computing torques with a feedback controller. The friction was not considered and simulations are presented for a 1-DOF and 4-DOF manipulators on a 6-DOF base, they worked on reducing actuator energy usage in 2009 [8]. Few important dynamic and control problems, uniquely in space robotics are discussed in [9]. They particularly considered the free flying and free-floating space robots for tasks as space station repair and construction. Three methods for planning and controlling the motion of space robotic systems are presented. They suggested that a thorough understanding of dynamics of these systems will result in effective solutions to their control problems.

An analysis of cooperating manipulators on a non-fixed base is considered; dynamic equations two 6-DOF manipulators and 6-DOF platform are presented and it is assumed that manipulator dynamics affect the platform but not actuate it [10]. A robot manipulator on a moving base was considered by C.M. Wronka and M.W. Dunnigan, The dynamic equations of robot manipulator on moving base were derived and they assumed that the base inertia was large enough not to be influenced by the manipulator motion [11].

This paper has the following sections. In section two the dynamic equations considering the base motion are derived by Newton-Euler and Lagrange-Euler methods. In section three the derived equations are applied to Mitsubishi PA10-6CE robotic manipulator. Finally, in section four equations are solved and results of two methods are compared.

2. Model derivation

The dynamic model of a manipulator on a moving platform is derived using the Newton-Euler and Lagrange-Euler approaches. In deriving the equations of the motions, the following assumptions are made.

- The manipulator motion has no influence on the platform, i.e. the inertia of the platform is very high,
- The motor inertias and friction terms are not considered in the model derivation.

2.1. Lagrange-Euler method

The general equation of motion of a manipulator can be conveniently expressed through the direct application of the Lagrange-Euler formulation of non-conservative systems. Different notation of the kinematic and kinetic of robot manipulator are presented [11, 12]. The Lagrange-Euler equations are as Eq. 1:

$$\frac{\mathbf{d}}{\mathbf{dt}}\left(\frac{\partial \mathbf{L}}{\partial \dot{\mathbf{q}}_\mathbf{i}}\right) - \frac{\partial \mathbf{L}}{\partial \mathbf{q}_\mathbf{i}} = \boldsymbol{\tau}_\mathbf{i} \qquad \mathbf{i} = \mathbf{1}, \mathbf{2}, \dots, \mathbf{n} \tag{1}$$

Where

L: Lagrangian = K (kinetic energy) – P (potential energy)

q_i: Generalized coordinate of robot arm in base coordinate

$\dot{q}_i$: First derivative of generalized coordinate q_i

τ_i: Generalized force (or torque) applied to the system at joint i to drive link i

The velocity $\left({}^{\mathbf{0}}\mathbf{V}_\mathbf{i}\right)$ of a point of a rigid link in inertial coordinate can be expressed as:

$$^{\mathbf{0}}\mathbf{V}_\mathbf{i} = \frac{\mathbf{d}}{\mathbf{dt}}\left({}^{\mathbf{0}}\mathbf{r}_\mathbf{i}\right) = \frac{\mathbf{d}}{\mathbf{dt}}\left(\mathbf{A}_\mathbf{b}\ {}^{\mathbf{0}}\mathbf{A}_\mathbf{i}\ {}^{\mathbf{i}}\mathbf{r}_\mathbf{i}\right) \tag{2}$$

Where

${}^{\mathbf{0}}\mathbf{A}_\mathbf{i}$: is the Denavit and Hartenberg (D-H) transformation matrix of the coordinates from i-th link to base.

${}^{\text{inertia}}\mathbf{A}_\mathbf{0}$ = $\mathbf{A}_\mathbf{b}$: is a transformation matrix of the base to the inertial coordinate.

${}^{i}\boldsymbol{r}_{\boldsymbol{i}}$ is a point within the i-th link expressed in the i-th link coordinates.

The time derivative of velocity can be found as:

$$\frac{\mathbf{d}}{\mathbf{dt}}\left(\mathbf{A}_\mathbf{b}\ {}^{\mathbf{0}}\mathbf{A}_\mathbf{i}\ {}^{\mathbf{i}}\mathbf{r}_\mathbf{i}\right) = \left(\dot{\mathbf{A}}_\mathbf{b}\ {}^{\mathbf{0}}\mathbf{A}_\mathbf{i} + \mathbf{A}_\mathbf{b}\sum_{\mathbf{j}=\mathbf{1}}^{\mathbf{i}} \frac{\partial\ {}^{\mathbf{0}}\mathbf{A}_\mathbf{i}}{\partial \mathbf{q}_\mathbf{j}}\dot{\mathbf{q}}_\mathbf{j}\right)\ {}^{\mathbf{i}}\mathbf{r}_\mathbf{i} \tag{3}$$

Where

${}^{\mathbf{0}}\mathbf{A}_\mathbf{i}$: is a function of joint angles $\mathbf{q}_\mathbf{j}$ for j ∈ (1, .. , i). Let

$$\mathbf{U}_\mathbf{ij} = \frac{\partial\ {}^{\mathbf{0}}\mathbf{A}_\mathbf{i}}{\partial \mathbf{q}_\mathbf{j}}, \qquad \mathbf{U}_\mathbf{ijk} = \frac{\partial \mathbf{U}_\mathbf{ij}}{\partial \mathbf{q}_\mathbf{k}} \tag{4}$$

$$\begin{aligned}\frac{\partial\ {}^{\mathbf{0}}\mathbf{A}_\mathbf{i}}{\partial \mathbf{q}_\mathbf{j}} &= {}^{\mathbf{0}}\mathbf{A}_{\mathbf{j-1}}\frac{\partial^{\mathbf{j-1}}\mathbf{A}_\mathbf{j}}{\partial \mathbf{q}_\mathbf{j}}\ {}^{\mathbf{j}}\mathbf{A}_\mathbf{i} \\ &= {}^{\mathbf{0}}\mathbf{A}_{\mathbf{j-1}}\mathbf{Q}_\mathbf{d}\ {}^{\mathbf{j-1}}\mathbf{A}_\mathbf{j}\ {}^{\mathbf{j}}\mathbf{A}_\mathbf{i}\end{aligned} \tag{5}$$

$$\mathbf{Q}_\mathbf{d} = \begin{bmatrix} \mathbf{0} & \mathbf{-1} & \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \text{ Algebraic operator}$$

The velocity vectors expressed in base coordinate results to:

$$\mathbf{{}^{0}V_{i}} = (\dot{\mathbf{A}}_{\mathbf{b}}\ {}^{\mathbf{0}}\mathbf{A}_{\mathbf{i}} + \mathbf{A}_{\mathbf{b}} \sum_{\mathbf{j=1}}^{\mathbf{i}} \mathbf{U}_{\mathbf{ij}} \dot{\mathbf{q}}_{\mathbf{j}})\ {}^{\mathbf{i}}\mathbf{r}_{\mathbf{i}} \tag{6}$$

1) *Kinetic energy*

The kinetic energy of a particle with mass dm of link *i* can be defined as (Tr = trace):

$$\mathbf{dk_i} = \frac{\mathbf{1}}{\mathbf{2}} \mathbf{Tr}\left({}^{\mathbf{0}}\mathbf{V}_{\mathbf{i}} ({}^{\mathbf{0}}\mathbf{V}_{\mathbf{i}})^{\mathbf{T}} \right) \mathbf{dm} \tag{7}$$

By substituting Eq. 6 in Eq. 7

$$\mathbf{dk_i} = \frac{\mathbf{1}}{\mathbf{2}} \mathbf{Tr}\Bigg(\Big(\dot{\mathbf{A}}_{\mathbf{b}}\ {}^{\mathbf{0}}\mathbf{A}_{\mathbf{i}} + \mathbf{A}_{\mathbf{b}} \textstyle\sum_{\mathbf{j=1}}^{\mathbf{i}} \mathbf{U}_{\mathbf{ij}} \dot{\mathbf{q}}_{\mathbf{j}} \Big)\ {}^{\mathbf{i}}\mathbf{r}_{\mathbf{i}} \Big(\Big(\dot{\mathbf{A}}_{\mathbf{b}}\ {}^{\mathbf{0}}\mathbf{A}_{\mathbf{i}} + \mathbf{A}_{\mathbf{b}} \textstyle\sum_{\mathbf{j=1}}^{\mathbf{i}} \mathbf{U}_{\mathbf{ij}} \dot{\mathbf{q}}_{\mathbf{j}} \Big)\ {}^{\mathbf{i}}\mathbf{r}_{\mathbf{i}} \Big)^{\mathbf{T}} \Bigg) \mathbf{dm} \tag{8}$$

After integrating over the link volume and simplifying total kinetic energy of system comes to Eq. 9

$$\begin{aligned} \mathbf{K} = & \sum_{\mathbf{i=1}}^{\mathbf{n}} \frac{\mathbf{1}}{\mathbf{2}} \mathbf{Tr}(\dot{\mathbf{A}}_{\mathbf{b}}\ {}^{\mathbf{0}}\mathbf{A}_{\mathbf{i}} \mathbf{J}_{\mathbf{i}}\ {}^{\mathbf{0}}\mathbf{A}_{\mathbf{i}}^{\mathbf{T}} \dot{\mathbf{A}}_{\mathbf{b}}^{\mathbf{T}}) \\ & + \sum_{\mathbf{i=1}}^{\mathbf{n}} \sum_{\mathbf{r=1}}^{\mathbf{i}} \mathbf{Tr}(\dot{\mathbf{A}}_{\mathbf{b}}\ {}^{\mathbf{0}}\mathbf{A}_{\mathbf{i}} \mathbf{J}_{\mathbf{i}}\ \mathbf{U}_{\mathbf{ir}}^{\mathbf{T}} \dot{\mathbf{A}}_{\mathbf{b}}^{\mathbf{T}})\ \dot{\mathbf{q}}_{\mathbf{r}} \\ & + \frac{\mathbf{1}}{\mathbf{2}} \sum_{\mathbf{i=1}}^{\mathbf{n}} \sum_{\mathbf{r=1}}^{\mathbf{i}} \sum_{\mathbf{s=1}}^{\mathbf{i}} \mathbf{Tr}\big(\mathbf{A}_{\mathbf{b}} \mathbf{U}_{\mathbf{ir}} \mathbf{J}_{\mathbf{i}} \mathbf{U}_{\mathbf{is}}^{\mathbf{T}} \mathbf{A}_{\mathbf{b}}^{\mathbf{T}}\big) \dot{\mathbf{q}}_{\mathbf{r}} \dot{\mathbf{q}}_{\mathbf{s}} \end{aligned} \tag{9}$$

2) *Potential energy*

Potential energy of each link can be defined as:

$$\mathbf{P_j} = -\mathbf{m_j g}\big(\mathbf{A_b}\ {}^{\mathbf{0}}\mathbf{A_i R_j}\big) \tag{10}$$

Where

$\mathbf{m_j}$: is mass of each link

$\boldsymbol{g}$: is earth gravity

$\mathbf{R_j} = {}^{\mathbf{i}}\mathbf{r_i}$: is a point within the *i*-th link expressed in the *i*-th link coordinates

While the total potential energy of system becomes

$$\mathbf{P} = -\sum_{\mathbf{j=1}}^{\mathbf{n}} \mathbf{m_j g}\big(\mathbf{A_b}\ {}^{\mathbf{0}}\mathbf{A_j R_j}\big) \tag{11}$$

3) *Lagrange function*

Lagrange function simplifies to Eq. 12

$$\begin{aligned} \boldsymbol{L} = & \sum_{\mathbf{i=1}}^{\mathbf{n}} \frac{\mathbf{1}}{\mathbf{2}} \mathbf{Tr}(\dot{\mathbf{A}}_{\mathbf{b}}\ {}^{\mathbf{0}}\mathbf{A}_{\mathbf{i}} \mathbf{J}_{\mathbf{i}}\ {}^{\mathbf{0}}\mathbf{A}_{\mathbf{i}}^{\mathbf{T}} \dot{\mathbf{A}}_{\mathbf{b}}^{\mathbf{T}}) \\ & + \sum_{\mathbf{i=1}}^{\mathbf{n}} \sum_{\mathbf{r=1}}^{\mathbf{i}} \mathbf{Tr}(\dot{\mathbf{A}}_{\mathbf{b}}\ {}^{\mathbf{0}}\mathbf{A}_{\mathbf{i}} \mathbf{J}_{\mathbf{i}}\ \mathbf{U}_{\mathbf{ir}}^{\mathbf{T}} \dot{\mathbf{A}}_{\mathbf{b}}^{\mathbf{T}})\ \dot{\mathbf{q}}_{\mathbf{r}} \\ & + \frac{\mathbf{1}}{\mathbf{2}} \sum_{\mathbf{i=1}}^{\mathbf{n}} \sum_{\mathbf{r=1}}^{\mathbf{i}} \sum_{\mathbf{s=1}}^{\mathbf{i}} \mathbf{Tr}(\mathbf{A}_{\mathbf{b}} \mathbf{U}_{\mathbf{ir}} \mathbf{J}_{\mathbf{i}} \mathbf{U}_{\mathbf{is}}^{\mathbf{T}} \mathbf{A}_{\mathbf{b}}^{\mathbf{T}})\ \dot{\mathbf{q}}_{\mathbf{r}} \dot{\mathbf{q}}_{\mathbf{s}} \\ & + \sum_{\mathbf{j=1}}^{\mathbf{n}} \mathbf{m_j g}(\mathbf{A_b}\ {}^{\mathbf{0}}\mathbf{A_j R_j}) \end{aligned} \tag{12}$$

After calculating $\frac{\partial L}{\partial \dot{q}_i}, \frac{\partial L}{\partial q_i}\ \ i = 1,2, \dots, n$ the torques applied to the system at joint *i* are

$$\begin{aligned} \boldsymbol{\tau_i} = & \sum_{\mathbf{j=i}}^{\mathbf{n}} \mathbf{Tr}(\ddot{\mathbf{A}}_{\mathbf{b}}\ {}^{\mathbf{0}}\mathbf{A}_{\mathbf{j}}\ \mathbf{J}_{\mathbf{j}} \mathbf{U}_{\mathbf{ji}}^{\mathbf{T}} \mathbf{A}_{\mathbf{b}}^{\mathbf{T}}) \\ & + \mathbf{2} \sum_{\mathbf{j=i}}^{\mathbf{n}} \sum_{\mathbf{r=1}}^{\mathbf{j}} \mathbf{Tr}(\dot{\mathbf{A}}_{\mathbf{b}} \mathbf{U}_{\mathbf{jr}} \mathbf{J}_{\mathbf{j}} \mathbf{U}_{\mathbf{ji}}^{\mathbf{T}} \mathbf{A}_{\mathbf{b}}^{\mathbf{T}})\ \dot{\mathbf{q}}_{\mathbf{r}} \\ & + \sum_{\mathbf{j=i}}^{\mathbf{n}} \sum_{\mathbf{r=1}}^{\mathbf{j}} \sum_{\mathbf{s=1}}^{\mathbf{j}} \mathbf{Tr}(\mathbf{U}_{\mathbf{jrs}}\ \mathbf{J}_{\mathbf{j}}\ \mathbf{U}_{\mathbf{ji}}^{\mathbf{T}})\dot{\mathbf{q}}_{\mathbf{r}} \dot{\mathbf{q}}_{\mathbf{s}} \\ & + \sum_{\mathbf{j=i}}^{\mathbf{n}} \sum_{\mathbf{r=1}}^{\mathbf{j}} \mathbf{Tr}(\mathbf{U}_{\mathbf{jr}} \mathbf{J}_{\mathbf{j}} \mathbf{U}_{\mathbf{ji}}{}^{\mathbf{T}})\ \ddot{\mathbf{q}}_{\mathbf{r}} \\ & + \sum_{\mathbf{j=i}}^{\mathbf{n}} \mathbf{m_j g}(\mathbf{A_b U_{ji} R_j}) \end{aligned} \tag{13}$$

$$\mathbf{D} = \sum_{\mathbf{j=i}}^{\mathbf{n}} \sum_{\mathbf{r=1}}^{\mathbf{j}} \mathbf{Tr}(\mathbf{U}_{\mathbf{jr}} \mathbf{J}_{\mathbf{j}} \mathbf{U}_{\mathbf{ji}}{}^{\mathbf{T}}) \tag{14}$$

D is inertial acceleration-related symmetric matrix (independent of base motion).

$$\mathbf{h} = \sum_{\mathbf{j=i}}^{\mathbf{n}} \sum_{\mathbf{r=1}}^{\mathbf{j}} \mathbf{Tr}(\mathbf{U}_{\mathbf{jrs}}\ \mathbf{J}_{\mathbf{j}}\ \mathbf{U}_{\mathbf{ji}}^{\mathbf{T}})\dot{\mathbf{q}}_{\mathbf{r}} \tag{15}$$

h is nonlinear Coriolis and centrifugal force vector (independent of base motion).

$$\mathbf{c} = \sum_{\mathbf{j=i}}^{\mathbf{n}} \mathbf{m_j g}\big(\mathbf{A_b U_{ji} R_j}\big) \tag{16}$$

c is gravity loading force vector.

$$\mathbf{Bmi} = \sum_{j=i}^{n} \mathbf{Tr}(\ddot{\mathbf{A}}_{\mathbf{b}}\ {}^{\mathbf{0}}\mathbf{A}_{\mathbf{j}}\ \mathbf{J}_{\mathbf{j}}\mathbf{U}_{\mathbf{ji}}^{\mathbf{T}}\mathbf{A}_{\mathbf{b}}^{\mathbf{T}}) \tag{17}$$

Bmi is inertial-like and Coriolis and centrifugal-like generalized forces induced by the platform motion on the manipulator.

$$\mathbf{Bmc} = \sum_{j=i}^{n}\sum_{r=1}^{j} \mathbf{Tr}(\dot{\mathbf{A}}_{\mathbf{b}}\mathbf{U}_{\mathbf{jr}}\mathbf{J}_{\mathbf{j}}\mathbf{U}_{\mathbf{ji}}^{\mathbf{T}}\mathbf{A}_{\mathbf{b}}^{\mathbf{T}}) \tag{18}$$

Bmc is Coriolis and centrifugal-like forces induced by the platform and manipulator motion on the manipulator.

$$\boldsymbol{\tau_i} = \mathbf{D}\ddot{\boldsymbol{q}} + \mathbf{h}\dot{\boldsymbol{q}} + \mathbf{c} + \mathbf{Bmi} + \mathbf{Bmc}\dot{\boldsymbol{q}} \tag{19}$$

4) *Base motion*

The base motion of the robot manipulator has not translational motion and is assumed to have only two rotations about X axis roll (α) and Y axis pitch (β) of inertia coordinate.

$$\mathbf{R_b} = \mathbf{Rot}(\mathbf{Y_{inertia}}, \boldsymbol{\beta}).\mathbf{Rot}(\mathbf{X_{inertia}}, \boldsymbol{\alpha})$$

$$\mathbf{R_b} = \begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{bmatrix}. \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix} \tag{20}$$

$$= \begin{bmatrix} \cos(\beta) & \sin(\alpha)\sin(\beta) & \cos(\alpha)\sin(\beta) \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ -\sin(\beta) & \sin(\alpha)\cos(\beta) & \cos(\alpha)\cos(\beta) \end{bmatrix}$$

The above rotation matrix appears in the rotation part of the base motion as:

$$\mathbf{A_b} = \begin{bmatrix} \cos(\beta) & \sin(\alpha)\sin(\beta) & \cos(\alpha)\sin(\beta) & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) & 0 \\ -\sin(\beta) & \sin(\alpha)\cos(\beta) & \cos(\alpha)\cos(\beta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{21}$$

First and second derivatives of the translation matrix are expressed as:

$$\dot{\mathbf{A}}_{\mathbf{b}} = \frac{\mathbf{d}}{\mathbf{dt}}(\mathbf{A_b}) = \frac{\partial \mathbf{A_b}}{\partial \boldsymbol{\alpha}}\dot{\boldsymbol{\alpha}} + \frac{\partial \mathbf{A_b}}{\partial \boldsymbol{\beta}}\dot{\boldsymbol{\beta}} \tag{22}$$

$$\ddot{\mathbf{A}}_{\mathbf{b}} = \frac{\mathbf{d}}{\mathbf{dt}}(\dot{\mathbf{A}}_{\mathbf{b}}) = \frac{\partial^2 \mathbf{A_b}}{\partial \boldsymbol{\alpha}^2}\dot{\boldsymbol{\alpha}}^2 + 2\frac{\partial^2 \mathbf{A_b}}{\partial \boldsymbol{\alpha}\partial \boldsymbol{\beta}}\dot{\boldsymbol{\alpha}}\dot{\boldsymbol{\beta}} + \frac{\partial^2 \mathbf{A_b}}{\partial \boldsymbol{\beta}^2}\dot{\boldsymbol{\beta}}^2 + \frac{\partial \mathbf{A_b}}{\partial \boldsymbol{\alpha}}\ddot{\boldsymbol{\alpha}} + \frac{\partial \mathbf{A_b}}{\partial \boldsymbol{\beta}}\ddot{\boldsymbol{\beta}} \tag{23}$$

In Lagrange-Euler method the base disturbances are imported to dynamic equations as two rotations; (2DOF) of base to inertia translation matrix A_b as sin trajectories expressed in Eq. 24, 25.

$$\boldsymbol{\alpha} = \mathbf{A_{roll}}\,\mathbf{sin}\left(\frac{\mathbf{2\pi t}}{\mathbf{T_{roll}}} + \boldsymbol{\varphi}_{\mathbf{roll}}\right) \tag{24}$$

$$\boldsymbol{\beta} = \mathbf{A_{pitch}}\,\mathbf{sin}\left(\frac{\mathbf{2\pi t}}{\mathbf{T_{pitch}}} + \boldsymbol{\varphi}_{\mathbf{pitch}}\right) \tag{25}$$

2.2. Newton-Euler method

In deriving the dynamic equations through this method, Iterative Newton-Euler formulation is used [8]. In this method constraint forces do not eliminate and they extend through the equations. Knowing the position, velocity and acceleration of joints and also mass distribution and kinematic of links, torques that cause this motion are calculated. There are two steps of calculation:

1) *Iterative outward to compute velocities and accelerations*

To compute forces acting on the links, first rotational and linear velocities and accelerations of links and then forces and torques exerted on the center of mass of each link should be calculated. For this purpose one should iteratively start from link 1 outward to link n and compute the required velocities and accelerations as:

$$^{\mathbf{i+1}}\boldsymbol{\omega}_{\mathbf{i+1}} = {}^{\mathbf{i+1}}_{\mathbf{i}}\mathbf{R}\ {}^{\mathbf{i}}\boldsymbol{\omega}_{\mathbf{i}} + \dot{\boldsymbol{\theta}}_{\mathbf{i+1}}\,{}^{\mathbf{i+1}}\mathbf{Z}_{\mathbf{i+1}} \tag{26}$$

${}^{i+1}\omega_{i+1}$: Angular velocity of link *i+1*

${}^{i+1}_{i}R$: Denavit and Hartenberg (D-H) transformation matrix of the coordinates from *i*-th link to *i+1*-th

${}^{i+1}Z_{i+1}$: Rotation axis component of link *i+1* expressed in *i+1* coordinate

$$^{\mathbf{i+1}}\dot{\boldsymbol{\omega}}_{\mathbf{i+1}} = {}^{\mathbf{i+1}}_{\mathbf{i}}\mathbf{R}\ {}^{\mathbf{i}}\dot{\boldsymbol{\omega}}_{\mathbf{i}} + {}^{\mathbf{i+1}}_{\mathbf{i}}\mathbf{R}\ {}^{\mathbf{i}}\boldsymbol{\omega}_{\mathbf{i}} * \dot{\boldsymbol{\theta}}_{\mathbf{i+1}}\,{}^{\mathbf{i+1}}\mathbf{Z}_{\mathbf{i+1}} + \ddot{\boldsymbol{\theta}}_{\mathbf{i+1}}\,{}^{\mathbf{i+1}}\mathbf{Z}_{\mathbf{i+1}} \tag{27}$$

${}^{i+1}\dot{\omega}_{i+1}$: Angular acceleration of link *i+1*

$$^{\mathbf{i+1}}\dot{\mathbf{v}}_{\mathbf{i+1}} = {}^{\mathbf{i+1}}_{\mathbf{i}}\mathbf{R}(\ {}^{\mathbf{i}}\dot{\boldsymbol{\omega}}_{\mathbf{i}} * {}^{\mathbf{i}}\mathbf{P}_{\mathbf{i+1}} + {}^{\mathbf{i}}\boldsymbol{\omega}_{\mathbf{i}} * (\ {}^{\mathbf{i}}\boldsymbol{\omega}_{\mathbf{i}} * {}^{\mathbf{i}}\mathbf{P}_{\mathbf{i+1}}) + {}^{\mathbf{i}}\dot{\mathbf{v}}_{\mathbf{i}}) \tag{28}$$

${}^{i+1}\dot{v}_{i+1}$: Linear acceleration of each link-frame origin

${}^{i}P_{i+1}$: Position vector of *i+1* coordinate origin with respect to *i* coordinate origin expressed in *i* coordinate

$$\mathbf{{}^{i+1}\dot{v}c_{i+1} = {}^{i+1}\dot{\omega}_{i+1} * {}^{i+1}Pc_{i+1} + {}^{i+1}\omega_{i+1} * ({}^{i+1}\omega_{i+1} * {}^{i+1}Pc_{i+1}) {}^{i+1}\dot{v}_{i+1}} \tag{29}$$

${}^{i+1}\dot{v}c_{i+1}$: Linear acceleration of the center of mass of each link

${}^{i+1}Pc_{i+1}$: Position vector of center of mass of link *i+1* expressed in *i+1* coordinate

Force and torque acting on a link could be extracted from Eq. 30

$$\mathbf{{}^{i+1}F_{i+1} = m_{i+1}{}^{i+1}\dot{v}c_{i+1}} \tag{30}$$

${}^{i+1}F_{i+1}$: Inertial force acting at center of mass of each link

m_{i+1}: Mass of each link

$$\mathbf{{}^{i+1}N_{i+1} = {}^{C_{i+1}}I_{i+1}{}^{i+1}\dot{\omega}_{i+1} + {}^{i+1}\omega_{i+1} * {}^{C_{i+1}}I_{i+1}{}^{i+1}\omega_{i+1}} \tag{31}$$

${}^{i+1}N_{i+1}$: Inertial torque acting at center of mass of each link

${}^{C_{i+1}}I_{i+1}$: The Inertia tensor written in the frame at center of mass

2) *Iterative inward to compute force and torque*

Having computed the force and torque acting on each link, joint torques need to be calculate. Next equations are evaluated link by link from link n inward to base of robot Eq. 32 to Eq. 34.

$$\mathbf{{}^{i}f_{i} = {}_{i+1}^{i}R\,{}^{i+1}f_{i+1} + {}^{i}F_{i}} \tag{32}$$

${}^{i}f_{i}$: Force exerted on link *i* by link *i-1*

$$\mathbf{{}^{i}n_{i} = {}^{i}N_{i} + {}_{i+1}^{i}R\,{}^{i+1}n_{i+1} + {}^{i}Pc_{i} * {}^{i}F_{i} + {}^{i}P_{i+1} * {}_{i+1}^{i}R\,{}^{i+1}f_{i+1}} \tag{33}$$

${}^{i}n_{i}$: Torque exerted on link *i* by link *i-1*

$$\boldsymbol{\tau}_{\mathbf{i}} = \mathbf{{}^{i}n_{i}^{T}\;{}^{i}Z_{i}} \tag{34}$$

τ_i: Joint torques

3) *Base motion*

In Newton-Euler method base rotational motions are imported as base parameters. Since it doesn't have translational motion, the base parameter becomes

$$\mathbf{{}^{0}w_{0}} = \begin{bmatrix}\boldsymbol{\alpha}\\ \boldsymbol{\beta}\\ \mathbf{0}\end{bmatrix},\ \mathbf{{}^{0}\dot{w}_{0}} = \begin{bmatrix}\dot{\boldsymbol{\alpha}}\\ \dot{\boldsymbol{\beta}}\\ \mathbf{0}\end{bmatrix},\ \mathbf{{}^{0}V_{0}} = \begin{bmatrix}\mathbf{0}\\ \mathbf{0}\\ \mathbf{0}\end{bmatrix} \tag{35}$$

2.3. Formula derivation

Formulas that are derived in both approaches describe the platform and manipulator arm dynamic. These are used to create codes in Maple software and simulated which are described in next section.

3. Dynamic Implementation on Mitsubishi PA10-6CE Robotic Manipulator Mounted on a Ship Platform

Dynamic derivation analyses in different references are in various coordinate attachments, notations, etc. Therefore for each of the methods different simulation techniques are employed. In this study the derived equations are simulated for Mitsubishi PA10-6CE which is 6-DOF revolute manipulator. The picture of this robot is shown in figure 1. Depending on method and its corresponding coordinated assignment technique, the inertial matrices are different for each link, it is mentionable that these discrepancies have been considered in simulation of this manipulator.

Figure 1. Mitsubishi PA10-6CE robotic manipulator [14]

3.1. Lagrange-Euler method

In this method at the beginning $\boldsymbol{{}^{0}A_i}$ is calculated which is Denavit and Hartenberg (D-H) transformation matrix of the coordinates from *i*-th link to base. According to this coordinate assignment is done on robot manipulator through the way which is expressed in [12]. We can see the coordinates assigned to robot manipulator in figure 2 (right) and Denavit and Hartenberg (D-H) table 1.

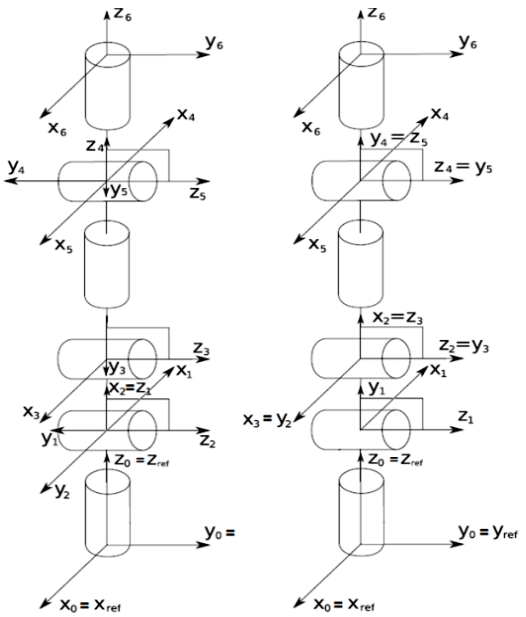


Figure 2. Coordinate assignment for Mitsubishi PA10-6CE robotic manipulator in Lagrange(right) and Newton(left) methods

Table 1. Denavit and Hartenberg (D-H) table for Lagrange method

Link number	α_i	a_i	d_i	θ_i
1	90	0	0.317	$\theta_1+\pi$
2	0	0.45	0	$\theta_2+\pi/2$
3	90	0	0	$\theta_3+\pi/2$
4	90	0	0.48	$\theta_4+\pi$
5	90	0	0	$\theta_5+\pi$
6	0	0	0.07	θ_6

In the next step inertias $(\mathbf{J_i})$, masses $(\mathbf{m_i})$ and vector of center of mass $(\mathbf{R_i})$ of links are imported to equations as company specifications of Mitsubishi PA10-6CE robot manipulator. Inertia term in both Lagrange and Euler methods are defined related to distinguished attached coordinates.

1) *Base motion*

After all transformation matrix of the base to the inertia coordinate should be calculated. For this aim base motions should be known (α , β) to calculate $\boldsymbol{A_b}$, α and β are the base motion parameters. When a ship is moving on the ocean, waves of water apply disturbances on it and due to the fact that the manipulator is placed on the ship the wave influences on manipulator's motion. There have been many researches on the ocean waves and frequency of waves which reaches to the surface of ship. The frequency depends on size and weight of the ships figure 3 [4]. Here specific type of ship is surveyed as an instance table 2.

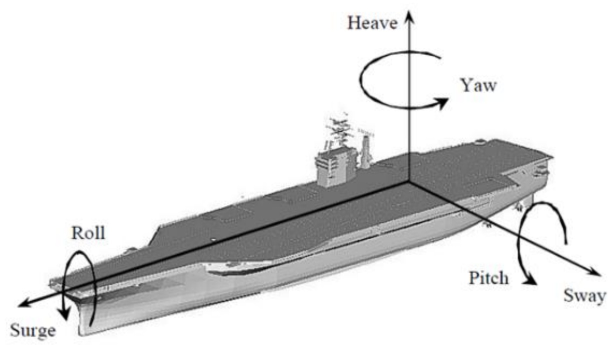


Figure 3. Ship displacements [4]

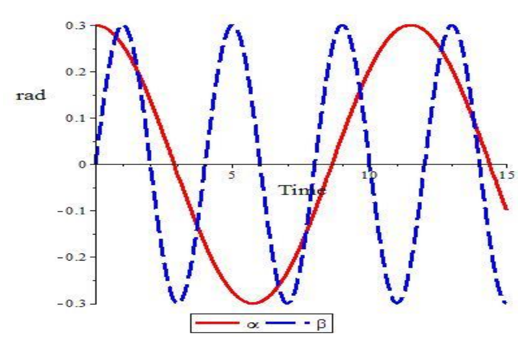
Table 2. Ship frequencies resonse as function of sea state

Parameter	Boat length (m)	Boat heave (m)	Pitch period (sec)	Roll period (Sec)
Value	45.72-76.2	0.67	4	11.5

For a boat with mentioned specifications, waves are acting with different periods in different directions, amplitudes specifications that are considered in roll and pitch frequencies on surface of ship are expressed in table 3.

Table 3. Parameters of base sinusoids wave frequency [4]

Parameter	A_{roll}	A_{pitch}	T_{roll}	T_{pitch}	φ_{roll}	φ_{pitch}
Value	0.3	0.3	11.5	4	$\pi/2$	0



2) *Gravity*

Gravity terms which influences the inertia of links is entered as

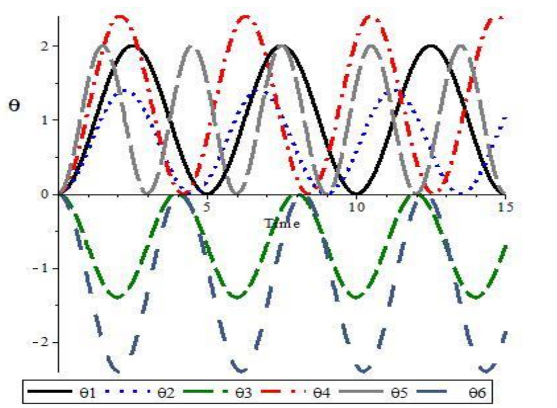
$$\mathbf{g} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ -\mathbf{9.8062} \\ \mathbf{0} \end{bmatrix} \quad (36)$$

Sinusoidal trajectory followed by all manipulator joints has been employed in all simulations table 4.

$$\boldsymbol{\theta_i = A_i \cos(\omega_i t) - 1} \qquad \boldsymbol{i = 1 \ldots 6} \quad (37)$$

Table 4. Joints sin trajectories specifications

	Joint 1	Joint 2	Joint 3	Joint 4	Joint 5	Joint 6
A_i	-1	-0.7	0.7	-1.2	-1	12
ω_i	$2\pi/5$	$2\pi/4.5$	$2\pi/4$	$2\pi/4.2$	$2\pi/3$	$2\pi/4.1$



3.2. Newton-Euler method

In this method like Lagrange method Denavit and Hartenberg (D-H) transformation matrix of the coordinates from *i*-th link to *i*+1-*th* link $^{i+1}_{\ i}\mathbf{R}$ are calculated and coordinate assignment is performed by manner expressed [13]. Figure 2 (left) and Denavit and Hartenberg (D-H) table 5.

Table 5. Denavit and Hartenberg (D-H) table for Newton method

Link number	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0.317	$\theta_1 + \pi$
2	90	0	0	$\theta_2 + \pi/2$
3	0	0.45	0	$\theta_3 + \pi/2$
4	90	0	0.48	$\theta_4 + \pi$
5	90	0	0	$\theta_5 + \pi$
6	90	0	0.07	θ_6

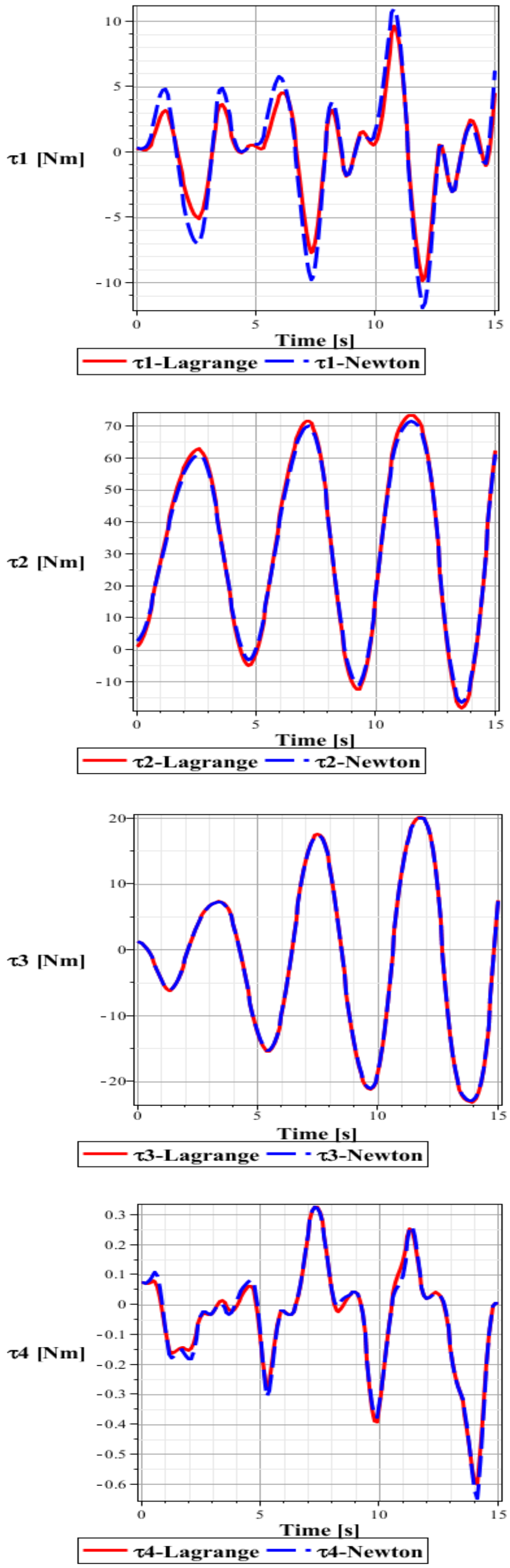
In the next step rotation axis $(^{i+1}\mathbf{Z}_{i+1})$, position vector of coordinate $(\,^{i}\mathbf{P}_{i+1})$, Position vector of center of mass $(^{i+1}\mathbf{Pc}_{i+1})$, link mass $(\mathbf{m_i})$, inertia of links about center of mass $(^{C_{i+1}}\mathbf{I}_{i+1})$ are calculated from geometry and mass distribution of links and are imported to computations.

3) *Base motion*

In this method base motion are imported in calculations different from the Lagrange method as Eq. 35 specifies 2-DOF motion of the base.

4. Result and Discussion

The dynamic equation of a manipulator arm with base motion is derived in two different methods. The derived equations of motion are simplified sufficiently and are solved in time domain with 100HZ frequency in Maple v15 software. In the dynamic simulations, the full capacity of an Intel PC core i5 CPU 2.8 GHz with 8 MB cash and 4 GB RAM is employed. Calculations in Newton method lasted 27.5 (sec) and it was increased to 57.53 (sec) in the Lagrange method due to more complicated equations. The results are shown and compared in Figure 4, and also the torque error between two methods is calculated and shown in Figure 5. It is observable that there are negligible differences in the results between two methods. Further, it is shown that the torque error between two methods is decreased from the base of the robot towards the end effector (from link 1 to link 6). At the end it is shown that the Lagrange method is better candidate to be used in cases where the base of the robot has translational and rotational acceleration and velocity motion simultaneously; however, it needs more calculation time. Further, it is demonstrated that the Newton-Euler method is easier to use for cases where there is only the rotational base motion and it needs less calculation time.



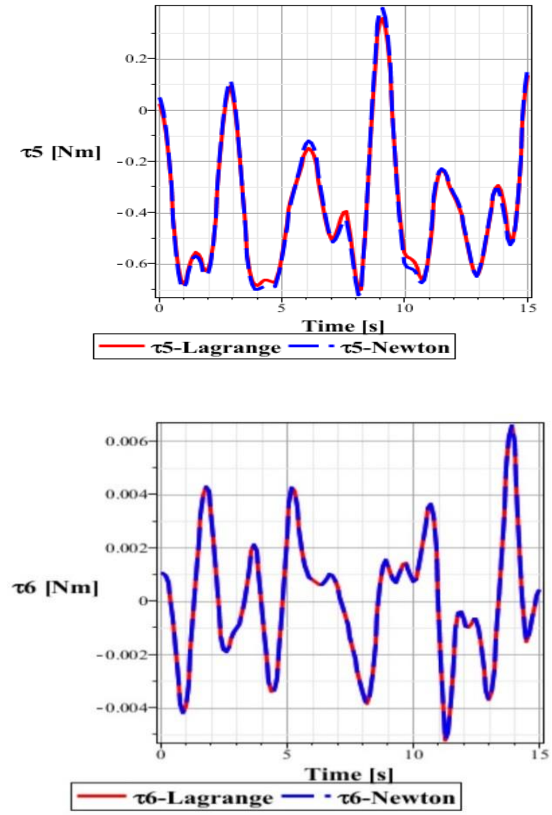


Figure 4. Joint torques under base motion

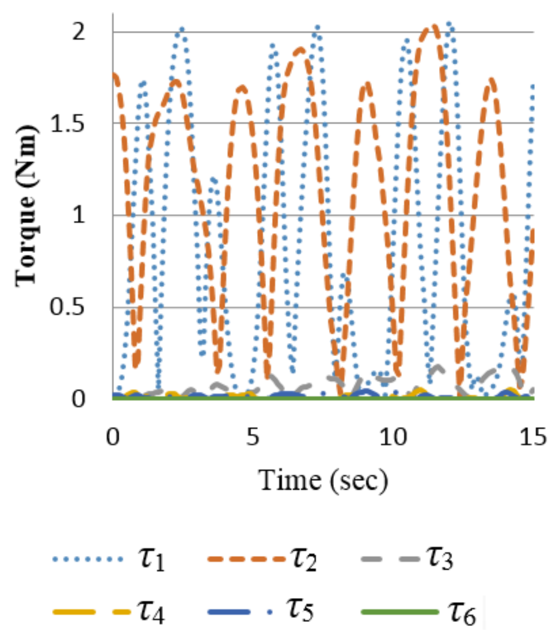


Figure 5: Joint Torque Errors between Newton Euler and Lagrange Methods

Reference

[1] J. Joshi and A. Desrochers, “Modeling and control of a mobile robot subject to disturbances”, Robotics and Automation. Proceedings. 1986 IEEE International Conference on, (1986), pp. 1508–1513.

[2] R. Jamisola, “Full dynamics identification and control of PUMA 560 and Mitsubishi PA-10 robots”, Master’s Thesis. National University of Singapore, Department of Mechanical and Production Engineering, (2001).

[3] N. Bompos, P. Artemiadis, A. Oikonomopoulos, K. Kyriakopoulos, “Modeling, full identification and control of the Mitsubishi PA-10 robot arm”, in: Advanced intelligent mechatronics, 2007 IEEE/ASME International Conference on, (2007), pp. 1–6.

[4] A. B. Tanner, “Study of robotic manipulators subjected to base disturbances”, Master’s Thesis. Army Military Personnel Center Alexandria VA, (1987).

[5] F. M. Carter, D. B. Cherchas, “Motion control of non-fixed base robotic manipulators”, Robotica, Vol. 17, (1999), pp. 143–157.

[6] L. Love, J. Jansen and F. Pin, “On the modeling of robots operating on ships”, in Robotics and Automation, 2004 Proceedings, ICRA’04, 2004 IEEE International Conference on, (2004), pp. 2436–2443.

[7] L. Love, J. Jansen, F. Pin, “Compensation of wave-induced motion and force phenomena for ship-based high performance robotic and human amplifying systems”, Technical Report ORNL/TM-2003/233. Oak Ridge National Laboratory, Oak Ridge, Tenn, (2003).

[8] P. J. From, V. Duindam, J. T. Gravdahl, S., “Modeling and motion planning for mechanisms on a non-inertial base”, in: Proceedings of the IEEE International Conference on Robotics and Automation, (2009), pp. 3320–3326.

[9] S. Dubowsky and E. Papadopoulos, “The Kinematic, Dynamic and Control of Free Flying Systems” IEEE Transactions on Robotic and Automation, Vol. 9(5), (1993).

[10] S. Murphy, J. Y. Wen, G. Saridis, “Simulation of cooperating robot manipulators on a mobile platform”, *Robotics and Automation, IEEE Transactions on*, (1991), pp. 468–478.

[11] C. M. Wronka, M. W. Dunnigan, “Derivation and analysis of a dynamic model of a robotic manipulator on a moving base”, *Robotics and Autonomous Systems*, Vol. 59 (2011), pp. 758–769.

[12] K. Fu, R. Gonzales, C. Lee, ROBOTICS, in: Control, Sensing, Vision and Intelligence, McGraw-Hill, 1987.

[13] John J. Craig: Introduction to Robotic Mechanics and Control.

[14] http://tams-www.informatik.uni-hamburg.de/research/robotics/ /hardware/index.php?content= manipulator.

Biography

Majid. M. Moghaddam, received the PhD degree in mechanical engineering from the University of Toronto, Canada, in 1996. He is a professor of mechanical engineering at the Tarbiat Modares University, Tehran, Iran. His current research, which focuses on applied robotics and robust control, is concerned with haptic robotics, rehabilitation robotics, inspection robotics, and rough terrain mobile robot design. He is a member of the Administrative Committee of Robotics and Mechatronics Societies of Iran.

Ehsan Sadraei received his B.Sc and M.Sc degrees in mechanical engineering from Iran University of Science and Technology and Tarbiat Modares University, respectively, both in Tehran, Iran. His master research was focused on haptic robotic simulation of soft tissue deformation. In 2014 Ehsan started his PhD in Institute for Transport Studies (ITS) in University of Leeds, England. His current research is on motion cuing fidelity of simulators and its effect on driver behavior.