Uludağ University Journal of The Faculty of Engineering, Vol. 23, No. 1, 2018 RESEARCH

DOI: 10.17482/uumfd.313881

MOTION OPTIMIZATION OF A 6 AXES ROBOT MANIPULATOR USED FOR MATERIAL HANDLING PURPOSES VIA DYNAMIC PROGRAMMING

Aytaç GÖREN [*]
Umut ÇAKIR [**]

Received: 16.05.2017; revised: 15.11.2017; accepted: 02.03.2018

Abstract: Due to the increasing cost of energy besides the environmental concerns, energy consumption is one of the hot topic in today's world. In this context, companies are searching for the ways to reduce their energy consumptions and their Carbon footprint. For a manufacturing company, one method to accomplish these is to increase their process efficiency. In this research, a production cell which contains a robot manipulator used for material handling purposes is tried to be modeled in computer environment, using dynamic programming over the computer model, system parameters increasing production capacity in addition to reducing the energy consumption and improvements depending on these are presented.

Keywords: Robot manipulator, motion optimization, dynamic programming, optimal control.

Malzeme Yükleme Amaçlı 6 Eksen Bir Robot Manipülatörün Dinamik Programlama ile Hareket Optimizasyonu

Öz: Çevresel kaygılarına ek olarak enerji maliyetinin artışı, günümüz dünyasında enerji sarfiyatı önemli başlıklardan birisi haline gelmiştir. Bu bağlamda, şirketler de enerji sarfiyatlarını ve karbon ayak izlerini düşürme yönünde araştırmalar yapmaktadırlar. Üretim yapan bir şirket için bu konuda başarı elde etmenin bir yöntemi verimliliklerini arttırmaktan geçmektedir. Bu çalışmada, üretimde malzeme aktarımı görevi olan bir robot manipülatör, bilgisayar yardımıyla modellenmeye çalışılmış, model üzerinde dinamik programlama ile üretim kapasitesinin arttırılması, enerji sarfiyatının düşürülmesini sağlayan sistem parametreleri ile geliştirmeler sunulmaya çalışılmıştır.

Anahtar Kelimeler: Robot manipülatör, hareket optimizasyonu, dinamik programlama, optimal kontrol.

1. INTRODUCTION

In this research, material handling process of a six DOF articulated robot manipulator for transferring a payload from one station to another one is conceived in a computer environment. By taking the productivity concerns into account, the aim is both to reduce the power consumption of robot manipulator while keeping, at worst, the process time as in the previous case. In this regard, for further increase in productivity, path length and path geometry are also the focal point of this analysis in reducing cycle time which also affects the power consumption.

The current process condition involves material transfer along a path having square shape on a horizontal plane in 8s. By changing the geometry and time history of joint variables, both capacity increases thanks to a reduction of process time and energy consumption of robot

[*] Dokuz Eylül Üniversitesi, Makina Mühendisliği Bölümü, 35397, Tınaztepe, Buca, Izmir.
[**] Dokuz Eylül Üniversitesi, Fen Bilimleri Enstitüsü, 35397, Tınaztepe, Buca, Izmir.
Correspondence Author: Aytaç GÖREN (aytac.goren@deu.edu.tr)

manipulator are aimed to be improved. In this context, vertical triangular like path is proposed to improve the process in both capacity and energy consumption.

After completing inverse kinematic analysis of robot manipulator on both current and proposed path, the time history of all joint displacements are obtained. Using differentiation formula, joint velocities and joint accelerations corresponding to each joint are calculated for the two path geometry.

The time history of joint configurations is used in the dynamic analysis of the robot manipulators which leads to the calculation of the required torque at each joint to transfer the part from one station to another station. The proposed path is further analyzed so as to obtain the optimized joint configuration in terms of energy consumption by utilizing from dynamic programming.

2. METHODOLOGY USED

In studies about industrial serial robot manipulators, researches are concentrated on the parameters from industry and this results with optimization of time and energy (Carter, 2009; Meike, 2011; Lewis, 2004). Machine vision (Staniak, 2010; Wang, 2012; Wang, 2012), kinematic analysis and implementation of computer based simulations (Wang, 2012; Rocha, 2011; Ding, 2014) are some topics which are valuable for applications in industry. The methodology steps in this research are written in order to give insight about the analysis and the calculations performed in the first part of the study. This work starts with the inverse kinematic analysis of robot manipulator for the given representative path for the current process condition. Inverse kinematic analysis is performed with Solidworks 2012. After obtaining the time history of joint angles for the current path in Figure 1, which is square-like path on the horizontal plane, joint velocities and joint accelerations are determined. According to these calculated joint variables, dynamic analysis of robot manipulator for the current process situation is completed using Robotic Toolbox in MATLAB. This toolbox provides ease in torque calculation. After having torque values corresponding to each joint configuration, power consumption for the current path is determined.

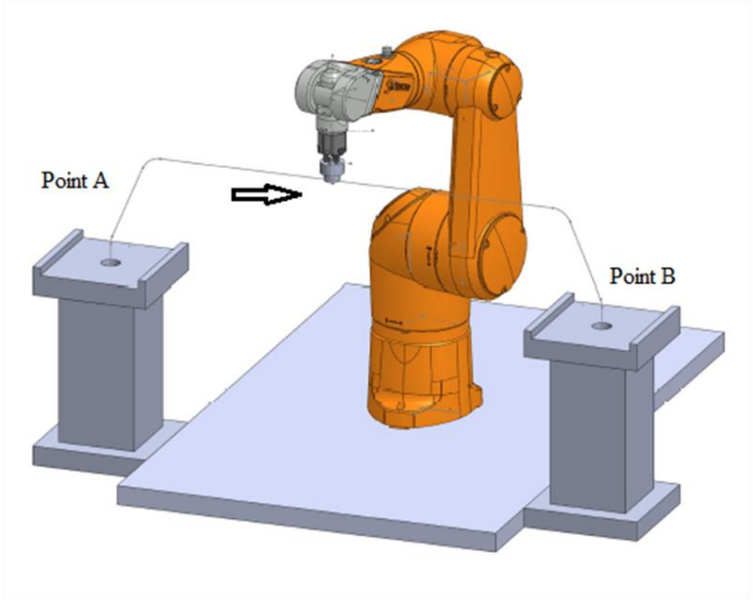


Figure 1:
Current path followed by the end-effector

The above calculation steps are valid for the optimized path calculation (Rodriguez, 2014; Owen, 2008). For the optimization purposes, triangular horizontal path is selected due to the

fact that it is the most energy efficient of all other possible path geometry in addition to the minimum path length which reduces the elapsed time for process (Carter, 2009). Furthermore the optimized path geometry is adjusted in such a way that no sharp corners exist on it. Leaving the corner points for the end effector path results in the stopping of the end effector at the corner point which in turn yields to non-smooth movement of the robot manipulator. In order to prevent this, utilizing from the rounded end effector profile creates smoother and quicker motion (Meike, 2011; Aldana, 2014; Gans, 2012). After defining the candidate optimized path represented in Figure 2, which is triangular horizontal path in this work, the minimum energy consumption path is further analyzed with dynamic programming. Dynamic programming yields to optimal values for joint configuration by taking the minimum energy performance index into account. These optimal values are the joint configuration values which leads to minimum energy consumption among the admissible joint configuration values.

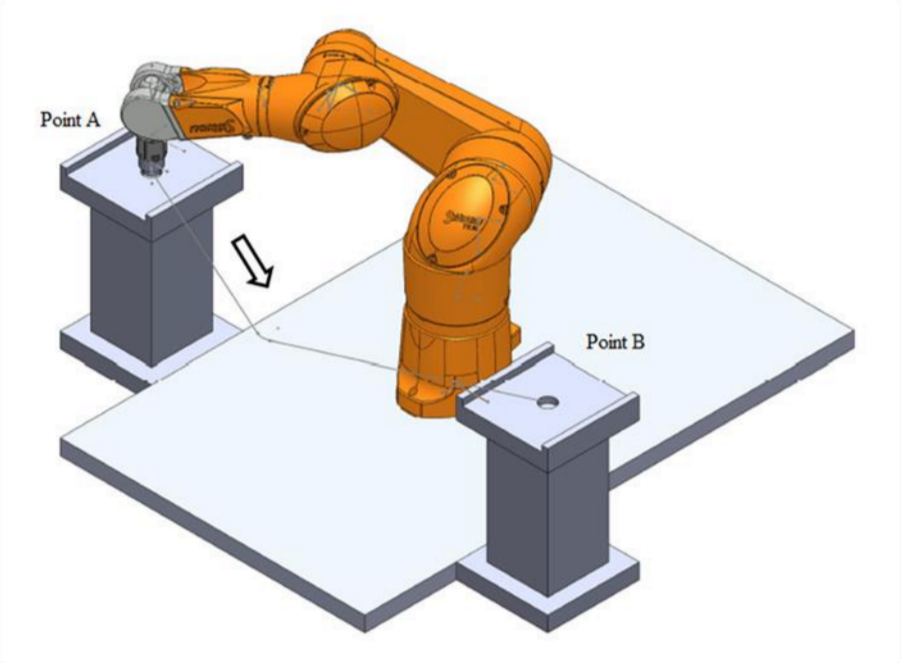


Figure 2:
Optimized path followed by the end-effector

3. MATHEMATICAL MODEL OF THE SYSTEM

3.1 Dynamic Equation of Robot Manipulator

Dynamic equation of robot manipulator can be represented with Euler-Lagrange equations as in equations (1),(2) and (3):

$$M(q).\ddot{q} + V(q,\dot{q}).\dot{q} + G(q) = B.\tau + {J_V}^T.F_{ext} \quad (1)$$

where q and $\dot{q}$ are joint angles and joint velocity of each link of robot manipulator respectively. *M(q)* and *V(q,q)* are the inertia and Coriolis/Centripetal matrix respectively. *G(q)* is the gravity matrix and τ is the required actuator torque to be applied at robot joints, J_V is the Jacobian matrix of the robot manipulator and F_{ext} is the externally applied force on robot links. The following mathematical modeling is based on the study (Aghanouri, 2011) in which DC motor is supposed to be used for the sake of the simplicity in mathematical equations.

$$V_a = i_a.R_a + L_a.\frac{di}{dt} + K_E.\dot{q}_m \quad (2)$$

$$J_m.\ddot{q}_m = K_T.i_a - B_m.\dot{q}_m - \tau \quad (3)$$

where L_a is the inductance of motor windings *(H)*, R_a is the armature resistance (Ω), V_a is the motor voltage (V), $\dot{q}_m$ is the motor velocity (rad/s), i_a is the armature current (A), K_E is the motor velocity constant (V.s/rad), *r* is the gear ratio ( ), *q* is the joint displacement (rad), K_T is the motor torque constant (N.m/A) and B_m is the friction constant of motor (). Equation (3) is the motor side torque equation. J_m is the motor inertia (kg.m2).

In this work, since $L_a << R_a$, motor inductance is ignored. By combining equation (2) and (3), we obtain:

$$J_m.r.\ddot{q} = \frac{K_T.V_a}{R_a} - (\frac{K_T.K_E+R_a.B_m}{R_a}).r.\dot{q} - \tau \quad (4)$$

Using $\dot{q}_m = \dot{q}.r$ and $\ddot{q}_m = \ddot{q}.r$, Equation (4) is further simplified into equation (5):

$$J_m.r.\ddot{q} = \frac{K_T.V_a}{R_a} - (\frac{K_T.K_E+R_a.B_m}{R_a}).r.\dot{q} - \tau \quad (5)$$

Equating equation(1) and (5) after remaining each τ term alone in the equation gives:

$$\frac{K_T.V_a}{R_a} - (\frac{K_T.K_E+R_a.B_m}{R_a}).r.\dot{q} - J_m.r.\ddot{q} = B^{-1}.(M(q).\ddot{q} + V(q,\dot{q}).\dot{q} + G(q) - {J_V}^T.F_{ext}) \quad (6)$$

3.2 Control Equation of Robot Manipulator

Computed torque control is a special application of feedback linearization of nonlinear systems. PID type completed torque control is selected for the controlling of robot manipulator since it includes zero steady state error and no stability problems. Furthermore, as a result of the unknown parameter existence such as link inertia, PID type controller eliminates the error resulting from this. In this way, computed torque control law is expressed by the summation of feed forward and feedback torque component (Murray, 1994) as follows:

$$\tau = \tilde{\tau} + \check{\tau} \quad (7)$$

where the feed forward component is:

$$\tilde{\tau} = B^{-1}.(M(q).\ddot{q}_d + V(q,\dot{q}).\dot{q} + G(q) - {J_V}^T.F_{ext}) \quad (8)$$

and the feedback component is:

$$\hat{\tau} = B^{-1}.M(q).(K_d.\dot{e} + K_p.e + K_i.\varepsilon) \quad (9)$$

in which q_d , $\dot{q}_d$, $\ddot{q}_d$ are the desired robot parameters, $e = q_d - q$ is the error, $\dot{e} = \dot{q_d} - \dot{q}$ is the derivative of error, K_d, K_p and K_i are the derivative, proportional and integral gains respectively.

Work actuators can be modeled as voltage source instead of being torque source (Ding, 2014). In this context, feed forward and the feedback equations are written as follows:

$$\tilde{V}_a = R_a/(B.K_T).\{(M(q) + J_m.B.r).\ddot{q}_d + (V(q,\dot{q}) + ((B.K_T.K_E + B.R_a.B_m)/R_a.r).\dot{q} + G(q) - {J_V}^T.F_{ext}\} \quad (10)$$

$$\hat{V}_a = R_a/(B.K_T).(M(q) + J_m.B.r).(K_d.\dot{e} + K_p.e + K_i.\varepsilon) \quad (11)$$

The actual voltage is expressed as:

$$V_a = \widetilde{V}_a + \widehat{V}_a \tag{12}$$

which is expanded into equation (13) as given below:

$$V_a = R_a/(B.K_T).\{(M + J_m.B.r).(\ddot{q}_d + K_d.\dot{e} + K_p.e + K_i.\varepsilon) + (V + C).\dot{q} + G - J_V{}^T.F_{ext}\} \tag{13}$$

where $C = (B.K_T.K_E.r + B.R_a.B_m.r)/R_a$.

4. OPTIMIZATION VIA DYNAMIC PROGRAMMING

Dynamic programming is applied for optimizing the robot parameters in our pick and place applications for the time optimized triangular horizontal path. This method is based on discretization of the equation and using the quantized state and input values. DP procedure begins with going from the last stage and iterating the equations using the quantized state and input values.

In this regard, the first thing to be used is the linearization of the nonlinear system equations into the following form:

$$\dot{X} = f(X(t), u(t), t) \tag{14}$$

Using the difference equations for the first derivative term in equation (14), equation (15) is obtained as follows:

$$(X(t + \Delta t) - X(t))/\Delta t = f(X(t), u(t), t) \tag{15}$$

substituting $t = k.\Delta t$ in equation (15) gives:

$$X(k + 1) = f(X(k), u(k), k).\Delta t + X(k) \tag{16}$$

For a 6 DOF robot manipulator, joint displacement and joint velocity are written as:

$$x_i(t) = \begin{cases} q_i(t), & i = 1,2,..,6 \\ \dot{q}_i(t), & i = 7,8,..,12 \end{cases}$$

The control input which is the actuator voltage is written as follows:

$$u(k) = [V_{a1}(k), V_{a2}(k), V_{a3}(k), V_{a4}(k), V_{a5}(k), V_{a6}(k)]^T \tag{17}$$

The state variables are denoted as X column vector and represented as:

$$X(k) = [x_1(k), x_2(k), \dots, x_6(k), x_7(k), \dots., x_{11}(k), x_{12}(k)]^T \tag{18}$$

Putting equation (17) and (18) into equation (16) gives:

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ . \\ . \\ x_{11}(k+1) \\ x_{12}(k+1) \end{bmatrix} = \begin{bmatrix} f_{11} \\ f_{21} \end{bmatrix}.\Delta t + \begin{bmatrix} x_1(k) \\ x_2(k) \\ . \\ . \\ x_{11}(k) \\ x_{12}(k) \end{bmatrix} \tag{19}$$

where f_{11} and f_{21} are the following column vectors:

$$f_{11} = [x_7(k), \dots., x_{11}(k), x_{12}(k)]^T \tag{20}$$

$$f_{21} = (M([x_1(k), .., x_6(k)]) + J_m.B.r)^{-1}.(A.[V_{a1}(k), ., V_{a6}(k)] - (V([x_1(k), .., x_6(k)], [x_7(k), .., x_{12}(k)]) + C).[x_7(k), .., x_{12}(k)] - G([x_1(k), .., x_6(k)]) + J_V{}^T.F_{ext}) \tag{21}$$

where and $A = (B.K_T)/R_a$ and $C = (B.K_T.K_E.r + B.R_a.B_m.r)/R_a$

By using equation (13), the control input is expressed as:

$$u(k) = A^{-1}.\{(M([x_1(k),..,x_6(k)]) + J_m.B.r).(\ddot{q}_d(k) + K_D.(\dot{q}_d(k) - [x_7(k),..,x_{12}(k)]) + K_p.(q_d(k) - [x_1(k),..,x_6(k)]) + K_i.(q_d(k) - [x_1(k),..,x_6(k)]).\Delta t) + (V([x_1(k),..,x_6(k)],[x_7(k),..,x_{12}(k)]) + C).[x_7(k),..,x_{12}(k)] + G([x_1(k),..,x_6(k)]) - J_V{}^T.F_{ext} \quad (22)$$

Then the state values which are $[x_1(k) \quad x_2(k) \ldots \quad x_6(k)]^T$ and $[x_7(k) \quad x_8(k) \ldots \quad x_{12}(k)]^T$ are quantized into S admissible values. In addition to this, control input values, $[u_1(k) \quad u_2(k) \ldots \quad u_6(k)]^T$ are also quantized into C admissible values.

$$X^{(m)}(k),\ m = 1, 2, \ldots, S \quad (23)$$

$$u^{(j)}(k),\ j = 1, 2, \ldots, C \quad (24)$$

At first, $X^{(m)}(N-1)$ is supposed and putting this assumed state values into $u(k)$ equation, we find $u^{(j)}(N-1)$. Again using both $u^{(j)}(N-1)$ and $X^{(m)}(N-1)$, $X^{(m)}(N)$ is calculated. In a similar iteration procedure, the assumed $X^{(m)}(N-2)$ is used to calculate $u^{(j)}(N-2)$, $u^{(j)}(N-2)$ and $X^{(m)}(N-2)$ are used to obtain $X^{(m)}(N-1)$. This iteration continues until $X^{(m)}(1)$ is gathered.

These obtained values are acceptable if one of them is equal to one of the admissible values which is determined from the performance index selected in accordance with the type of optimization desired to have. Performance index is a mathematical expression which indicates that the system is performing in a desirable way when minimized. In brief, if there are 2 admissible control values which lead the system to work and causing the admissible state values, shown in the Figure 3, the better control value is selected. In this regard, if $J^{(1)} < J^{(2)}$, then $u^{(1)}$ is nominated as better control.

In this analysis, for the minimum power consumption with little deviation from the starting and terminal point of the motion, the following performance index represented in the equation 23 (Kirk, 1998) is used in which k is the stage number, N is the last stage, X(k) is the state values (joint angle/velocity) at kth stage and R(k) is the desired state values at kth stage

$$J = (X^{(m)}(N) - R^{(m)}(N))^2 + \sum_{k=1}^{N}(X^{(m)}(k) - R^{(m)}(k))^2 + \sum_{k=1}^{N}(u^{(j)}(k))^2 \quad (23)$$

In each stage for m and j, different performance index values are calculated. The minimum of these calculated performance index values is selected. The state values corresponding to this minimum performance index denoted by J^* are defined as the optimized robot parameters which yield to minimum power consumption of robot manipulator in pick and place application.

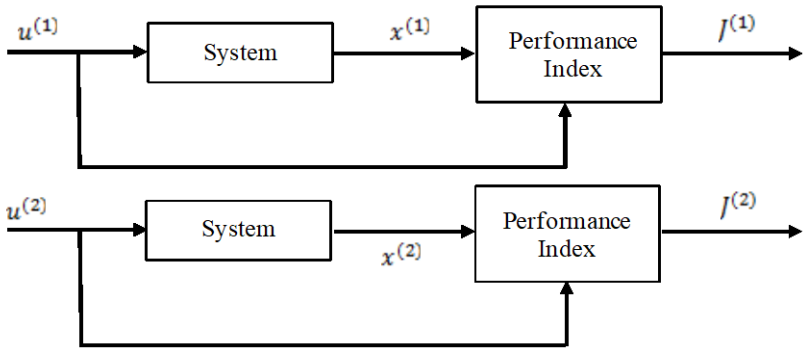


Figure 3:
Two system control pair evaluation using performance index

5. SIX DOF MANIPULATOR RESULTS

5.1. MATLAB Modeling

Modeling of the robot manipulator and the physical parameters used for mathematical calculations are made using Robotic Toolbox in MATLAB.

Time history of joint angle and joint velocity obtained in the analysis of the initial path and the optimized path using the MATLAB are represented in Figure 1 and Figure 2 in this section. Admissible range is selected as small as possible in order not to deviate much from the geometrically optimized path. Furthermore, to eliminate the positional error encountered in this motion analysis, we set the initial and final joint coordinates of the robot manipulator the same as the ones obtained after geometrical optimization.

The optimization procedure yields to the optimal parameters shown in Figure 4 and Figure 5 which reduces the energy consumption of robot manipulator among several admissible joint angle and joint velocity values. The stated energy consumption stems from the mechanical losses due to friction of motors, link inertia and mechanical load to be transferred within work station. Table 1 and Table 2 show the parameter values used to analyze the optimization where m_i is the mass of links, Jm_i is the moment of inertia of motors , I_i is the moment of inertia of links, $rcom_i$ is the center of gravity of links with respect to coordinate frame i, r_i is the gear ratio of motors used to control robot manipulator, R_a is the armatur circuit resistance, L_a is the inductance circuit inductance, B is the viscous friction, K_t is the torque constant, K_p is the proportional constant of PID, K_i is the integral constant of PID, K_d is the derivative constant of PID, K_e is the speed constant of the motor, B_m is the friction constant of motor. Table 3 represents process parameters to make a comparison in between initial condition, simple path geometry optimization and parameter optimization obtained with dynamic programming applied on the geometrically optimized path taking minimum energy performance index into account.

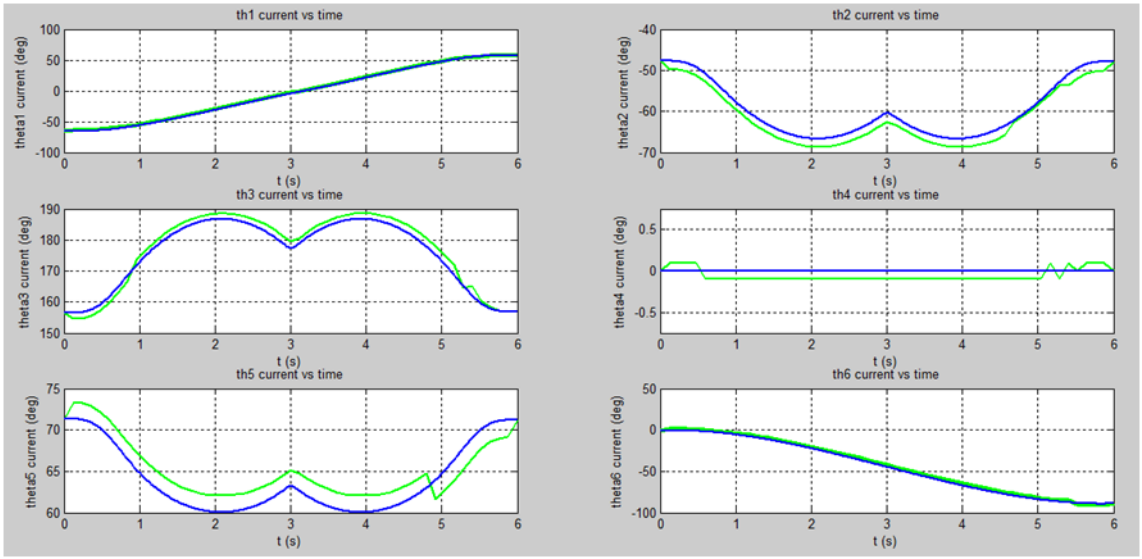


Figure 4:
Joint angle vs. time corresponding to initial path (blue) and optimized path (green)

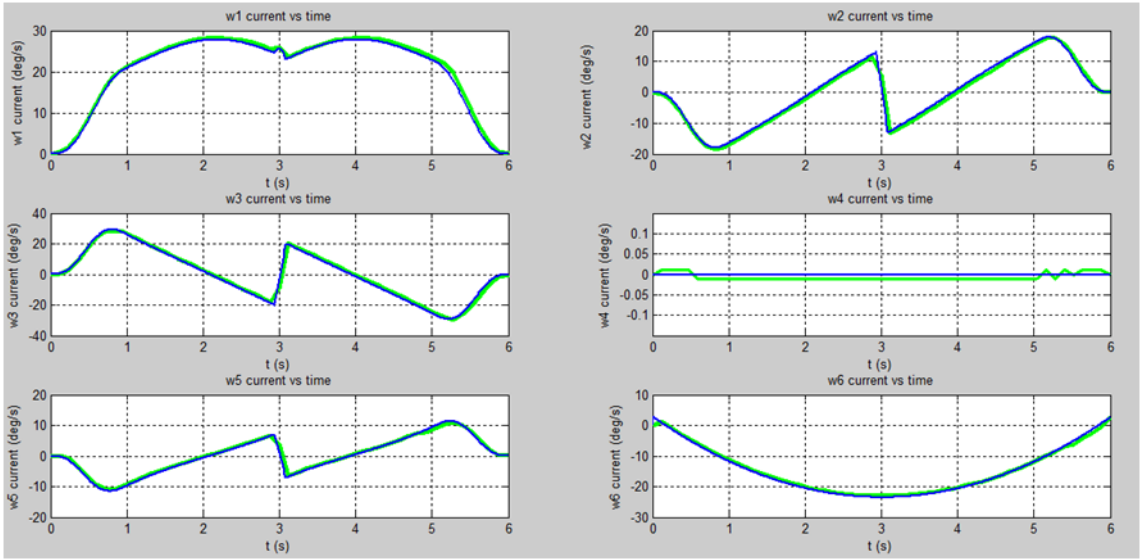


Figure 5:
Joint velocity vs. time corresponding to initial path (blue) and optimized path (green)

Table 1. Numerical Analysis

I	m_i [kg]	Jm_i [kg.m^2]	I_i [kg.m^2]	$rcom_i$ [m]	r_i [ ]
1	33.935	$1.35*10^{-4}$	$\begin{bmatrix} 0.4446 & -0.0476 & 0.033 \\ -0.0476 & 0.3142 & -0.0634 \\ 0.033 & -0.0634 & 0.418 \end{bmatrix}$	$\begin{bmatrix} 0.0191 \\ 0.0248 \\ 0.4106 \end{bmatrix}$	0.2
2	25.937	$1.35*10^{-4}$	$\begin{bmatrix} 0.8506 & 0 & 0 \\ 0 & 0.109 & 0.0058 \\ 0.033 & -0.0634 & 0.418 \end{bmatrix}$	$\begin{bmatrix} 0 \\ -0.1731 \\ 0.2202 \end{bmatrix}$	0.2
3	15.502	$1.35*10^{-4}$	$\begin{bmatrix} 0.1178 & 0 & 0 \\ 0 & 0.0686 & 0.0074 \\ 0 & 0.0074 & 0.1138 \end{bmatrix}$	$\begin{bmatrix} -0.0036 \\ 0 \\ 0.0151 \end{bmatrix}$	0.2
4	9.540	$0.451*10^{-4}$	$\begin{bmatrix} 0.0864 & 0 & -0.0014 \\ 0 & 0.0768 & 0 \\ -0.0014 & 0 & 0.029 \end{bmatrix}$	$\begin{bmatrix} -0.0025 \\ 0 \\ 0.2952 \end{bmatrix}$	0.2
5	2.569	$0.451*10^{-4}$	$\begin{bmatrix} 0.0022 & 0 & 0 \\ 0 & 0.0008 & 0 \\ 0 & 0 & 0.0022 \end{bmatrix}$	$\begin{bmatrix} 0 \\ -0.0228 \\ -0.0001 \end{bmatrix}$	0.2
6	1.19148	$0.451*10^{-4}$	$\begin{bmatrix} 0.0002 & 0 & 0 \\ 0 & 0.0004 & 0 \\ 0 & 0 & 0.0002 \end{bmatrix}$	$\begin{bmatrix} -0.0005 \\ 0.0005 \\ 0.1263 \end{bmatrix}$	0.2

Table 2. Numerical Values

R_a [Ω]	$\begin{bmatrix} 9.4 & 0 & 0 & 0 & 0 & 0 \\ 0 & 9.4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 9.4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.832 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.832 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.832 \end{bmatrix}$
L_a [H]	$\begin{bmatrix} 12.77*10^{-3} & 0 & 0 & 0 & 0 & 0 \\ 0 & 12.77*10^{-3} & 0 & 0 & 0 & 0 \\ 0 & 0 & 12.77*10^{-3} & 0 & 0 & 0 \\ 0 & 0 & 0 & 2.73*10^{-3} & 0 & 0 \\ 0 & 0 & 0 & 0 & 2.73*10^{-3} & 0 \\ 0 & 0 & 0 & 0 & 0 & 2.73*10^{-3} \end{bmatrix}$
B[ ]	$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$

K_T [N.m/A]	$\begin{bmatrix} 0.487 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.4871 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.487 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.375 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.375 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.375 \end{bmatrix}$
K_p [ ]	$\begin{bmatrix} 300 & 0 & 0 & 0 & 0 & 0 \\ 0 & 300 & 0 & 0 & 0 & 0 \\ 0 & 0 & 300 & 0 & 0 & 0 \\ 0 & 0 & 0 & 300 & 0 & 0 \\ 0 & 0 & 0 & 0 & 300 & 0 \\ 0 & 0 & 0 & 0 & 0 & 300 \end{bmatrix}$
K_d [ ]	$\begin{bmatrix} 40 & 0 & 0 & 0 & 0 & 0 \\ 0 & 40 & 0 & 0 & 0 & 0 \\ 0 & 0 & 40 & 0 & 0 & 0 \\ 0 & 0 & 0 & 40 & 0 & 0 \\ 0 & 0 & 0 & 0 & 40 & 0 \\ 0 & 0 & 0 & 0 & 0 & 40 \end{bmatrix}$
K_i []	$\begin{bmatrix} 100 & 0 & 0 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 & 0 & 0 \\ 0 & 0 & 100 & 0 & 0 & 0 \\ 0 & 0 & 0 & 100 & 0 & 0 \\ 0 & 0 & 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 0 & 0 & 100 \end{bmatrix}$
K_E [V/(rad/s)]	$\begin{bmatrix} 0.28 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.28 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.28 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.22 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.22 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.22 \end{bmatrix}$
B_m [N.m.s]	$\begin{bmatrix} 0.1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.1 \end{bmatrix}$

To reduce the time elapsed during calculation, the time history of each joint angle values are divided into NN1=50 values. Each of these values is called as quantized state value. After defining the each quantized state (joint angle) values, admissible range is determined. Admissible range for joint angles is selected as ± 2° so as not to deviate from the given path for the end effector movement. Larger admissible joint angle values for each joint lead to larger error in between the original and the optimized path. By considering the working environment and the possibility of interference of robot manipulator with the other objects (fixtures, tools, etc.) it is better to limit the range of admissible values in order not to obtain a path that is different from the given path. Positional deviation in between the geometrically optimized path and the parametrically optimized path is kept at minimum and shown in Figure 6. Furthermore

the number of admissible joint angle and admissible joint velocity is chosen as N1=3 to decrease the calculation time.

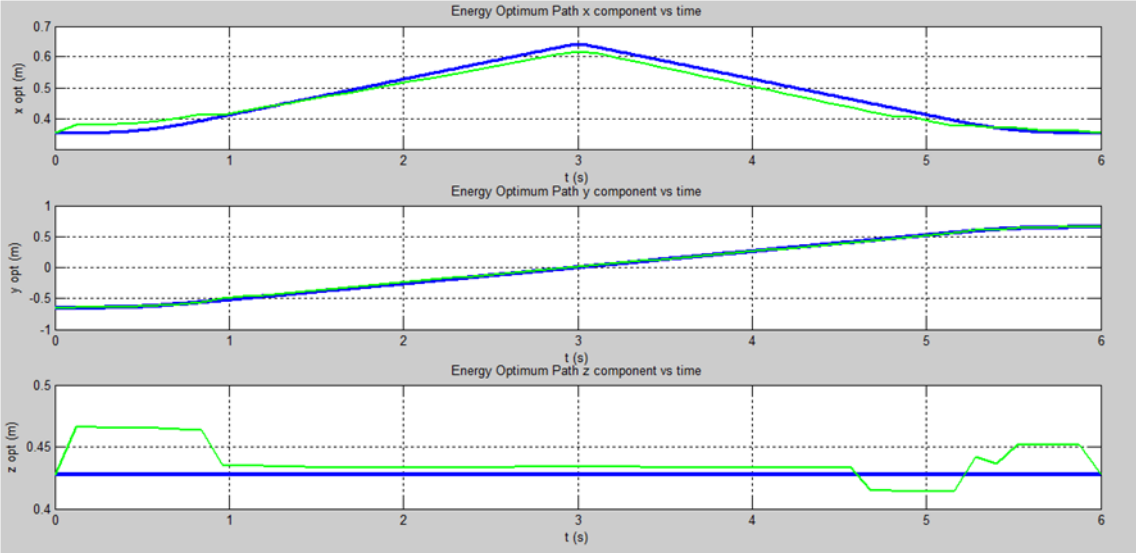


Figure 6:
x, y, z coordinate change of end effector tip vs. time for geometrically optimized path (blue) and energetically optimized path (green)

The results tabulated in Table I show that by only changing the path geometry, the process time is be reduced by an amount of 25% and improved from 8s to 6s. In addition to the capacity increase, energy saving by an amount of 40% compared to the initial path is achieved thanks to the elimination of the work done aganist the gravity. When parameter optimization is applied on this geometrically optimized path, this can also lead to energy consumption savings which is approximately 25%. In brief, 420[joule] energy saving is possible with this approach.

Table 3. Results containing data regarding to the initial case and optimized case

	Initial Case	***Geometry Optimization***	***Parameter Optimization with DP***
Path Length [mm]	1708	1422	1422
Process Time [s]	8	6	6
Avg. End Effector Speed [mm/s]	244	284,4	284,4
Energy Consumption [J]	700,7	386,3	280,5

6. CONCLUSION

In this research, it is aimed to reduce both process time and energy consumption to satisfy the desire in decreasing manufacturing cost. To start with, initial manufacturing conditions are modeled in computer environment by taking process time and end effector motion into account and power consumption of this movement is calculated. Then, geometry of path is optimized for which a substantial amount of process time reduction is possible. After the geometry of path for optimized case is defined without going beyond robot manipulator limits, optimal process parameters in implementing material transfer from one station to another in an energy optimal way are determined with dynamic programming.

To sum up, thanks to this analysis, by only changing the path geometry, the process time can be reduced by an amount of 25% in addition to the elimination of work done aganist the gravity which also contributes the power consumption by an amount of 40%. When parameter optimization is applied on the geometrically optimized path, this can also lead to energy consumption savings which is approximately 25%. In total, 420 [joule] energy saving is possible with both geomerty and parameter optimization. These savings and improvements provide a company a substantial amount of improvement in material handling processes.

REFERENCES

1. Aghanouri, M., Habibollahi, A., Esmaeili, A., Faghihian, H., and Koloushani, M. (2011) Optimization of robotic manipulators parameters modeled with integrated equations of actuators and links, *3rd International Students Conference on Electrodynamics and Mechatronics (SCE III)*, Opole, Poland, 31-36, DOI: 10.1109/SCE.2011.6092120.

2. Aldana, C.I., Nuño, E., Basañez, L., and Romero, E. (2014) Operational space consensus of multiple heterogeneous robots without velocity measurements, *Journal of the Franklin Institute* 351(3), 1517-1539. DOI: 10.1016/j.jfranklin.2013.11.012.

3. Carter, T. (2009) The modeling of a six degree of freedom industrial robot for the purpose of efficient path planning, *M.Sc. Thesis*, The Pennsylvania State University, USA.

4. Ding, W.H., Deng, H., Li, Q.M. and Xia, Y.M. (2014) Control-orientated dynamic modeling of forging manipulators with multi-closed kinematic chains, *Robotics and Computer-Integrated Manufacturing*, 30(5), 421-431. DOI: 10.1016/j.rcim.2014.01.003

5. Gans, N.R., Hu, G., Shen, J., Zhang, Y. and Dixon, W.E. (2012) Adaptive visual servo control to simultaneously stabilize image and pose error, *Mechatronics*, 22(4), 410-422. DOI: 10.1016/j.mechatronics.2011.09.008

6. Kirk, D.E. (1998) Optimal Control Theory: An Introduction, N.Y., Dovel Publication Inc.

7. Lewis, F.L., Dawson, D.M. and Abdallah, C.T. (2004) Robot Manipulator Control Theory And Practice (2nd ed), N.Y., Marcel Dekker Inc.

8. Meike, D. and Ribickis, L. (2011) Analysis of the energy efficient usage methods of medium and high payload industrial robots in the automobile industry, *10th International Symposium: "Topical Problems in the Field of Electrical and Power Engineering"*, Pärnu, Estonia, 62-66.

9. Murray, R.M., Li, Z. and Sastry, S.S. (1994) A Mathematical Introduction to Robotic Manipulation (2nd ed.), USA, CRC Press.

10. Owen, W.S., Croft, E.A. and Benhabib, B. (2008) A multi-arm robotic system for optimal sculpting, *Robotics and Computer-Integrated Manufacturing*, 24(1), 92-104. DOI: 10.1016/j.rcim.2006.08.001

11. Rocha, C.R., Tonetto, C.P. and Dias, A. (2011) A comparison between the Denavit–Hartenberg and the screw-based methods used in kinematic modeling of robot manipulators. *Robotics and Computer-Integrated Manufacturing*, 27(4):723-728. DOI: 10.1016/j.rcim.2010.12.009

12. Rodríguez, C., Montaño, A. and Suárez, R. (2014) Planning manipulation movements of a dual-arm system considering obstacle removing, *Robotics and Autonomous Systems*, 62(12),1816-1826. DOI: 10.1016/j.robot.2014.07.003

13. Staniak, M. and Zieliński, C. (2010) Structures of visual servos, *Robotics and Autonomous Systems*, 58(8), 940-954. DOI: 10.1016/j.robot.2010.04.004

14. Wang, D., Bai, Y. and Zhao, J. (2012)Robot manipulator calibration using neural network and a camera-based measurement system, *Transactions of the Institute of Measurement and Control*, 34(1), 105–121. DOI: 10.1177/0142331210377350

15. Wang, H., Liu, Y.H. and Chen, W. (2012) Visual tracking of robots in uncalibrated environments, *Mechatronics*, 22(4), 390-397.DOI: 10.1016/j.mechatronics.2011.09.0006