Reinforcement Learning Based Adaptive Control of a Flexible Manipulator

Bidyadhar Subudhi, *Senior Member, IEEE* and Santanu Kumar Pradhan
Center for Industrial Electronics & Robotics, Department of Electrical Engineering, National Institute of Technology, Rourkela, India-769008

***Abstract*— In this paper a new nonlinear adaptive controller using actor-critic based Reinforcement Learning (RL) is proposed to adapt the load pick-up and release operation while following a desired trajectory by the end effector for a Two-Link Flexible Manipulator (TLFM). Simulation results show that the proposed RL based adaptive control gives better trajectory tracking performance and suppression of link vibration compared to conventional adaptive controllers with time varying payload.**

***Index Terms*— Flexible Manipulator, Reinforcement Learning adaptive control, Time varying payload.**

I. Introduction

Tip-trajectory-tracking with variable payload in a flexible-link manipulator is a complex task compared to its rigid counterpart because of its structural flexibility. Conventional control methods, e.g., state-feedback and proportional-derivative are generally used for such manipulators [1]-[3]. However, very often, the payload of a manipulator does not remain constant. Use of such fixed-gain controllers provides poor performance because dynamics of manipulator with payload variation is not effected by the control gains.

To overcome the problem associated with fixed-gain controllers, one may employ adaptive controllers [4]-[7]. Among the several different types of adaptive control schemes, the use of direct adaptive control is useful due to less computational burden involved into it. However, in case of nonlinear flexible manipulators, employing direct adaptive control has two steps: first, one requires linearizing the plant dynamics and then, secondly, implementing the controller based on the linear model [5]. Moreover, this approach heavily depends on finite-dimensional model of the actual infinite-dimensional flexible manipulator system. Therefore it would be expected that an adaptive controller is performing with under approximate models. The contribution of this paper is to solve the problem of direct adaptive control using actor-critic based RL for end point trajectory tracking for a TLFM robot under time varying payload with minimum computational effort.

II. Dynamic Model of TLFM

A TLFM schematic along with the coordinate frames (X,Y) represent the rigid body moving frame and (x,y) represent the shift in frame due to flexibility is shown in Fig.1. It is required that the joint angles θ_1 *and* θ_2 to follow certain desired angular positions even with variations in the payload while suppressing the vibrations in the links arising due to their flexibility in the links.

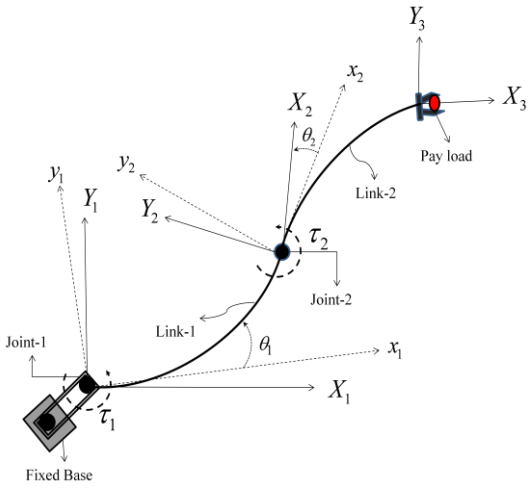

Fig. 1.Schematic diagram of the TLFM

The dynamic model for the above TLFM can be written using assume-mode method as follows [10]:

$$M(q)\ddot{q}+C(q,\dot{q})+Kq=\tau \tag{1}$$

where $q=[\theta_1,\theta_2,\delta_{11},\delta_{12},\delta_{21},\delta_{22}]^T$ are the generalized coordinate vector comprising of rigid joint angles, joint velocities and flexible deflection modes; M is the inertia matrix; C is the Coriolis and Centrifugal force vector; K is the stiffness matrix, and τ is the control torque applied to TLFM.

III. Non-Adaptive and Adaptive Controllers for TLFM

A. Non Adaptive Case

Structure of a non-adaptive feedback linearization based PID controller [11] to suppress the link deflection is shown in Fig.2. In this control strategy, a dynamic inversion torque is applied so that the outer-loop PID controller works on a composite linear model. The corresponding torque may be written as:

$$\tau = M(\ddot{q}_d - u) + N, \tag{2}$$

where $u = K_P e + K_I \int e dt + K_D \dot{e}$ and K_P, K_I *and* K_D are the control gains, the error e in tracking is a vector and is defined as

$$e = \begin{bmatrix} \theta_{d1} - \theta_1(t) \\ \theta_{d2} - \theta_2(t) \end{bmatrix} \tag{3}$$

and

$$N = C(q, \dot{q}) + Kq$$

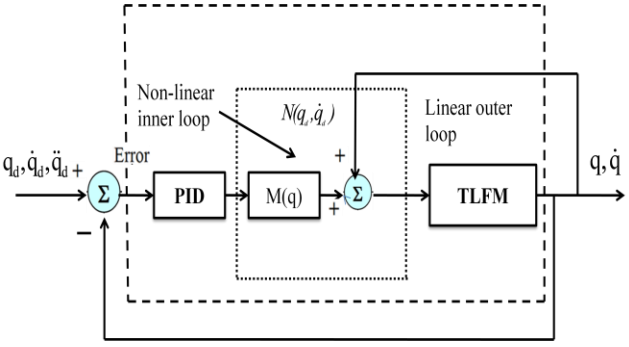

Fig. 2.Nonlinear Feedback Linearization Control

B. Adaptive Case

Since for time-varying tip mass case, the plant parameters of TLFM are unknown, one may resort to direct adaptive controller that estimates these parameters online. Such an adaptive control scheme is shown in Fig. 3. Let a be a parameter vector, and $\hat{a}$ be its estimate, by substituting the estimated parameter and putting in (1), we get [5]

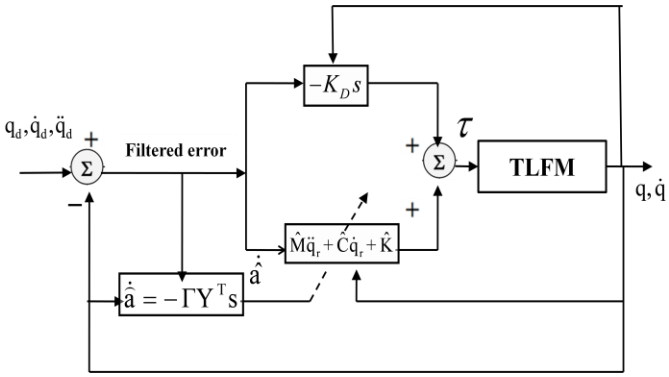

Fig.3.Nonlinear Direct Adaptive Control

$$\hat{M}(q)\ddot{q} + \hat{C}(q, \dot{q}) + \hat{K}q = \hat{\tau} \tag{4}$$

The input adaptive toque is give as

$$\hat{\tau} = Y\tilde{a} - K_D s \tag{5}$$

$\tilde{a} = \hat{a} - a$ is the parameter estimation error. Y is a matrix independent of dynamic parameter known as equivalent regressor matrix of the dynamics and consists of system states and errors.

$$\dot{\hat{a}} = -ZY^T s \tag{6}$$

and the filtered error is given as

$$s = \dot{\bar{e}} + \Lambda \bar{e} \tag{7}$$

where Λ , Z being constant positive definite matrices.

IV. The Reinforcement Learning Controller

A reinforcement learning method is applied to tune the gains online to give an optimal gain in order to adapt the payload changes on the tip. The Actor-critic base RL structure is shown in Fig.4 Applying (2) to (1), the error dynamics is obtained as

$$\frac{d}{dt}\begin{bmatrix} e \\ \dot{e} \end{bmatrix} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix}\begin{bmatrix} e \\ \dot{e} \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix} \tag{8}$$

Involving the system model into (8), one obtains a nonlinear error dynamics

$$\dot{x} = f(x) + g(x)u \tag{9}$$

with the state vector defined as $x = \begin{bmatrix} e \\ \dot{e} \end{bmatrix}$

The RL based adaptive control is achieved using the policy iteration method applied to (9). Let the sate-feedback control which stabilize ($f(x), g(x)$) be,

$$u(t) = h(x_t) = -Kx_t \tag{10}$$

The quadratic cost function to be minimized is expressed as

$$V(x(t)) = \int_t^\infty (x(t)^T Qx(t) + u(t)^T Ru(t))dt \tag{11}$$

The cost function parameters Q and R are chosen as identity matrices. It may be noted that the cost function is continuous in time, considering the one step cost function the utility can be written as

$$r(x(t), u(t)) = Qx(t) + u(t)^T Ru(t) \tag{12}$$

Eq. (12) can be written below as one step cost function and the system output state cost [12].

$$V(x(t)) = r(x(t), u(t)) + \gamma V(x(t+\tau)) \tag{13}$$

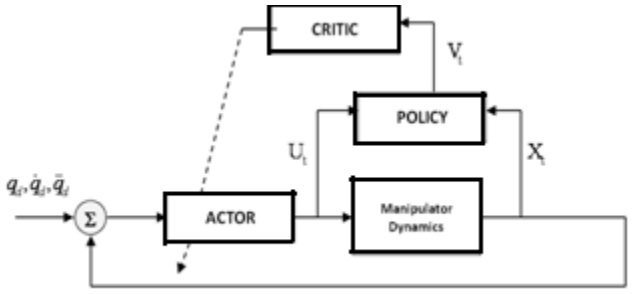

Fig.4.Reinforcement Learning Adaptive Control

with $0 < \gamma \le 1$ a discount factor and (13) is a nonlinear Lyapunov equation known as *Bellman equation*. In this proposed RL based adaptive control the (13) will be solved using policy iteration using data measured along system trajectories without the knowledge of system matrix. Discretetising (13) one gets

$$V(x_k) = \sum_{i=k}^{\infty} (x_i^T Qx_i + u_i^T Ru_i) \tag{14}$$

Substituting LQR (Linear Quadratic Regulator) feedback gain in (17) the *Bellman equation* LQR becomes

$$x_k Px_k = x_k Qx_k + u_k Ru_k + x_{k+1} Px_{k+1} \tag{15}$$

$$e_K = x_k Qx_k + u_k Ru_k + x_{k+1} Px_{k+1} \tag{16}$$

where $P = Q + K^T RK$ Equation (15) is linear in the unknown parameter matrix P .To further simplify Eq. (16) we use Kronecker product [9] may be used to give the new LQR-TD (Temporal Difference) error.

$$x_k Px_k = \bar{b}x_k = W^T \psi(x) \tag{17}$$

Substituting the above result to the *Bellman TD equation* we get

$$e_K = r(x_k, u_k) + \gamma W^T \psi(x_{k+1}) - W^T \psi(x_k) \tag{18}$$

Solving (18) using the policy iteration to minimize the TD error is the solution for online RL.

The TD equation (18) can be written as,

$$\left\{ \begin{array}{l} e_k = r(x_k, u_k) + \gamma W^T \psi(x_{k+1}) - W^T \psi(x_k) \\ for, e_k = 0; \\ \gamma W^T \psi(x_{k+1}) - W^T \psi(x_k) = r(x_k, u_k) \\ W^T (\psi(x_k) - \gamma \psi(x_{k+1}) = r(x_k, u_k) \end{array} \right\} \tag{19}$$

Fig. 5 shows the continuous time online policy iteration based RL.

Step 1: Initialize. Stabilizing control policy $K_0 x(t)$ is selected.

Step 2: Policy Evaluation. Weight were updated using gradient method, for policy evaluation

$$W_{l+1}{}^T (\psi(x_k) - \gamma \psi(x_{k+1}) = r(x_k, u_l(x_k))$$

Step 3: Policy Improvement. Determine a improved policy using for utility being (13),

$$u_l(x_k) = -\frac{\gamma}{2} R^{-1} g^T(x) \frac{\partial \psi(x)}{\partial x} W_{l+1}$$

Fig.5. On-line Policy Iteration Algorithm [8]

where, $\frac{\partial \psi(x)}{\partial x}$ is the Jacobian of the activation function vector. For the *Gradient decent tuning algorithm* the policy improvement can be written as

$$W_{l+1}^{i+1} = W_{l+1}^{i} - \Gamma \psi(k)((W_{l+1}^{i})^{T} \psi(k) - r(x_k, u_l(x_k)) \quad (20)$$

V. Results And Discussion

Taking $\Gamma > 0$ a tuning parameter here (100), and the iteration of i incremented at each discrete events. The above RL method provides the solution of the optimal control using Policy Iteration by measuring data along the system trajectories. As soon as the parameter has converged, the loop to update the control policy again activates to $l+1$ and update the policy. This updating of the policy ends when the gain and as well as ARE solution matrix converge to optimal values. The above derived algorithm for RL based adaptive control is implemented online to the TLFM so as it track a desired tip trajectory while performing load pick-up and release operation using Actor Critic structure. The policy which is defined as (20) in Critic then the gradient method is used to update the policy matrix elements and then the control is updated in the Actor as

$$u_l(x_k) = -\frac{\gamma}{2} R^{-1} g^{T}(x) \frac{\partial \psi(x)}{\partial x} W_{l+1} \quad (21)$$

the, γ is taken as 0.98.Then at each time k one measures the data set $(x_k, x_{k+1}, r(x_k, u_k))$ which consist of current state, the next state and the utility (14) then one step of parameter update is performed and this is repeated till convergence to the optimal value, Simulation was done in Simulink©/Matlab© environment. The physical parameters of the TLFM are shown in Table-1.The desired tip trajectory is computed using inverse kinematics deduced in (22)-(25).

$$q_d = (\theta_1(t), \theta_2(t)) \quad (22)$$

where,

$$\theta_1(t) = a\tan(y, x) - a\tan(a_2 \sin\theta_2(t), a_1 + a_2 \cos\theta_2(t)) \quad (23)$$

TABLE I-TLFM Material Properties

Link length	L1, L2	0.3493m, 0.2975m
Elasticity	E1=E2	2.0684×10^{11}(Pa)
Rotor moment of Inertia	Ks1, Ks2	6.28×10^{-6}, 1.03×10^{-6}(kg.m^2)
Drive moment of Inertia	J11, J21	7.361×10^{-4}, 444.55×10^{-6} (kg.m^2)
Link moment of Inertia	J12, J22	0.17043, 0.0064387(kg.m^2)
Gear ratio	N1, N2	100, 50
Maximum Rotation	R1, R2	+/-90, +/-90
Drive Torque constant	Kt1, Kt2	0.119; 0.0234(N.m/A)

$$\theta_2(t) = a\tan(\cos\theta_2, \pm\sqrt{1-\cos^2}\theta_2) \quad (24)$$

$$\theta_{d1} = \theta_{d2} = 10\sin(0.05t) \quad (25)$$

Figs. 5(a), 6(a) and 7(a) show the joint tracking error for TLFM on comparing we find that after the 10 sec learning time the tracking error is almost zero. Figs. 5(b), 6(b) and 7(b) show the joint-1 tracking and Figs. 5(c), 6(c) and 7(c) show joint-2 tracking. Figs. 5(d), 6(d) and 7(d) show the tip deflections for adaptive, non-adaptive and reinforcement learning respectively. Figs. 5(e), 6(e) and 7(e) show the link-1 mode deflection and Figs. 5(f), 6(f) and 7(f) show link-2 mode deflection and we see that the link deflection as discussed due to inner loop feedback linearization the dynamics cancel out leaving behind (8). Figs. 5(g), 6(g) and 7(g) show the joint-1 torque and Figs. 5(h), 6(h) and 7(h) show joint-2 torque From the results, it can easily be seen that the RL base control performance is better compared to adaptive and non-adaptive case in minimizing the tip trajectory tracking error as well as link vibrations. A step-type time-varying tip mass as shown in Fig.8 is considered, which represents both the load pickup and release operation. The simulation was carried out for 50 sec. Table 2 shows the gains computed for all the above discussed controllers.

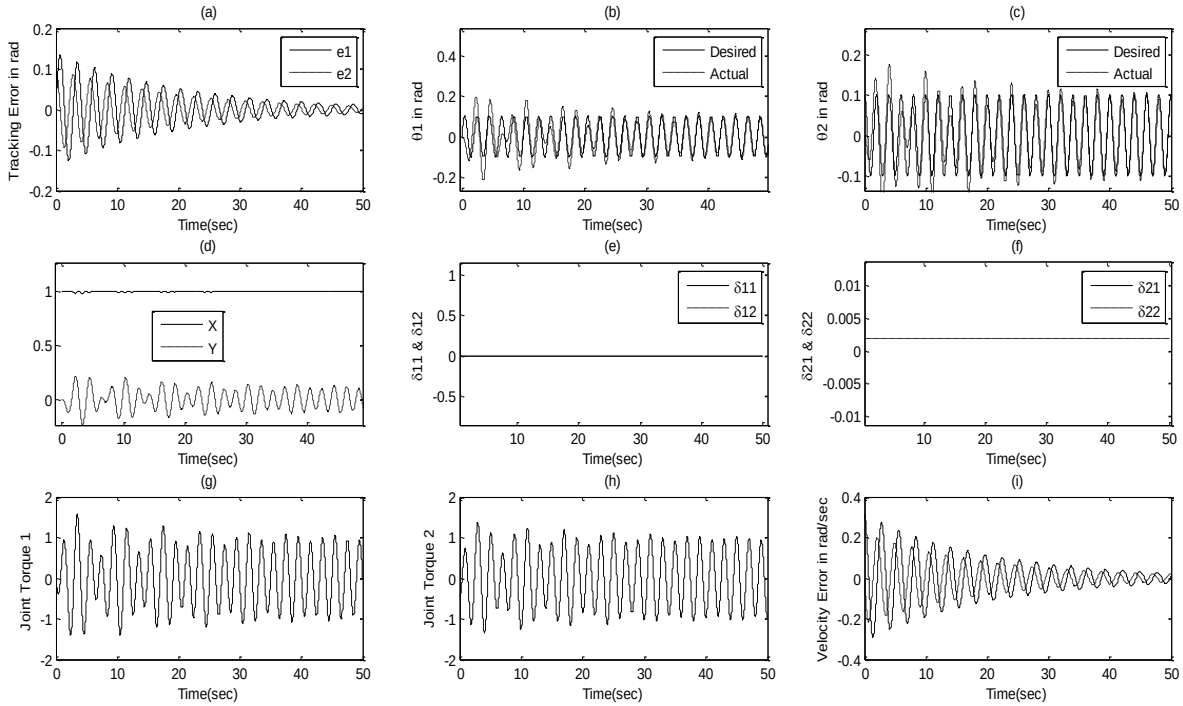

Fig.5. Feedback Linearization control

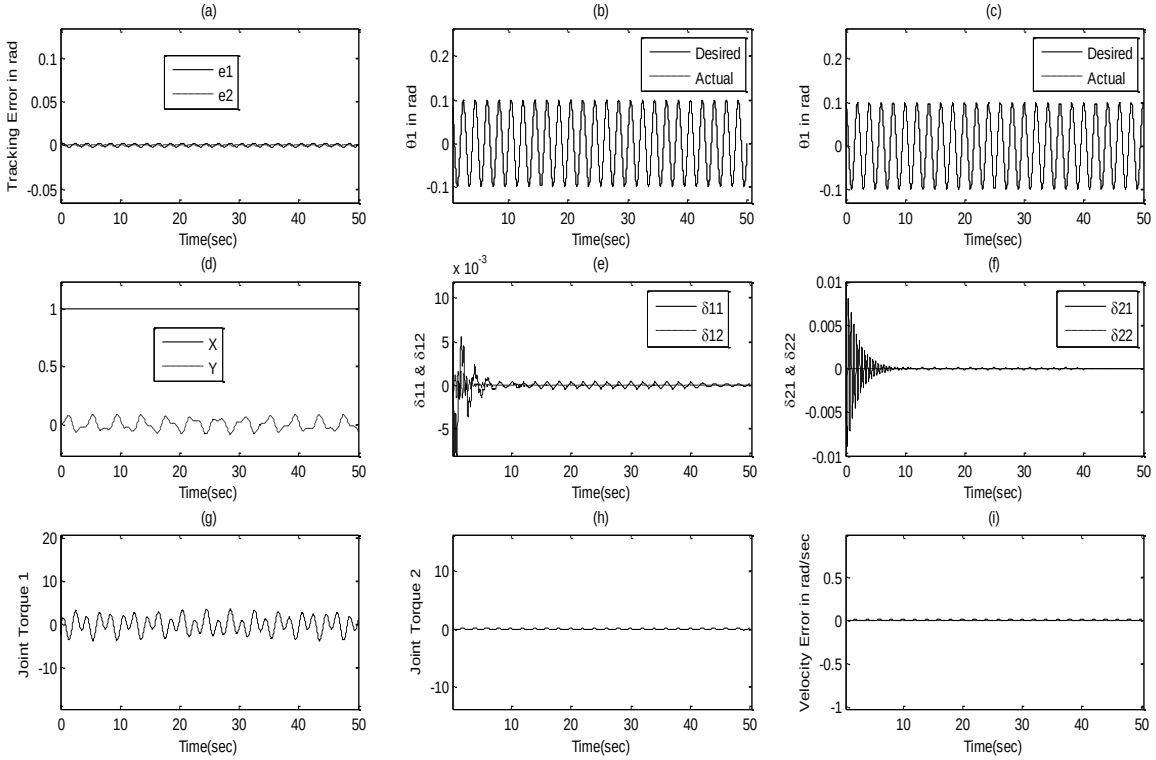

Fig.6. Nonlinear adaptive control

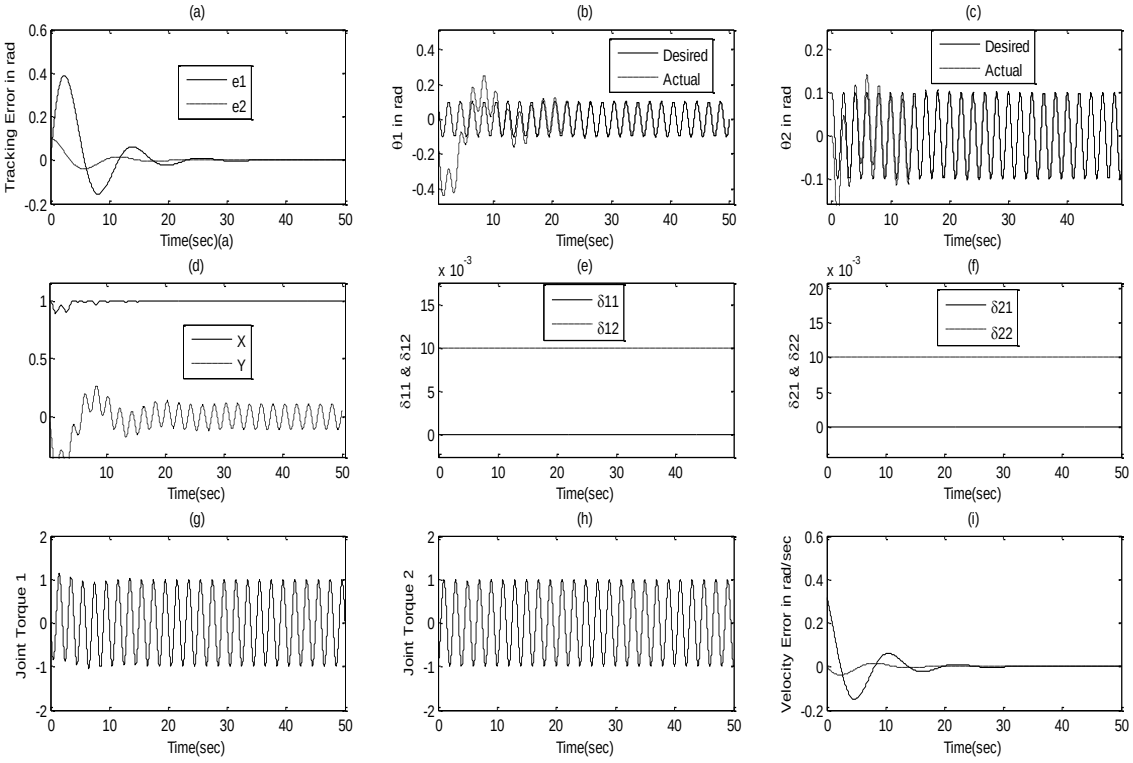

Fig.7. Reinforcement learning based adaptive control

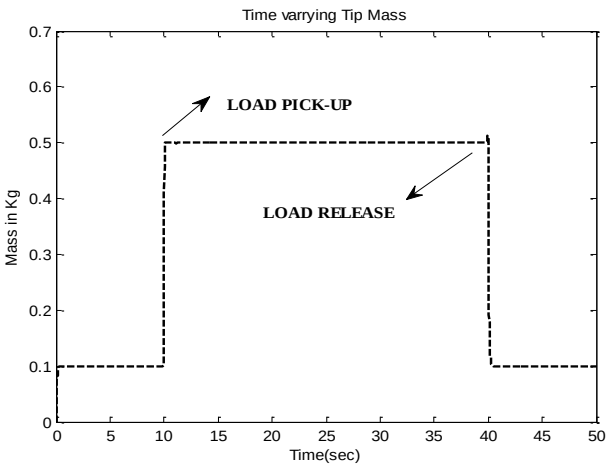

Fig.8.Time varying Payload

TABLE II-TLFM Control Parameters [11]

Controllers	Joint-1 gains	Joint-2 gains
Feedback Linearization(Computed Torque Control)	$K_P = K_{P1}$ $K_D = K_{D1}$ =100(approx)	$K_P = K_{P2}$ $K_D = K_{D2}$ =100(approx)
Linear Parametric Adaptive Controller	$K_P = K_{P1}$ $K_D = K_{D1}$ =50(approx)	$K_P = K_{P2}$ $K_D = K_{D2}$ =50(approx)
Reinforcement Learning based Adaptive controller	$\hat{K}_{P1} = \sqrt{QR^{-1}}$ $\hat{K}_{D1} = \sqrt{2K_{P1} + QR^{-1}}$ =4.5	$\hat{K}_{P2} = \sqrt{QR^{-1}}$ $\hat{K}_{D2} = \sqrt{2K_{P2} + QR^{-1}}$ =5.5

VI. CONCLUSION

In this paper, it is shown that, in presence of time-varying tip-mass as external disturbance, reinforced learning based adaptive controller has improved performance over two other non-adaptive and adaptive controllers for TLFM. What make this control more efficient than that of compared methods is that it also suppresses the link as well as tip deflection. Moreover, the controller adapts the feedback gains by not involving the manipulator dynamics hence reducing the computational difficulty and error due to modeling approximation. Such a performance is desirable in case of industrial task to get high-speed operation using light weight manipulators with accurate tip trajectory tracking under external disturbance.

REFERENCES

[1] M. A. Artega and B. Siciliano,"On Tracking Control of Flexible Robot Arms,"IEEE Trans. Aut. Control, vol.45, pp 520-527, 2000

[2] M. Mollem, K. Khorsani,'An Intrigal Manifold Approach for Tip-Position Tracking of Flexible Multi –Link Manipulator,'IEE Trans. Robotics and Auto., vol. 13, pp 823-836,1997

[3] R. N. Banavar and P. Domonic,"An LQG/Hinf Controller for a Flexible Manipulator,"IEEE Trans. Control Sys. Tec., vol. 3, pp 409-416,1995

[4] T. C. Yang, J. C. Yang, P. Kudra,"Adaptive Control of a Single-Link Flexible Manipulator with Unknown Load,"IEEE Proceedings, vol. 138, pp 153-159,1991

[5] J. J. E. Slotine and W. Li,"Adaptive Augmented Manipulator Control: Case Studyr,"IEEE Trans. on Auto. Control, vol. 33, pp 995-1003, 1988

[6] M. S. De Queiroz, D. M. Dawson, m. Agarwal and F. Zang, "Adaptive Nonlinear Boundary Control of a Flexible-Link Robot Arm,"IEEE Trans. on Robotics and Automation, vol. 15, pp 779-787, 1999

[7] J. H. Yang, F. Li Lian and Li Chen Fu," Nonlinear Adaptive Control for Flexible-Link Manipulators,"IEEE Trans. on Robotics and Automation, vol. 13, pp 140-148, 1997

[8] R. S. Sutton and A. G. Barto, *Reinforcement Learning—An Introduction*. Cambridge, MA: MIT Press, 1998.

[9] F. L. Lewis and Draguna Vrabie,"Reinforcement learning and Adaptive Dynamic Programming for Feedback Control,"IEEE Circuits and System magazine, 2009

[10] Alessandro De Luca, and Bruno Siciliano," Closed-Form Dynamic Model of Planar *Multilink* Lightweight Robots"IEEE Transaction on System, Man, And Cybernetics, vol. 21, August 1991

[11] J.J.Slotine and W.Li, Applied Nonlinear Control, Eaglewood Cliffs NJ: Prentice-Hall, 1991.

[12] K. Ogata, Modern Control Engineering, Prentice-Hall International, Upper Saddle River, NJ, 1997.