Modelling and Control of a Complex System Using a New Approach

SOHEIL DAMANGIR
School of Mechanical Engineering
Sharif University of Technology, Tehran
IRAN

GHAZALEH JAFARIJASHEMI
TAYLOR'S COLLEGE (subang jaya)
Taylor's College, Kuala Lumpur
MALAYSIA

HASSAN ZOHOOR
Fellow, the Academy of Sciences of I.R. Iran.
Member, Center of Excellence in Design, Robotics, and Automation
Sharif University of Technology, Tehran
IRAN

Abstract: - Presented in this paper are a method, named the Dynamic Rule Prediction (DRP), which predicts the behavior of a system and its application in designing a controller. The aim of the study is to overcome some of the limitations and shortcoming of the other controllers. The effectiveness of this method is verified. The controller based on DRP possesses two main features. It can control the system without any prior knowledge of the controlled plant. It is, also, superior as its high-speed prediction. This paper focuses on the robot manipulator controllers and applications of this approach in it.

Key-Words: - Dynamic Rule Prediction; Predictive controller; Robust control; Robot manipulator;

1 Introduction

Since the dynamics of the systems are highly nonlinear and may contain uncertain elements, many efforts have been made in developing modeling control schemes to achieve the precise model of the system. Conventionally, many control techniques for robot manipulators in industrial operation rely on proportional-integral-derivative (PID)-type controllers s due to their simple control structure, ease of design, and low cost [1,2,8]. However, systems have to face various uncertainties in practical applications such as, internal friction, and external disturbance (in mechanical systems) [11,13,18]. All the uncertain or time-varying factors could affect the system control performance seriously. Many control techniques have been investigated as viable means to improve the shortcomings of the conventional PID-type controllers [4,12,15,20]. Sun and Mills [20] proposed an adaptive-learning control scheme to improve control performance and could guarantee convergence in single and repetitive operational modes. But the control scheme requires the system dynamics in detail. A model-based PID controller was presented by Li et al. [15] to achieve the time-varying control of a robot manipulator tracker system. However, it is difficult to establish an appropriate mathematical model for the design of a model-based control system. Thus, the general claim of traditional intelligent control approaches is that they can attenuate the effects of structured parametric uncertainty and unstructured disturbance using their powerful learning ability without prior knowledge of the controlled plant in the design processes.

In the past decade, the applications of intelligent control techniques (fuzzy control or neural-network control) have received considerable attention [5 – 7,9,10,14,17,22,23]. A control system, which comprises PID control and neural network control, was presented by Chen et al. [5] for improving the control performance of the system in real time. Clifton et al. [6] and Misir et al. [17] designed fuzzy–PID controllers. Huang and Lee [9] suggested a stable self-organizing fuzzy controller. This approach has a learning ability for responding to the time-varying characteristics. However, the fuzzy rule learning scheme has a latent stability problem. Yoo and Ham [23] presented two kinds of adaptive control schemes via fuzzy compensator in order to confront the unpredictable uncertainties. Though the stability of the whole control system can be guaranteed, some strict constrained conditions and prior system knowledge are required in the control process. On the other hand, Kim and Lewis [10] dealt with the application of quadratic optimizations for motion control of robotic systems using cerebellar model arithmetic computer neural networks. However, the functional reconstructed error, the neural tuning weights and the high-order term in Taylor series are assumed to be known bounded functions, and some inherent properties of the system are required in the design process (e.g., skew-symmetry property, bounded system parameters and disturbances). In the whole design process, no strict constraints and prior knowledge of the controlled plant are required, and the asymptotic stability of the control system can be guaranteed. To accomplish the mentioned motivation, a SMNN control system is developed by Wai to control the joint position of an n rigid-link robot manipulator for periodic motion[34].

The aim of this study is to propose a method, named the Dynamic Rule Prediction (DRP), to map the input function into a virtual space in which the values of the function in --- is predictable. In DRP modeling process, no prior knowledge of the system is required. To design a system using DRP, the system, first, modeled by means of DRP. Thereafter the behavior of the system could be predicted. So the time function of the control variables is calculated.

2 Formulation

The main idea of the Dynamic Rule Prediction (DPR) is to map the input variable into a mapped space (MS), in which the transformed behavior of the input variables is predicable. In the other word, DRP is used to determine the MS so that it has the mentioned features.

In order to determine such MS, a transformation of the input variables with the following condition is considered. The basis of this transformation must (1)map zero and initial values of the input variables into nonzero and finite values. In addition, (2)outside the convergence margin, the output must approach zero. Moreover, (3)the transformation function should predict the input variables in a short period of time. This is determined in terms of the type of sampling and the length of the final prediction interval.

The transformation function is defined as follows

$$T\left(f(x),\lambda\right)=\int_{-\infty}^{x}\left[\left(\int_{-\infty}^{+\infty} f(y)e^{-ixy}dy\right) f(x-\lambda)\right]\times e^{i(x-\lambda)S}dx$$

eq.1

Then

$$T\left(f(x),\lambda\right)=\int_{0}^{x}\left[F\left(f(x)\right) f(x-\lambda)\right]\times e^{i(x-\lambda)S}dx$$

eq.2

clearly

$$T\left(f(x),\lambda-C\right)=\int_{0}^{x}\left[F\left(f(x)\right) f(x-\lambda+C)\right]\times e^{i(x-\lambda+C)S}dx$$

$$T\left(f(x+C),\lambda\right)=\int_{0}^{x}\left[F\left(f(x)\right) f(x-\lambda+C)\right]\times e^{i\left(x-\lambda+\frac{CS-C}{S}\right)S}dx$$

eq.3

For the small values of C respect to S, it could be proven that

$$T\left(f(x+C),\lambda\right)=T\left(f(x),\lambda-C\right)$$

eq.4

Therefore the eq.4 can be used to predict f at each step from its previous values.

The initial value of λ is calculated the "feeding step" procedure in order to fulfill the above-mentioned condition for C and S. As shown in figure 1, the feeding step requires some measured values of the system behavior to set the initial value of λ. First the input value of the system is transformed by using an arbitrary value of λ in eq.2. Thereafter the deviation between the predicted and actual behavior of the system is calculated and it is used to correct the selected value of λ.

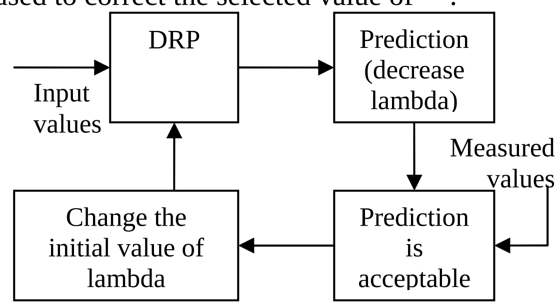

Figure 1. Flow chart of the feeding step

Subsequently, for any time sequence sampling input the procedure is repeated and the system behaviour is predicted, during which minor correction is made to previous value of λ in order the prediction to be be acceptable.

In other word the transformation function learns the system behavior by selecting the appropriate initial value of λ and its subsequent correction.
In addition the transformation function passes the "feeding step" rapidly when the input variable is a suitable controlling one.

3 Controller Design

In order to design a controller, the DRP model should pass the feeding step. The controlled variable should be fed to the system. Passing this step the model has authority to predict the behavior of the system.
The controller predicts the behavior of the controlled plant in term of controller variables. The controller sets the controller variables in order to achieve the most closely behavior to the desired behavior.
Schematic of the controller is shown in fig. 2.

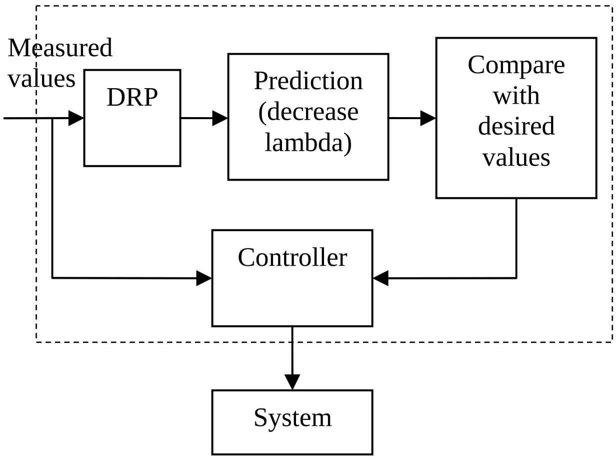

Figure 2 schematic of the controller

4 Numerical Example

An example herein is provided to verify the effectiveness of this approach. This method is applied to MK-3s robot for real-time trajectory tracking that guarantees the high speed and precise end-effector positioning control of robot arms, within the maximum joint torque constrain. Here the proposed method, named The Dynamic Rule Prediction is used to predict the value of the joints torque and the most closely end-effector trajectory to the desired path, without any optimization techniques.

4.1 Robot Arms Architecture

A schematic of Performer MK-3s industrial robot arm and its 2D world co-ordinate system are shown in Fig. 3(a) and Fig. 1(b). Out of the five joints of the robot arm, the first three joints from the base are shown. The end-effector is attached to the hand of the arm, which consists of three concentrated joints which are manipulated for orientation control of the end-effector tool.

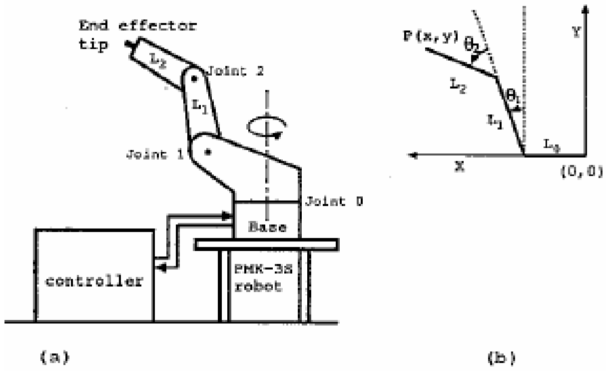

Figure 3 Performer MK-3s articulated robot arm (a) Control system schematic, (b) Convention of robot arm 2D co-ordinate system

Fig. 3(b) shows 2D world co-ordinate system and link placement where L0=0.135 [m], L1=O.250 [m], and L2=0.215 [m]. This dimension is considered by Munasinghe et al. [9]. The joint motors are actuated with current or voltage controllers that implement torque control of joint motors, according to the DRP control algorithm. The required input, i.e., the time-bases sequences of either position or velocity is given by the reference input generator that generates the joint trajectories of all joints, according to a specific trajectory tracking algorithm.

4.2 Modelling of the robot

The trajectory of the end-effector at random torque, random position and random initial velocity is calculated from the dynamics of the robot. These data feed off-line to the system. Figure 4 shows how the model is feed.

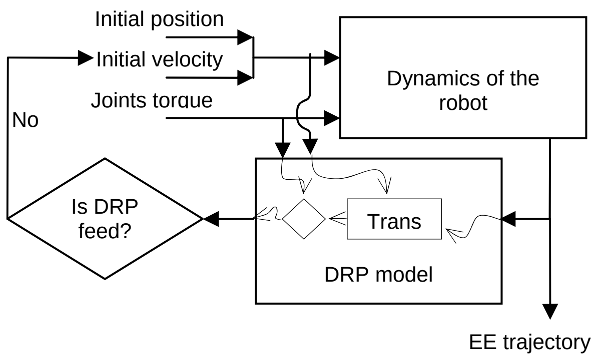

Figure 4 Scheme of the feeding step

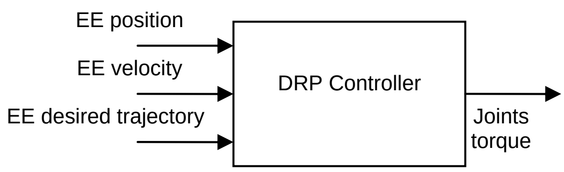

Figure 5 on-line mode of the control system

After the feeding step, the model could be used to predict the joints torque for the desired end-effector trajectory, at the initial conditions. Figure 5 illustrated the schema of the control system at on-line mode.

4.3 Tested Trajectory and Constrains

Objective trajectory was specified by a 10 [cm] radius circle centred O= (0, 0.35, 0.1) in the YZ plane. The global acceleration limit of torque saturation was set to 0.72 [rad/s2] for all joints, , and the absolute velocity of the end-effector was set to be between 0.18 and 0.25 [m/s]. The desired trajectory was contained 300 points. The input data were the three next desired end-effector position, which is approximately was 30.5 [ms].

5 Results

The controller was programmed in MATLAB. After the off-line feeding, the model was used to control the robot. The obtained results are shown in Fig. 6 and Fig.7

Fig.6 shows the desired and the produced path; note that the deviation is scaled by 10.

The root mean square error of the produced trajectory (with respect to the desired one) is shown in Fig.7. as shown in figures the maximum distance between the objective and produced trajectory is not exceeded more than 1.1 mm.

The average time require predicting the prepared joints torque was 1.2 [ms]. It shows that the proposed method has superior in fast controlling, so can be used for high-speed high frequency on-line control of the industrial robots.

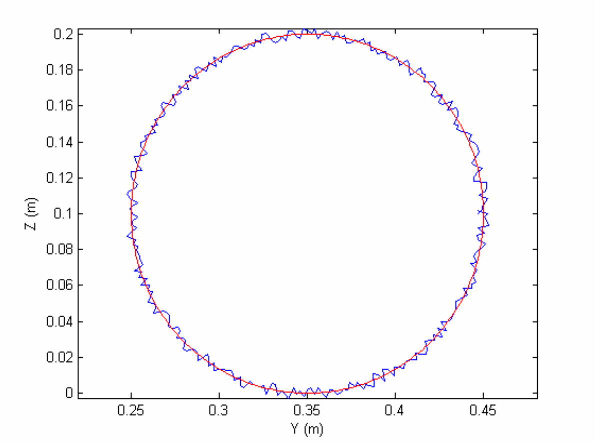

Figure 6 Desired and produced trajectory, Deviation scaled by 10

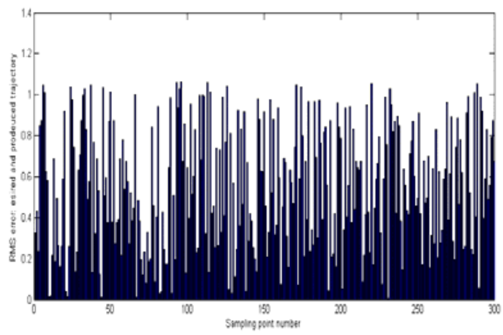

Figure 7 RMS error, desired and produced trajectory

6 Conclusion

The Dynamic Rule Prediction is illustrated and shown how it could be employed as an on-line controller without prior knowledge of the controlled system and exploiting optimization technique. This method is verified by applying to control of the MK-3s robot for path tracking that produces high speed and accurate end-effector point-to-point motion. The joints are actuated within the maximum joint torque constrain when the absolute velocity was set to be in an interval. It is shown this method is faster and therefore could be used for fast trajectory decision and fast movement of end-effector.

Although, herein, this method is applied to control the MK-3s robot for trajectory tracking, the method can be easily applied to other controllers.

References:

[1] K.J. Astrom, T. Hagglund, Automatic tuning of simple regulators with speciLcation on phase and amplitude margins, Automatica 20 (1984) 645.

[2] K.J. Astrom, T. Hagglund, PID Controller: Theory, Design, and Tuning, Research Triangle Park, NC, USA, 1995.

[3] K.J. Astrom, B. Wittenmark, Adaptive Control, Addison-Wesley, New York, 1995.

[4] I. Cha, C. Han, The auto-tuning PID controller using the parameter estimation, IEEE=RSJ International Conference on Intelligent Robots andSystems, 1999, p. 46.

[5] P.C.Y. Chen, J.K. Mills, G. Vukovich, Neural network learning andgeneralization for performance improvement of industrial robots, Canadian Conference on Electrical and Computer Engineering 1996,p. 566.

[6] C. Clifton, A. Homaifar, M. Bikdash, Design of generalized Sugeno controllers by approximating hybrid fuzzy–PID controllers, IEEE International Conference on Fuzzy Systems, 1996, p. 1906.

[7] L.B. Gutierrez, F.L. Lewis, J.A. Lowe,

Implementation of a neural network tracking controller for a
single Texible link: comparison with PD andPID controller IEEE Trans. Ind. Electron. 45 (1998) 307.
[8] C.C. Hang, K.J. Astrom, W.K. Ho, ReLnements of the Ziegler-Nichols tuning formula, IEE Proc. Control Theory Appl. 138 (1991) 111.
[9] S.J. Huang, J.S. Lee, A stable self-organizing fuzzy controller for robotic motion control, IEEE Trans. Ind. Electron. 47 (2000) 421.
[10] Y.H. Kim, F.L. Lewis, Optimal design of CMAC neural-network controller for robot manipulators,
IEEE Trans. Systems Man Cybernet. 30 (2000) 22.
[11] A.J. Koivo, Fundamentals for Control of Robotic Manipulators, Wiley, New York, 1989.
[12] T.Y. Kuc, W.G. Han, Adaptive PID learning of periodic robot motion, IEEE Conference on Decision andControl, 1998, p. 186.
[13] F.L. Lewis, C.T. Abdallah, D.M. Dawson, Control of Robot Manipulators, Macmillan, New York, 1993.
[14] F.L. Lewis, A. Yesildirek, K. Liu, Multilayer neural-net robot controller with guaranteed tracking performance, IEEE Trans. Neural Networks 7 (1996) 388.
[15] Y. Li, Y.K. Ho, C.S. Chua, Model-basedPID control of constrainedrobot in a dynamic environment
with uncertainty, IEEE International Conference on Control Applications, 2000, p. 74.
[16] F.J. Lin, W.J. Hwang, R.J. Wai, Ultrasonic motor servo drive with on-line trained neural network
model-following controller, IEE Proc. Electric Power Appl. 145 (1998) 105.
[17] D. Misir, H.A. Malki, G. Chen, Graphical stability analysis for a fuzzy PID controlledrobot arm model,IEEE International Conference on Fuzzy Systems, 1998, p. 451.
[18] R.J. Schilling, Fundamentals of Robotics: Analysis andControl, Prentice-Hall, EnglewoodCli4s, NJ,1998.
[19] J.J.E. Slotine, W. Li, AppliedNonlinear Control, Prentice-Hall, EnglewoodCli4s, NJ, 1991.
[20] D. Sun, J.K. Mills, High-accuracy trajectory tracking of industrial robot manipulator using adaptive-learning schemes, American Control Conference, 1999, p. 1935.
[21] D. Taylor, Composite control of direct-drive robots, Proceedings of the IEEE Conference on Decision
andControl, 1989, p. 1670.
[22] A.T. Vemuri, M.M. Polycarpou, Neural-network-basedrobust fault diagnosis in robotic systems, IEEE Trans. Neural Networks 8 (1997) 1410.
[23] B.K. Yoo, W.C. Ham, Adaptive control of robot manipulator using fuzzy compensator, IEEE Trans.Fuzzy Systems 8 (2000) 186.
[23] J.Y.S. Luh and C.S. Lin, “Optimum path planning for mechanical manipulators”, ASME %zns. Dyna. Syst., Meas., and Cdr., vol. 14, no. 3, pp. 444-451, June, 1984.
[24] K.G. Shin, and N. D. Mckay, LLMinimum-Time Control of Robotic Manipulators with Geomertic Path Constraints”, IEEE Trans. Automat. C&r., vol. 30, no. 6, pp. 531-541, June, 1985.
[25] K.G. Shin, and N. D. Mckay, “A dynamic programming approach to trajectory planning of robotic manipulators”, IEEE Trans. Automat. Cdr., vol. 31, no.6, pp. 491511, June, 1986.
[26] C.S.G. Lee, and M. Zeigler, “Geometric Approach in Solving Inverse Kinematics of PUMA Robots”, IEEE Aero. and Elect. Syst., vol. 20, no. 6, pp. 695-705, November, 1984.
[27] S.Goto, M.Nakamura and N.Kyura, “Modified taught data method for industrial mechatronic servocontroller to achieve accurate contour control performance,” IEEE/ASME Int. Conf., on Advanced Intelligent Mechatronics (AIM), 525B, June, 1997.
[28] Watson, I. Applying case-based reasoning: techniques for enterprise systems. Morgan Kaufmann, California, 1977
[28] F. Jatta, G. Legnani, A.Visioli, and G. Ziliani, “On the use of velocity feedback in hybrid force/velocity control of industrial manipulator,” Con. Eng. Prac. 2006, in press
[29] Y. Koike, H. Hirose, Y.Sakurai, and T. Iijima, “Prediction of arm trajectory from a small number of neuron activation in the primery motor cortex,” Neu. Res., 2006, in press
[30] S.R.Munasinghe, M.Nakamura, S.Aoki, S.Goto, and N. Kyura, ” High Speed Precise Control of Robot Arms with Assigned Speed Under Torque Constraint by Trajectory Generation in Joint,” 1999
[31] A. L. Omirou, and A. C. Nearchou, ”CNC machine tool interpolator for surface cros-section design,” 2006
[32] H. Diken, “Trajectory control of mass balanced manipulators,”
[33] R. V. Mayorga, and P. Sabnongboon,” Inverse kinematic and geometrically bounded singularities prevention of redundant manipulators: An Artificial Neuron Network approach”, 2005
[33] K.Y. Tu, and J. Baltes, “Fuzzy potential energy for a map approach to robot navigation,” 2006
[34] P.R. Ouyang, W.J. Zhang , and M.M. Gupta, “ An adaptive switching learning control method for trajectory tracking of robot manipulator,” 2005