Networked Control Of An Invert Pendulum Robot: Design And Performance Analysis

Meng Siqing, Meng Ping, KrishJayaram
School of Control Science and Engineering, Hebei University of Technology, Tianjin,
Department of Electrical Engineering, Delft University of Technology, Netherlands
School of Science and Engineering, Malysia University of Science and Technology, Malysia

Abstract

In this paper we study the problem of controlling a robot cart mounted with invert pendulum over unreliable wireless communication channel. The wireless channel is modeled with time-varying packet drop probability. A modified optimal controller for state feedback control is proposed based on the dynamic property of both the robot and the channel. The proposed controller adjusts the control objective according to the instantaneous packet drop probability. Experiments through simulation demonstrated good results of the proposed control design.

1. Introduction

Nowadays networked control systems (NCSs) are more and more used because they have less wiring and are easier to deploy and easier to maintain [1, 2, 17]. In a typical NCS the components are spatially isolated from one another, operating in synchronous or asynchronous manner. The sensor measurement data and control commands are all sent via wired or wireless links. Typical examples of NCSs are intelligent traffic systems, mobile sensor arrays, multiple autonomous mobile robots, large-scale decentralized flexible manufacturing systems, platoons of unmanned vehicles, aircraft networks, etc. Many of them involve controlling one or a group of intelligent agent, or robots over communication networks.

A robot is usually defined as an electro-mechanical machine, or sometime a program (virtual robot), which is capable of completing a specific task or to do a whole range of tasks or actions, under the guidance of remote computer, on-panel computers or single-chip micro-computers. It may also have some ability to perceive and absorb data through mounted sensors along with its local computational environment, or to respond to various stimuli reflected by sensor data. This is in contrast to a simple mechanical device such as a fan or a gear which has no processing ability and simply does tasks through purely mechanical motion.

Robots today are being utilized in a wide variety of industrial applications. Most of these applications are jobs that involve repetitiveness, accuracy, speed, and reliability. These tasks can be carried out much better by robots than human. Another type of tasks are those involves high risk field environment such as high radiation, extremely high or low temperature, etc., for which robots are usually deployed to the field and human operators can monitor its actions remotely. Nowadays the automated production line of the automobile industry is a good example of robots' successful application in production industry. Some of the other industrial applications of robots are packaging goods, dispensing jobs, and electronic component mounting and soldering on printed circuit boards in the semiconductor industry.

The need to control robots remotely through communication networks has inspired a lot of research work. Indeed, network controlled robots possess many advantages of NCSs, making it a favorable choice over traditional industrial robots that are controlled using field buses. For example, flexibility, mobility, low maintenance cost, etc. However, it is challenging to implement high-precision, high performance control algorithms over network, especially wireless channels. This is because the insertion of communication networks in the feedback loop complicates the application of standard results in analysis and design of controllers for NCSs: many assumptions made in the traditional control theory may not hold for NCS. For example, due to the limitations introduced by the undesirable behaviors of the network, it cannot be safely assumed that sensor measurement data are feed back to the controller reliably. This severely limits the control performance and may sometimes lead to instability [3-6, 8-10, 13, 14].

In this paper we designed a networked state feedback controller for wireless invert-pendulum robots control. Experiments through simulation have yielded good results of the proposed control design. The rest part of this paper is organized as follows: Section 2 introduces the properties of key components in a wireless robotic control system, with emphasis on their requirement on the reliability of the wireless network. Section 3 discuss networked control algorithms and described our controller design. This controller is further tested in an experimental setup and the simulation results are presented Section 4.

2. System Description

A wirelessly controlled robotic system has many components, as depicted in **Figure.1.** The characteristics of these components largely decide the type of control strategy that should be used, the sensitivity of the solution to the network congestion status, and the performance of the whole solution. In this section we brief review several key components of such a typical robotic system before investigating the control algorithms.

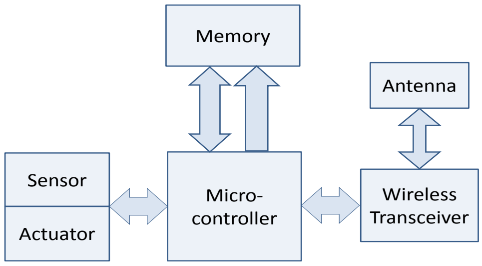

Figure.1 Typical wireless controlled robotic system structure

2.1 Servo motors for robots

Servo motor is the most common method to drive a robot. It is a motor which has a feedback mechanism to sense its position. The control command tells the motor to go to a certain position, then the built in logic in the servo motor will position it. Current continuous servo can rotate up to 360 degrees in both clockwise and counterclockwise directions. Typical servo system consists of a DC motor, a gearhead, a potentiometer for position feedback, and a circuit to read the potentiometer and position the motor.

The servo motor can also use PWM signals for controlling the DC motor. Most servo motor work well on 50 Hz of PWM frequency (PWM signal has a period of 20ms). Usually the 0.7ms to 1ms (i.e. Duty cycle of approximately 5%) PWM width will make the servo motor to turn clockwise (CW), the 1.7ms to 2ms (Duty cycle= 10%) PWM width will make the servo motor to turn counterclockwise (CCW). For the standard servo the 1.5ms (Duty Cycle=7.5%) PWM width will turn the servo motor to its center. However the exact PWM width depends on the servo motor types and brands.

2.2 Sensors

Sensors provide a robot with the means to observe characteristics of the external environment, and internal state. They are the key for robotics to determine its state and thereby act intelligently. Sensors vary in cost, complexity, and precision, most commonly used sensors in robotic system include: infrared sensors, ultrasonic sensors, strain gauge, accelerometers, gyroscopes, contact and proximity sensors, video camera, vibration sensors, etc. Sensors usually have various degree of nonlinearity, meaning that if we treat true value as input and sensor reading as output, they may not form an exact linear map (c.f. Figure.2.). This is usually taken care of in control design stage. Different sensors collect data at different rates, and therefore have different requirement on the network capacity.

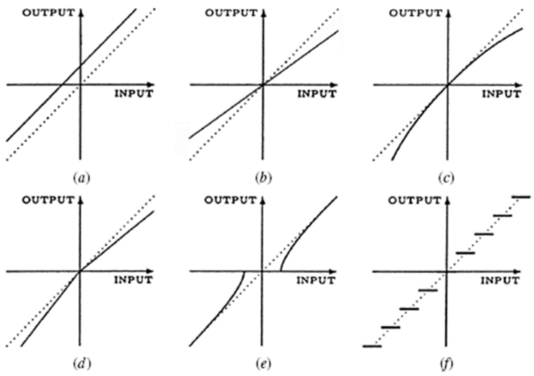

Figure.2. Different types of sensor errors

For servo control purposes, it's of vital importance to feed the state of robot back to controller in a timely manner. For example battery capacity, kinematic and dynamic parameters of the robot such as joint positions, velocities, and accelerations. These sensors usually don't require a bandwidth as high as video camera. From system design point of view it is also recommended that when a sensor solution with reasonable accuracy and low bandwidth requirement is available, do not resort to other sensors that may have higher accuracy. The actual control performance is highly likely to be worse due to bandwidth limitations in the wireless network.

2.3 Wireless transmission protocols

Many wireless communication protocols are available for industrial control applications. ZigBee is a low-cost, low-power, wireless standard that has been successfully applied to industrial control applications such as building automation, process control, etc. It is also widely used for NCS prototyping. Building upon the physical layer and medium access control defined in IEEE standard 802.15.4, it adds the network layer, application layer, ZigBee device objects (ZDOs) and manufacturer-defined application objects which allow for customization.

It is important to note that we only focus on servo control of robots. Solutions such as wireless SCADA, although it is relevant to networked robot control, is not within the scope of this paper. This is because wireless SCADA is more for supervisory control and therefore has less strict requirement on the real-time property of the communication network and the whole control system. For example in particular types of industry like Oil & Gas or Water & Wastewater, wireless SCADA is often used for the remoteness of the sites. One advantage of wireless SCADA systems is that it can be built on private radio (licensed or unlicensed), cellular or satellite communications. If the plant is hundreds or thousands of miles away and there is a TCP/IP LAN speed requirements then satellite communications become the only solution. However for our applications the plant (for example robot arm) has a much fast time scale and the wireless SCADA solution become poor choice.

3. Networked Control Algorithm Design

The key components described in last section are analyzed to extract mathematical models for control system design. In particular, the wireless communication network is modeled as a single link with random time varying packet drop with probability $p(t)$, we also assume the current packet drop probability $p(t)$ is available to the controller. This can be achieved through wireless channel estimation. The nonlinearity in the sensor and actuators are ignored so that relatively simple mathematic models are available for analysis. After mathematical abstraction, the structure of the control problem is shown in Figure.3.

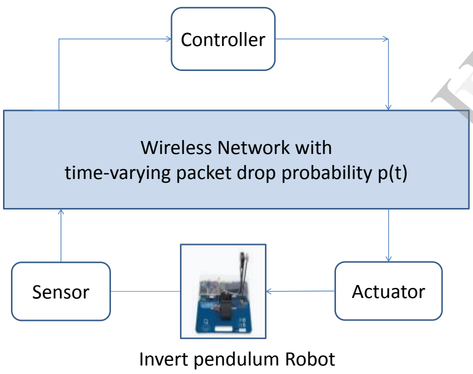

Figure.3: Simplified control problem scheme

The invert-pendulum robot model (c.f. Figure.4) is obtained by analyzing the potential energy and kinetic energy of the system with Lagrange's method. Choosing state of the system as

$$x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} \alpha \\ \dot{\alpha} \\ x_c \\ \dot{x}_c \end{pmatrix}$$

This leads to a nonlinear system representation which is difficult to analyze. Linearizing the nonlinear state space model at equilibrium point $x = [\,0, 0, 0, 0\,]$, $u = 3$, we can get simple linear time invariant model.

$$d\dot{x} = Adx + Bdu$$

where

$$A = \begin{bmatrix} 0 & 100 \\ a & 000 \\ 0\ \ 0 & 01 \\ b\ \ 0 & 00 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ c \\ 0 \\ d \end{bmatrix}$$

The coefficients a, b, c, d are computed from system physical parameters.M_p, M_c, l_p and g.

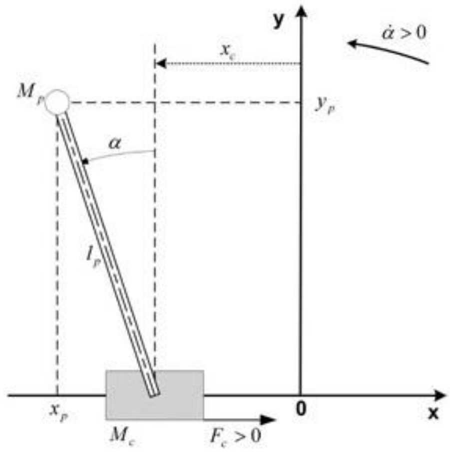

Figure.4: An invert pendulum robot.

We then discretize this linearized robot model to obtain a discrete time version of the LTI system, an optimal full state feedback controller is then obtained using linear quadratic regulator design. To accommodate the packet drop effect in the unreliable wireless channel, we added additional terms to the cost function which depends on the time varying packet drop probability $p(t)$. The cost function therefore becomes

$$J(x,u) = \sum_{t=1}^{N} x(t)^T Q(t) x(t) + u(t)^T R(t) u(t)\ )$$

Here N is the optimization horizon, and

$$Q(t) = Q_0 - \mu p(t)\, I$$
$$R = R_0 + \beta p(t)$$

Note that I is the 4-by-4 identity matrix, Q_0 is the un-modified kernel matrix for state cost and R_0 is the un-modified kernel matrix for control cost. The heuristic here is to make the control action against state less aggressive when packet drop rate p(t) is high.

4. Experiment results

In this section we implement the algorithm proposed in section.3 and compare its performance to that of ordinary optimal control approach. The discrete time natural and finite horizon solution has in effect made the optimal control result a model predictive control (MPC) approach: at each time step the finite horizon optimal control problem is solved for a sequence of control actions, and only the first control is applied. At the next time step the same procedure repeats. A theoretical treatment of stability and performance of MPC falls outside the scope of this work. In our experiments we simply choose the planning time horizon $N = 50$ which is long enough for stability. With a sampling time

T=10ms, this is equivalent to 0.5 seconds. The packet drop probability p(t) is chosen to be i.i.d. random variable sequence that is drawn from uniform distribution on the interval [0, 0.2].

Choosing Q_0 as identity matrix, $R_0 = 0.02$, an ordinary MPC algorithm that stabilize the invert-pendulum robot is implemented in MATLAB. Simulations show that its performance under packet drop probability is not as good as our proposed algorithm with parameters μ=1 and $\beta = 0.05$.

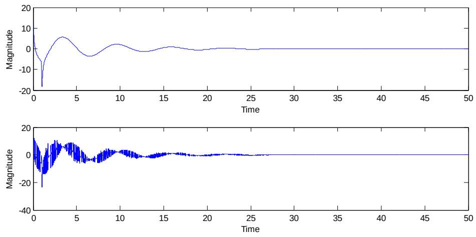

Figure 5. Control action generated by proposed method (top) and infinite horizon LQR (bottom)

The control action generated by the proposed control algorithm and infinite horizon LQR is compared in Figure 5. Since we used the moving horizon implementation (MPC), when a packet is dropped, the control action can be taken from the optimal control sequences generated at the last time step. Therefore the smoothness of control is much better. Figure 6. Showed the result of our proposed control algorithm under the given packet drop rate. The effect of dropped packet can hardly be recognized from the plots.

The parameters μ and β has an indirect impact on the performance of the closed loop system. As we observed, the larger these parameters are, the more conservative the control actions tend to be. But in choosing these parameters we should observe certain thresholds to avoid undesirable behavior. For example if $\beta > 0.1$ the cost for control action becomes negative.

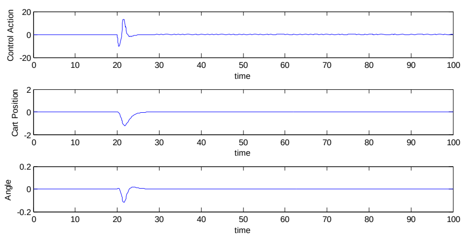

Figure 6. Simulation Result for the proposed method

The packet drop probability $p(t)$ is constrained to be small. When it is allowed to take large values, consecutive packet losses happen more frequently. In that case both algorithms perform poorly. An investigation on how to choose parameters μ and β according to current probability $p(t)$ so that best possible performance can be achieved is part of our future work.

5. Conclusion

In this paper we elaborated on a networked state feedback controller designed for wireless invert-pendulum robots. Important issues to consider in the design procedure are highlighted, especially those pertaining to the unreliable feature of wireless networks. A modified optimal control algorithm is proposed that takes the packet drop probability into both state and control cost functions. Simulations have shown good results of the proposed control design. In the future we will combine the proposed approach with the so-called event-triggered control method [11, 12, 15, 16].

6. References

[1] Baillieul, John and Antsaklis, Panos J., “Control and Communication Challenges in Networked Real-Time Systems”, Proceedings of the IEEE, 95(1) pp.9 - 28, 2007

[2] Hespanha, J.P.; Naghshtabrizi, P.; YonggangXu, "A Survey of Recent Results in Networked Control Systems," Proceedings of the IEEE , vol.95, no.1, pp.138-162, Jan. 2007

[3] SekharTatikonda, S. Mitter, “Control Under Communication Constraints”, IEEE Transactions on Automatic Control, 49, 1056-1068., 2004,

[4] SekharTatikonda, S. Mitter, “Control Over Noisy Channels”, IEEE Transactions on Automatic Control, 49, 1196-1201, 2004,

[5] G. N. Nair & R. J. Evans, “Stabilization with data-rate-limited feedback: tightest attainable bounds”, Systems & Control Letters, vol. 41, no. 1, pp. 49-56, 2000.

[6] Sun Yu, Prashant G. Mehta, “Fundamental performance limitations with Kullback-Leibler control cost,” in Procs. of Conference on Decision & Control, pp.7063-7068, 2010.

[7] Tabuada, P., "Event-Triggered Real-Time Scheduling of Stabilizing Control Tasks", IEEE Transactions on Automatic Control, 52(9), pp.1680-1685, 2007

[8] Nicola Elia: Remote stabilization over fading channels. Systems & Control Letters, vol.54, no.3, pp.237-249, 2005.

[9] Martins, Nuno C., Dahleh, Murither A., "Feedback Control in the Presence of Noisy Channels: 'Bode-Like' Fundamental Limitations of Performance", IEEE Transactions on Automatic Control, vol.53, issue.7, pp. 1604 – 1615, 2008

[10] Yu. Sun and P. G. Mehta, “Fundamental performance limitations via entropy estimates with hidden Markov models,” in Procs. of Conference on Decision & Control, pp. 3982–3988, 2007.

[11] Dimarogonas, D.V.; Johansson, K.H., "Event-triggered control for multi-agent systems," IEEE Conference on Decision and Control, pp.7131-7136, 2009

[12] X. Wang and M. Lemmon, Event-Triggering in Distributed Networked Control Systems, IEEE Transactions on Automatic Control, vol. 56, no. 3, 586-601, 2011

[13] Sun Yu, Prashant G. Mehta, “Bode-Like Fundamental Performance Limitations in Control of Nonlinear Systems”. IEEE Transaction on Automatic Control, vol. 55, pp.1390-1405, 2010.

[14] G. Zang, and P. A. Iglesias,"Nonlinear extension of Bode's integral based on an information theoretic interpretation", Systems and Control Letters, vol.50, pp.11–29, 2003.
[15] Wang, Xiaofeng; Yu. Sun; Hovakimyan, N., "Relaxing the consistency condition in distributed event-triggered networked control systems," IEEE Conference on Decision and Control (CDC) pp.4727-4732, 2010
[16] Xiaofeng Wang, Yu. Sun, Naira Hovakimyan: Asynchronous task execution in networked control systems using decentralized event-triggering. Systems & Control Letters 61(9), pp. 936-944, 2012
[17] Braslavsky, J.H.; Middleton, R.H.; Freudenberg, J.S., "Feedback Stabilization Over Signal-to-Noise Ratio Constrained Channels," IEEE Transactions on Automatic Control, vol.52, no.8, pp.1391-1403, Aug. 2007