PRACTICAL EDUCATION OF CONTROL ENGINEERING
Using Open OS, Free Software and Actual Plant

Sunao Tanimoto, Kiyoshi Yoshida
Nippon Institute of Technology, Miyashiro, Minami-Saitama, Saitama, 345-8501, Japan

Jyunichiro Tahara
Japan Agency for Marine-Earth Science and Technology (JAMSTEC), Japan

Keywords: Experiment Devices, Computer Control, Modern Control Theory, Plant Control, Linux, Octave, Control CAD.

Abstract: The importance to educate control engineers is to make them experience the modern control theory by the actual devices instead of pure digital simulator. During these applications, students will face to every existing problem and digest the fruits of control theory by tangible impression. This paper reports the development of experiment devices for computer control engineering education, where a pendulum is controlled by a General Linux P/C through a DC motor every 1ms, realizing most kinds of advanced control schemes linked with free control CAD, Octave. Controlled examples with various schemes are reported.

1 INTRODUCTION

Control engineering has an important role in industries. Authors feel that the education for control engineers has the problem that it mainly depends on just the computer simulations. This seldom offers tangible advanced control experiences to engineers or students, and valuable theories are not applied to actual fields well. To cope with this problem, authors developed actual motor driven pendulum/arm real-time control experimental systems with an open source code OS; Linux, CAD/CAE; Octave and “C” Language (Figure 1). Using the developed devices, students have flexibly managed to realise tens of schemes of tangible theories in the college lab. during last four years.

This paper explains the concepts and specifications of developed devices, plant modelling with mathematical linearization. Some developed applications of modern control theory are introduced. Finally, three types of discrete–time control methods for industry uses are applied and their performances are evaluated in each other.

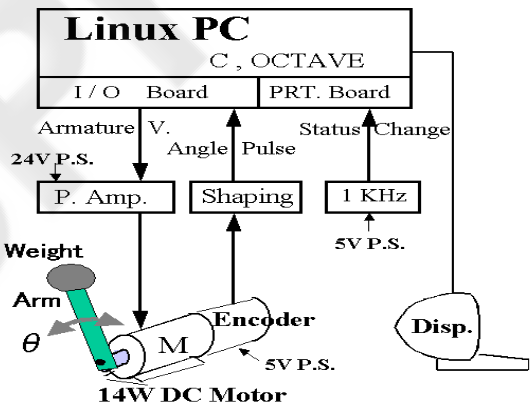


Figure 1: System configuration.

2 DESIGN CONCEPT AND SYSTEM CONFIGLATION

The basic component of plant control or Robot can be reduced to control the position of a arm driven by a motor. Key points of the design concept and the spec. of this control system are discussed below.

Tanimoto S., Yoshida K. and Tahara J. (2010).
PRACTICAL EDUCATION OF CONTROL ENGINEERING - Using Open OS, Free Software and Actual Plant.
In *Proceedings of the 2nd International Conference on Computer Supported Education*, pages 265-270

2.1 Design Concept

2.1.1 Real Time Control

Conventional real time control systems in industries are controlled every 10 ms. Thanks to the progress of computer capability today, it is possible to realize 1 ms DDC (Direct Digital Control) by just a P/C. Once the developed control program is turned on, the program senses the status change of 1 KHz oscillator input. this program is executed every 1 ms, without running OS area at all, achieving the perfect real-time function.

The sampling time should not be disturbed by the control calculation time. Because of the 2 GHz P/C clock, most kinds of optimum calculation are acceptable for control. Authors did not adopt commercial real time OS to avoid their complexity and ambiguity.

2.1.2 Accuracy, Simplicity and Reliability

Actual arm angle can be detected by 14,400 pulses per rev. through I/O board of the P/C. Control input to the motor armature is in analogue through 10-bit DAC. A backlash-less motor without a gear is adopted. No redundant devices are included.

A simple system configuration directly corresponds to its reliability and accuracy. These devices are neatly packed on a single board.

2.1.3 Maintainability

In plant control applications in industries, the traceable-ness is quite important. Once a software bug stalled the plant and destroyed the machinery, the plant cannot be restarted without fixing the bug.

In case of Windows PC, whose source codes are not disclosed, this bug cannot be fixed. Therefore, Authors adopted Linux OS of open source code.

2.1.4 Flexible Control Design Capability

Although, AC motors are commonly used instead of DC motors today, Authors adopted a DC motor for the arm control, because a DC motor can be easily expressed by differential equations, which results in better understanding for users or students.

The controller is designed in C-language and its PID control reference software is initially installed by a author. Users can only modify this software for their own purpose. The input and output of this controller are the pulse count of the pendulum angle and the armature voltage of the DC motor, respectively.

2.1.5 Computer Aided Engineering

This system includes a control CAD/CAE called by Octave of free software instead of expensive Matlab. Users can get the chart of controlled result by storing the control variables in the software table every 1 ms. Octave also offers tools of control theory such as eigen value, Ricatti solution and transfer function. Users can compare the actual control results with the modelled theoretical result by both charts.

2.1.6 Exclusion of Non-linearity in Devises

Authors believe that the mechanical non-linearity such as backlash, friction and dead time are difficult to be evaluated and also its fine repeatability is not realized in actual plants. In the devices here, those non-linearity are excluded as much as possible. The gearless motor and the directly coupled encoder with fine resolution are adopted.

In case of developing those non-linear control methods, non-linearity should be realized in the P/C through its software with repeatability. And then, developed control scheme for pure non-linearity will be applied to the actual field, and analyzed.

2.1.7 Cost and Open Documents

4 same systems are developed. Its cost is \$3,000/set excluding authors' labour. This reasonable cost can mainly be realized by adopting the free software of Linux and Octave. Its specification is open by authors to any other groups for education and non-business use.

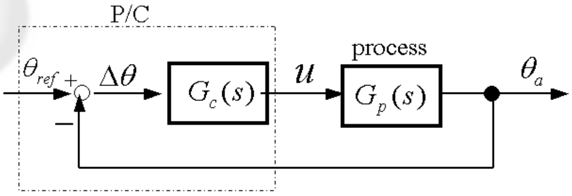


Figure 2: Control Block Diagram.

2.2 Hardware

This system consists of controlled process or plant and a controller (Figure 2).

2.2.1 Process

a. Motor and Encoder

A small DC motor is adopted considering users' safety. Rating specifications are on Table 1. Low inertia-ed rotary encoder directly coupled with the motor shaft is adopted. Pulse rate is 14,400 pulses/rev.

b. Amplifier and Power Supply

Large capacity-ed power amp. Is adopted considering the motor stall current (table 2). Switching power supply for the power amp. Is of 100w with 24vdc and 4.5a.

c. Oscillator

1 KHz oscillator is made as a sampler.

Table 1: Motor spec.

Power	14 W
Voltage	DC100
Speed	2500 pm
Torque	0.054 m

Table 2: Amp. spec.

Power	125 W
Voltage	100 V
Current	15 A

2.2.2 Controller

a. P/C

P/C has a CPU of 2 GHz, 256MB memory and 80GB H/D.

b. I/O Board

1 ms sampling signal is input to the printer port .

- Analogue i/o; PCI- 3523A has DAC with 12bits, ADC and a few DI/DO.
- Encoder counter; PCI-6204 has a 32-bit counter.

2.3 Software

Linux Fedora Core 1 is adopted for OS. Application task can access I/O through developed sub-systems without OS.

Octave for Linux is down loaded. Its application manual is openly supplied by Internet.

Reference PID control program is prepared. It has the strict correspondence with its documentation, block diagram and naming rule. It has blocks of control parameter initial input, periodical run loop, data acquisitions, PID calculation, Output to i/o board and data storing for Octave. Users can modify this program for their own purposes.

3 PLANT MODELING

Once the plant H/W is designed, then the advanced control needs its mathematical modelling

3.1 Mathematical Model of the Plant

A motor with a pendulum arm is shown by Figure 3.

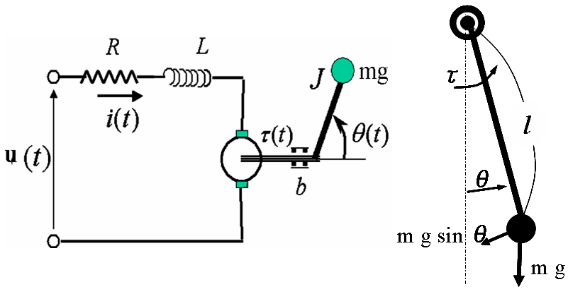


Figure 3: DC motor and a pendulum.

These plants are mathematically expressed by

$$J\ddot{\theta}(t) + b\dot{\theta}(t) = \tau(t) - mgl\sin\theta(t) \quad (1)$$

$$\tau(t) = K_T\, i(t) \quad (2)$$

$$u(t) = L\dot{i}(t) + Ri(t) + K_V\dot{\theta}(t) \quad (3)$$

Definition of variables and parameters are as follow:

u : Armature voltage R : Armature resistance
L : Armature inductance i : Armature current
b : Friction loss J : Arm inertia
τ : Motor torque θ : Arm angle
l : Arm length m : Mass of Weight
g : Gravity acc. K_T : Torque const.
K_V : B.E.F. const.

3.2 Linearization of the Plant

This plant has a non-linear term of $\sin\theta$. We need a linear model to apply the linear control theory. In conventional method, this term is approximated by θ . Another method adopted here is the strict linearization by non-linear term compensation.

3.2.1 Approximated Linear Model

Assuming $\sin\theta \cong \theta$, 3rd order model is expressed by the state variable expression bellow:

$$\dot{\mathbf{x}} \equiv \begin{pmatrix} \dot{\theta} \\ \ddot{\theta} \\ \dot{i} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ -\dfrac{mgl}{J} & -\dfrac{b}{J} & \dfrac{K_T}{J} \\ 0 & -\dfrac{K_V}{L} & -\dfrac{R}{L} \end{pmatrix} \begin{pmatrix} \theta \\ \dot{\theta} \\ i \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \dfrac{1}{L} \end{pmatrix} u \quad (4)$$

$$= \mathbf{A}\mathbf{x} + \mathbf{b}u$$

Incidentally, the inductance L in the DC motor is ignored in many applications considering the large amount of the inertia of the arm, which is called 2nd order model.

3.2.2 Strictly Linearized 2nd Order Model

The original system has the serious non-linearity of $\sin\theta(t)$. Authors applied the input linearization to delete non-linearity. (1) can be modified below:

$$J\ddot{\theta} + b\dot{\theta} = \tau - mgl\sin\theta \; + mgl(\sin\theta - \theta) \quad (5)$$

The term $(\sin\theta(t) - \theta(t))$ is compensated to the motor armature voltage every 1 ms (Figure 4), getting the linear model as follow:

$$J\ddot{\theta} + b\dot{\theta} + mgl\theta = \tau \quad (6)$$

By adding this compensation, control theory in textbooks can be applied without any contradiction.

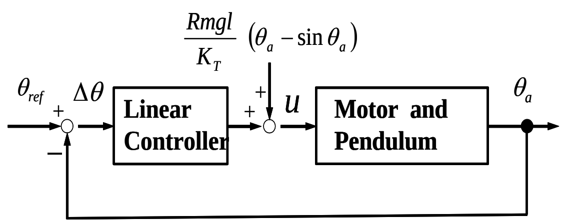


Figure 4: Real time input linearization.

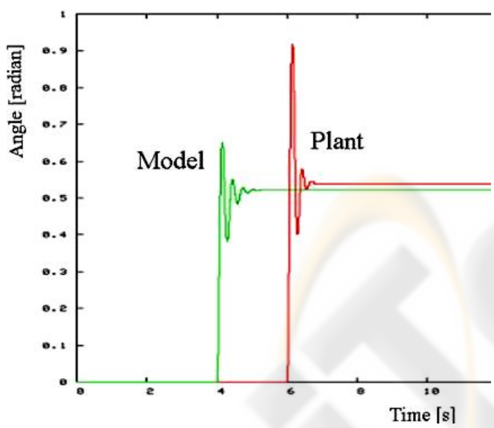


Figure 5: Step response by PID controller.

3.2.3 Identification of the Plant

The derived 2nd order model is identified to the actual plant (Figure 5). Both model and actual plant are controlled by the same PID controller. The model is executed every 1 ms using Euler difference method. Incidentally, the dead time of this system is confirmed to less than 2ms.

4 CONTROL APPLICATIONS

Authors examined several control schemes in text-books. Basically, control is executed every 1ms, which is almost regarded as a continuous system.

Every nonlinear function such as dead time, sampling period, discrete time, saturation, dead zone, static friction is realized by “C” in P/C. The lists of developed schemes are as follow:

- PID control.
  - Position control.
  - Angle velocity control.
  - Dead time control.
  - Sampling control.
  - Control with low resolution sensor.
  - Control with less number of effective digits.
- Modern control with 2nd order.
  - State feedback control.
  - Pole assignment control.
  - Optimum regulator.
  - Integral type control.
  - Observer coupled control.
  - Discretized State feedback control.
- Modern control with 3rd order, under work.
  - Observer coupled control (armature current observance).

4.1 PID Performance by Sampling Time

In actual plants, control performance mainly depends on the sampling period. Continuous feedback control is desirable.

Authors checked how the control performance depends on the sampling period quantitatively. Maximum stable proportional gain Kp for the horizontal pendulum position control depending on every sampling period is shown by Figure 6.

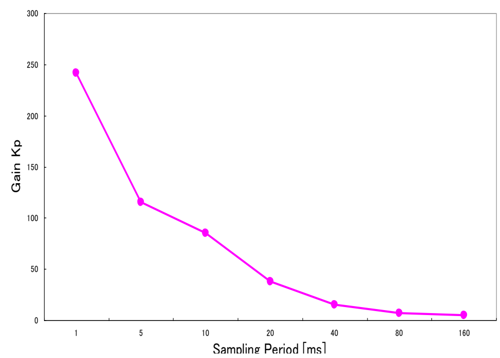
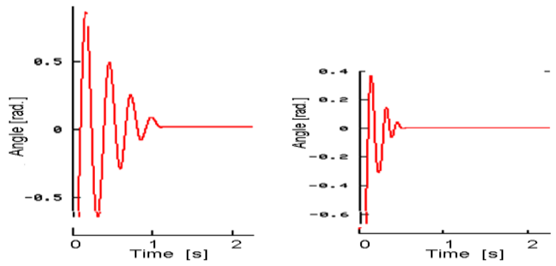


Figure 6: Maximum proportional feedback gain vs. sampling period.

4.2 Pole Assignment Control

During the state feedback control using 2nd order plant model, poles are assigned with $p=-5\pm j20,\ \ -10\pm j40$ using Octave. Control results are shown in Figure 7. This means the linear modern control is realized theoretically enough.



$p=-5\pm j20$ $\qquad$ $p=-10\pm j40$

Figure 7: Pole assigned step responses.

4.3 Integral Type Optimum Regulator

As control theory says the optimum regulator is just a gain feedback and offset cannot be deleted depending on the weight matrix Q as follow:

$$\min_{u}\int_0^{\infty}(\mathbf{x}^T\mathbf{Q}\,\mathbf{x}+u^T ru)dt \tag{7}$$

To delete this offset, optimum regulator with integral element is tested. New integral variable is defined and sate equation is modified as follow:

$$\dot{\Sigma}=\theta \tag{8}$$

$$\dot{\mathbf{x}}=\begin{pmatrix}0 & 1 & 0\\ -\dfrac{mgl}{J} & -\dfrac{1}{J}\left(b+\dfrac{1}{R}K_TK_V\right) & 0\\ 1 & 0 & 0\end{pmatrix}\begin{pmatrix}\theta\\ \dot{\theta}\\ \Sigma\end{pmatrix}+\begin{pmatrix}0\\ \dfrac{K_T}{JR}\\ 0\end{pmatrix}u \tag{9}$$

Solving the Ricatti equation using Octave, integral type method is executed getting a satisfied result. Here, the optimum regulator without integral variable is compared (Figure 8).

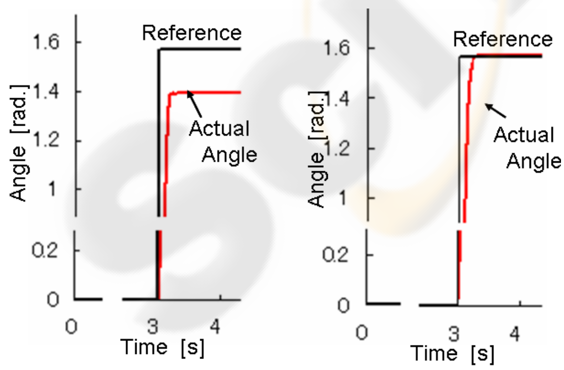


Figure 8: Optimum regulators, right: integral type.

4.4 Discrete Time Control Performances

4.4.1 Three Kinds of Discretizations

In industry applications, DDC is adopted, where the sampling time is from a few to 50 ms depending on the plant dynamics. Figure 9 shows its discrete-time control block diagram, where T_c denotes the calculation time in the sampling period. Authors checked how the control performance depends on the sampling period quantitatively in modern control. Three kinds of discrete-time control schemes are considered.

4.4.2 Conventional Method

The most common method is just to feedback the state vector with gain matrix derived from the continuous type Ricatti Equation as follow:

$$\begin{aligned}&\dot{\mathbf{x}}=\mathbf{A}\mathbf{x}+\mathbf{b}u\\ &y=\mathbf{c}\mathbf{x}\qquad u_i=-\mathbf{k}\mathbf{x}_i\\ &\min_{u}\int_0^{\infty}(\mathbf{x}^T\mathbf{Q}_c\mathbf{x}+u^T r_c u)dt\end{aligned} \tag{10}$$

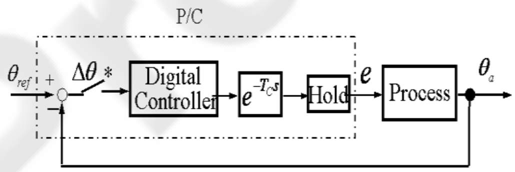


Figure 9: Discrete time Control System.

This method is not discrete and may be called by periodical feedback. This is not optimum any longer, which doesn't utilize the information of sampling period. But this is widely applied in industries because it doesn't need the plant discrete model.

4.4.3 Euler Difference Method

Euler Difference Method can easily discretize the continuous plant model even in a plant controller. Optimal feedback gain matrix **f** is obtained by discrete-time Ricatti Equation through Octave.

$$\begin{aligned}&\mathbf{x_{i+1}}=(\mathbf{I}+\mathbf{A}\Delta T)\mathbf{x_i}+\mathbf{b}u_i\Delta T\\ &y_\mathbf{i}=\mathbf{c}\mathbf{x_i}\qquad u_i=-\mathbf{f}\mathbf{x}_i\\ &\min_{u_\mathbf{i}}\sum_0^{\infty}(\mathbf{x_{i+1}}^T\mathbf{Q}_d\mathbf{x_{i+1}}+u_\mathbf{i}{}^T r_d u_\mathbf{i})\end{aligned} \tag{11}$$

4.4.4 Strict Discretization

The strict discretization needs to solve the continues state variable differential equation :

$$\mathbf{x}(\mathbf{t}) = e^{\mathbf{A}\mathbf{t}}\mathbf{x}(\mathbf{0}) + \int_0^{\mathbf{t}} e^{\mathbf{A}(t-\tau)}\mathbf{b}u(\tau)d\tau \tag{12}$$

$$\therefore \quad \mathbf{x}_{\mathbf{i+1}} = e^{\mathbf{A}\Delta T}\mathbf{x}_{\mathbf{i}} + (\int_0^{\Delta T} e^{\mathbf{A}(\Delta T-\tau)}d\tau)\mathbf{b}u_{\mathbf{i}} \tag{13}$$

Discrete-time state variable plant model is derived:

$$\begin{aligned} &\mathbf{x}_{\mathbf{i+1}} = \mathbf{F}\mathbf{x}_{\mathbf{i}} + \mathbf{g}u_{\mathbf{i}} \\ &u_i = -\mathbf{f}\mathbf{x}_i \qquad y_{\mathbf{i}} = \mathbf{H}\mathbf{x}_{\mathbf{i}} \\ &\min_{u_{\mathbf{i}}} \sum_{i=0}^{\infty}(\mathbf{x}_{\mathbf{i+1}}{}^T\mathbf{Q}_d\mathbf{x}_{\mathbf{i+1}} + u_{\mathbf{i}}{}^T r_d u_{\mathbf{i}}) \end{aligned} \tag{14}$$

Optimum feedback gain matrix $\mathbf{f}$ is also applied here using Octave. This method is strict as a discrete-time modification, but it is pretty difficult to re-calculate F and g in on-line controller in case of the product lot change or parameter change of models in industries

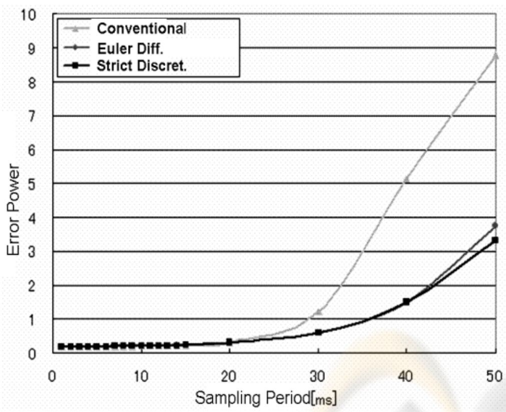


Figure 10: Three performances comparison.

4.4.5 Comparison of Three Performances

Three types of above discrete-time methods have been designed and executed. The step reference change was given from horizontal arm angle to the top dead point. Performances are evaluated by the angle error power. Each weight matrix Q, r of the evaluation function of quadratic form in optimum regulator was chosen to get a good result for 1 ms sampling period control. And this was fixed during different sampling period tests.

There may be some discussion about fixing the evaluation matrix for each to keep the fair comparison. Basically, the control sampling time should be related to the plant time constant. In authors' plants, velocity response time constant is 0.6 s.

Figure 10 shows the results where different sampling periods were adopted for these three methods. These tests gave us the considerable information that the conventional method resulted in poor performance especially when the sampling period is moderately long ,above 20ms. This is why this method does't have the information of sampling period ΔT .

On the other hand, authors got relatively good result by Euler method which is easy to apply compared to the Strict descretization.

In each discrete time control method, it is important that the sampling period should be small under coping over data acquisition and output time through network.

5 CONCLUSIONS

Authors developed practical educational devices for students to make them experience the on-line real time plant control.

Advanced control theories were applied and those results were checked tangibly. Every trouble derived from actual plant such as saturation of the amplifier, cogging of the DC motor, vibration of the plant, static friction and "C" coding is experienced by students.

Authors are now working for 3rd order discrete model with a observer which is giving us delicate problems of sampling period and pole assignments of the regulator and observer

Authors confirmed the importance of tangible control experiment devices for the control engineering education by the 4 year experience of sending out about 40 students to industry fields.

REFERENCES

Hanyuda, S. et. al. 2007. Applications of Pendulum Control System for Developed Educational Experiment Devices, *Risp International Workshop on Nonlinear Circuits and Signal Processing, Shanghai*, pp.189-192, 2007/3

Tahara, J. et al. 2003. Design of Real Time Controller with General Linux , *Society of Signal Processing Applications and Technology of Japan,* pp 17-21 Vol.6, No.2, 2003/6