

A Revised View on Teaching Basic Process Control

Finn Haugen* and Kjell Erik Wolden**

**Telemark University College, Kjolnes ring 56, N-3914, Norway
(Tel: +47 97019215; e-mail: finn.haugen@hit.no).*

***Telemark University College, Kjolnes ring 56, N-3914, Norway
(Tel: +47 35575173; e-mail: kjell.e.wolden@hit.no).*

Abstract: A revised view on teaching basic process control for bachelor and master students is presented. Practical topics are emphasized at the expense of a number of traditional theoretical topics. New teaching material in the form of interactive simulators, instructional videos, and a number of identical portable laboratory rigs have increased the pedagogical quality. The main results are an increase of the practical applicability of the course and more content students and instructors.

Keywords: Control; Teaching; Curriculum; Laboratory; Simulations; Practice; Theory.

1. INTRODUCTION

Teaching a basic process control course is a challenging task. There are lots of practical and theoretical topics to include or exclude. Which topics should be included in an undergraduate course? We, the authors, were brought up on linear systems and frequency response theory. It is easy to mainly reproduce this knowledge in our own teaching, and we did for many years. However, by evaluating our and other's experiences from solving practical control problems and listening to comments from critical students have made us gradually change our view on the teaching basic process control. In this paper we present our changed teaching practices, and the experiences we have made.

The control courses which we discuss here are in the bachelor and master studies at our university college. The students have completed basic courses in calculus, physics, and chemistry before they they start their control course.

2. HOW OUR CONTROL COURSES USED TO BE

We used to teach basic control courses with structure and contents as follows:

- The courses were theory-oriented. The theory covered mathematical phenomenological ("first-principles") modelling, state-space models, linearization of non-linear models, Laplace transform, transfer functions, analytical calculation of dynamic responses, frequency response, stability analysis based on eigenvalues, poles, and frequency response, feedforward control design using transfer function models.
- The first parts of the courses covered systems theory giving the necessary background for the subsequent mainly theory-based control topics.
- Students created simulators or used premade simulators provided by the instructor in SIMULINK.

These simulators did not give a full "real-time" or dynamic experience since the simulation runs are completed in just a few moments.

- The laboratory assignment was based on experiments for frequency response analysis and design.
- Very few laboratory stations were available, so there was a relatively large number of students per group (4 - 6).

3. WHAT WERE THE PROBLEMS?

Every year the courses were evaluated by the students, with the following main student comments:

- The course is relatively demanding because of the theory (mathematics) involved.
- The basic principles of control are difficult to grasp as they are distracted and obscured by theory.
- Students with industrial background claim that certain theoretical topics are not useful since they are actually not applied in practical implementations.

From an instructor's point of view there were also a few problematic issues:

- Due to the very limited possibilities for running laboratory exercises, the number of students per group was as large as 4 - 6 while we assume that the optimal group size is 2 students.
- There was little or practically no time available in the curriculum for covering sensor and actuator technology.

Hence, both students and instructors conceived problems with the courses. In the light of the feedback control principle we felt that changes had to be implemented based on these "offsets".

4. HOW WE CHANGED THE COURSE

4.1 A new overall structure: More practice and control system structure - less theory

Revising a course must of course take into account the aims of the course. Simply stated, the main aim of a basic course in process control, which for a number of the students is their only control course, is to make the students able to *solve practical, real control problems*. Students who intend to prepare for higher, theoretical studies in control, will take other courses which prepare them for these studies.

Over several years we have gradually implemented many changes of the basic control course at the undergraduate level to increase the practical applicability of the course. These changes are described in the following respective subsections. The course amounts to 5 ECTS.

It turns out that after these changes were implemented, students do not convey critical comments like those mentioned in Section 3.

Formerly our courses used to have the following chronological overall structure which roughly is based on "theory before practice":

1. Course overview and motivation
2. Systems theory based on phenomenological mathematical modelling, Laplace transform, transfer functions, and frequency response
3. Principle of feedback control
4. The PID controller
5. Feedback control theory based on transfer functions and frequency response
6. PID controller tuning in both time-domain and frequency domain.
7. Control structures, e.g. cascade control and plantwide control

We have now flipped this order into "practice before theory". The most prevalent changes are that control structures are taught early in the course, and that many theoretical topics are left out. The new structure is as follows.

1. Course overview and motivation
2. Principle of feedback control, incl. several practical cases which the students can simulate using premade simulators.
3. Control structures, e.g. cascade control and plantwide control.
4. Systems theory based on phenomenological mathematical modelling and only first-order linear differential equations
5. The PID controller

6. PID controller tuning with only experimental methods, including Skogestad's model-based method (2003) used as an open-loop step response method.

4.2 Course changes in detail

Interactive real-time simulators

We have included relatively simple, focused simulators in our courses. The simulators are in the SimView library (Haugen, 2012a) which at the present time consists of forty-seven freely available simulators for signals, dynamic systems and control. The simulators are implemented in LabVIEW (National Instruments), but can be run on Windows PCs with the freely available LabVIEW Run-time Engine installed.

The simulators run in real time, or in scaled real time. They are interactive as adjustments of inputs and parameters are immediately made effective.

The SimView simulators are used extensively in the lectures and in the exercises.

Examples of simulators are:

- Time constant
- Liquid tank with valve and pump outlets
- Level control system of wood-chip tank, see Fig. 3.
- Temperature control of liquid tank
- Speed servo
- Control of separator train
- Reverse and direct action of a PID controller, see Fig. 4.
- Anti wind-up
- Cascade control
- Feedforward control

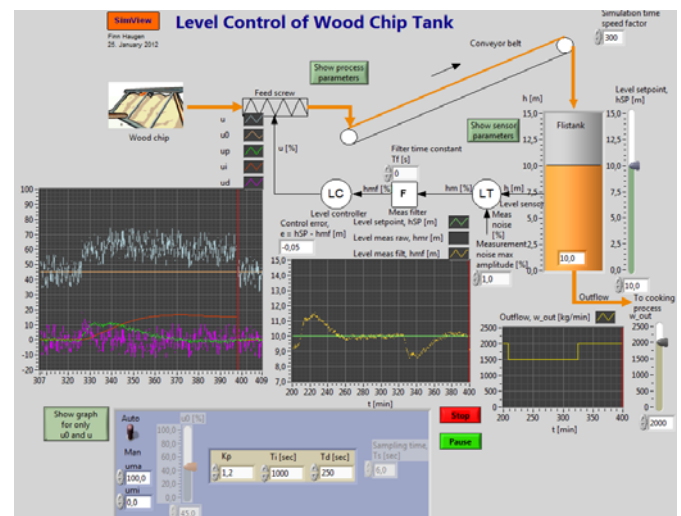


Fig. 3. Front panel of a simulator of a level control system for the wood-chip tank in a paper pulp factory. (The noisy control

signal (on the left) is due to the random measurement noise added to the pure level measurement signal.)

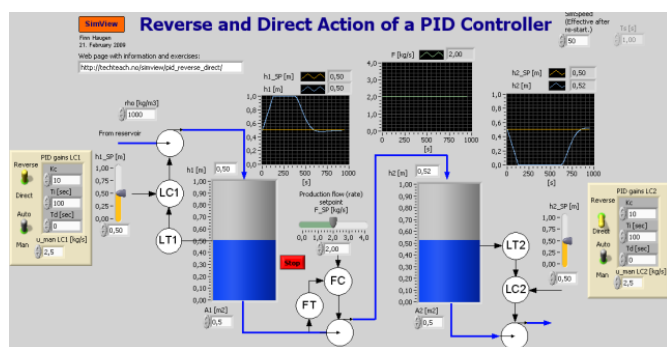


Fig. 4. Front panel of simulator demonstrating reverse and direct action of a controller.

Instructional videos supplementing the lectures

Instructional videos are made available to the students as supplementary learning material, to facilitate "flipped classroom" teaching for students who want to learn topics "at home". The videos are collected in a library named TechVids (Haugen, 2011) which at the present time contains nineteen freely available instructional streaming videos covering dynamics and control topics. The library also includes instructional videos for computer tools as MATLAB, SIMULINK, and LabVIEW.

Most videos are recordings of the instructor explaining orally the topics while displaying Powerpoint slides, followed by the instructor running simulators. The same simulators are available for the students to run while playing the video.

Examples of video titles:

- Feedback control
- Feedforward control
- How a control system may become unstable
- Reverse or direct action in the PID controller?
- Anti windup in a PI(D) controller
- PID controller tuning with Ziegler-Nichols' method
- Gain scheduling
- Cascade control
- Flow smoothing with buffer tank
- Time-constant and integrator dynamics

Parallel laboratory rigs

The basic control course should include a practical laboratory exercise covering basic control topics. Furthermore, to optimize the learning outcome, the students should be organized in groups with two students per group. To these ends, we have constructed in-house eleven parallel (identical) portable laboratory rigs in the form of an air heater, see Fig. 5.

The temperature of the air outlet is to be controlled with a PI controller. (We also have six equal portable water tanks at our disposal for similar exercises.)

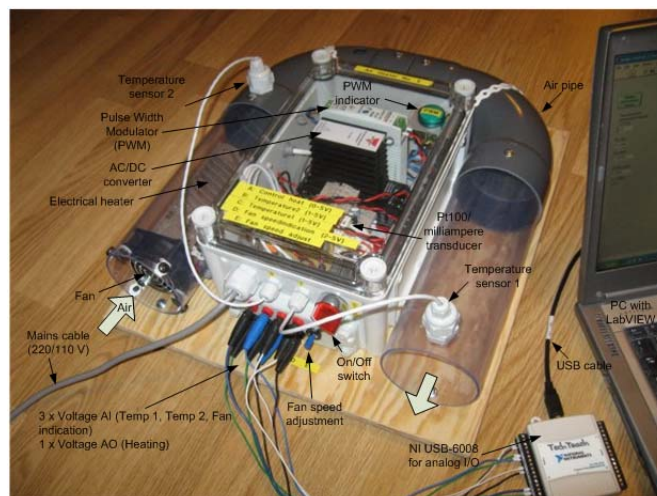


Fig. 5. Air heater + I/O device + PC used in a laboratory exercise about basic PID control

The controller is implemented in a premade LabVIEW program which the students download from the Internet, see Fig. 6. The computer program communicates with the air heater via a USB-based I/O device (NI USB-6008). The exercise covers controller tuning, comparing closed loop control with open-loop control ("blind control"), stability, reverse and direct controller action, measurement noise filtering, and data logging. The exercise lasts for 5 hours. The students are required to write a short report to be delivered at the end of the scheduled exercise time. No LabVIEW programming skills are necessary.

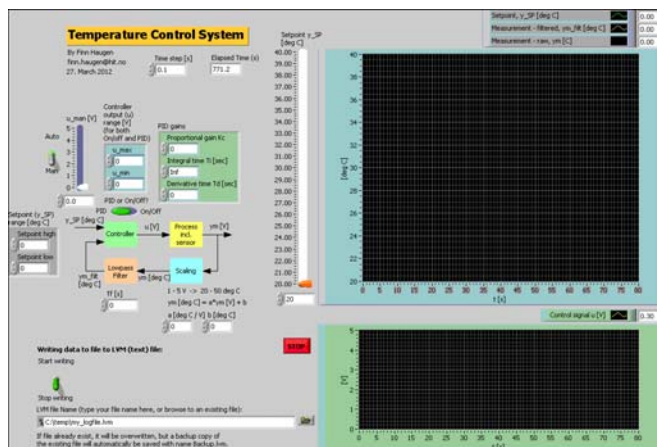


Fig. 6. Front panel of LabVIEW program used in the laboratory exercise about temperature control.

Documenting control structures using Piping & Instrumentation Diagrams with ISA 5.1 standard

In the beginning of the revised course the students become familiar with the ISA (International Society of Automation) 5.1 standard for Piping & Instrumentation Diagrams (P&ID)

available at the home page of ISA, and presented by e.g. McAviney *et al.* (2004). Later in the course a small project assignment about drawing a P&ID depicting the structure of a control system using Visio (Microsoft) as the drawing tool. Visio contains symbols for P&IDs.

Understanding the PID controller in discrete-time domain rather than in continuous-time domain

In the course we introduce the continuous-time PID controller as

$$u(t) = u_{man} + K_c e(t) + \frac{K_c}{T_i} \int_0^t e(\tau) d\tau + K_c T_d \frac{de(t)}{dt} \quad (1)$$

However, we have found that is easier to explain the behaviour of the PID controller function in the discrete-time domain than in the continuous-time domain. Moreover, the discrete-time PID controller is more realistic than the continuous-time controller. Therefore, we show the students how to discretize (1) by applying simple numerical approximations to the integral and derivative terms. The result is the following (here assumed well-known) PID algorithm which is (almost) ready for implementation in a computer:

$$u(t_k) = u_{man} + u_p(t_k) + u_i(t_k) + u_d(t_k) \quad (2)$$

where

$$u_p(t_k) = K_c e(t_k) \quad (3)$$

$$u_i(t_k) = u_i(t_{k-1}) + \frac{K_c T_s}{T_i} e(t_k) \quad (4)$$

$$u_d(t_k) = K_c T_d \frac{e(t_k) - e(t_{k-1})}{T_s} \quad (5)$$

where T_s is the sampling time. We have found that it easier to explain the ability of the integral term to force the steady-state control error to zero using (4) than using (1), and that it is easier to explain the ability of the derivative term to speed up, but also to dampen (stabilize), the control using (5) than using (1). Also, explaining amplification of measurement noise and the derivative kick is easier using (5) than using (1).

Measurement noise and how to cope with it

Measurement noise is prevalent in most practical control loops. The derivative term of a PID controller may cause the control signal to become very noisy. We let the students get experience with the problematic measurement noise on simulators and on real experiments, and let them observe the importance of using a low-pass measurement filter and/or deactivating the derivative term to cope with the problems. A measurement filter is typically available in industrial controllers in the form of either a time constant filter or a moving average filter. The filters are presented in the form of discrete-time algorithms (difference equations). For the time-constant filter we start out with the differential equation defining the filter, and then show how to discretize the filter. As for the PID controller, we think that it easier to explain the discrete-time filter than its continuous-time counterpart. We present filter attenuation of random noise standard deviation

as a function of filter time-constant and sampling time. We compare the time-constant filter and the moving average filter.

Experimental PID tuning methods

Previously we taught students how to tune a PID controller in the frequency domain. However, we must admit that we do not know many practical process control loops where the PID controller is tuned in the frequency domain. Therefore, it is hard to defend spending time on these methods in our basic courses. We decided instead to focus on simple, experimental methods like the Ziegler-Nichols' Ultimate Gain method (1942), in combination with Åström-Hägglund's Relay-tuner which we relate to industrial auto-tuners, the Good Gain method for PI tuning (Haugen, 2012b) which copes with a couple of problematic issues in the Ziegler-Nichols' method, and Skogestad's model-based method (2003) used as an experimental open-loop step response method.

One strong aspect of Skogestad's method which we exploit is that the PID parameters are expressed as functions of the process parameters, thereby providing useful rules for adjusting the controller parameters if parameters in the sensor, the actuator, or the process change.

Feedforward control with nonlinear models

We no longer teach transfer function based feedforward control. Instead we show how to use nonlinear dynamic phenomenological models directly in the feedforward controller. Simulated applications are dynamic positioning of ships and temperature control of heated tanks. We also teach experimental feedforward design based on table-lookup. The table contains a number of disturbance measurement values and the corresponding values of the feedforward control signal found as the PID control signal in steady state. A simulated application is temperature control of an heat exchanger.

Leaving out Laplace transform and transfer functions

In our short 5 ECTS basic process control course we have taken the dramatic step of leaving out the Laplace transform and transfer functions from the curriculum. In our experience these theoretical tools do not play an important role in practical control design and implementation. One argument for keeping these tools is that they are used to calculate time responses. However, we limit ourselves to first-order dynamics, and for such systems, differential equations are sufficient for calculating time responses.

Transfer functions are very useful for frequency response and stability analysis. While this is true, we have even left out frequency response and theoretical stability analysis, see below.

Leaving out time-domain analysis using frequency response

Traditionally, Bode diagrams of frequency response are used to interpret time-domain properties of control systems.

Although the frequency response provides some insight in the dynamic behaviour of dynamic systems and control systems, the real world signals are seldom sinusoidal, and hence the usefulness of the frequency response for conveying time-domain behaviour of practical control systems is questionable. We believe more in obtaining insight into dynamic and steady-state (static) behaviour through simulations and experiments with more realistic signal forms like steps, constants, and random noise.

Leaving out frequency response is in agreement with the low priority given to this topic as pronounced by industry people in the reports by Edgar *et al.* (2006) and Haugen (2009).

Leaving out theoretical stability analysis, but retaining the notion of stability

We have left out theoretical stability analysis from our basic process control course. This is motivated by the perceived lack of direct applicability of theoretical stability analysis - whether based on poles/eigenvalues or frequency response - to given practical control systems.

Still, stability is a very important concept for real control systems which we, therefore, do include in our course. We introduce stability in an intuitive, experimental way by assuming that a PI(D) control loop may get stability problems due to an increase in at least one of the following loop parameters:

- Loop gain
- Loop time-delay
- Loop time-constant

We emphasize that these increases may come in any of the blocks in the control loop, e.g. in the actuator, in the process (plant), in the sensor, in the measurement filter, or in the controller. Often we use the simulator of the level control system shown in Fig. 3 to demonstrate these effects.

We also apply Skogestad's PID controller method to see how the controller can be adjusted to retain the control loop stability if control loop parameter change. For example, assume that for some reason the transport delay of the conveyor belt in the wood-chip tank system shown in Fig. 3 is increased by a factor of three. This will make the control system unstable. How should you adjust the PI controller parameters to retain stability? Using Skogestad's method you can find that the controller gain should be reduced by a factor of three and the integral time should be increased by a factor of three.

If you leave out stability analysis with frequency response from a course, you lose one traditional tool for defining stability margins like gain margin and phase margin. However, the gain margin can still be defined using the experimental behaviour of a practical or a simulated control loop. Even the phase margin can be defined using the experimental behaviour of a practical or a simulated loop (Haugen, 2012b), since the phase margin, PM, can be calculated as $360 \cdot T_{\text{delay}} / P_u$ where T_{delay} is the time-delay increase that causes marginal stability with sustained oscillations and P_u is the period of these oscillations.

However, up to now, we have not introduced the notion of experimental phase margin in our course, but we plan to do so Fall 2013.

Although we leave out much theoretical analysis (and design), we emphasize to the students the importance of such analysis (and design) in control applications like motion control, and in research. Automation students do learn these theories, but after the "theory-free" introduction.

Sensor and actuator technology

Leaving out several theoretical topics releases time for covering sensor and actuator technology, which makes the course appear more complete from a practical point of view.

Implementation of simulators

Dynamic simulators can be very useful for control system testing and training. Simulators can even be used for controller tuning if the underlying model is sufficiently accurate.

In a short course about basic process control there is virtually no time to cover implementation of dynamic simulators for control systems. In courses which do allow for simulator implementation, we recommend block diagram based tools like SIMULINK and LabVIEW because of the graphical model representation. Our experience from courses which cover simulator building is, roughly said, that it suffices to teach the students how to represent a scalar nonlinear differential equation in a block diagram. We have found that transfer function blocks are hardly needed; A time-constant measurement filter can be implemented from its differential equation directly, making it easy to implement a proper initial value of the filter output (typically the initial value is a non-zero value, equal to the initial value of the pertinent process variable to be filtered). Built-in blocks for PID controllers, time-delays, etc. should be used unless the aim is to build such blocks from scratch.

An effective way to implement a scalar nonlinear differential equation is using only two blocks, namely an integrator block which integrates the time-derivative, and one "text code block" in which you write the right-side of the nonlinear differential equation. The output of this text-code block is the time-derivative which is connected to the input of the integrator block. Depending on the differential equation, there may be a feedback from the integrator output to the block. Fig. 7 illustrates this approach. The figure shows a SIMULINK block diagram of the level control system for the wood-chip tank depicted in Fig. 3. The block named MATLAB Function contains the following MATLAB code which comprises the right-side of the mass balance of the chip in the tank:

```
function dh_dt = func_dh_dt(w_out,w_in,params)
rho=params(1);
A=params(2);
dh_dt=(1/(rho*A))*(w_in/60-w_out/60);
```

The MATLAB function is a versatile block in SIMULINK as almost any MATLAB code can be written in the block, and

the input and output terminals of the block are created automatically as you edit the function header.

In a LabVIEW implementation of this level control system the Formula Node can be used in the same way as the MATLAB Function block in SIMULINK. In the Formula Node, C code can be written.

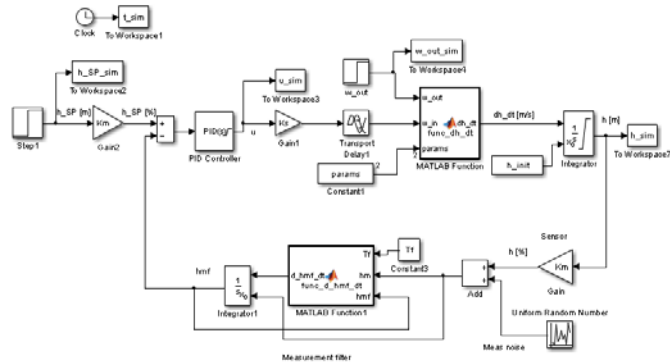


Fig. 7. A SIMULINK block diagram of the level control system for the wood-chip tank depicted in Fig. 3.

5. EXPERIENCES

Our experiences from changing the focus of the teaching of basic control from theory to practice are positive. This is mainly because we now find that the topics we teach are more applicable to real control problems than were many of the topics we used to teach. Furthermore, the course evaluations by students are overall more positive than before. We assume this is due to the reduced theoretical contents of the course, but also to the applicability of the selected topics.

REFERENCES

- Edgar, Th.F., Ogunnaike, B.A., Downs, J.J., Muske, K.R. Bequette, B.W. (2006). Renovating the undergraduate process control course. *Computers & Chemical Engineering*, 30, 1749–1762.
- Haugen, F. (2009). Industrifolks syn på automatiseringsutdanningen, in Norwegian. (English title: Industrial perspective on control education.) *AMNytt*, 2, 2009. The article is available at <http://techteach.no/publications/amnytt/web>.
- Haugen, F. (2011). TechVids. <http://techteach.no/techvids>.
- Haugen, F. (2012a). SimView. <http://techteach.no/simview>.
- Haugen, F. (2012b). The Good Gain method for simple experimental tuning of PI controllers, *Modeling, Identification and Control*, 33, 4:141-152. doi: 10.4173/mic.2012.4.3.
- McAviney, Th., and Mulley, R. (2004). *Control System Documentation*. International Society of Automation (ISA).
- Skogestad, S. (2003). Simple Analytical Rules for Model Reduction and PID Controller Tuning, *J. Process Control*, 13, pp. 291-309.
- Ziegler, J.G., Nichols, N.B. (1942). Optimum Settings for Automatic Controllers, *Trans. ASME*, Vol. 64, pp. 759-768.