CIS009-2, Mechatronics
Control Systems

David Goodwin

Department of Computer Science and Technology
University of Bedfordshire

15^{th} November 2012

Outline

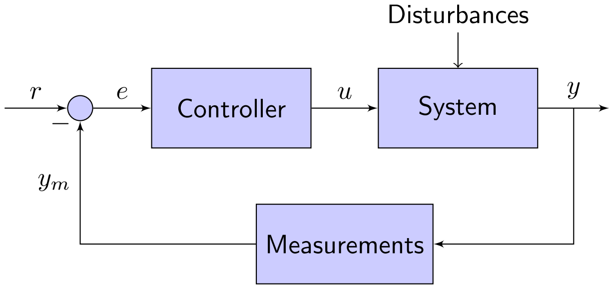
Department of Computer Science and Technology
University of Bedfordshire

Control systems

Control system principles

Block Diagrams

- Electronic systems and circuits are often represented by block diagrams.
- Shorthand pictorial representation of the cause-and-effect relationship between the input and output in a real system.
- Characterise functional relationships between components.
- Not necessary to understand the functional details of a block to manipulate a block diagram.



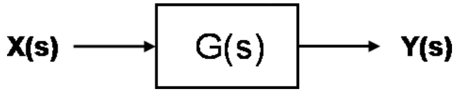
Department of Computer Science and Technology University of Bedfordshire

Block Digram Basics

Transfer function

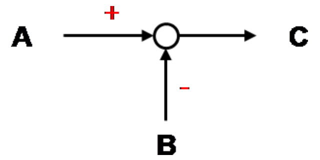
Department of Computer Science and Technology
University of Bedfordshire

- The basic building block of a control system block diagram.
- The transfer function block represents a portion of the system - plant, sensor, controller, or pre-filter.
  - $G(s) = \dfrac{Y(s)}{X(s)}$
  - $G(s)$ must be a Linear Time-Invariant System (output does not depend explicitly on time).
  - $G(s)$ is property of the system independent of the input to that system.



Block Digram Basics
Summing junction

- A summing junction is represented by a circle.
- The input signals are signed (can be + or -) and the sign with which they are added to the output is displayed on the signal line leading to the junction.
    - $C = A + (-B)$
    - A is a signed input, B is a signed input, and C is the output.



Department of Computer Science and Technology University of Bedfordshire

Block Digram Basics
Cascade blocks

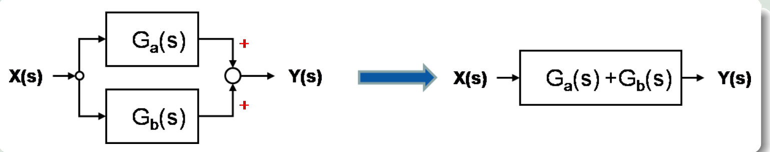
- The 2 blocks in series (also known as cascade blocks) can be combined (multiplication) into a single block as long as there is nothing between them.
  - The way these two blocks were combined is only valid for SISO (single input single output) systems.

Department of Computer Science and Technology
University of Bedfordshire

Block Digram Basics

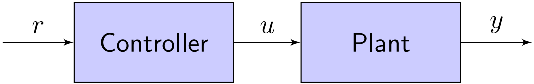
Parallel blocks

- The 2 blocks in parallel can be combined (added) into a single block as long as there is nothing between them.



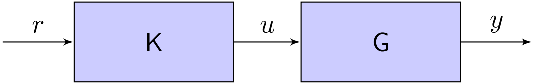
Department of Computer Science and Technology
University of Bedfordshire

Open-loop control



Department of Computer Science and Technology
University of Bedfordshire

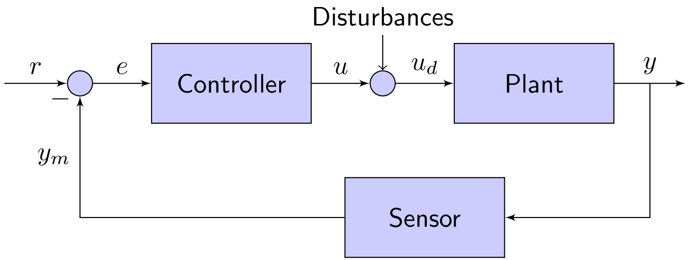
Open-loop control



- Then the resulting Loop Transfer Function can be found by inspection:
- Open-loop transfer function $= \dfrac{y}{r} = GK$
- The Open-Loop Transfer Function is GK which appears to be backwards given that the blocks appeared as KG when the system is read left to right.
- For SISO systems the order is arbitrary. However, for MIMO systems this backwards order must be maintained.
- This is because the system is represented with matrices and in Linear Algebra (matrix math) the order of execution is vital - there is no Commutative property for matrix multiplication.

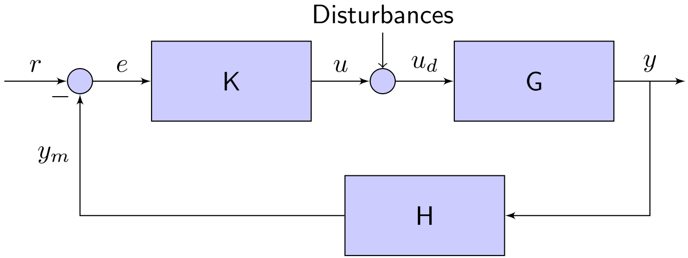
Department of Computer Science and Technology
University of Bedfordshire

Closed-loop control



Department of Computer Science and Technology University of Bedfordshire

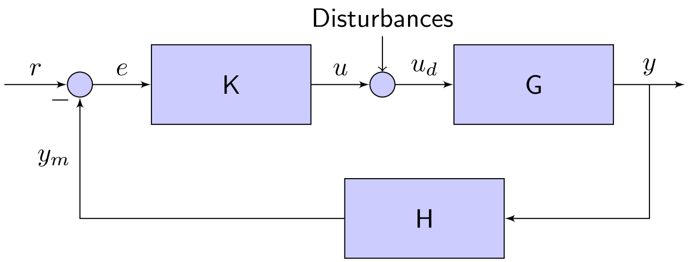
Closed-loop control



- The Closed Loop Transfer Function is
- $y = GKe = GK(r - y_m)$
- where $y_m = Hy$

Department of Computer Science and Technology
University of Bedfordshire

Closed-loop control



$$
\begin{aligned}
y =& GK(r - Hy) \\
=& GKr - GKHy \\
y + GKHy =& GKr \\
(1 + GKH)y =& GKr
\end{aligned}
$$

Closed-loop Tansfer Function $= \dfrac{y}{r} = \dfrac{GK}{(1 + GKH)}$

Department of Computer Science and Technology
University of Bedfordshire

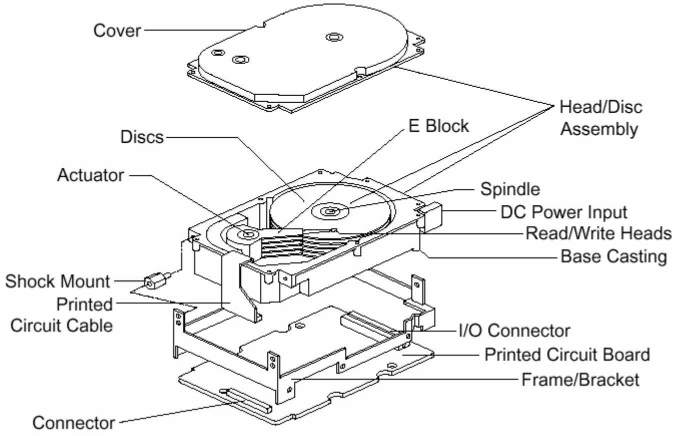
Closed-loop control

Department of Computer Science and Technology
University of Bedfordshire

$$\text{Closed-loop Tansfer Function} = \frac{y}{r} = \frac{GK}{(1 + GKH)}$$

- The numerator is the forward (open-loop) gain from r to y.
- The denominator is one plus the gain in going around the feedback loop, the so-called loop gain.
  - If $|GK| \gg 1$, and if $|H| \approx 1$, then y is approximately equal to r and the output closely tracks the reference input.

An example of Closed-loop systems control
Disk drive read system



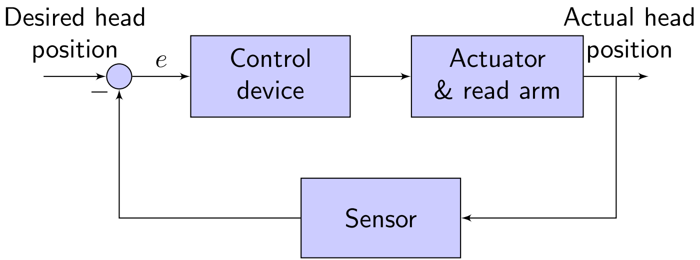
An example of Closed-loop systems control

Disk drive read system

- A hard disk uses round, flat disks called platters, coated on both sides with a special media material designed to store information in the form of magnetic patterns.
- The platters rotate at high speed, driven by a special spindle motor connected to the spindle.
- Special electromagnetic read/write devices called heads are mounted onto sliders and used to either record information onto the disk or read information from it.
- The sliders are mounted onto arms, all of which are mechanically connected into a single assembly and positioned over the surface of the disk by a device called an actuator.
- A logic board controls the activity of the other components and communicates with the rest of the computer.

An example of Closed-loop systems control
Disk drive read system

- We can model the system as a closed-loop control system



Department of Computer Science and Technology University of Bedfordshire

System Responses

Department of Computer Science and Technology
University of Bedfordshire

System Responses
Controllers

- To satisfy our design criteria we might need to apply feedback to our open loop system.
- Engineers use basically four techniques to achieve this. These techniques are::
  - PID
  - Root-locus
  - Frequency response
  - State space
- Controllers are not magic. It should not be assumed that every system can be controlled or that desired performance can be achieved for any given system if only the correct controller were employed. Proper system design is always required.

PID controller

A generic controller

- A Proportional-Integral-Derivative controller (PID controller) is a generic controller widely used in industrial control systems.
- The PID control equation involves three separate parameters;

  Proportional term - responds instaneously to the current error (providing instaneous response)

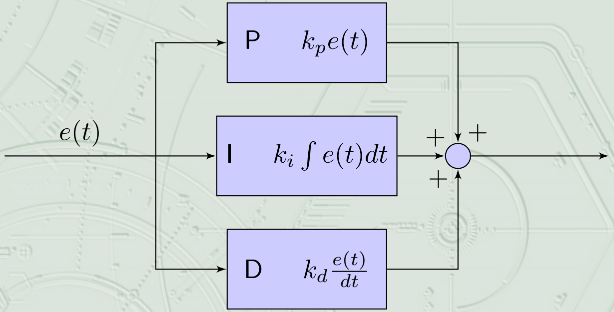
  Integral term - responds to the accumulation of errors (providing a slow response that drives the steady-state error towards 0)

  Derivative term - responds to the rate at which the error is changing (providing some anticipatory response)

- Their respective weighting determines the controls response.

Department of Computer Science and Technology
University of Bedfordshire

PID controller
A generic controller

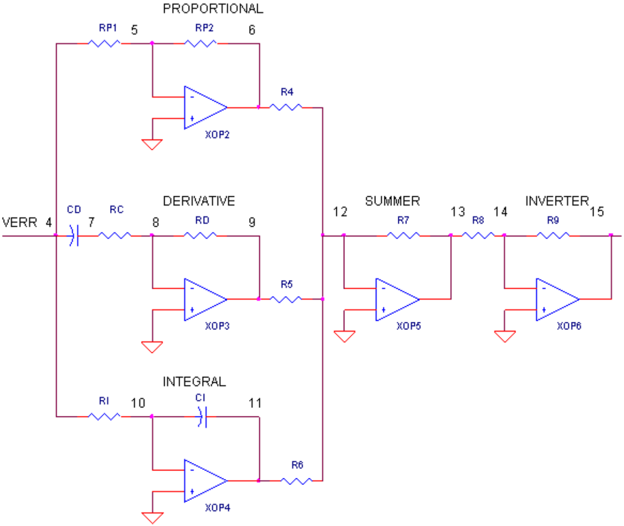


- PID control combines proportional control, integral control, and derivative control in parallel.

$$K = k_p e(t) + k_i \int e(t)dt + k_d \frac{e(t)}{dt}$$

Department of Computer Science and Technology University of Bedfordshire

PID controller
Schematic with Op Amps



Department of Computer Science and Technology
University of Bedfordshire

Step Response

A test for system stability

- Step response is the time behaviour of the outputs of a general system when its inputs change from zero to one in a very short time.
- knowing how the system responds to a sudden input is important because large and possibly fast deviations from the long term steady state may have extreme effects on the component itself.
- The overall system cannot act until the component's output settles down to some vicinity of its final state, delaying the overall system response.
- Knowing the step response of a dynamical system gives information on the stability of such a system, and on its ability to reach one stationary state when starting from another.
- `http://www.youtube.com/watch?v=cNZPRsrwumQ` shows a good example of a step response (around 35 second into the video when the robot is kicked from the side).

Step Response

A test for system stability

- The step response can be described by the following quantities related to its time behavior:

Overshoot - refers to an output exceeding its final, steady-state value

Rise time - time taken by a signal to change from a specified low value to a specified high value

Settling time - the time required for the response curve to reach and stay within a range of certain percentage (usually 5% or 2% - characterised by the ringing, *see below*) of the final value.

Ringing - unwanted oscillation of a signal, closely related to overshoot, generally occurring following overshoot. This is characterised by the steady-state error (S-S error)

Department of Computer Science and Technology
University of Bedfordshire

Effects on controller parameters

- The effects of each of controller parameters, K_p, K_d, and K_i on a closed-loop system are summarized in the table below.

K RESPONSE	RISE TIME	OVERSHOOT	SETTLING TIME	S-S ERROR
K_p	Decrease	Increase	Small Change	Decrease
K_i	Decrease	Increase	Increase	Eliminate
K_d	Small Change	Decrease	Decrease	No Change

Department of Computer Science and Technology
University of Bedfordshire

Matlab & Octave

Department of Computer Science and Technology
University of Bedfordshire

MATLAB
with Control System Toolbox

- Numerical computation, visualization, and programming.
- Analyze data, develop algorithms, and create models and applications.
- The language, tools, and built-in math functions enable you to explore multiple approaches and reach a solution faster than with spreadsheets or traditional programming languages, such as C/C++ or Java.
  - Control System Toolbox provides industry-standard algorithms and tools for systematically analyzing, designing, and tuning linear control systems.
  - Transfer function, state-space, pole-zero-gain, or frequency-response model.

GNU Octave
compatible with MATLAB

- GNU Octave is a high-level language for numerical computations.
- Language is mostly compatible with MATLAB.
- As part of the GNU Project, it is free software under the terms of the GNU General Public License (GPL).
  - The Octave language is an interpreted programming language. It is a structured programming language (similar to C) and supports many common C standard library functions, and also certain UNIX system calls and functions.
  - Octave programs consist of a list of function calls or a script.
  - The syntax is matrix-based and provides various functions for matrix operations. It supports various data structures and allows object-oriented programming.
  - Syntax is very similar to MATLAB, and careful programming of a script will allow it to run on both Octave and MATLAB.
  - Can be used together with QtOctave to gave a MATLAB-like IDE.

Resources

Department of Computer Science and Technology
University of Bedfordshire

Resources

Useful literature

- R. Manchini (2002) - Op Amps for everyone: http://www.ti.com/lit/an/slod006b/slod006b.pdf
- ControlTheoryPro.com (2012) - http://wikis.controltheorypro.com/index.php?title=Main_Page
- Op Amp PID controller scematics - http://www.ecircuitcenter.com/Circuits/op_pid/op_pid.htm
- MATLAB community http://www.mathworks.co.uk/matlabcentral/ including forums, file exchange, and full documentation on syntax (http://www.mathworks.co.uk/help/matlab/)

GNU Octave

Setup instructions for windows

- Download Octave from `http://octave.en.softonic.com/download` and run the exe, then follow the instructions from the wizard.
- Download qtoctave, I got mine from `http://www.outsch.org/wp-content/uploads/2011/01/qtoctave-0.10.1-win32.zip`
- Open qtocatve, select config; general configuration from the drop down menu at the top of the window, then select the octave menu
- In the octave path click select, then navigate to where you installed octave then select the octave.exe in the bin folder
- Create a working directory, where ocatve will search for your .m files to run (this is important)