Welcome to SENG 480B / CSC 485B / CSC 586B Self-Adaptive and Self-Managing Systems

Dr. Hausi A. Müller
Professor
Department of Computer Science
University of Victoria

http://courses.seng.uvic.ca/courses/2013/summer/seng/480b
http://courses.seng.uvic.ca/courses/2013/summer/csc/485b
http://courses.seng.uvic.ca/courses/2013/summer/csc/586b

1

Announcements

- Marking
  - A2 graded
  - Midterm graded
  - A3 being graded
  - Marks are posted on website
- Group presentations of A3
  - Excellent
  - One more today
- A4
  - posted
  - Due Thu, Aug 1
- Grad student presentations
  - July 23 – July 31
- Teaching Eval
- Review for final exam
  - Wed, Aug 7
- Last day of classes
  - Wed, Aug 7
- Final exam
  - Tue, Aug 13, 9:00-12:00 am in ECS 124
  - Closed books, closed notes
  - Materials: entire course
  - Format: like midterm

2

Graduate Student Research Paper Presentations

- Garlan, D., Cheng, S.-W., Huang, A.-C., Schmerl, B., Steenkiste, P.: Rainbow: Architecture-Based Self-Adaptation with Reusable Infrastructure. *IEEE Computer* 37(10):46-54 (2004) — **Angela Rook, July 23**
- Kramer, J., Magee, J.: Self-Managed Systems: An Architectural Challenge. In: *ACM /IEEE International Conference on Software Engineering 2007 Future of Software Engineering (ICSE),* pp. 259-268 (2007) — **Pratik Jain, July 23**
- Oreizy, P., Medvidovic, N., Taylor, R.N.: Runtime Software Adaptation: Framework, Approaches, and Styles. In: *ACM/IEEE International Conference on Software Engineering (ICSE 2008),* pp. 899-910 (2008) —**Alessia Knauss, July 24**
- Brun, Y., Di Marzo Serugendo, G., Gacek, C. Giese, H. Kienle, H.M., Litoiu, M., Müller, H.M., Pezzè, M., Shaw, M.: *Engineering Self-Adaptive Systems through Feedback Loops.* SE for Self-Adaptive Systems, pp. 48-70 (2009) — **Samra Ramandeep, July 24**
- Kaushik, R.T., Cherkasova, L., Campbell, R.H., Nahrstedt, K.: Lightning: self-adaptive, energy-conserving, multi-zoned, commodity green cloud storage system, *ACM International Symposium on High Performance Distributed Computing (HPDC 2010),* 332-335 (2010) — **Andi Bergen, July 26**

3

Graduate Student Research Paper Presentations

- Villegas, N.M., Müller, H.A., Tamura, G., Duchien, L., Casallas, R.: A framework for evaluating quality-driven self-adaptive software systems. In: *Proc. 6th Int. Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS 2011),* pp. 80-89 (2011) — **Lorena Castaneda, July 30**
- Ebrahimi, S., Villegas, N.M., Müller, H.A., Thomo, A.: SmarterDeals: a context-aware deal recommendation system based on the SmarterContext engine. *CASCON 2012:* 116-130 (2012) — **Nina Taherimakhsousi, July 30**
- McKinley, P.K., Sadjadi, M., Kasten, E.P., Cheng, B.H.C.: Composing Adaptive Software. *IEEE Computer* 37(7):56-64 (2004) — **Carlos Gomez, July 31**
- Tewari, V., Milenkovic, M.: Standards for Autonomic Computing, *Intel Technology Journal,* 10(4):275-284 (2006) — **Nitin Goyal, July 31**

4

Guidelines for Grad Presentations

- Format of presentation
  - Presentation 15-20 mins
  - Q&A 5 mins
  - Practice talk (!)
- Slides
  - High quality
  - Submit slides 2 days before presentation to instructor for approval
  - Submit final slides 1 day after presentation for posting on website
- Talk outline
  - Motivation
  - Problem
  - Approach
  - Relation to what we heard in the course so far
  - Contributions of the paper
- Avoid plagiarism!!
  - Prepare your own talk
  - Critical

5

Assignment 4

Part I

In Part I (a) you are to write a summary of the following paper:

H.A. Müller and N.M. Villegas. Runtime Evolution of Highly Dynamic Software Systems." in *Evolving Software Systems*, T. Mens, A. Serebrenik, and A. Cleve (eds.), Springer, 39 pages, July 2013. In Press.

In Part I (b) you are to write a recommendation on how to improve the paper.

The answers to this question should fit into approximately 3-4 typeset pages.

Do not copy verbatim from any source. Cite your sources.

Additional motivation: This paper summarizes a significant part of this course self-adaptive and self-managing systems. You will have access to a hard copy of this paper during the final exam. The answers to selected final exam questions can be found in this paper.

6

Assignment 4 — Groups

Part II - Group Project (3-4 people per group)

Control theory offers several reference models for realizing *adaptive control* where the target system but also the controller is adjusted over time guaranteeing global stability and convergence. Two famous models are *reference adaptive control (MRAC)* and *model identification adaptive control (MIAC)*.

In Part II you are to design an *innovative* application that uses an MRAC or MIAC reference model. Immerse yourself in adaptive control and then design a truly innovative application that could be platform for a company.

Document your design experience in the form of a tutorial.

Implementation (e.g., using Matlab) is optional.

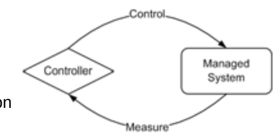
The answers for this question should fit into approximately 3–4 typeset pages.

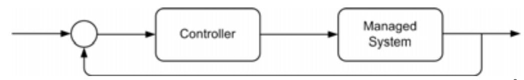
Do not copy verbatim from any source. Cite your sources.

7

Feedback Control System

- **Merriam-Webster's Online Dictionary**
  the return to the input of a part of the output of a machine, system, or process
  - producing changes in an electronic circuit to improve performance
  - an automatic control device to provide self-corrective action





8

Control Theory

- A theory that deals with influencing the behavior of dynamical systems
- An interdisciplinary subfield of science, which originated in engineering and mathematics

9

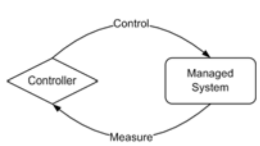
Origins of Control Theory

- Control systems date back to antiquity
- James Maxwell (1831-1879) started the field in 1868 analyzing the dynamics analysis of the centrifugal governor
- Routh (1831-1907) abstracted Maxwell's results for the general class of linear systems in 1877
- Hurwitz (1859-1919) analyzed system stability using differential equations in 1877
- Laplace (1749-1827) invented the Z-transform used to solve discrete-time control theory problems. The Z-transform is a discrete-time equivalent of the Laplace transform.
- Alexander Lyapunov (1857–1918) developed stability theory.
- Harry Nyquist (1889–1976), developed the Nyquist stability criterion for feedback systems in the 1930s.

10

Control Systems are Ubiquitous

- Cruise control
- Fuel injection
- Flight control
- Climate Control
- Health Care
- Model Helicopters
- Quadcopters
- Rumba
- iRobots
- Radiotheraphy



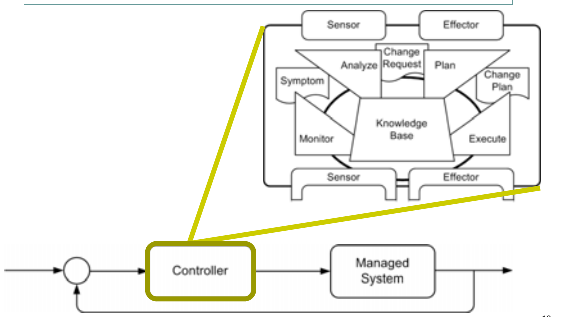
11

Control System Goals: Self-Management

- Regulation
  - Thermostat, target service levels
- Tracking
  - Robot movement
  - Adjust TCP window to network bandwidth
- Optimization
  - Best mix of chemicals
  - Minimize response times

12

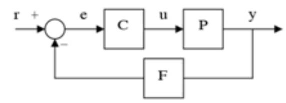
Controller as an Autonomic Element



13

Closed Loop Controller or Feedback Controller

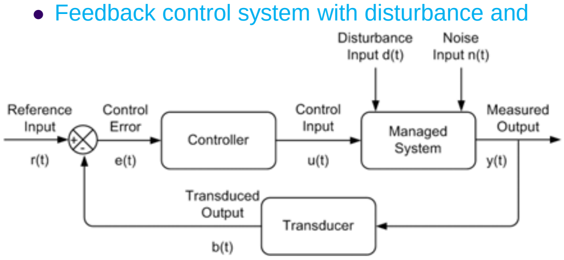
- The output *y(t)* of the feedback system is fed back through a sensor measurement *F* to the reference value *r(t)*.
- The controller *C* then takes the error *e* (difference) between the reference and the output to change the inputs *u* to the control process *P*.
- SISO
  - Single-input-single-output (*SISO*) control system
  - Variables are simple scalar values (i.e., r(t), e(t), u(t), y(t)
- MIMO
  - Multi-Input-Multi-Output systems, with more than one input/output, are common
  - Variables are vectors



14

Realization of a Dynamic Architecture

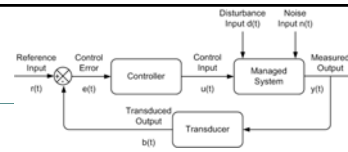
- Feedback control system with disturbance and



Hellerstein, Diao, Parekh, Tilbury: Feedback Control of Computing Systems. John Wiley & Sons (2004)

15

Realization of a Dynamic Architecture



- Reference input
  - Goal, objectives, specified desired output
- Control Error
  - Reference input minus transduced output
- Control Input
  - Parameters which affect behavior of the system—number of threads, CPU, memory
- Disturbance input
  - Affects control input—arrival rate
- Controller
  - Change control input to achieve reference input—design is based on a model of the managed system
- Managed system
  - Dynamical system, process, plant—often characterized by differential equations
- Measured output
  - Measurable feature of the system—response time
- Noise input
  - Affects measured output
- Transducer
  - Transforms measured output to compare with reference

Controller Algorithm based on Managed System Model

- *"All models are wrong, some models are useful."*
  - generally attributed to the statistician George Box
- The design of the controller algorithm is based on a model of the managed system or process
- Approaches
  - Analytical modeling: physical and mathematical laws
  - Experimental modeling: data fitting from observed input and output
- The control algorithm changes *u(t)* based on the error *e(t) = r(t) - b(t)*
  - Proportional—if *e(t)* is high, then *u(t)* should be high
  - Integrative—eliminates transients; sum of all previous errors
  - Derivative—anticipate the trends; rate of change of the error
  - PID—computation based on the error (proportional), the sum of all previous errors (integral) and the rate of change of the error (derivative)

17

PID Controller

- The PID algorithm is the most popular feedback controller algorithm used
- It is a robust easily understood algorithm that can provide excellent control performance despite the varied dynamic characteristics of processes
- PID algorithm consists of three basic modes:
  - **P**roportional mode
  - **I**ntegral mode
  - **D**erivative mode

18

P, PI, or PID Controller

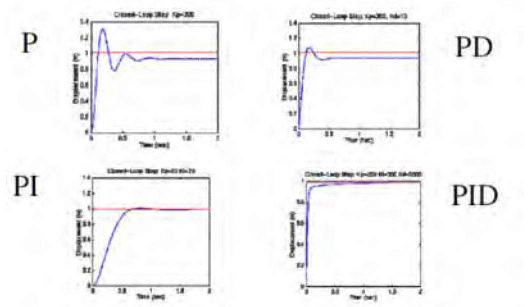
- When utilizing the PID algorithm, it is necessary to decide which modes are to be used (P, I or D) and then specify the parameters (or settings) for each mode used.
- Generally, only three basic algorithms are used: P, PI or PID

$$u(t) = \underbrace{K_p}_{\text{proportional gain}} e(t) + \underbrace{K_i}_{\text{integral gain}} \int_0^t e(t)dt + \underbrace{K_d}_{\text{derivative gain}} \frac{de(t)}{dt}$$

Controller Effects

- A proportional controller (P) reduces error responses to disturbances, but still allows a steady-state error.
- When the controller includes a term proportional to the integral of the error (I), then the steady state error to a constant input is eliminated, although typically at the cost of deterioration in the dynamic response.
- A derivative control typically makes the system better damped and more stable

PID Controller



Closed-Loop Response

	Rise time	Max overshoot	Settling time	Steady-state error
P	Decrease	Increase	Small change	Decrease
I	Decrease	Increase	Increase	Eliminate
D	Small change	Decrease	Decrease	Small change

PID Controller

- Output feedback
  - From Proportional action
  - Compare output with set-point
- Eliminate steady-state offset or error
  - From Integral action
  - Apply constant control even when error is zero
  - Eliminates transients; sum of all previous errors
- Anticipation
  - From Derivative action
  - React to rapid rate of change before errors grows too big
  - Anticipate the trends; rate of change of the error

Adaptive Control

- Adaptive control is the idea of “redesigning” the controller while online, by
  - looking at its performance and
  - changing its dynamic in an automatic way
- Motivated by aircraft autopilot design
  - Allow the system to account for previously unknown dynamics
- Adaptive control uses feedback to observe the process and the performance of the controller and reshapes the controller closed loop behavior autonomously.

Adaptive Control

- Modify the control law to cope by changing system parameters while the system is running
- Different from Robust Control in the sense that it does not need *a priori* information about the uncertainties
  - Robust Control includes the bounds of uncertainties in the design of the control law.
  - Therefore, if the system changes are within the bounds, the control law needs no modification

25

System Identification Model Building

- Mathematical tools and algorithms to build dynamical models from measured data
- A dynamical mathematical model in this context is a mathematical description of the dynamic behavior of a system or process in either the time or frequency domain
- Theories and processes
  - Physical
  - Computing
  - Social
  - Engineering
  - Economic
  - Biological
  - Chemical
  - Therapeutic

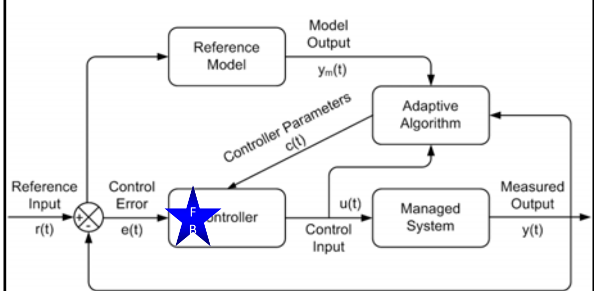
26

Model Reference Adaptive Controllers—MRAC

- Also referred to as Model Reference Adaptive System (MRAS)
- Closed loop controller with parameters that can be updated to change the response of the system
- The output of the system is compared to a desired response from a reference model (e.g., simulation model)
- The control parameters are updated based on this error
- The goal is for the parameters to converge to ideal values that cause the managed system response to match the response of the reference model.

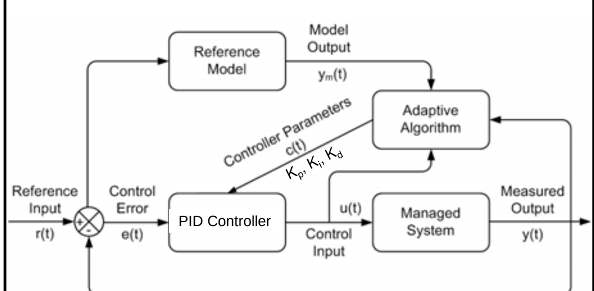
27

Model Reference Adaptive Controllers—MRAC



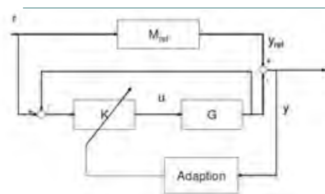
28

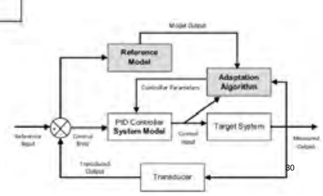
Model Reference Adaptive Controllers—MRAC



29

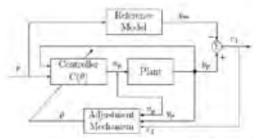
MRAC Diagrams





30

The MIT Rule for Adaptive Control: Adaptive Flight Control System – X15



Pilot Observations

The true superiority of the X-15 AFCS was that it unburdened the pilot. The airplane was stable at any dynamic pressure and at any angle of attack. The AFCS inspired confidence and allowed the pilot to spend time cross-checking flight instruments, checking subsystems, and "sightseeing."

Flight Research Center,
National Aeronautics and Space Administration,
Edwards, Calif., November 3, 1970.

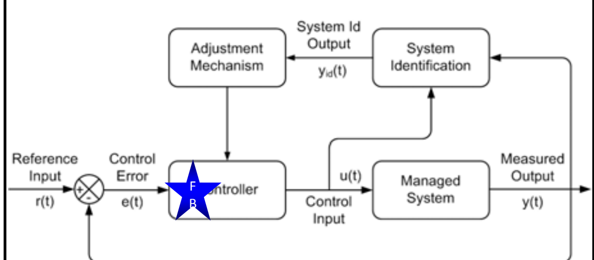
31

Model Identification Adaptive Controllers—MIAC

- Perform system identification while system is running to modify the control laws
  - Create model structure and perform parameter estimation using the Least Squares method
- Cautious adaptive controllers
  - Use current system identification to modify control law, allowing for system identification uncertainty
- Certainty equivalent adaptive controllers
  - Take current system identification to be the true system, assume no uncertainty
  - Nonparametric adaptive controllers
  - Parametric adaptive controllers

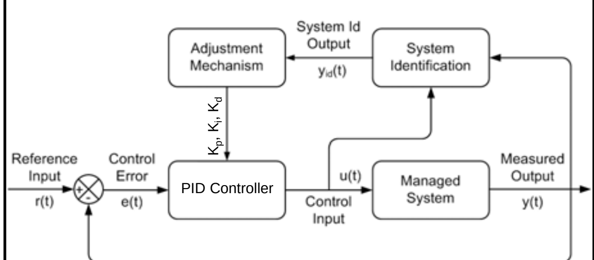
32

Model Identification Adaptive Controllers—MIAC



33

Model Identification Adaptive Controllers—MIAC



34

MIAC versus MRAC

- In the MRAC approach, the reference model is static (i.e., given or pre-computed and not changed at run-time)
- In the MIAC approach, the reference model is changed at run-time using system identification methods
- The goal of both approaches is to adjust the control laws in the controller

35