

Verification and Validation of Adaptive Systems

Johann Schumann, RIACS
NASA Ames

–DRAFT VERSION–

Versions

- 2/29
- 3/5, 3/9

Synopsis

Goals

Adaptive and learning systems are nowadays found in many safety-critical areas (e.g., aircraft, automotive industry, chemical and nuclear industry). Often based on neural networks such system provide advanced capabilities of data analysis and control in the face of change (e.g., to control a damaged aircraft). Due to their nonlinear and dynamic nature, however, verification and validation of such systems pose substantial issues and require novel methods and tools.

This course will provide a background on the basic principles of artificial neural networks and machine learning systems and its applications, in particular in safety-related areas and it will provide detailed information on methods and tools for the verification and validation of such systems.

Schedule

- **Course:**
 - Monday - Friday 9AM - 12PM
- **Tutorial:**
 - Mo, Tue, Thurs 2PM - 5PM
- **Student's Seminar**
 - Wed 2PM - 5PM
- **Seminar**
 - Fri 2PM - 5PM

Overview & Resources

The material for this introductory course on V&V of adaptive control systems is exclusively taken from (a) textbook material, (b) publicly available scientific papers, and (c) material from papers co-authored by J. Schumann and for which a 1676 form exists.

- Introduction: Why adaptive control?: Material from [IEEE2008]
- Safety-critical software systems, Verification and Validation. Material from [Storey96,Neumann95]
- Neural Networks introduction. Material from [Bishop95, Reed98,Norgaard2000]
- Control systems, Lyapunov stability. Material from [G. Franklin] and papers [Calise98,SchumannGupta2004]
- Dynamic monitoring, Confidence tool [GuptaSchumann2004]

Materials - Literature

- [ReedMarks]R. Reed and R. Marks, *Neural Smithing*", MIT Press, 1998
- [Bishop95] C. Bishop, "Pattern Recognition", Oxford, 1995
- [Norgaard2000] M. Norgaard etal. *Neural Networks for Modelling and Control of Dynamic Systems*", Springer, 2000
- [Storey96] N. Storey, *Safety-critical Computer Systems*, Addison-Wesley, 1996
- [Calise98] A. Calise and R. Rysdyk. *Fault tolerant Flight Control via Adaptive Neural Network Augmentation*, AIAA, 1998
- [SchumannGupta04] J. Schumann P. Gupta. Monitoring the performance of a neuro-adaptive controller, Maxent, 2004.
- [GuptaSchumann04] P. Gupta, J. Schumann, A Tool for Verification and Validation of Neural Network based Adaptive Controllers, HASE, 2004 IEEE.
- [IEEE2008] J. Schumann Y. Liu Tools and Methods for the Verification and Validation of Adaptive Aircraft Control Systems.

Day I

1. Introduction
2. Damage-adaptive Control
3. Course Topic Overview

Introduction

- Damaged aircraft:
 - Examples
 - mid-air collision pictures and video
 - What does the pilot need to do?
 - Difficulty of *controlling* aircraft
 - Novel and unpracticed control commands might be necessary, e.g., use ailerons instead of elevator

Introduction

- Can the pilot be helped in controlling such a damaged aircraft?
- Goals:
 - Perform mission as much as possible
 - Get home safely
- Support by:
 - Identification of problem
 - *Automatic adaptation of the AC controller*

Introduction

- Applications of adaptive control
 - Aircraft: damage, wear and tear, configuration
 - Space:
 - Unforeseen events: e.g., not unfolded solar panel or obstructed thruster changes dynamics of spacecraft
 - Novel and unexplored environments: aircraft on Mars, Europa submarine, landing/balloon on Titan
 - Automotive:
 - Wear and tear
 - Flat tire
 - (Chemical) industry
 - Power plants

Damage-adaptive Control-I

- Day I: High-level view; details on Day IV
- Traditional feedback control
 - History & examples
 - Watt's Steam engine controller (fly-wheel)
 - Thermostat: bang-bang heater control
 - Some simple linear example

Feedback Control: Intro

- Notions to be explained with example:
 - Plant
 - Dynamics
 - Model of plant P
 - Controller
 - Control input and sensor input
 - Control output (calculated)
 - Controller model P must be designed for P
 - Controller parameters (“gain”)
 - Feed back
- Important: Plant model $M =$ controller model M

Damage-adaptive Control-II

- Why does traditional feedback control fail
 - $P \neq M$ (too much deviation)
 - P is unknown
 - P is changing
- Idea: add an “adaptive component” to modify the controller

Damage-adaptive Control-III

- NN-based controller:
 - NN stores and adapts the model M
 - NN control augmentation

Here we introduce the NN only as a black box, which gets some inputs and produces “good” outputs, based on its “training”

Damage-adaptive Control-IV

- Questions to ask for all applications
 - Does it work?
 - Is it safe?
 - Does it perform?

Topics

- Safety: What is safety? How to design a safe software system? Can we construct safe adaptive controllers? (Day II)
- Need to know: What is a NN? How does the NN adapt? Safety measures for NNs? (Day III)
- Need to know: safety of a controller? What is that? Stability, robustness, ... (Day IV)
- Putting everything together (Day IV)
- Safety and Safety Certification (Day V)

Day I - Tutorials

- Octave (or Matlab if available to students)
 - Introduction into system & programming
 - Some simple examples
- A simple control application
- Watertank control example w/NN to play around

Day I - Books

- R. Reed and R. Marks, *Neural Smithing*”, MIT Press, 1998
- C. Bishop, “Pattern Recognition”, ZZZZ
- M. Norgaard et al. *Neural Networks for Modelling and Control of Dynamic Systems*”, Springer, 2000
- N. Storey, *Safety-critical Computer Systems*, Addison-Wesley, 1996

Day I - Books (cont)

- P. Neumann, *Computer related risks*, ACM press
- P. Gill et al., *Practical Optimization*, new edition(??)
- Controls intro (-> Karen)
- Standards (DO178B etc) ??

Day II

1. Safety-Critical Systems
2. Verification and Validation

Safety-critical systems

- Safety-critical systems and computers
 - What are SCS? Examples and areas
 - Short Classification
 - System failures
 - Hardware
 - Software
 - Operator errors

Safety-critical systems

- Level of criticality
 - According to FAA
 - Level A .. F
 - Applicability for other areas
- Definition of
 - Failure, fault
 - Redundancy, fault avoidance, fault diagnosis

Designing a safety-critical system

- Here: software only (but also discuss integration)
- The software lifecycle
 - General remarks
 - The waterfall model
 - Iterative models
- Software life cycle and safety (overview)
- Software processes

Software V&V Process

- Verification
 - Definition
 - Examples: different levels of formality
 - Requirements <-> implementation “informal”
 - Formal termination proof
 - Code review - coding standards
- Validation
 - Testing
 - What is testing?
 - Unit vs. system vs. integration testing
 - Interactive, automatic, regression testing
 - Testing for code coverage
 - Functional testing
 - Parametric testing
- Documents: safety-case, certification, PDR, CDR, FRR

The V-Shape

- When to do which V&V activity?
- Typical verification and validation tasks throughout the software process

Language/System requirements

- Real-time constraints
 - Computer hardware / rad-hard
 - WCET: no cache
- Coding Standards (examples)
 - No malloc
 - No multiple inheritance
 - For(;;) or for(I=0;I<4;I++)
 - Parallel/multi-threaded execution and synchronization
- Interface constraints
 - Buses (e.g., 1553)
- Operating system
 - VxWorks, Arinc 653, RT-Linux

What is not good

- Recursion
 - Memory and termination
- While(converging) do
 - termination
- AI algorithms
 - Often “search” involved
- Self-modifying code
- “Non-deterministic” code
 - What is non-deterministic?
 - Computer science
 - Engineering

Day II -- Tutorials

- Analysis of certain incidents with respect to software safety
- Analysis of a SoA coding standard
- Looking at a software standard
- Practical V&V
 - Small verification task
 - How to test a piece of software
 - Code coverage: statement/branch with gcov
 - Testing a finite state machine or nested if-then-else
 - Testing a numerical routine

Day III

- Neural Networks
 - Introduction
 - Architectures
 - Learning Algorithms
 - Problems with Learning
 - Network Performance

NN-I

- Neural Networks I
 - Introduction and History
 - Historical view: Neurons in the Brain
 - History of artificial neural networks
 - The linear Perceptron
 - Non-linear network models
 - Running a network

NN-II

- Unsupervised vs. supervised learning
- Principles of supervised learning (Error minimization)
- Short excursion into quadratic minimization
- Back Propagation
- Example

NN-III

- Problems in Training a Neural Network
 - Selection of training data
 - Convergence
 - Non-convergence
 - Convergence to wrong (local) minimum
 - Overtraining
 - Network size and architecture
 - Numerical Issues
 - Accuracy/round-off errors
 - Divergence
 - Different learning methods
 - 2nd order
 - Variable step

NN-IV

- How good does the network perform?
 - MSE, residual
 - Sensitivity
 - Confidence

Day III - Student Seminars

Day IV

1. Control Systems
2. Aircraft Control
3. Stability Analysis

Control Systems

- Feedback control
 - Recap of major notions
- Types of Controllers
 - Bang-Bang control
 - P, PI, PID control basics

Linear Control Systems

- Example:
 - Plant
 - Plant model
 - The control law
 - Simulation and curves

Aircraft control

- Aircraft
 - Major components: aileron, elevator, rudder
 - Three AC axes
 - Roll
 - Pitch
 - Yaw

Aircraft control

- Flying the AC
 - The stick
 - Rudder Pedal
- Simple example maneuvers
 - Straight flight
 - Doublets
 - Pitch doublets
 - Chase

Aircraft control

- Aircraft control
 - Mechanical
 - Hydraulic
 - Fly-by-wire
- Control separate for each axis

Stability of a Controller

- Stability as a major performance and safety measure for a control system
- What is stability, eventual stability

Stability of a Controller

- Example of stable/unstable control
- Discussion of how to prove stability
 - Linear: picture of phase and gain margin

Problem: damaged AC = nonlinear control

Stability Analysis

- For non-linear plant: Lyapunov stability
- At example: pendulum
- Lyapunov function candidate
- Graphical representation
- What does this mean (formally)?
- Why is Lyapunov analysis complicated & limited

Safety of Adaptive Control

- Adaptive controllers are non-linear with non-constant gains under a non-constant (nonlinear) plant
- The Three Laws
 - Stability
 - Convergence speed
 - Acceptable solution

Stability of adaptive control

- Lyapunov analysis of NN-adaptive controller
- Overview of the Math
- Results:
 - Bounded error
 - Update rule

Day IV - Tutorials

- Neural network: practical training
 - Some Open source NN software
 - Simple / difficult training data
 - Training speed: grad-descent vs. Levenberg-M.
 - Selection of training data
 - Over-training
 - Interpreting a trained neural network
 - Network performance

Day V

1. Dynamic NN monitoring
2. Certification

Dynamic Monitoring

- What does this mean?
- Bayesian consideration of NN learning
- The Confidence Tool
- The Parameter Confidence Tool

Certification

- What is certification?
 - Relationship to V&V
 - Development and Certification Process
 - PDR, CDR, FRR
 - FAA-guided
 - Standards
 - IEEE, DO 178B, ...
 - Tools for certification support
 - Tool qualification

Day V - Seminar

- Certification and Certification Standards
- Qualified tools
- Processes
- New and upcoming standards