About This Course

Course Administrative Information

- Instructors: Dr. Jason E. Hicken[1] and Prof. Juan J. Alonso[2]
- Course Assistant: TBA
- Office Hours: J. Hicken: Fridays 2:00-3:30pm (Prof. Alonso: TBA)
- Schedule: Mon/Wed 2:15–3:30pm, McCullough 126
- Course Web Page: `http://adl.stanford.edu/aa222`
- Course Mailing List: `aa222-class`. Subscribe by going to `https://itservices.stanford.edu/service/mailinglists/tools` and entering `aa222-class` in the *Subscribe and unsubscribe to a list* box.
- Pre-requisites: multivariable calculus, basic linear algebra, familiarity with a programming language (C, C++, F90/95, Python, MATLAB).
- AA222 satisfies the MS and PhD Math Requirement. It is therefore going to deal with the mathematics of optimization and MDO.
- Homeworks & Projects: You can discuss ideas in groups but you MUST carry out and write up solutions on your own. Late policy is 10% of grade per day / fraction of day. Assignments will start from more mathematical and will shift towards more applied.
- Grading: 75% Homework, 25% Final Project.

[1] Durand 359, *jehicken@stanford.edu*
[2] Durand 252, *jjalonso@stanford.edu*

Textbooks and Resources

No book is required. Detailed course notes will be handed out for every lecture. However, if you are interested in more details, the following books have been placed on reserve in the library and I would recommend that you get the first one (if you are likely to continue to work in optimization).

- Optimization Concepts and Applications in Engineering. Belegundu, A. and Tirupathi, R., Prentice Hall, 1999.
- Introduction to Engineering Design Optimization. Onwubiko, C., Prentice Hall, 2000.
- Applied Optimization with MATLAB programming. Venkataraman, P., Interscience, 2001.

Some useful programming references are listed below.

- Octave: Open source, mostly compatible with Matlab. `http://www.gnu.org/software/octave`
- Python: Open source, interpreted language with many scientific computing and graphing packages.
  - `http://www.python.org`
  - `http://wiki.python.org/moin/NumericAndScientific`
  - Langtangen, H. P. *Python Scripting for Computational Science*, 3rd ed. Springer, 2007.
  - Langtangen, H. P. *A Primer on Scientific Computing with Python.* Springer, 2009.

Acknowledgements & Additional Courses

The first version of these notes was jointly put together by Prof. Joaquim R. R. A. Martins, from the University of Michigan, and Prof. Juan J. Alonso, from Stanford University. This is the fifth+ iteration of these notes which contains a tailoring for the subject matter covered in AA222, additional lectures in approximation theory, hierarchical decomposition, and setup of the MDO problem. In addition, a number of additional interactive optimization problems have been added, together with new homework problems and graphics for the explanation of concepts. Prof. Ilan Kroo and Dr. Dev Rajnarayan have contributed greatly to the current status of these notes.

AA222 is not meant to be a replacement for optimization courses offered elsewhere at Stanford. If you are interested in optimization you are strongly encouraged to get a depth of knowledge from some of the following courses: MS&E 111, 112, 120, 211, 212, 310, 311, 312, 318, Math 151 (or more advanced), EE 364A/B/S, and STATS 310 A,B,C, among others.

Course Outline

Introduction:

1. What is "MDO"?
2. Terminology and Problem Statement
3. Classification of Optimization Problems
4. Methods of Solution
5. Practical Applications

Single Variable Optimization:

1. Optimality Conditions
2. Line Search Methods

Gradient-Based Optimization:

1. Optimality Conditions
2. Steepest Descent and Conjugate Gradient Methods
3. Quasi-Newton Methods

Sensitivity Analysis:

1. Finite Differences
2. Complex-Step Derivative Approximation
3. Algorithmic Differentiation
4. Semi-Analytic Methods

Handling Constraints:

1. Karush–Kuhn–Tucker (KKT) Conditions
2. Penalty and Barrier Methods
3. Reduced Gradient and Gradient Projection Methods
4. Sequential Quadratic Programming (SQP)
5. Constraint Agglomeration

Gradient-Free Optimization:

1. Nelder–Mead Simplex
2. DIRECT Method
3. Genetic Algorithms and Pareto Optimality
4. Particle Swarm Algorithms

Function Fitting and Regression:

1. Polynomial approximations
2. Design of Experiments

3. Gaussian processes / Kriging
4. Multi-fidelity approximations
5. Other topics

MDO Architectures:

1. Collaborative Optimization (CO)
2. Concurrent Subspace Optimization (CSSO)
3. Bi-Level Integrated System Synthesis (BLISS)
4. Coupled-Sensitivity Analysis

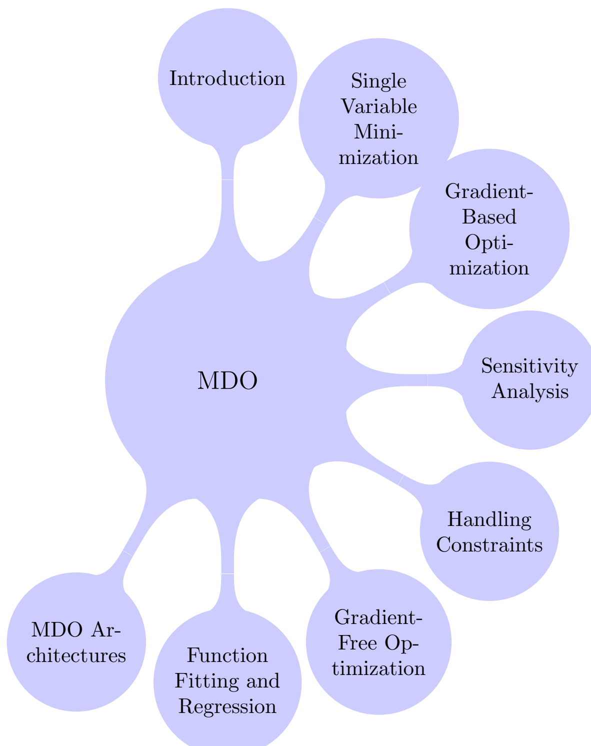

Figure 1: Course content