MA417
Operations Research II
Decision Making Under Uncertainty

1 Instructor

Robert Molzon

Office: POT 933

E-Mail: molzon@ms.uky.edu

Phone: 859 257-1480

Office Hours: MWF 10:00-11:00

2 Room and Time

FB213 309 on TR at 11:00 AM

3 Text

The primary text we shall follow is Dynamic Programming by Eric V. Denardo. I also suggest you buy the book, Dynamic Programming by Richard Bellman. You can get both books for about $25 on Amazon.

There are many notes online about dynamic programming. Here are a few you might find useful.

I also recommend the text by Bertsekas, Dynamic Programming and Optimal Control. You can get this from Amazon for about $119.

- Dynamic Programming Wiki. Wiki http://en.wikipedia.org/wiki/Dynamic_programming
- Optimization and Control. Optimization and Control http://www.statslab.cam.ac.uk/~rrw1/oc/index.html
- Alte Stout http://www.eecs.umich.edu/~qstout/dynamprog.html

4 Grading

Your grade for the course will be based on a midterm exam, a project, a final exam, and homework. Each of the four components will count toward 25% of your grade. I will post sample problems for the midterm and final exams.

4.1 Midterm Sample Problems

exam_problems exam_probs.xhtml

5 Software

First some comments on programming. Many students who have taken this course in

the past expressed a reluctance to do much programming. It is difficult to get started, but the huge advantages basic programming skills will give you in a job or in graduate school cannot be overestimated. One of the first questions I am asked when companies contact me about prospective employees is "Does the individual have good programming skills?" I shall set aside enough time during the course of the semester for us to work problems in the computer lab. If you use one of the "interpreted" languages described below it is fairly easy to get started once you see a few examples.

Only the very simplest examples of the problems we shall consider can be solved by paper and pencil. Therefore you will need access to a programming language on a computer. I shall provide code examples in some of the languages. Here is a list of possibilities.

1. R: This language is very heavily used in statistical analysis. It is the open source version of S+ and is available free for download at Cran CRAN http://cran.r-project.org/. There is very good online documentaion and help. I shall provide sample code in R to help you get started. It is available for Linux, Mac, and Windoz. One big advantage of R is its "interpreted language" nature. This means that you can type in commands directly from a shell and get the output immediately - in other words the code does not have to be compiled as in C or Fortran. This feature makes it much easier to learn.
   Here are a few links that may help with learning R.
   a. Programming in R Programming in R http://zoonek2.free.fr/UNIX/48_R/02.html
   b. Wiki Wiki http://en.wikipedia.org/wiki/R_(programming_language)
   c. A Brief Introduction to R A Brief Introduction http://xweb.geos.ed.ac.uk/~hcp/r_notes/r_notes.html
   d. R Reference Card Reference Card http://www.rpad.org/Rpad/Rpad-refcard.pdf
   e. Simple for loops and if statements in R for_loop R/for_loop.r and if_statement R/if_statement.r
2. AMPL: "A Mathematical Programming Language". A student version is available free for download from the AMPL website AMPL http://www.ampl.com/. This language makes it fairly easy to set up mathemtical models - especially linear programming models. Since many dynamic programming problems can be solved by using linear programming as the final step, this language can be quite useful. One has the usual looping tools available as well. I shall provide code examples in AMPL. It is also an interpreted language an commands can be entered from a shell.
3. Perl: Unless you are already fairly familiar with programming, I would not suggest using Perl to start with. However, if you have good programming skills, Perl is extremely useful.
4. C and Fortran: Same comments as above. These are essentially required for solving big industrial strength problems. However, I think for a beginner R would be much easier to pick up quickly.
5. Maple or Mathematica: These are high level languages that are aimed more at symbolic computation. They are both fairly easy to learn - at least for basic programming. If you are familiar with either one and have access to them, then they will work fine.
6. Excel or Calc (the Open Office version of Excel). These are spreadsheet programs and you could work many of the problems using them. They have builtin solvers for optimization that work fine.

6 Topics and Goals

We will cover the material in the eight chapters of Denardo's book. Follwing his suggestion, we shall skip over most of the * sections that cover more advanced topics. In addition to the material in the book, I shall give a brief introduction/review to basic Markov processes. You may have seen these processes in a probability course under a different name. For example, a random walk left and right determined by a coin flip is a Markov process. I shall also give enough of an introduction to R with code examples so that you can write simple programs to gain experience with the algorithms of dynamic programming. Here are three basic goals for the course.

- Gain an understanding of the basic ideas of dynamic programming,
- Learn to formulate and solve moderately complicated practical optimization problems that can be approached using dynamic programming,
- Gain a basic working knowledge of the use of software for optimization of sequential decision problems.

Here is a list of the topics in the order that they will be covered. Each topic will take approximately one week.

- Introduction to Sequential Decision Processes: Examples and Introduction to R.
- The Prototype Sequential Decision Process: Shortest Path Problems.
- Allocation of a Resource: Knapsack Problems and Marginal Analysis.
- Stages, Grids, and Discretizing Control Problems.
- Review of Probability and Introduction to Basic Markov Processes.
- Hidden Markov Models and Shortest Paths. The Viterbi Algorithm, Gene Sequencing.
- A Markov Decision Model.
- Production Control and Network Flow
- Inventory Control
- Discounted Markov Decision Models

7 Notes

I shall place some notes and computer code online in this section.

Introduction notes/introduction

Shortest Paths notes/shortest_paths/shortest_path.xhtml

Resource Allocation notes/resource_allocation/resource_allocation.xhtml

Stages and Grids notes/stages/stages.xhtml

Probability and Markov Processes notes/probability_markov_process

Hidden Markov Models notes/hidden_markov

Markov Decision Models notes/markov_decision

Production Control notes/production

Inventory control notes/inventory

Discounted Decision Models notes/discounted_decision

8 Projects

As part of your course grade you are to do a project that goes beyond the standard homework in complexity. Your project must be typed as a report and include information about any software you used. I've included a link below that outlines a project for inventory control.

inventory inventory_project.xhtml

8.1 Inventory Control

inventory_control inventory_control.xhtml

8.2 Cyclic Networks and Shortest Paths.

The problem of finding the shortest route from point A to point B is a cyclic network problem. It is solved in real time by Google Maps. An introduction is given in the second chapter of the text.

9 Additional Links

- INFORMS: The Operations Research Website. Informs http://www.informs.org
- Government Jobs in Operations Research Government Jobs. http://www.usajobs.opm.gov/. Search Operations Research.
- Math-Jobs.com Math-Jobs http://www.math-jobs.com/

File translated from TEX by TTM, version 3.72.
On 31 Mar 2008, 11:32.