

**AMSC/CMSC 661 Term Project**  
**Due at 4pm Monday 05/17/2010**  
100 points

There are many interesting algorithms that we do not have time to cover in this course. Choose a paper that discusses one of them, implement the algorithm in MATLAB, test it, and evaluate it.

**How to get started:**

- Each person is required to have a unique project, chosen from among a list of papers posted on the website.
- Look at the list of papers (posted at 7pm April 5) and send me email containing a list of your top choice(s).
- I will assign the papers in the order in which I receive email requests. I will respond to your email, either assigning a paper to you or telling you that your choice was previously assigned to someone else. I will update the website as papers are assigned.
- Beginning April 19, you may ask to change to a different paper, in case you are not happy with your original choice.

**What to submit:**

- Your MATLAB software. Documentation should include an explanation of what the algorithm does and how it does it. In many cases, good MATLAB software already exists; in that case you should definitely make use of it and put your effort into experiments or improvements to the software.
- A discussion, including
  - what you implemented, (i.e., a special case of the method discussed in the paper, exactly the method, or a generalization of it)
  - what experiments you performed.
  - what you compared the algorithm to.
  - how the algorithm performed.
  - what advantages, disadvantages, and limitations the algorithm has,
  - your conclusions about its usefulness. On what class of problems do you expect it to perform better than alternatives? On what class would it perform worse?
  - a list of open research problems inspired by the paper or by your experiments.
  - any errors you found in the paper.
  - a list of all references that you used.

Note that the discussion need not contain a description of the algorithm; this description should be clear from your program and its internal documentation.

Warning: Plagiarism will merit a failing grade. (See the course syllabus and website for further information.) If you use someone's ideas or programs, cite

the source. If you use a direct quote, use quotation marks and cite the source. And don't expect a good grade on a project that is mostly someone else's work.

Note: A few of the papers concern a problem rather than an algorithm. In that case, discuss the relationship between your problem and one discussed in class, applying as much of our theory and computational knowledge as possible. The instructions and grading criteria will thus be modified by replacing algorithm with problem wherever it makes sense.

#### **Grading criteria:**

- The program:
  - Is the algorithm implemented correctly?
  - Is the program clearly documented, including enough explanation to understand the algorithm?
  - Is software provided to demonstrate that the algorithm works correctly?
  - Is software provided to evaluate how well the algorithm works?
  - Is the output clearly presented?
- The discussion:
  - Is the algorithm compared with an appropriate alternative?
  - Are the test problems appropriate and the tests well designed?
  - Is the design of the tests explained clearly?
  - Are the reasons behind the design explained clearly?
  - Is the discussion of results clear and correct?
  - Is the evaluation of the algorithm correct and well justified?
  - Are the open problems interesting?
  - Was the document spell-checked?
- Overall:
  - Does the project along with the original paper give enough information to be self-contained?
  - Does the project show sufficient effort? (In general, the effort should match that of 100 points of homework. For some papers, it is fine to implement a special case, while for others, the full algorithm can be implemented. Use your common sense, and if you have questions, ask me.)
  - Does the problem show sufficient thought and some evidence of creativity?

A grade of 80-89 means a good job, with sound design and good discussion. A grade of 90-100 means all of this with above average effort, creativity, and clarity.

**How to submit:** Submit your project by email. A legitimate time stamp on the email will determine whether the project is on-time or late. I will acknowledge receipt of each project after I check that the attachment arrived safely.

- The MATLAB programs should be in plain text, stored in files that can actually be run by MATLAB.
- Data files should be in .mat format or plain text.
- The discussion should be in pdf format. **Microsoft-formatted documents (Word, Excel, etc) will not be accepted; submit a pdf file for these.**
- Encode all of this in a tar, zip, or gzip file.
- There will be a 15% penalty for projects turned in up to 24 hours late, 30% penalty for projects turned in 24–48 hours late, etc.
- My workstation runs Linux, so Microsoft-specific features are unlikely to work properly. For example, don't use dll.