CSC801-002

Stochastic Optimization I: State of the Art and Beyond

Spring 2018

Course Syllabus (tentative)

3 credits

Instructor: Franc Brglez, Ph.D.
Office: Monteith Research Center, Room 456
Phone: 919-515-9675
Instructor E-mail: brglez@ncsu.edu

Lectures:
- Fridays 1:45 PM – 3:00 PM ; 3:15 PM – 4:30 PM
- room in EB2 TBD
- enrollment is limited, see below

Office Hours: TBD

Summary: This is a research-oriented, project-based, and limited-enrollment course on stochastic optimization. The textbook provides a ready-to-start framework in R. The course will not only cover the relevant algorithms/solvers and optimization problems introduced in the textbook but also explore new solver prototypes in R that will challenge and potentially outperform algorithms published in the book and elsewhere.

The prototypes rely on the markov chain framework by introducing, in discrete and continuous variable domains, a number of walk strategies. The experimental designs measure, in the context of standardized families of problems that are increasing in size, the asymptotic complexity of these algorithms in terms of the *uncensored mean first-passage-time.*

In the continuous domain, such families may include

- a variety of hard test functions,
- instances of Lennard-Jones clusters,
- clusters of folded proteins,
- hard instance brought to class by students.

In the discrete domain, such families may include

- instances of the knapsack problem,
- the maximum graph clique problem,
- the maximum satisfiability problem,
- the linear ordering problem,
- the job-shop scheduling problem,
- hard instance brought to class by students.

Textbooks:

Modern Optimization with R, Paulo Cortez, Springer-Verlag, 2014. The pdf file is freely downloadable from the NCSU library. For a review, see https://www.jstatsoft.org/article/view/v070b03

Markov Chains and Random Walks: two chapters from *Introduction to Probability* by Charles M. Grinstead and J. Laurie Snell, 2-nd revised edition, AMS, 2000. The pdf file of this book is freely available at http://www.dartmouth.edu/~chance/teaching_aids/books_articles/probability_book/amsbook.mac.pdf

Supplementary Resources:

The R-language environment and packages described in the textbook on optimization are freely available from http://www.r-project.org

The material that complements and extends the chapters on markov chains and random walks will be sourced from publications authored and co-authored by instructor during 2007 – 2017; under https://people.engr.ncsu.edu/brglez/publications.html

The material relevant to this course will include articles in public domain such as https://en.wikipedia.org/wiki/Lennard-Jones_potential. Student in the class will also critically review articles such as *Effect of hybridizing Biogeography-Based Optimization (BBO) technique with Artificial Immune Algorithm (AIA) and Ant Colony Optimization (ACO)*, under http://www.sciencedirect.com/science/article/pii/S1568494614001173 and articles such as *Metaheuristics – the metaphor exposed* under http://onlinelibrary.wiley.com/doi/10.1111/itor.12001/abstract; all downloadable from NCSU library.

Prerequisites: Undergraduate level knowledge of Discrete Mathematics [CSC 226], Data structures [CSC 316], Linear Algebra, and programming.

Learning outcomes:

- Understand the fundamental principles of stochastic optimization, both in continuous and discrete domains.
- Learn how to design computational experiments to rigorously evaluate performance difference between two or more solvers on well-defined sets of problem instances.
- Critically review a number of articles and learn about search strategies that have a good chance to improve the current generation of optimization solvers.
- Raise the level of curiosity and motivation for research in experimental algorithmics while also getting experience in rapid prototyping of new solvers in R while using LaTex to efficiently produce beautiful documents.

Limited enrollment: Enrollment is limited to a maximum of 10 graduate and senior-level undergraduate students. This course is a prerequisite for a follow-up 3-credit hour independent study course on Stochastic Optimization II where students will continue to pursue challenges and latest advances in stochastic optimization, including prototyping state-of-the-art stochastic solvers in `Julia`, `Swift`, and `C/C++`. Interested students should email a request for an interview to brglez@ncsu.edu, attaching an informal description of relevant courses, projects, and any references.

Course structure and schedule: The class will meet once a week for a total of 2.75 hours (including a 15-minute break) in a projector-equipped small seminar room, with participants bringing their own laptop computers (macOSX and linux preferred). The instructor will initiate a series of informal seminar-like lectures that will cover material from textbooks and relevant publications. Each week, students will be paired as a team to work on homework assignment selections. During the class on the following week, students will briefly initiate the discussion with few slides, outlining solutions in R. By Week 4, research projects will be identified and assigned to each student pair. Students will be able to initiate and test the computational projects on their own personal computers. For more extensive computational experiments, they will make use of the resources under `https://vcl.ncsu.edu/`.

On GoogleDrive, instructor will aggregate and maintain a class directory of lecture notes, homework/project slides, relevant publications, and a LaTex directory of a collaborative draft manuscript that will merge and compile contributed sections from the instructor and student teams. Students will be expected to maintain similar structures of their own course-related material on their GoogleDrive, including the final R-code and latex draft of their projects. When the course is completed, instructor will archive the relevant code under `https://github.com`, publish the jointly-authored manuscript under `https://arxiv.org/` and submit the manuscript for review to `https://www.jstatsoft.org/`.

The schedule in the table below is tentative. An updated schedule, along with all course material, will be maintained on the course home page under `https://drive.google.com/open?id=0B6mJTvscUcVfdnVwcGdRdEQwdm8`.

Again, in the table below, homeworks are shown on the date they will be assigned to pairs of students. The following week, students will prepare a few slides and report in class about results of their assignment.

Dates:	Topics:	ToolKits:	Homeworks:
Jan 12	Introduction	R-packages	Exercises with R
	DE algorithm	DEoptim	Exercises with DEoptim
Jan 19	Genetic algorithm	GA	Exercises with GA
	Evolutionary algorithm	genalg	Exercises with genalg
Jan 26	Particle swarm algorithm	pso	Exercises with pso
	Classification of problems	xyzLib, diceLib	Exercises with *Lib
Feb 2	Comparative experimentation		
	Q&A and project assignments		Projects: pass 1
Feb 9	Markov chains & dice graphs		
	Minimum first-passage walkLength	stpm.spectral	Projects: pass 2
Feb 16	Significance of sparse dice graphs	landscapeGraph	
	Merits of self-avoiding walks	stpm.fpt.walk	Projects: pass 3
Feb 23	New walkXYZ algorithm	walkXYZ	
	New walkDice algorithm	walkDice	Projects: pass 4
Mar 2	Initiating experiments under VCL		
	Mid-term project reports in latex		Projects: pass 5, VCL
Mar 9	Spring break		
Mar 16	Can we solve 'harder' problems?		More 'harder' problems?
	Problems to be added to projects		Projects: pass 6, VCL
Mar 23	*Walk versus project solvers		team tests of *Walk
	Project updates under VCL		Projects: pass 7, VCL
Mar 30	Spring Holiday		
Apr 6	Are there more 'harder' problems?		
	Project updates under VCL		Projects: pass 8, VCL
Apr 13	Project drafts in latex due		
	Final project updates under VCL		Final project report
Apr 20	manuscript editing begins		Final project edits
Apr 27	manuscript editing ends		

Grading: There is only one grade in this course: Pass. Students who are failing to participate with homeworks as described in the table above, will be asked to withdraw before the drop-course deadline. The final project report and participation in editing the joint manuscript is in lieu of the final exam.

Examples of projects (details and decisions will be discussed in class): The metaphor of *the uncensored mean first-passage-time* is central to the performance evaluation of two optimization solvers in this project. We will have learned about this metaphor when reviewing markov chains and random walks. In our experiments with R, we do not measure the actual runtime directly since this would not provide us with a platform-independent metrics. Just like when comparing two sorting algorithms, *we count the number of primitive operations*, such as the number of function evaluations. For each instance, we invoke the optimization solver from a different randomly selected starting point $O(100)$ times and stop the search as soon as the best-known target value is reached for the first time.

- Evaluate *the uncensored mean first-passage-time* of the solver while searching for optima in continuous domains using using a progression of test functions from the same family but of increasing size: in addition to 'standard' functions, consider also new and hard test functions such as represented by Lennard-Jones clusters, clusters of folded proteins, etc.

- Evaluate *the uncensored mean first-passage-time* of the solver while searching for optima in discrete domains using using a progression of test instances from the same but of increasing size: the knapsack problem, the maximum graph clique problem, the maximum satisfiability problem, the low autocorrelation binary sequences problem, the optimum Golomb ruler problem, the linear ordering problem, the job-shop scheduling problem, the TSP problem, etc.
- In class, we shall rigorously compare, on identical instances, asymptotic performances of various stochastic solvers identified in the textbook with the new solvers walkXYZ and walkDice introduced by the instructor.