CSE 598: Project Description

Joshua J. Daymude

DUE FRIDAY, NOVEMBER 22, 11:59PM

Overview

This course project gives you an opportunity to work with Markov chain and Monte Carlo methods in a way that's relevant to you. You can work in groups of 1–3, but note that larger groups (especially groups of three) will be expected to produce significantly more work. There will be three components to this project:

- A 1-page *project proposal* including (*i*) your project title, (*ii*) a list of your group members, (*iii*) the type of project chosen, (*iv*) details on the problem being considered, and (*v*) a general outline of the MCMC approaches to be used/investigated. Due Friday, October 18, 11:59pm.
- A 6-page *written report* detailing your work (more details on the content can be found below). Due Friday, November 22, 11:59pm.
- A *presentation* showing your work in the style of a conference talk, roughly 15–25 minutes each (depending on the total number of groups). Presentation time slots will be assigned randomly for the last two weeks of classes. Due Friday, November 22, 11:59pm.

All three deliverables must be submitted on Canvas by their respective deadlines. As stated in the syllabus, the written report is worth 25% of the overall course grade and the presentation is worth 30%. The project proposal will be counted as a fifth of the written report. Students can choose between three project types:

1. **Implementation:** Students will find a suitable research paper (or papers) utilizing Markov chains and/or Monte Carlo methods and will perform a rigorous validation of the results therein using an original implementation. Preferably, the students will extend the results of the paper to new application domains or datasets, designing and executing their own experiments. Code can be written in a language of the student's choosing, but must be well documented (both with in-code comments and external documentation, if necessary).
2. **Survey Paper:** Students will choose a technique or application area related to Markov chains and Monte Carlo methods and will write a rigorous survey paper on it. The survey must be unified by a specific line of questioning or investigation; it should not simply list a long summary of various papers. The final paper must cite at least 20 relevant references.
3. **Original Research:** Students will pose an original research question relating to or using Markov chains and/or Monte Carlo methods and will write a paper investigating the research question. The format of a standard academic paper for the topic area is expected, including an introduction, related work, etc.

Report Guidelines

The written report should be 6 pages long, including references, prepared in the two-column ACM conference style (`sigconf`).[1] As such, descriptions should be clear, succinct, and detailed. Reports should be written using the standard format for research papers, including an abstract (summarizing the work), an introduction (with motivating context, related work, and a clear problem statement), relevant sections for the chosen project type, a discussion/conclusion, and references. Some of these specifications vary according to the type of project.

An implementation project should motivate the problem(s) considered and the MCMC approaches taken in the papers of choice. Relevant details on the chosen papers' techniques should be given in a background section. Areas where you went beyond the confines of the original paper should be discussed in detail. All datasets, experimental setup, and results should be justified and described at length. The inner workings of your implementation should stand for themselves (i.e., *document your code!*), but you may highlight important implementation details in your report if they merit discussion. All code should be submitted alongside the written report.

A survey paper project should be a detailed discussion of a technique or application area related to MCMC. The survey should be unified around a specific line of inquiry, and its structure should reflect that investigation. The final report should cite and discuss at least 20 relevant references (note that a survey does not need a dedicated related work section). Do not simply parrot (copy/paste) the papers that you are reading! A survey paper should thoughtfully combine, compare, and contrast different papers and results in order to answer a new question.

Finally, an original research paper should take the form appropriate for your area of research. Carefully motivate your problem and why MCMC is a relevant approach. Discuss relevant literature — including, if applicable, why other approaches are insufficient or suboptimal — and give context for why this problem is interesting. Explain your MCMC technique or approach, providing either proofs or experimental results, depending on what is relevant to your problem.

Your report grade is broken down as follows:

- **[5pts]** *Format.* Is the report 6 pages long, including references? Is it prepared in the two-column ACM conference style (`sigconf`)? Does it follow the standard format for an academic paper?
- **[5pts]** *Clarity.* Does the report clearly communicate its main points, insights, and results? Is the report easy to read? Is there a logical flow between ideas and different parts of the report? Does the report cushion heavily technical details with motivating context? Can a reader understand how each part of the report relates to the whole? Does the report reflect an effort to state things understandably?
- **[10pts]** *Depth.* Is there meaningful technical depth, work, or insights related to MCMC in the report? Does the report reflect a deep understanding of the area being discussed? Are the techniques (or, in the case of a survey paper, the synthesis) non-trivial and insightful, going beyond simply restating existing results and ideas? Is there sufficient detail for a reader be convinced of the conclusions and synthesis the report is claiming?
  - Implementation: Do the experiments conducted go beyond the work of the original paper(s) in an interesting way? (New datasets, new algorithm variants, etc.?) Are the

[1] ACM Master Template: `https://www.acm.org/publications/proceedings-template`.

experiments comprehensive and convincing?

  - Survey: Does the survey make thoughtful connections between the papers surveyed? Are the surveyed papers connected to the overarching question?
  - Research: Are non-trivial new results presented? Are challenges and obstacles encountered discussed?

Presentation Guidelines

The presentation will be given in class. Each will last 15–25 minutes long, depending on the number of project groups. Presentations should be given in the style of a conference talk (i.e., given to an informed audience that may not be aware of your specific area of research). If you have multiple group members, you will each be expected to speak a reasonable amount during the presentation. Your presentation grade is based both on instructor and peer evaluation, broken down as follows:

- **[2pts]** *Motivation.* Is the project topic well motivated using real applications? Does the audience understand why they should be interested in the work?
- **[5pts]** *Clarity.* Are the main points well presented and communicated? Is it clear what the main ideas are? Does the presentation flow easily? Do the slides avoid large walls of text, instead acting as complementary support to what is being said?
- **[5pts]** *Depth.* Are the details of the project well presented and communicated? Are the relevant details, methods, themes, etc. discussed? If the audience were to be quizzed on the content of the problems, papers, or methods discussed, would they score well?
- **[3pts]** *Quality.* Is the project a relevant and interesting problem? Are the results and insights non-trivial, or do they simply mimic what already exists? Does the presentation communicate a thorough understanding of the work and related areas?
- **[5pts]** *Peer Evaluation.* Each audience member will be answering "How clear was this presentation?" on a scale of 1–5. This is an average of those scores for your presentation.
- **[10pts]** *Completed Peer Evaluations.* You will receive 2.5pts for completing a peer evaluation on each of the four presentation days.

Project Ideas

The following is a list of topic areas and relevant papers to get you started on finding a project topic. This list is not comprehensive, and you are not limited to these topic areas and ideas.

- *More examples of MCMC random sampling.* Generating random spanning trees [1, 8, 14, 53]. Sampling on matroids [4, 16, 30].
- *More examples of mixing time analysis.* Coupling and path coupling for graph coloring [15, 25, 51]. Coupling for lozenge tilings and card shuffling [54]. Conductance (and the impossibility of efficient coupling) for graph matchings [33]. Coupling from the past for perfect rejection sampling [17, 18]. On trees [38].

- *More methods for bounding mixing times.* Evolving sets [40] and applications to graph clustering [2, 3]. The log-Solobev inequality [11, 19, 30]. Nash inequalities [12]. The Dobrushin uniqueness condition [23, 52].

- *Proving slow mixing.* On the Swendsen-Wang process [22]. For independent sets [7, 20]. A lower bound for Glauber dynamics [24].

- *More on the Ising model.* Polynomial-time approximation algorithms [29]. On trees [37, 38, 44, 46].

- *More on card shuffling and magic tricks.* Generating random permutations [13]. Top-to-random shuffling [10]. The Dovetail shuffle [5]. The Thorp shuffle [41, 42]. The Kruskal Count [34, 39].

- *Computational geometry.* Volume [35] and surface area [6] computation of convex bodies.

- *Cryptography and security.* Catching wild kangaroos [34,39]. The discrete log problem [31,32].

- *Computer graphics.* Scene rendering and path space MCMC integration [27, 28].

- *Optimization and machine learning.* The knapsack problem [43]. Simulated annealing [48,49] and simulated tempering [21,36]. Asynchronous and parallel MCMC [9,50]. Deep learning [26, 45, 47].

References

[1] David J. Aldous. The random walk construction of uniform spanning trees and uniform labelled trees. *SIAM Journal on Discrete Mathematics*, 3(4):450–465, 1990.

[2] Reid Andersen, Shayan Oveis Gharan, Yuval Peres, and Luca Trevisan. Almost optimal local graph clustering using evolving sets. *Journal of the ACM*, 63(2):15:1–15:31, 2016.

[3] Reid Andersen and Yuval Peres. Finding sparse cuts locally using evolving sets. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing*, STOC '09, pages 235–244, 2009.

[4] Y. Azar, A. Z. Broder, and A. M. Frieze. On the problem of approximating the number of bases of a matriod. *Information Processing Letters*, 50(1):9–11, 1994.

[5] Dave Bayer and Persi Diaconis. Trailing the dovetail shuffle to its lair. *The Annals of Applied Probability*, 2(2):294–313, 1992.

[6] Mikhail Belkin, Hariharan Narayanan, and Partha Niyogi. Heat flow and a faster algorithm to compute the surface area of a convex body. In *47th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '06, pages 1–10, 2006.

[7] Antonio Blanca, Yuxuan Chen, David Galvin, Dana Randall, and Prasad Tetali. Phase coexistence for the hard-core model on $\mathbb{Z}^2$. *Combinatorics, Probability and Computing*, 28(1):1–22, 2019.

[8] A. Broder. Generating random spanning trees. In *Proceedings of the 30th Annual Symposium on Foundations of Computer Science*, FOCS '89, pages 442–447, 1989.

[9] Christopher De Sa, Kunle Olukotun, and Christopher Ré. Ensuring rapid mixing and low bias for asynchronous Gibbs sampling. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning*, ICML '16, pages 1567–1576, 2067.

[10] Persi Diaconis, James Allen, and Jim Pitman. Analysis of top to random shuffles. *Combinatorics, Probability and Computing*, 1(2):135–155, 1992.

[11] Persi Diaconis and Laurent Saloff-Coste. Logarithmic Sobolev inequalities for finite Markov chains. *The Annals of Applied Probability*, 6(3):695–750, 1996.

[12] Persi Diaconis and Laurent Saloff-Coste. Nash inequalities for finite Markov chains. *Journal of Theoretical Probability*, 9(2):459–510, 1996.

[13] Persi Diaconis and Mehrdad Shahshahani. Generating a random permutation with random transpositions. *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete*, 57(2):159–179, 1981.

[14] David Durfee, Rasmus Kyng, John Peebles, Anup B. Rao, and Sushant Sachdeva. Sampling random spanning trees faster than matrix multiplication. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, STOC '17, pages 730–742, 2017.

[15] Martin Dyer, Alan Frieze, Thomas P. Hayes, and Eric Vigoda. Randomly coloring constant degree graphs. *Random Structures & Algorithms*, 43(2):181–200, 2013.

[16] Tomás Feder and Milena Mihail. Balanced matroids. In *Proceedings of the 24th Annual ACM Symposium on Theory of Computing*, STOC '92, pages 26–38, 1992.

[17] James Allen Fill. An interruptible algorithm for perfect sampling via Markov chains. *The Annals of Applied Probability*, 8(1):131–162, 1998.

[18] James Allen Fill, Motoya Machida, Duncan J. Murdoch, and Jeffrey S. Rosenthal. Extension of Fill's perfect rejection sampling algorithm to general chains. *Random Structures & Algorithms*, 17(3–4):290–316, 2000.

[19] Alan Frieze and Ravi Kannan. Log-Sobolev inequalities and sampling from log-concave distributions. *The Annals of Applied Probability*, 9(1):14–26, 1999.

[20] David Galvin and Prasad Tetali. Slow mixing of Glauber dynamics for the hard-core model on regular bipartite graphs. *Random Structures & Algorithms*, 28(4):427–443, 2005.

[21] Charles J. Geyer and Elizabeth A. Thompson. Annealing Markov chain Monte Carlo with applications to ancestral inference. *Journal of the American Statistical Association*, 90(431):909–920, 1995.

[22] Vivek K. Gore and Mark R. Jerrum. The Swendsen-Wang process does not always mix rapidly. In *Proceedings of the 29th Annual ACM Symposium on Theory of Computing*, STOC '97, pages 674–681, 1997.

[23] Thomas P. Hayes. A simple condition implying rapid mixing of single-site dynamics on spin systems. In *47th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '06, pages 1–8, 2006.

[24] Thomas P. Hayes and Alistair Sinclair. A general lower bound for mixing of single-site dynamics on graphs. *The Annals of Applied Probability*, 17(3):931–952, 2007.

[25] Tom Hayes and Eric Vigoda. Coupling with the stationary distribution and improved sampling for colorings and independent sets. In *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '05, pages 971–979, 2005.

[26] Geoffrey E. Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7):1527–1554, 2006.

[27] Wenzel Jakob. Path space Markov chain Monte Carlo methods in computer graphics. *Monte Carlo and Quasi-Monte Carlo Methods*, pages 107–141, 2016.

[28] Wenzel Jakob and Steve Marschner. Manifold exploration: A Markov chain Monte Carlo technique for rendering scenes with difficult specular transport. *ACM Transactions on Graphics*, 31(4):58:1–58:13, 2012.

[29] Mark Jerrum and Alistair Sinclair. Polynomial-time approximation algorithms for the ising model. *SIAM Journal on Computing*, 22(5):1087–1116, 1993.

[30] Mark Jerrum and Jung-Bae Son. Spectral gap and log-Sobolev constant for balanced matroids. In *43rd Annual IEEE Symposium on Foundations of Computer Science*, SFCS '02, pages 1–9, 2002.

[31] Jeong Han Kim, Ravi Montenegro, Yuval Peres, and Prasad Tetal. A birthday paradox for Markov chains with an optimal bound for collision in the Pollard rho algorithm for disrete logarithm. *The Annals of Applied Probability*, 20(2):495–521, 2010.

[32] Jeong Han Kim, Ravi Montenegro, and Prasad Tetali. Near optimal bounds for collision in pollard rho for discrete log. In *48th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '07, pages 215–223, 2007.

[33] V.S. Anil Kumar and H. Ramesh. Markovian coupling vs. conductance for the Jerrum-Sinclair chain. In *40th IEEE Annual Symposium on Foundations of Computer Science*, FOCS '99, pages 1–11, 1999.

[34] Jeffrey C. Lagarias, Eric Rains, and Robert J. Vanderbei. The Kruskal count. In *The Mathematics of Preference, Choice and Order*, pages 371–391. Springer-Verlag, 2009.

[35] László Lovász and Santosh Vempala. Simulated annealing in convex bodies and an $O^*(n^4)$ volume algorithm. *Journal of Computer and System Sciences*, 72(2):392–417, 2006.

[36] Neal Madras and Mauro Piccioni. Importance sampling for families of distributions. *The Annals of Applied Probability*, 9(4):1202–1225, 1999.

[37] Fabio Martinelli, Alistair Sinclair, and Dror Weitz. Glauber dynamics on trees: Boundary conditions and mixing time. *Communications of Mathematical Physics*, 250(2):301–334, 2004.

[38] Fabio Martinelli, Alistair Sinclair, and Dror Weitz. Fast mixing for independent sets, colorings, and other models on trees. *Random Structures & Algorithms*, 31(2):134–172, 2007.

[39] Ravi Montenegro and Prasad Tetali. How long does it take to catch a wild kangaroo? In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing*, STOC '09, pages 553–560, 2009.

[40] B. Morris and Yuval Peres. Evolving sets, mixing and heat kernel bounds. *Probability Theory and Related Fields*, 133(2):245–266, 2005.

[41] Ben Morris. The mixing time of the Thorp shuffle. *SIAM Journal on Computing*, 38(2):484–504, 2008.

[42] Ben Morris. Improved mixing time bounds for the Thorp shuffle and L-reversal chain. *The Annals of Probability*, 37(2):453–477, 2009.

[43] Ben Morris and Alistair Sinclair. Random walks on truncated cubes and sampling 0-1 knapsack solutions. *SIAM Journal on Computing*, 34(1):195–226, 2005.

[44] Elchanan Mossel and Yuval Peres. Information flow on trees. *The Annals of Applied Probability*, 13(3):817–844, 2003.

[45] Simon Osindero and Geoffrey Hinton. Modeling image patches with a directed hierarchy of markov random fields. In *Proceedings of the 20th International Conference on Neural Information Processing Systems*, NIPS '07, pages 1121–1128, 2007.

[46] Robin Pemantle and Yuval Peres. The critical Ising model on trees, concave recursions and nonlinear capacity. *The Annals of Probability*, 38(1):184–206, 2010.

[47] Ruslan Salakhutdinov and Geoffrey Hinton. Deep boltzmann machines. In *Proceedings of the 12th International Conference on Artificial Intelligence and Statistics*, pages 448–455, 2009.

[48] Galen H. Sasaki and Bruce Hajek. The time complexity of maximum matching by simulated annealing. *Journal of the ACM*, 35(2):387–403, 1988.

[49] Daniel Stefankovic, Santosh Vempala, and Eric Vigoda. Adaptive simulated annealing: A near-optimal connection between sampling and counting. In *Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '07, pages 183–193, 2007.

[50] Alexander Terenin, Daniel Simpson, and David Draper. Asynchronous Gibbs sampling. Available online at `https://arxiv.org/abs/1509.08999`, 2019.

[51] Prasad Tetali, Juan C. Vera, Eric Vigoda, and Linji Yang. Phase transition for the mixing time of the Glauber dynamics for coloring regular trees. In *Proceedings of the 21st Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '10, pages 1646–1656, 2010.

[52] Dror Weitz. Combinatorial criteria for uniqueness of Gibbs measures. *Random Structures & Algorithms*, 27(4):445–475, 2005.

[53] David Bruce Wilson. Generating random spanning trees more quickly than the cover time. In *Proceedings of the 28th Annual ACM Symposium on Theory of Computing*, STOC '96, pages 296–303, 1996.

[54] David Bruce Wilson. Mixing times of lozenge tiling and card shuffling Markov chains. *The Annals of Applied Probability*, 14(1):274–325, 2004.