Coversheet: Gaussian Processes for Efficient Numerical Simulations in Physics

In this preamble to the final project report we describe progress since the midterm report and respond to comments and questions made by reviewers on that midterm report. Code for this project can be found at `https://drive.google.com/open?id=18CeHPPehNv9HcdRpsnpZowbKgjLSNbuN`.

We would like to acknowledge the fact that our final report is 9 pages instead of the specified 8. Due to the two-part nature of our project, we required a significant amount of background and methodological description, and as such were unable to further reduce the content of the report without impacting clarity and completeness. Some less important details are relegated to our appendix.

1 Progress

This project is separated into two components, both based on Gaussian process techniques. The first, investigated by Christophe Bonneville, is the interpolation of the solution of stationary linear partial differential equation (PDE) systems with respect to domain geometry and boundary conditions. The second, investigated by Maxwell Jenquin, is the numerical integration of time-dependent nonlinear PDE systems learned from two adjacent snapshots of the system's behavior and an assumption on the functional form of governing equation.

In the realm of domain and boundary condition interpolation, a number of new experiments and extensions have been made. As of the midterm report, estimation of the maximum temperature value for the heat equation on rectangular and arbitrary quadrilateral domains had been completed. In the final report, we also report maximum temperature results for arbitrary convex octagonal domains, and begin to explore predicting the temperature on a sparse set of systematically placed points within the domain using a number of independent GP models. Results for more complex boundary conditions are also described. In addition we apply this method to the linear elasticity equation, obtaining encouraging performance overall. One goal, extrapolation of characteristics beyond the range of training data, was not achieved, and is relegated to future work.

In the realm of time extrapolation, one core extension has been made. As of the midterm report, the model discovery scheme detailed in section 4.3 of the report had been used with an RBF kernel to extrapolate the dynamics of several 1-dimensional systems. In the final report, we derive and implement the corresponding model for the spectral mixture (SM) kernel, and test it on those same systems. We also report error statistics and detail the process of adapting the model to the SM kernel, with mixed results.

2 Responses to Selected Review Comments and Questions

- Some confusion on the part of the reviewers seems to have come from connecting domain interpolation to time extrapolation. These two sections of the project address different problems entirely, with different methods developed and employed. The common thread between them is the use of GPs to simulate PDE system solutions, but they do so in very different contexts and should be considered separate efforts with their own distinct methodology. We have included further description of that methodology in order to clarify this distinction.

- Both models have been explained in more detail, including data generation and kernel choice for the domain interpolation section. The functional form of the RBF kernel is assumed to be known and much of the theory of Gaussian processes is relegated to a citation of Rasmussen, but overall much more background is explicit.

- Error quantification in the time extrapolation section has been included, both in graphical and tabular form.

- With regards to computational cost of our proposed methods: In general GPs provide $\mathcal{O}(n^3)$ inference (n being the number of training points), but can be massively sped up using new techniques such as Structured Kernel Interpolation. Here we do not implement such techniques, but note that they provide the potential for significant speed improvements in future work. Regardless, naïve GPs tie general methods for time integration (without problem-specific structure, which is rare in practice and will be nearly impossible to beat with these methods). They can also massively outperform existing methods for FEM simulation, which are $\mathcal{O}(n_p^3)$, where n_p is the number of discretized domain points, easily much larger than the size of the training set. In fact, the GP methods described in the domain interpolation section are $\mathcal{O}(1)$ with respect to n_p.

- Further justification for several design choices (i.e. squared OU kernel vs no exponent or cubed) has been included.

- To the best of our knowledge, the time extrapolation framework is not extensible to true nonlinearity of the governing equation involved. This is due to the reliance on linearity of the operator $(I - \Delta t \mathcal{L}_x)$ to induce a joint GP prior by placing only a simple GP prior on one timestep.

- In the domain interpolation portion of this project we compare our methods only to FEM simulation, and not to finite difference methods. This is due to the fact that finite difference methods are only effective on very regular domains, whereas finite element approaches are adaptable to arbitrary domain geometry. Thus we have only compared computational efficiency of the GP approach to a similarly flexible option.