MATH 676 Project for Logan Harbour

Proposed project: You project consists of implementing a preconditioner for the single group) radiation transport equation in 2D. This will involve to first simplify the neutron transport equations

$$\vec{\Omega}_d \cdot \nabla \vec{\Psi}_d + (\sigma_s + \sigma_a)\vec{\Psi}_d = \frac{1}{4\pi} \int \sigma_s(\vec{\Omega}\cdot\vec{\Omega}')\,\vec{\Psi}(\vec{\Omega}')\,d\Omega' + \frac{q}{4\pi}$$

to a single group equation by assuming that the $\vec{\Omega}_d$ dependence of $\vec{\Psi}_d$ is given by a single spherical harmonic function. (This could simply by the identity, i. e. an isotropic distribution in the beginning - but later on I would like to see that your program can also handle higher-order harmonic functions.) As a first step you will need to formulate the problem and eliminate the interaction term by enforcing the spherical harmonic behavior.

One example step that comes close (in topic) to your problem is step 28. However, this example step is mainly concerned about data structures for a multigroup problem (and we don't need this here). It might be best to just read through the example step to get an idea about what is going on and start implementing your own code from scratch based on step 9 (that solves a pure advection problem). You can change the solver to a direct solver temporarily in order to get the implementation of the equations validated. Depending on the discretization you choose you might also want to explore step 39 that demonstrates how to use the Meshworker framework for a discontinuous Galerkin approximation.

The goal of your project is to implement a preconditioner based on artificial diffusion. This will require that you first introduce artificial diffusion in your problem; what is a good strategy to choose the diffusion coefficient? And then write an approximate inverse of this problem with the `LinearOperator` framework (`https://www.dealii.org/9.0.0/doxygen/deal.II/group__LAOperators.html`). How do you want to (approximately) solve the diffuse problem that serves as a preconditioner? Can you identify a strategy to adaptively decrease the diffusion constant the closer the approximation converges to the actual solution of the transport problem?

Your project should also contain a discussion of the "acceleration" properties of your preconditioner (possibly benchmarking it against an analytic, or known solution and a parameter study that explores a bit what optimal diffusion constants are.

Note: This project is subject to modification by mutual agreement between you and me. The goal is to have a project that fits your research interests. If you would like to deviate from the path outlined here, talk to me!

Relevant tutorial programs: For your project, you will want to read through tutorial programs **9, 28, 39, 55** . You should consider starting the program you will write for your project as a variation of one of these. (Links to all tutorial programs can be found at `https://www.dealii.org/developer/doxygen/deal.II/Tutorial.html`.)

Relevant video lectures: In addition to the resources already posted on the course website (Google Drive), you should watch lectures **34, 35, 37, 38** for background material necessary for your project. Links to all videos can be found at `http://www.math.colostate.edu/~bangerth/videos.html`.

Keep notes on these resources and all background reads in your journal as on all other external resources you consult!

Project tasks: As part of your project, you will need to meet the following milestones:

– *Milestone 1 (March 18, 2019 – the week after spring break):*

  - Implement a single group (spherical harmonic) neutron transport code in 2D. Identify a corresponding diffusion problem that serves as the basis for a preconditioner. Start implementing an efficient, approximate diffusion solver.
  - You will need to prepare and give a 10 minute presentation in class on your progress so far.
  - You will also need to prepare a 1–2 page reflection essay outlining things you have learned so far. Guidance for the content is provided in the first day handout. This essay should be part of (and be clearly marked in the table of contents in) your online journal.

- *Milestone 2 (April 29, 2018):*

  - Implement an iterative solver for the neutron tranport equation with a preconditioner based on artificial diffusion. Benchmark your preconditioner (time versus accuracy and number of outer iterations).
  - You will need to prepare and give a 15 minute presentation in class or during the final exam time on the results of your project.
  - You will also need to prepare a report on your project in the style of the tutorial program – i.e., including an introduction, a results section, and commented code that contains everything to run a simulation you show in your result.

Deliverables: Your deliverables at the end of semester include the following items:

- Your final report (as discussed above in Milestone 2) as a PDF file, emailed to me or checked into a github repository to which you give me access.
- Your finished, documented code and all input files necessary to run it, checked into a github repository to which you give me access.
- Your second 1–2 page reflection essay, as outlined in the first day hand out. This essay should be part of (and be clearly marked in the table of contents in) your online journal.

Grading: I will determine your grade in this class based on the following criteria:

- Sophistication of the code beyond the program from which it was started.
- Extent of documentation in the code.
- Extent of the documentation surrounding the program, i.e., description of the equation and its properties, description of the principles used in the implementation, and documentation of worked-out examples computed with the program.
- Sophistication and realism of the testcases to which you apply your numerical scheme.

As an example of how these reports should look like (though maybe not quite as extensive), take a look at the deal.II tutorial programs.

If you are interested, good projects may be published as part of the library and distributed with future versions (see for example the step-21, step-24, and step-25 tutorial programs that were created by students of a prior class), or as part of the code gallery (see `http://dealii.org/code-gallery.html`). Of course, you will then also be credited publicly for your work.