Literature:

- Glaudin/PLC - Asynchronous implementation has not been studied
- Eckstein/PLC - Proven to work for asynchronous, but noone has programmed it. It is the first-in-class for fully nonsmooth problems.
- Application areas: low-rank + sparse decomposition – smooth, although there are likely problem-specific algos to compare.

Preliminary To-Dos:

- Keep a daily log of the research you work on – tasks completed, progress made, etc. When you learn to write in latex, you will submit this report compiled in latex. For starting out, .txt is fine.
- Acquire a method to turn .tex files to PDF; (e.g., Overleaf, pdfLaTeX, TeXShop, TeXStudio, LaTeX $\to$ dvips $\to$ ps2pdf, etc. If you use vim, I can recommend vim-latex.)
- Install Julia; familiarize yourself with Julia by running a basic experiment with the theoretical tools below (e.g., implement a few basic proximity operators from here `http://proximity-operator.net/indicatorfunctions.html` and compare with the premade MATLAB/Octave scripts available in their repo)
- **Theory and Vocabulary Questions** (if a question is not provided, please define the word **and** provide an intuitive description/example.):
  - What is the definition of a convex subset of $\mathbb{R}$? What about a convex subset of $\mathbb{R}^n$?
  - What does it mean for a function $f\colon \mathbb{R}^n \to \mathbb{R}$ to be convex?
  - *Proximity operator* (of a convex function): (hint: `http://proximity-operator.net/proximityoperator.html`)
  - *Projection operator* (of a convex set): (hint: `https://en.wikipedia.org/wiki/Hilbert_projection_theorem`)
  - An *indicator function* of a closed convex set $C \subset \mathbb{R}^n$ is given by

    $$\iota_C\colon \mathbb{R}^n \to \mathbb{R}\colon x \mapsto \begin{cases} 0 & \text{if } x \in C \\ +\infty & \text{otherwise.} \end{cases} \tag{1}$$

    Prove that the proximity operator of ι_C is the projection operator of C.
  - Define the *Gradient* of a convex function $f\colon \mathbb{R}^n \to \mathbb{R}$.
  - Prove or disprove (via counterexample): The gradient of a convex function $f\colon \mathbb{R}^n \to \mathbb{R}$ always exists.
  - What is the difference between a gradient and a proximity operator?
  - What "direction" does the gradient point towards?

- Provide a geometric description of where the vector $x - \nabla f(x)$ points.

One of the main things we are going to be comparing is various algorithms which combine gradients, proximity operators, and projection operators. A special case of these algorithms is optimization algorithms, which are at the backbone of machine learning. We are specifically going to research **asynchronous** algorithms involving these tools – we are going to implement them on challenging problems, determine the best hyperparameters for running them, and compare the algorithms between each other.

Many of the papers provided in the Literature folder are involve *firmly nonexpansive operators* – this is a class which includes proximity operators, projections, and rescaled gradient operators (we must rescale by $1/L$ where L is known as the "Lipschitz constant" of the gradient. Don't worry too much about where L comes from for now).

Computational goals:

- Implement the main algorithms in Glau20 and Comb18 with the ability to activate the operators $(T_i)_{i \in I}$ asynchronously.

Contact people

Mathieu Besançon (research) Heike Balliunet (General admin questions)