Cornell current directions and next steps

David Bindel

17 Mar 2026

Department of Computer Science
Cornell University

Two goals

- Talk about ongoing work at Cornell (and sell collaboration)
- Set up conversations re grant-writing (and Genesis)

Some recurring challenges

- Coil (and magnet) sets we can engineer
- Fast and accurate MHD equilibrium codes
- Mapping vacuum fields
- Optimized confinement in the core
- Confining energy of fast alpha particles
- Effective neutron shields and blankets
- Bulk MHD stability
- Control of turbulence
- Divertor design

(We at Cornell have worked on all but the last two!)

Recent projects

- Walk on spheres (Michael Czekanski, Adelle Wright, Ben Faber, Meg Fairborn)
- Differentiable neutronics (Xi Deng, Steve Marschner, Maosen Tang, Michael Czekanski)
- BO for particle loss (Neil Dhir. Michael Churchill, Michael Czekanski, Misha Padidar, …)
- Bulk MHD stability and optimization under stability constraints (Caira Anderson, Adelle Wright)
- More advanced spectral calculations (Adelle in 2027)
- **Other mesh-free methods for anisotropic diffusion** (Kelly Leiby, Michael Czekanski, Dennis Corraliza)
- **Cost-aware multi-objective BO** (Alex Terenin, Sihwa Park)

Mesh-free methods

Idea: Represent an unknown field in (+ PDE) a "mesh-free" way

- Often coupled with a (partly) Lagrangian discretizations (SPH, RKPM, MPM)
- Variants for strong or weak form, using collocation, finite difference, or finite volume style discretizations
- Does need a geometry representation for boundaries
- May need a background mesh for quadrature

Mesh-free methods

Example: Solve

$$\nabla \cdot (\kappa \nabla u) = f$$

where $u(x) \approx \sum_i c_i \phi(\|x - x_i\|)$ ("mesh-free").

Classic Kansa approach:

- Enforce PDE by collocation at x_i points
- Enforce (Dirichlet) BCs at boundary collocation points

Issues (partially mitigated by RBF-FD and PUM methods)

- Loss of symmetry
- Dense matrices
- Ill-conditioning

Some numerical progress

- Analytical preconditioning for floating point stability
- Lasso-style adaptive point selection

… but this is mostly orthogonal to the current case

Anisotropic case

Example:

$$\nabla \cdot (\kappa \nabla u) = f$$

What about a different ansatz for u?

$$u = \sum_i c_i \phi(\|x - x_i\|_{\tilde{\kappa}}^{-1})$$

where $\tilde{\kappa} = \kappa((x_i + x)/2)$.

- This is *not* a pos def kernel (for κ varying)
- Still may make a good approximation space with good x_i
- What are consistency and stability properties?

Bayesian optimization setup

Basic BO setup:

- Model objective with a probabilistic surrogate
- Adaptively sample to balance *exploration* and *exploitation*
- Usually balance by optimizing an *acquisition function*

Beyond: Multiple objectives and constraints, varying costs, dependencies, etc

Markov decision process framework

Ingredients:

- States $\mathcal{S}$
- Actions $\mathcal{A}$
- Transitions $P(S_{t+1} = s' | S_t = s, a_t)$
- Rewards

Output: policy $\pi_t : \mathcal{S} \rightarrow \mathcal{A}$
Goal: Choose policy to maximize expected rewards
Common: Dynamic programming for value function $\mathcal{S} \rightarrow \mathbb{R}$

BO with costs

Different cost to sample different places? Solve a Markov decision process!

- Dynamic programming gets super expensive (curse of dimensionality)
- Solve in practice with *approximate* dynamic programming
- Special case: *index*-based policies (e.g. Gittins index)

More complicated case

Have a harder problem when we want

- Multiple objectives and constraints...
- With diferent costs...
- But maybe computational dependencies between them...
- And lots of cheap correlated proxies?

Problem in *decision making under uncertainty*.

Where we are

- Gradient-based policy iteration for BO (Darian Nwankwo)
- Index-based policies for cost-aware (Peter Frazier, Alex Terenin, Ziv Scully, Qian Xie)
- Multi-objective BO (Peter, Alex, Ziv, Qian, Raul Astudillo, and Sihwa Park)
- Model spaces with trend terms (Calvin Tolbert, Darian Nwankwo, Ibrohim Nosirov)
- Incorporating bound constraints in BO (Alex Terenin, Tejal Nair)

What next?

I would love

- Collaborators and connections on current projects
- Supporting framework for current and new projects

Let’s turn to the latter.

Fusion + AI

How do we pitch stellarators + AI when Genesis proposal requests appear?

Preferred mode

> *"Machine learning is a great tool when the cost of being wrong is not too high" – Jamie Sethian*

… also, let's agree AI $\gg$ LLMs

Where does it fit?

Can benefit from:

- Community seeking to do “AI for science”
- Enabling tech (differentiable programming, faster GPUs)
- Trained models
    - For proxies and end-to-end prediction?
    - For learned closures inside deterministic models
    - For mechanistic model reduction (e.g. learned bases?)
- Learned regularizers (for design or diagnostics)
- Use in decision making (RL, control, “learning to optimize”)

Next steps

- Cornell has an “AI for science” initiative (and many others do as well) — can collaborate with this community
- Many of us will likely submit to Genesis calls — let’s collaborate!
- May be some industrial support for AI for science, too

Would love to have some large joint collaborations proposed before we (maybe) meet in March 2027.