

Differentiable physics analyses

Alexander Held¹, Gordon Watts²

¹ New York University

² University of Washington

[Snowmass 2021 Computational Frontier Workshop](#)

Aug 11, 2020

Analysis optimization

- Analysis design includes **optimization for sensitivity**
 - Typically done by manually comparing limited set of possible analysis decisions
 - Grid scans of selected parameters
 - Example: jet p_T cut: 25 or 30 GeV, which muon reconstruction algorithm to use
 - Generally not possible to simultaneously optimize all free parameters in analysis design
 - Computationally prohibitive
- **Lack of end-to-end optimization in HEP limits sensitivity and physics reach.**
- **Manual optimization is inefficient.**

Differentiable programming

- **Differentiable programming:**

- ▶ Writing programs that can be differentiated throughout, especially using automatic differentiation
- ▶ Enables end-to-end optimization

- **In HEP:**

- ▶ Want to be able to calculate gradients wrt. analysis decisions
 - e.g.: $\partial(\Delta\mu)/\partial\phi$ where $\Delta\mu$: uncertainty of parameter of interest, ϕ : any parameter in the analysis
- ▶ Requires (auto-) differentiable software
- ▶ Sensitivity optimization becomes minimization problem: use gradient descent methods
 - Can optimize millions of parameters at once
 - Benefit from experiences in machine learning

Challenges and the future

- **Challenge:** write all relevant code in an auto-differentiable way
 - ▶ Need to replace non-differentiable parts like histograms with differentiable alternatives
- **Challenge:** interfacing different simulation tools and libraries
 - ▶ Ensure gradients can be passed through
- **Challenge:** picking optimization objective and constraints
 - ▶ Best limit / lowest uncertainty may not be desirable when they come with unphysical constraints in fits
- **Benefit from developments outside of HEP** (research & industry)
 - ▶ Differentiable drop-in replacements for familiar libraries (example: numpy→[JAX](#))
 - ▶ Differentiable alternatives to common algorithms (example: sorting/ranking <https://arxiv.org/abs/2002.08871>)

Related activities and resources

- **Related activities**

- ▶ [HEP Software Foundation group on differential programming](#)
- ▶ [IRIS-HEP analysis challenge](#) for end-to-end optimized analysis



- **Further resources**

- ▶ [Talk at PyHEP about neos](#), which uses [pyhf](#) and [JAX](#) for differentiable inference
- ▶ [Differentiable Tracking in ACTS](#)
- ▶ INFERNO method: <https://arxiv.org/abs/1806.04743>

