technische universität
dortmund

Numerical methods for exploring high-dimensional phase spaces

Cornelius Grunwald

AG Kröninger - TU Dortmund

IAL Kick-Off Meeting

March 20, 2024

The Bayesian Analysis Toolkit - BAT.jl

- collection of state-of-the art algorithms for Bayesian data analysis
- widely extended re-write of a C++ tool in Julia language
- provides modern sampling approaches & new algorithms

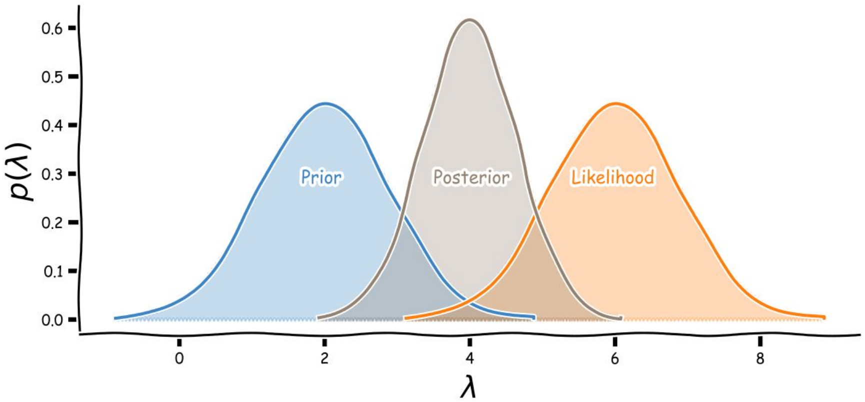
Bayesian Analysis Toolkit

https://github.com/bat/BAT.jl

Bayes' Theorem: (simple on paper, but numerics are hard)

$$P(\lambda|D) = \frac{P(D|\lambda)P(\lambda)}{\int P(D|\lambda)P(\lambda)\ \mathrm{d}\lambda}$$

D - data $\qquad \lambda$ - parameters



Development of BAT.jl partially funded by PUNCH4NFDI

The Bayesian Analysis Toolkit - BAT.jl

- collection of state-of-the art algorithms for Bayesian data analysis
- widely extended re-write of a C++ tool in Julia language
- provides modern sampling approaches & new algorithms

Bayesian Analysis Toolkit
https://github.com/bat/BAT.jl

user-specified:

- likelihood & data
- parameters & prior

provided by BAT.jl:

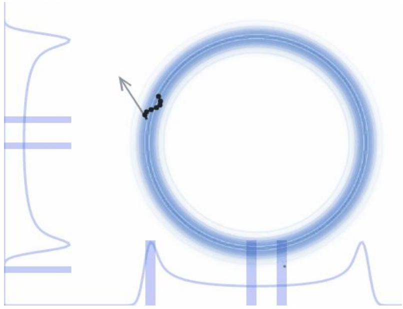
- sampling algorithms
  - MCMC sampling
  - Nested Sampling
- integration algorithms
- optimization algorithms

automated posterior exploration

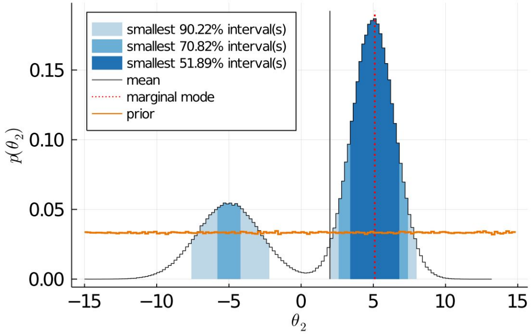
(tuning, parameter space transformations, parallelization, ...)

outputs

- samples
- plots
- modes, mean values, intervals



https://github.com/chi-feng/mcmc-demo



Features of BAT.jl

- use custom target distributions
- collection of sampling algorithms:
  - MCMC: Metropolis-Hastings, Hamiltonian-MC (e.g. NUTS)
  - Importance Samplers
  - Nested Sampling
  - in development: Normalizing flow enhanced MCMC sampling
- automated initialization, tuning & convergence tests of markov chains
- parameter space transformations
- integration algorithms: Nested Sampling, Adaptive Harmonic Mean Integration (AHMI), CUBA
- design idea: offer reasonable default settings for ease of use, but allow fine-grained control for experienced users

Interfacing BAT.jl & Sherpa for MC event generation

- HEP analyses rely heavily on MC event generators
- sampling complex processes can become very inefficient
  (e.g. when using approaches such as importance sampling)

Idea:

- use advanced algorithms for exploring phase spaces, such as MCMC & ML-based methods, for generating MC events

Our approach: Build an interface between BAT.jl (Julia) & Sherpa (C++) to easily test & develop sampling algorithms for MC event generation in HEP

First simple example - Event generation with Sherpa + BAT.jl

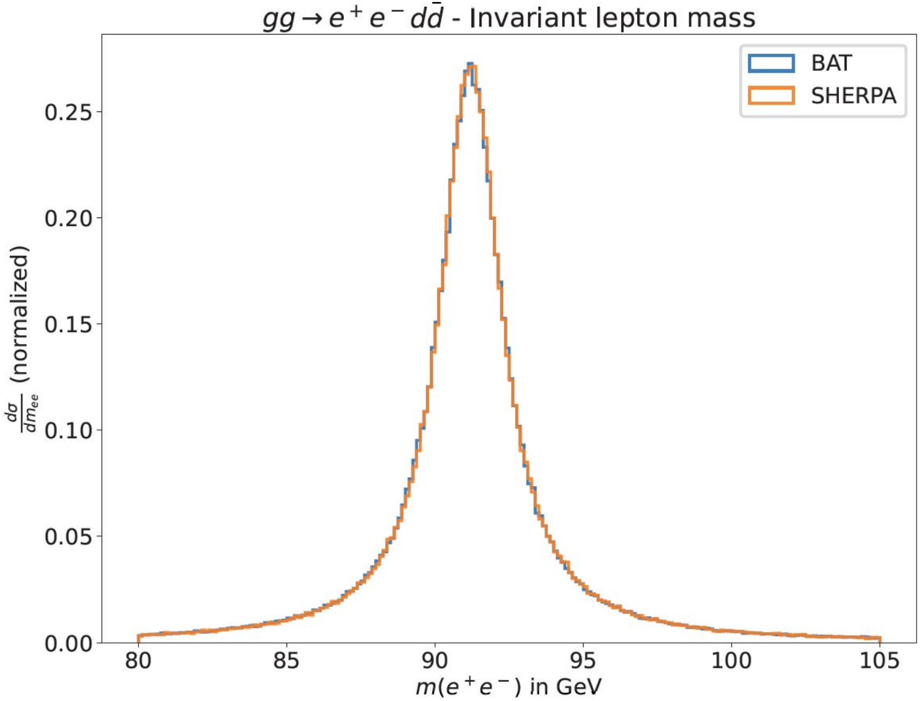
Process: $g\, g \longrightarrow d\, \bar{d}\, e^{+}\, e^{-}$ @13GeV

Phase space: 10 dimensions

Sherpa (importance sampling)

with BAT.jl (MH) interfaced

Invariant lepton mass:



First simple example - Event generation with Sherpa + BAT.jl

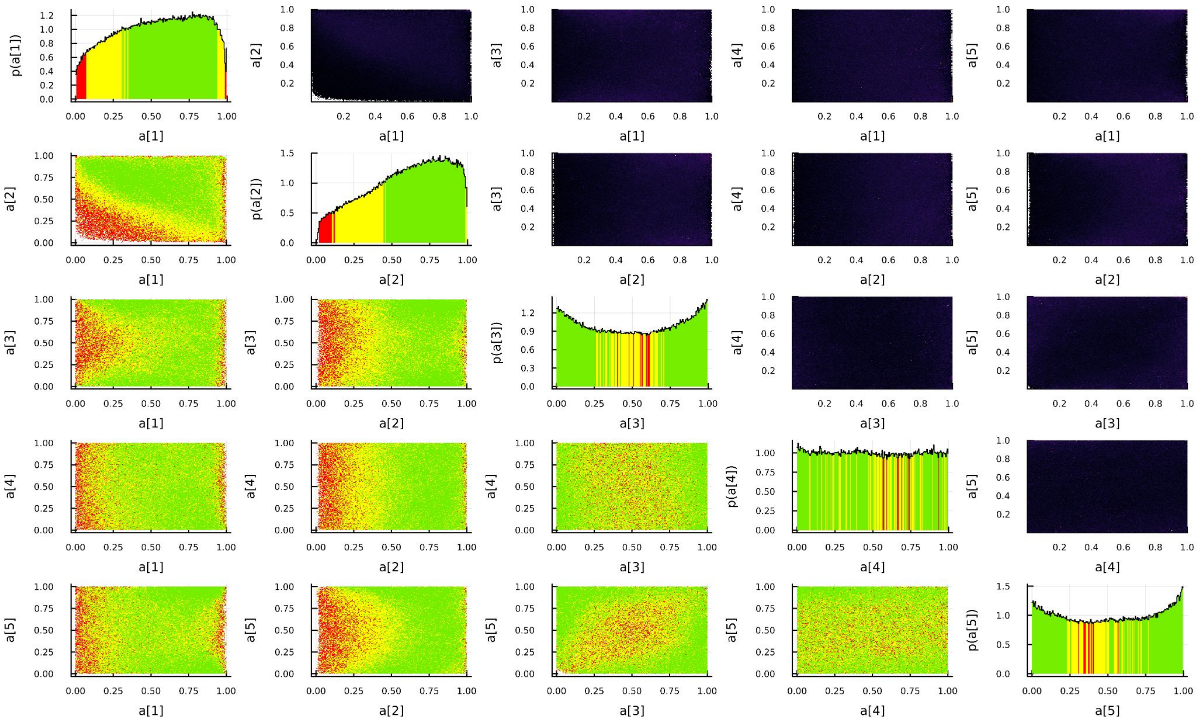
Process: $g\, g \longrightarrow d\, \bar{d}\, e^{+}\, e^{-}$ @13GeV

Phase space: 10 dimensions

Rambo mapping [1308.2922]:

map physical phase space (four-vectors) onto a *3n-4* dimensional [0, 1]-hypercube

Dimensions 1-5



First simple example - Event generation with Sherpa + BAT.jl

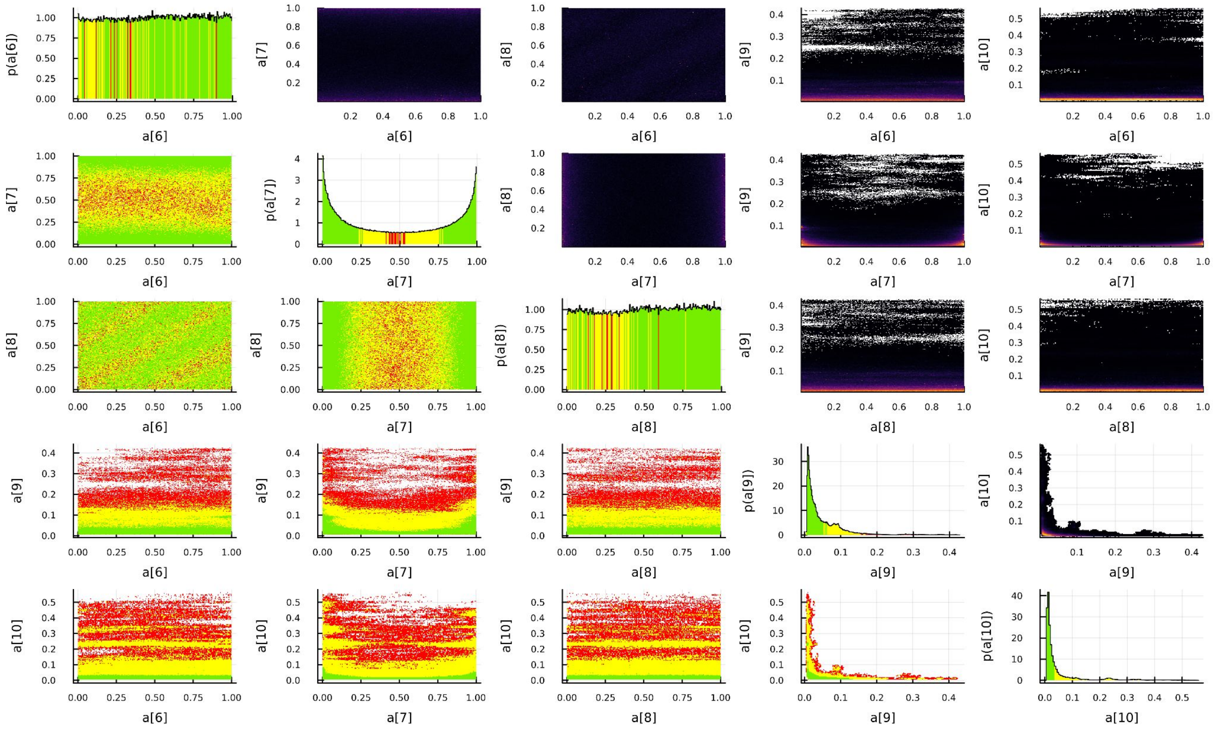
Process: $g\, g \longrightarrow d\, \bar{d}\, e^+\, e^-$ @13GeV

Phase space: 10 dimensions

Rambo mapping [1308.2922]:

map physical phase space (four-vectors) onto a *3n-4* dimensional [0, 1]-hypercube

Dimensions 6-10



Current research topics

- improving the performance of high-dimensional sampling by combining MCMC & ML methods ⇨ normalizing flow enhanced MCMC
- using advanced sampling techniques to improve efficiency of MC event generators (interfacing BAT.jl & Sherpa)
- building a test suite to validate and compare the performance of different sampling algorithms