ECHEP Generators area summary

Andy Buckley, Marek Schoenherr

~~ECHEP mini-workshop, 6 July 2020~~

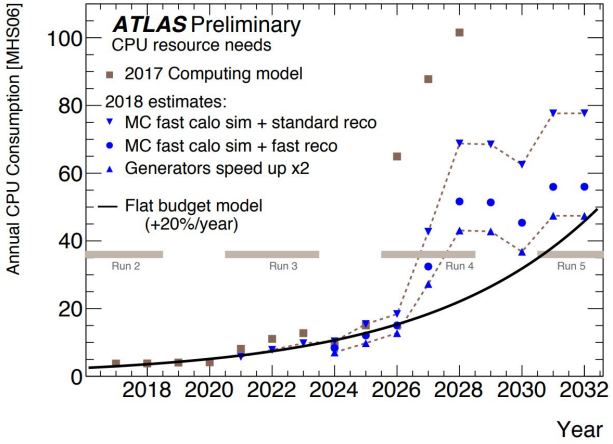
HSF Generator WG Meeting, 9 Jul 2020

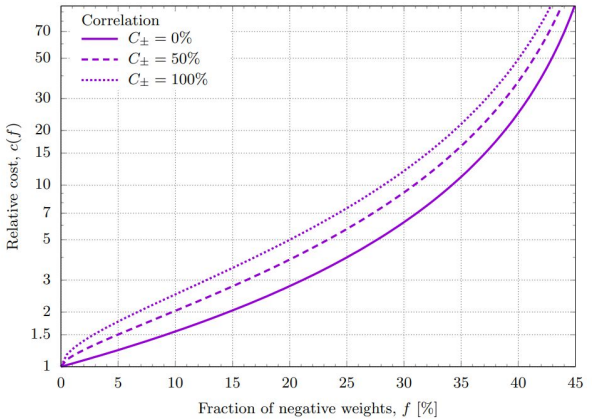
Big issues in MC event generation

Challenge: generation of large, high-precision SM samples alone will saturate HL-LHC compute budget!

Main issues are physical, technical, and logistical:

- low efficiency in ME phase-space sampling
  NNLO ~ 10x NLO ~ 100x LO!
- high rates of negative event weights from NLO ME/shower matching kill statistical power
- legacy code designs mismatch modern architectures and new usage patterns (e.g. O(100) weights/event)
- low-scale physics can also be problematic when scaled up: heavy quarks, exclusive hadron production & decay channels, colour reconnection algorithms
- evgen duplication between experiments





ECHEP

“Efficient Computing for HEP”: UK short-term project

- Minimal manpower in ECHEP itself: three small-fraction FTEs for 6 months Jan-July… but started late! Extended to late 2020 due to COVID-19

- Idea is to prototype & evidence a bid (deadline ~2 weeks, coordinated by Davide Costanzo) for a larger project: ~10 FTEs over 2 years

- Generators is on the radar: Tim Martin (Warwick) has been exploring Sherpa performance impacts, ~1.5 FTE requested in bid to start Oct 2021… long latency :-/

- Aim to complement HSF, not duplicate or step on toes!

Ways forwards

Machine learning for better ME-sampling proposal functions

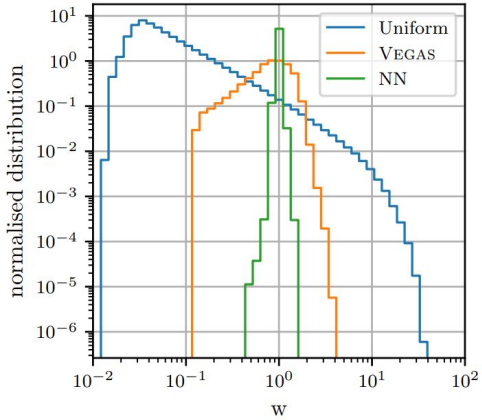
- Promising, but theoretical depth essential: already active and academically valued e.g. https://arxiv.org/abs/2001.05478

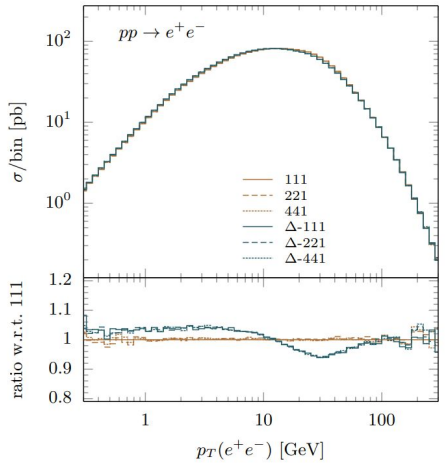
New matching schemes for negative weights

- Academically active (e.g. https://arxiv.org/abs/2002.12716)
  ⇒ public codes will be investigated by experiments

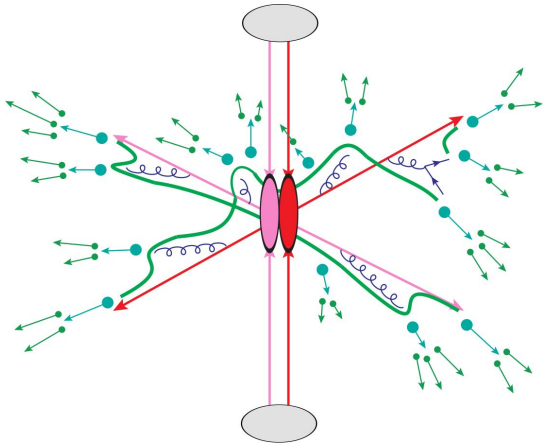
Computational efficiency and architecture mismatches

- Not high priority for theory, amenable to sw engineer and experimental physicist effort: profiling, caching, vectorising
- Similar for efforts to take advantage of HPC facilities, hardware accelerators, etc.: US HPC efforts, HDF5 formats
- good match to UK HEP software initiatives ✓





Ways forwards (2)



Inefficient low-scale physics modelling

- Efficient hadronisation biasing (decays not a bottleneck) ✓
- Colour reconnection algorithms: latest Pythia model better, but too expensive for LHCb. Convert to use accelerators? ✓

Evgen duplication between experiments

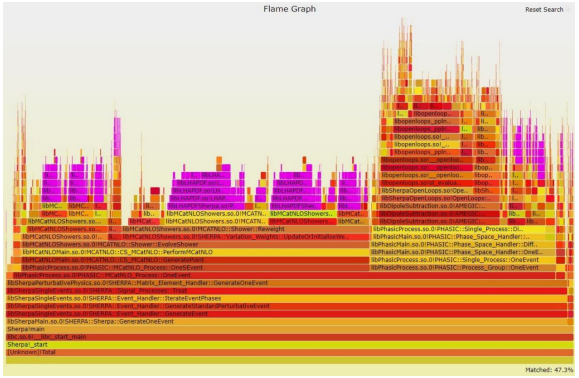
- not so bad since ATLAS=Sherpa/CMS=MadGraph, but physics impact would be better if ME events shared
- experiment-side initiatives exist, cf. HSF and HDF5 format
- multiple output streams for flavour filtering: ~avoid cost of light & charm samples — *b* unavoidably (?) expensive. Any current HSF effort??

Need to be realistic about goals vs. FTE, but can dream…

Activity under ECHEP

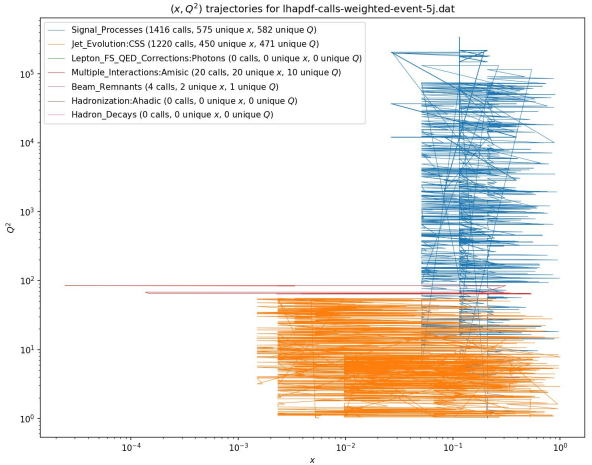
Profiling Sherpa in experiment-like setups:

- study of multiweight, NLO V+jets (thanks, HSF!)
- main CPU sinks = PDFs & transcendental functions
- 10% gain from Intel math lib, but already in LHC use
- new LHAPDF 6.3 introduced caching, but needs to be *called* efficiently, too: potentially O(50%) gain
- now working on fast approx log and exp functions
- for details see Tim Martin's talk



Next activities:

- investigate accelerator API for Pythia8's colour reconnection model as simple system
- time permitting, investigate hadronisation and hadron decay biasing



Proposed Event Generator programme

Performance study and technical optimisation

1) Detailed report on CPU-performance profiling of high-precision Monte Carlo event generation to identify bottlenecks for cost-driving setups used by experiments
2) Optimisation of the PDF evaluation through major contributions to software packages (e.g. LHAPDF)
3) Technical generator optimisations and refinements for usage by experiments through major contributions to software packages (e.g. Herwig, Sherpa)

Development of generator models

1) Develop a prototype of a biased hadronisation method for Pythia8
2) ~~Development of similar functionalities in Herwig and Sherpa~~

Broader programme under widened scope

JIT optimisation

- event generators are universal codes at prototype stage, featuring plenty of recurring branch points of constant evaluation outcome in a given run (eg. check that hadron collider setup and both initial states need PDFs, ME has n legs, etc)
- in many cases (probably) not caught/anticipated by current CPUs (esp. In light of spectre/meltdown fixes)

Large-scale refactorings of generator codes

- different parts of event generation lend themselves to different types of acceleration
  ME -- constant program flow, large expressions (multithreading, vectorisation, GPU)
  rest -- variable program flow (Markov Chains, etc), best parallelised trivially
- needs software engineering expertise, probably also target machine specific opts

Common decay package (including systematics)

- hadron decays mostly non-controversial, provide common hadron decay package

Real world problems

Incentives are often misaligned: need to design around this

- MC generators are organisationally developed under HEP theory grants
- Career progression still strongly coupled to “theory” papers, not sw performance
- Less institutional room & reward than in experiment for technical work
- If hiring fractional FTEs (we are!), how to ensure career path?
  Essential to attract right people, which is crucial.
- Organising how (remote?) fractional FTEs will embed in MC gen groups.
- If experimentalists, need their gen work FTE to earn service credit. Work on external tools which primarily benefit experiments needs to be allowed and recognised — perhaps by classifying as software, not physics