New gen level efficiencies

- Recomputed prompt gen-level efficiencies for those generator particles which are not from long-lived secondaries.
- Adopted existing algorithms using GaudiPython/Bender to run over generator-level simulation and record nTuples with bools for the `DaughtersInLHCb` cut.
- Bools evaluated with BenderTools `GenCutTool`.
- Access the `HepMC::GenParticle` objects to obtain the information on our Λ_c ancestry.
- From Appendix A in note - all secondaries come from either long-lived beauty or from double charm baryons.
- So, if any Λ_c ancestor is a beauty particle, or is a double charm baryon, don't include the Λ_c in our nTuple.

Algorithm tweaking

- DecayFinder searches intermediate resonances - can include decays of the form:
  - $\Lambda_c^+ \to pK^-\pi^+\pi^0$ where the π^0 decays via. $\gamma\ \gamma$.
  - $\Lambda_c^+ \to pK_S^0$ where the K_S^0 decays via $\pi^+\pi^-$.
- Explicitly check each Λ_c decendant at all tree levels to check for presence of K_S^0 or π^0 and veto accordingly.
- Encountered odd functionality in algorithm - `AlgoMC::gselect(tag, decayFinder)` used to return the list of Λ_c `HepMC::GenParticles` in event matching DecayFinder - but it returns duplicates of each Λ_c candidate.
- Use Λ_c energy as unique identifier while iterating over `GRange` container.

Secondary fractions in simulation

- Cross check this against secondary fractions from Appendix A.
- Some entries missing due to hanging jobs.
- Don't expect these to be equal - MC11a has gen-level cuts so per-mode acceptance should vary due to kinematic differences.
- At gen level secondary fraction per-mode should be compatible.
- This is broadly the case. Looks like gen-level cuts are cutting more secondary than prompt.

Mode	Polarity	MC11a secondary frac [%]	Gen-level secondary frac [%]
$\Lambda_c^+ \to pK^-\pi^+$	MagDown	8.93 ± 0.03	10.9 ± 0.1
	MagUp	8.92 ± 0.03	11.0 ± 0.1
$\Lambda_c^+ \to pK^-K^+$	MagDown	9.02 ± 0.03	10.9 ± 0.1
	MagUp	8.99 ± 0.03	10.9 ± 0.1
$\Lambda_c^+ \to p\pi^-\pi^+$	MagDown	8.87 ± 0.03	11.2 ± 0.1
	MagUp	8.85 ± 0.03	11.3 ± 0.2
$\Lambda_c^+ \to p\pi^-K^+$	MagDown	8.87 ± 0.03	11.1 ± 0.1
	MagUp	8.91 ± 0.03	-

Prompt vs all gen level effs

- Compare prompt-only Λ_c gen-level effs to those from taken from logfiles with `MCStatTools` package - which are all Λ_c.

Polarity	Part/Antipart	Decay mode	Logfile Eff	True Prompt Eff
MagUp	Particle	$\Lambda_c^+ \to pK^-\pi^+$	0.208 ± 0.001	0.211 ± 0.002
		$\Lambda_c^+ \to pK^-K^+$	0.234 ± 0.001	0.237 ± 0.002
		$\Lambda_c^+ \to p\pi^-\pi^+$	0.194 ± 0.001	0.199 ± 0.003
		$\Lambda_c^+ \to p\pi^-K^+$	0.211 ± 0.001	-
	Antiparticle	$\Lambda_c^- \to \overline{p}K^+\pi^-$	0.219 ± 0.001	0.218 ± 0.002
		$\Lambda_c^- \to \overline{p}\pi^+\pi^-$	0.242 ± 0.001	0.244 ± 0.002
		$\Lambda_c^- \to \overline{p}K^+K^-$	0.200 ± 0.001	0.201 ± 0.004
		$\Lambda_c^- \to \overline{p}\pi^+K^-$	0.217 ± 0.001	-
MagDown	Particle	$\Lambda_c^+ \to pK^-\pi^+$	0.210 ± 0.001	0.210 ± 0.002
		$\Lambda_c^+ \to pK^-K^+$	0.234 ± 0.001	0.235 ± 0.002
		$\Lambda_c^+ \to p\pi^-\pi^+$	0.193 ± 0.001	0.195 ± 0.002
		$\Lambda_c^+ \to p\pi^-K^+$	0.211 ± 0.001	0.210 ± 0.002
	Antiparticle	$\Lambda_c^- \to \overline{p}K^+\pi^-$	0.214 ± 0.001	0.220 ± 0.002
		$\Lambda_c^- \to \overline{p}\pi^+\pi^-$	0.243 ± 0.001	0.242 ± 0.002
		$\Lambda_c^- \to \overline{p}K^+K^-$	0.199 ± 0.001	0.203 ± 0.002
		$\Lambda_c^- \to \overline{p}\pi^+K^-$	0.217 ± 0.001	0.217 ± 0.002

Summary

- Efficiencies change outside of errors but not by much.
- $\Lambda_c^+ \to p\pi^- K^+$ efficiencies don't follow this trend - although we only have two values to base this on.
- Most efficiencies increase slightly compared to before - overall effects on relative branching fractions should be low to negligible.
- As a cross check want to invert my filter's logic and produce gen-level tuples for the secondaries to check these results make sense compared to our MC11a.