x-boa - Feature #1152

Query Re: applying cuts through xboa

08 October 2012 14:24 - Carlisle, Timothy

Status:	Closed	**Start date:**	08 October 2012
Priority:	Normal	**Due date:**	
Assignee:	Rogers, Chris	**% Done:**	100%
Category:		**Estimated time:**	0.00 hour
Target version:			

Description

Hi Chris,

I'm trying to exclude an outlying particle at all virtual planes in order to calculate a sensible emittance. Using code lifted from xboa Example 3, I'm cutting on y using:

bunch_list[100].cut({'y':-300.}, operator.le)
bunch_list[100].cut({'y':300.}, operator.ge)

which works fine at one plane (as shown by running the plotter on the data file (both attached)). But using:

bunch_list[100].cut({'y':-300.}, operator.le, global_cut=True)
bunch_list[100].cut({'y':300.}, operator.ge, global_cut=True)

removes all the events, rather than simply applying to cut to all the planes...

What am I doing wrong?

Thanks,

Tim

Associated revisions

Revision 90 - 30 January 2013 12:29 - Rogers, Chris

weight uses hash of spill/event/particle (fixes #1152, #612); add muon1_output.csv to tests; some messing with system tests

History

#1 - 10 October 2012 13:43 - Rogers, Chris

Sorry, am in US for some meeting so communication is difficult... let me have a look.

#2 - 10 October 2012 14:08 - Rogers, Chris

I think I know what is going on, Can you check - in MAUS output what is the event number and particle number? I think that I do global cuts by event number only (i.e. was this particle in bunch 1 the same as the particle in bunch 2?), but in MAUS I ID hits by *event number = Spill* and *particle number = primary* - which comes from retrofitting the MAUS data structure to an existing python code.

I can make a feature to add *particle number* to the global cut logic. It is a little complicated as I need to go into the C code... workaround is to assign manually (sorry) event numbers. But can you verify that this is what is going on?

#3 - 10 October 2012 16:06 - Carlisle, Timothy

Yep looks like event_number = Spill Number for the Virtual Hits (i.e. 1 in the example I posted).

Thanks

#4 - 22 October 2012 10:21 - Carlisle, Timothy

Hey Chris,

At the moment I'm looping over all hits in all planes & applying rather rudimentary cuts (w/o a transmission cut)...and it's all rather slow. Will the C code mod. speed things up also? Do you have an eta? I can then get stuck into checking step length, and [hopefully] present @ the next Analysis meeting.

Thanks

#5 - 28 October 2012 20:18 - Rogers, Chris

What are your cuts? Is the slow bit assigning event numbers, making the cuts or looking at the output? Or something else?

#6 - 29 October 2012 11:18 - Carlisle, Timothy

I simply loop over all hits in all planes & cut (v. basic, slow):
@
bunch_list = Bunch.new_list_from_read_builtin(filetype, filename)

for b in range(len(bunch_list)):
print "Plane " + str(b)
for a in range(len(bunch_list[b])):
bunch = bunch_list[b]
bunch.cut({'y':-300.}, operator.le, global_cut=False)
bunch.cut({'y':300.}, operator.ge, global_cut=False)
bunch.cut({'x':-300.}, operator.le, global_cut=False)
bunch.cut({'x':300.}, operator.ge, global_cut=False)
bunch.cut({'pz':150.}, operator.le, global_cut=False)
@
Is there a simpler/faster way, and one that enables me to make a transmission cut?

#7 - 26 November 2012 16:03 - Carlisle, Timothy

is this too fiddly/time consuming, or still live? Thanks

#8 - 26 November 2012 17:28 - Rogers, Chris

Nope, sorry I was distracted onto other things. Will try to get it done this week.

#9 - 26 November 2012 17:28 - Rogers, Chris

- Tracker changed from Support to Feature

#10 - 30 November 2012 15:15 - Rogers, Chris

- Project changed from MAUS to x-boa

Moved to x-boa...

#11 - 30 November 2012 15:19 - Rogers, Chris

So I implemented a native C routine for the inner loop of cut(...). Code and tests are committed to xboa launchpad as revision 89, and will go in the next release when it is ready.

bzr://xboa.bzr.sourceforge.net/bzrroot/xboa

Note that this is my development version - I think it is okay, failing tests to do with loading data from an obscure file format, but otherwise should be okay. I will just run some time trials on it also to see whether it is really any faster.

#12 - 30 November 2012 16:08 - Rogers, Chris

- Status changed from Open to Closed

- % Done changed from 0 to 100

So I did some time tests - I timed taking a cut on 1e6 particles, tried it 10 times, got the following results.

	python	native C
Mean	2.66	0.14
Standard Deviation	0.31	0.005

I should also say, I only implemented the optimised code for cuts on float types e.g. position and momentum variables. I can do the same for integer types e.g. event number, pid, etc. Let me know if you need it. Closing the issue for now.

#13 - 30 November 2012 16:10 - Carlisle, Timothy

Thanks! Don't think I'll use it for anything other than 6D variables. I'll try it out next week.

#14 - 07 December 2012 12:34 - Carlisle, Timothy

- File Sim_10mm_20MeVc_50mu_1e-08nsec_IV_LH2i__cutTest2_.root added

- File Test_Cuts.py added

Hey Chris, can you advise on how apply these global cuts to virtual hits? I'm following the approach of Example 3, but cutting on Pz, and not getting anywhere. Do I have to take a different approach? I've looked at the repository for your modifications but don't really get where to start with them. A sample data file & the test script are attached.

i) Plot Pz Distribution at plane 0.

ii) Local cut (Pz < 199 MeV/c) --> plot, observe effect (plot & print local weight etc)

clear weights

iii) Global cut (same condition, global_cut = True)--> all global weights go to zero omitting all events so nothing plotted...

Thanks

#15 - 07 December 2012 12:50 - Rogers, Chris

- Status changed from Closed to Open

- % Done changed from 100 to 0

#16 - 28 January 2013 11:58 - Rogers, Chris

See also #612

#17 - 30 January 2013 13:12 - Rogers, Chris

- Status changed from Open to Closed

- % Done changed from 0 to 100

Applied in changeset commit:chris.rogers@stfc.ac.uk-20130130122928-9udbdz6zaftbv7nz.

Files

Plot.py	4.13 KB	08 October 2012	Carlisle, Timothy
Sim_4mm_1MeVc_108mu_1e-08nsec_IV_LH2i__RUNTEST_.root	998 KB	08 October 2012	Carlisle, Timothy
Sim_10mm_20MeVc_50mu_1e-08nsec_IV_LH2i__cutTest2_.root	469 KB	07 December 2012	Carlisle, Timothy
Test_Cuts.py	1.7 KB	07 December 2012	Carlisle, Timothy