Subject: Re: Position calculations on start/stop scintillators
Posted by miree on Tue, 19 May 2015 12:29:11 GMT
View Forum Message <> Reply to Message

Hi Tayfun,

The second calibration coefficient (the slope) corresponds to how many ns delay per mm scintillator material the light has. Yes, it should be the inverse of the speed of light in that medium. But empirically the slope is in the order of 0.01 ns/mm. If the speed of light inside the scintillator would be c/n, the value of n would be n=3 which is twice the expected value. That means there must be other mechanisms that delay the light. Perhaps it is simply the geometry of the multiple reflections...

Quote:
Regarding to the part that you explained how should one treat the data without any reference interaction point, I think I'm lost when you say;

Quote:
... all possible combinations of the 64 calibration coefficients ...

What are the all possible combinations? Do you mean all combinations of the PMT pairs?

I mean all possible calibration coefficients of all PMTs.
I'll give an example: the first calibration coefficient (the offset) has values around 0ns in the range of approx. -10 ns ... +10 ns.
The second coefficient for a PMT is in the range of 0.005 ns/mm .... 0.015 ns/mm.
If you know nothing else about the calibration coefficients, you can (theoretically) do the following: choose some values (32 offsets and 32 slopes) in the given ranges for each PMTs and analyze some data with this calibration. Then you see how good the chosen calibration coefficients are. Then do the same choosing a different calibration for all PMTs (again 32 offsets and 32 slopes) and analyze some data again. This you continue until you have found a good set of calibration coefficients.

In principle you randomly choose a set of calibration coefficients for all PMTs and hope that it is the right one. Of course, it is extremely unlikely to get by chance the correct calibration coefficients. In practice one would choose an algorithm for this kind of high-dimensional search, for example: http://en.wikipedia.org/wiki/Nelder%E2%80%93Mead_method , is what I meant with "simplex method" in my earlier post, but apparently it is also known as "NelderMead method". If one has reasonable starting values, maybe set all offsets to 0 and all slopes to 0.01, the algorithm will make small modifications to all calibration coefficients and eventually find better ones. If you let it run for long enough, you can hope to get the best possible calibration.

I don't know if that was any more clear than the previous post... It is a problem of multidimensional optimization. Maybe you can read something about this technique in general.

Best regards,
Michael