Absorber Splitter

Josiah D. Kunz

02.10.14

Flat Absorber

- 32 cm flat absorber preceded and followed by 10 cm drifts
- "Accepted values" based on 5 different ICOOL runs
- 25 splits ≈ 2 mins

SPLIT=10	COSY	AVG ICOOL	%	Accepted %
pz	188.22	188.19	0.014	0.005
σpz	20.76	20.79	0.149	0.035
t	1.99	1.99	0.004	0.001
σt	0.053	0.053	0.291	0.266

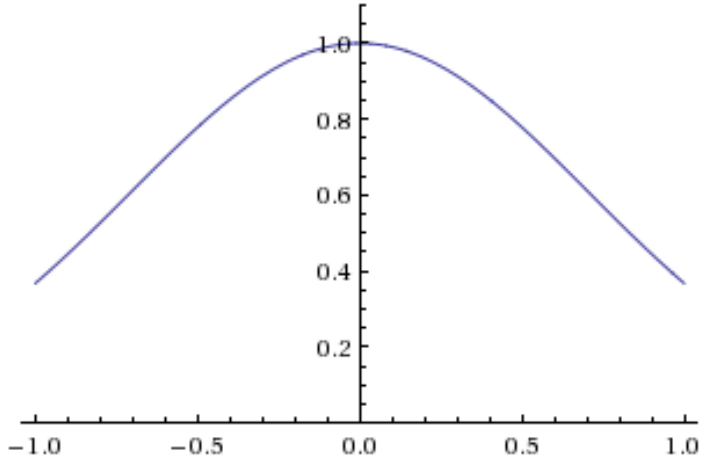
SPLIT=25	COSY	AVG ICOOL	%	Accepted %
pz	188.21	188.19	0.010	0.005
σpz	20.76	20.79	0.187	0.035
t	1.99	1.99	0.001	0.001
σt	0.053	0.053	0.282	0.266

SPLIT=50	COSY	AVG ICOOL	%	Accepted %
pz	188.20	188.19	0.005	0.005
σpz	20.77	20.79	0.140	0.035
t	1.99	1.99	0.001	0.001
σt	0.053	0.053	0.119	0.266

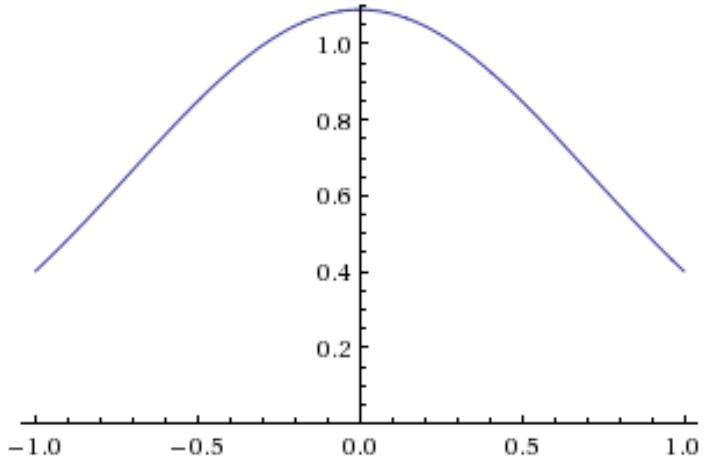
SPLIT=100	COSY	AVG ICOOL	%	Accepted %
pz	188.19	188.19	0.001	0.005
σpz	20.77	20.79	0.104	0.035
t	1.99	1.99	0.006	0.001
σt	0.054	0.05	0.221	0.266

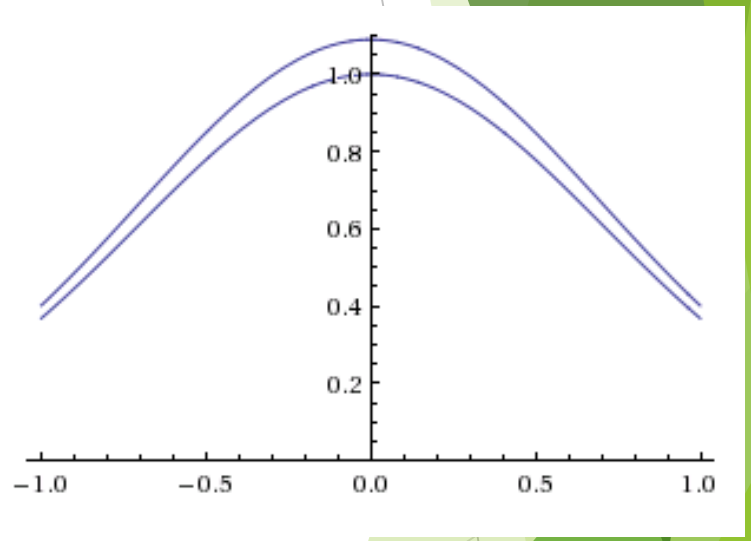
Other Ideas: Distribution Method (1/3)

COSY:

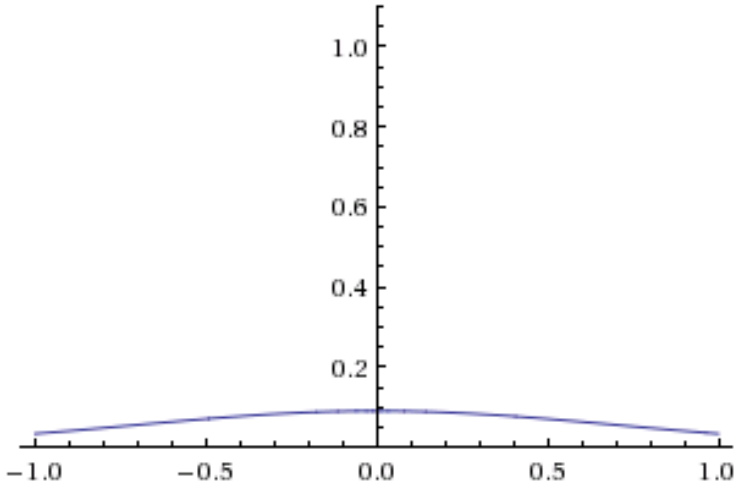


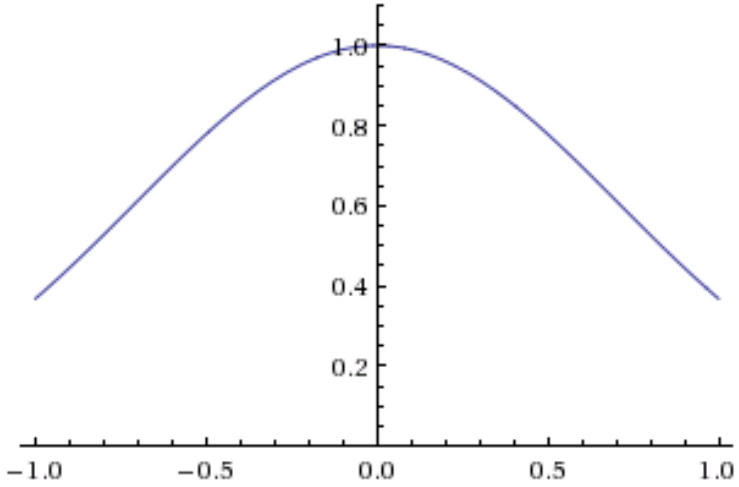
ICOOL:





Distribution Method (2/3)

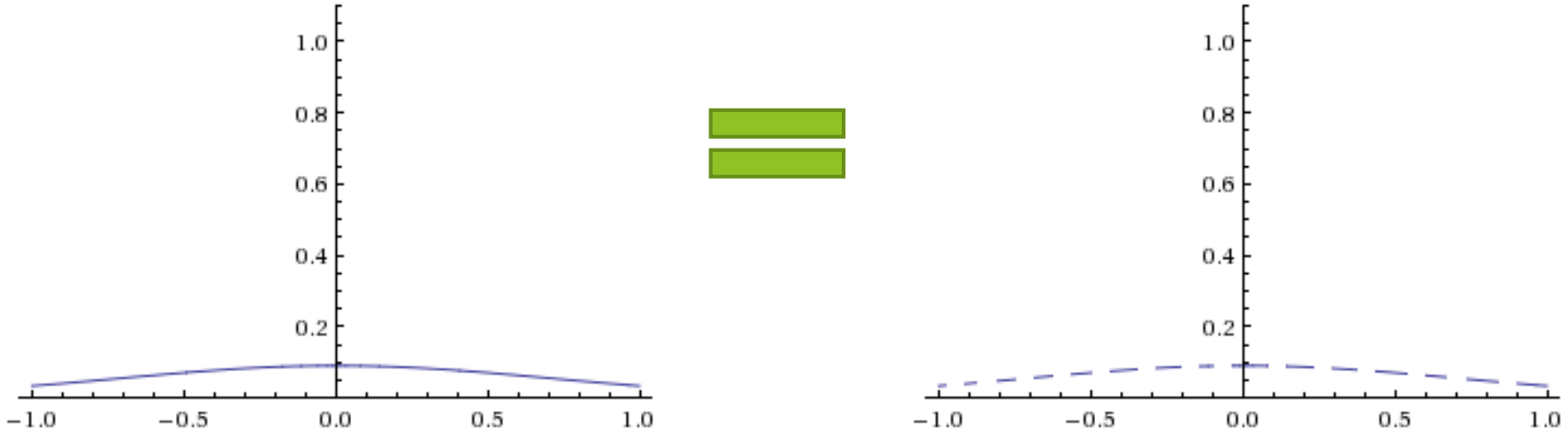




- Adding the difference back to COSY will create a realistic final distribution

Distribution Method (3/3)

- To get the difference distribution, we may not need to run the full number of particles



- We only need the distribution's average and standard deviation to reconstruct it (provided that the difference distributions are Gaussian)
- This can be approximated by using a relatively small number of particles

Other Ideas: Decay

- 1) Decay particles in ‘UPDATE’. Requires user to run external command (i.e. ‘DECAY’), with the particle vector as the argument.
- 2) Decay particles in ‘POLVAL’. Users worried about decays are already using vectors, and are necessarily using POLVAL. This would let POLVAL call another subroutine that simply updates whether or not the particle has decayed, giving the user more control and fewer commands to worry about.
- Other: assign mean lifetime values automatically based on reference particle mass and charge (e.g. anything between 510 and 512 KeV skips the decay routine altogether, while anything between 105 and 107 MeV has $\tau = 2.197$ μs).