MRI Simulator Notes

Liam

Friday 29th November, 2019

Tasks

- I looked into distributed computing using Python.
- I looked into random number generation.
- I implemented excitation pulses in our simulator.

Distributed computing using Python

- The module Ray seems well-suited to our application.
- Ray allows you to modify serial code for distributed computing (using a cluster or in the cloud) by simply adding decorators to existing functions and classes (reference).
- See the reference page here.

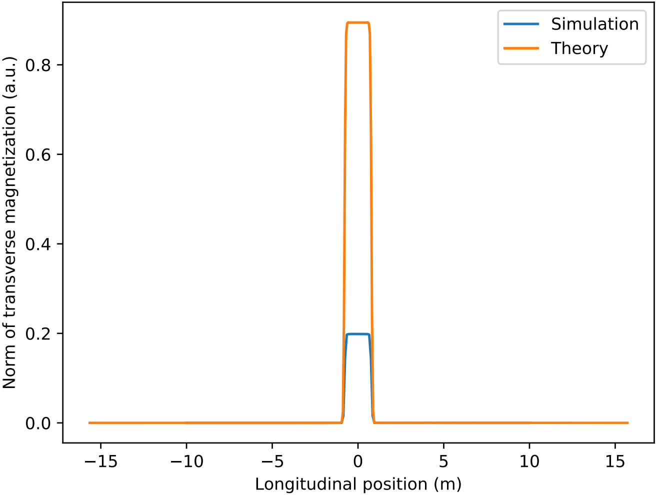
Random number generation

- If we want fast random number generation, I think we are better off with a software pseudorandom number generator.
- Hardware random number generators create a sequence of numbers that is more truly random; however, software pseudorandom number generators have a higher output rate.
- This Wikipedia entry says "hardware random number generators generally produce only a limited number of random bits per second."
- Cherkaoui et. al. 2013 report on a hardware random number generator that outputs 200 Mb/sec.
- Compare with the Xorshift class of software pseudorandom number generators, which outputs (200 million 32-bit ints)/sec = 6.4 Gb/sec (reference) .

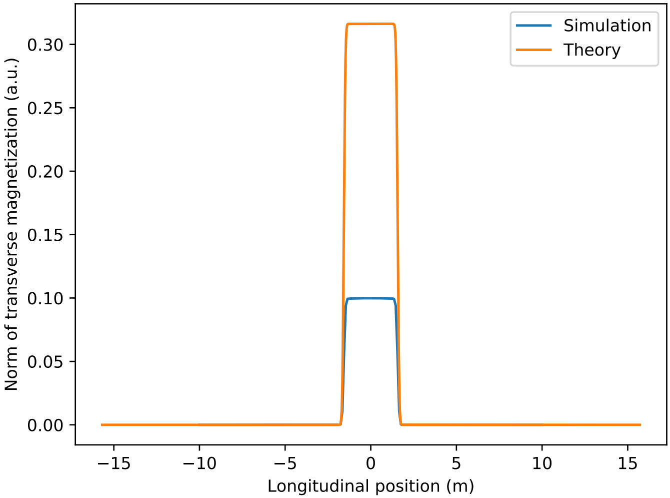
Excitation pulses in our simulator

- To test my excitation routine, I initialized a simulation with 200 stationary ems evenly spaced along the z axis between -10 m and 10 m.
- I applied a Gaussian-windowed sinc excitation pulse with duration 5.0 seconds, 10.0 seconds, and 20.0 seconds.
- The transverse magnetization profile from the simulation ("Simulation") and from the small-tip-angle solution ("Theory") are shown in the following figures.
- I am not sure why the transverse magnetization norms are different between simulation and theory, but there is agreement between the two on the spatial extent of the excitation.
  - I think the error lies in my coding of the small-tip-angle solution. When I compute the tip angle at the origin by $\alpha = \int \gamma B_1(t) dt$ and compute $\mu_0 \sin(\alpha)$ I get agreement with the simulation ($\mu_0 =$ equilibrium longitudinal magnetization).

Pulse duration 5.0 seconds



Pulse duration 10.0 seconds



Pulse duration 20.0 seconds