MRI Simulator Notes

Liam

November 26, 2019

Tasks

- I looked into the use of quaternions to represent rotations in 3D space.
- I looked into the means by which two different MRI simulators, SIMRI and JEMRIS, implement a numerical solution to the Bloch equation.

Quaternions

Quaternions

- A quaternion q is a number that can be written in the form

$$q = a + bi + cj + dk\,,$$

  where a, b, c, d are real numbers and i, j, k are the imaginary units.
- The real part of q is the number a and the vector or imaginary part of q is the triple (b, c, d).
- The imaginary units follow rules for multiplication $i^2 = j^2 = k^2 = ijk = -1$.
- Multiplication of two quaternions is associative but not commutative.

Orientation and Rotation Quaternions

- Quaternions give a compact way to represent orientations and rotations in 3D space.
- Let $(p_1, p_2, p_3) \in \mathbf{R}^3$ be a position vector representing the orientation of an object in 3D space. The corresponding orientation quaternion is

$$p = p_1 i + p_2 j + p_3 k\,.$$

- Let $(u_1, u_2, u_3) \in \mathbf{R}^3$ be a unit vector denoting an axis of rotation let and $\theta \in [0, 2\pi]$ be an angle of rotation. The corresponding rotation quaternion is

$$\begin{aligned} q &= \exp(\theta/2(u_1 i + u_2 j + u_3 k)) \\ &= \cos(\theta/2) + (u_1 i + u_2 j + u_3 k)\sin(\theta/2)\,. \end{aligned}$$

The latter expression follows from a generalization of Euler's formula.

Orientation and Rotation Quaternions

- The rotation of (p_1, p_2, p_3) about the axis (u_1, u_2, u_3) by angle θ yields the vector (p_1', p_2', p_3'), which may be computed using quaternions as follows:

$$(p_1', p_2', p_3') = \mathbf{vec}\{qpq^{-1}\}\,.$$

- The rotation is clockwise if viewed along the direction of (u_1, u_2, u_3).
- The composition of rotations q_1 and q_2 may be represented by the single rotation quaternion $q_2 q_1$.
- reference

Quaternions versus Rotation Matrices

- The composition of rotations using quaternions requires *fewer* computations than the composition of rotations using rotation matrices.
- Rotating a vector using a quaternion requires *more* computations than rotating a vector using a rotation matrix.
- reference

My thoughts on quaternions

- Is it more computationally efficient to use quaternions or rotation matrices for our application? It's not clear to me which offers the advantage.

Numerical Solution of the Bloch Equation

Bloch Equation

- Represent the magnetization by $\mathbf{m} = m_x \mathbf{x} + m_y \mathbf{y} + m_z \mathbf{z}$.
- Let the equilibrium longitudinal magnetization be m_0.
- We wish to solve the Bloch equation

$$\frac{d\mathbf{m}}{dt} = \mathbf{m} \times \gamma \mathbf{B} - \frac{(m_z - m_0)}{T_1}\mathbf{z} - \frac{(m_x \mathbf{x} + m_y \mathbf{y})}{T_2}$$

numerically.

SIMRI

- SIMRI is an MRI simulator from 2005.
- SIMRI stores $\mathbf{m}(\mathbf{r}, t)$ on a grid.
- The magnetization vector is updated according to

$$\mathbf{m}(\mathbf{r}, t + \Delta t) = Rot_z(\theta_g) Rot_z(\theta_i) R_{\mathrm{relax}} R_{\mathrm{RF}} \mathbf{m}(\mathbf{r}, t)\,.$$

- θ_g and θ_i are precession angle changes due to the applied gradients and field inhomogeneities.
- R_{relax} captures relaxation effects on the magnitude of m_x, m_y, m_z.
- R_{RF} is a rotation by a specified flip angle about a specified axis in the transverse plane due to an RF pulse.
- reference

My thoughts on SIMRI

- Advantage: Bloch equation is implemented using matrix multiplication alone.
- It might be more efficient to implement $Rot_z(\theta_g)Rot_z(\theta_i)R_{\mathrm{RF}}$ using quaternions.
- Disadvantage: Cannot simulate arbitrary excitations, so "selective excitation cannot be studied [with the SIMRI approach]" reference.

JEMRIS

- JEMRIS stores the magnetization $\mathbf{m}(\mathbf{r}, t)$ on a grid.
- JEMRIS solves the Bloch equation in cylindrical coordinates using the ODE solver CVODE.
- The magnetic field $\mathbf{B}(\mathbf{r}, t)$ can be an arbitrary function of position and time.
- Each position and time is associated with a set of physical parameters including m_0, T_1, T_2, $T_2^\star$.
- reference

My thoughts on JEMRIS

- Advantage: JEMRIS can simulate arbitrary excitations, so we can simulate selective excitation.
- Disadvantage: Using the CVODE solver may be too computationally intensive for our purposes.

We could use a hybrid approach?

- Maybe we can simulate the excitation phase with the JEMRIS method and the readout phase with the SIMRI method?
- Maybe we can simulate the excitation phase with the small-tip-angle solution to the Bloch equation? I'm not sure how this would be implemented, but it might be faster than the JEMRIS method.