Bloch Simulation Acceleration for Fast Pulse Design in Parallel Transmit

S-K. Lee[1], D. Xu[2], S. M. Lechner[3,4], and M. W. Vogel[3]

[1]GE Global Research, Niskayuna, NY, United States, [2]Applied Science Laboratory, GE Healthcare, Waukesha, WI, [3]Advanced Medical Applications Laboratory, GE Global Research, Munich, Bavaria, Germany, [4]Department of Scientific Computing in Computer Science, Technical University Munich, Munich, Bavaria, Germany

Introduction: Iterative calculation of parallel transmit (pTx) RF pulses based on the optimal control theory [1] is versatile but slow since it relies on a large number of Bloch simulations for cost function and gradient calculations. Recent work by Grissom et al [2] focused on reducing the design time by linear approximation of the Bloch equations around a particular solution. Here we propose analytical and computational methods to improve the speed of the Bloch simulation itself. In the first method, we formulate the optimal control theory in a "local rotating frame" which transforms away all longitudinal magnetic fields present in the system and thereby simplifies the numerical workflow. In the second method, we reduce the arithmetic calculation time by shifting the calculation load from the conventional Central Processing Unit (CPU) to a Graphics Processing Unit (GPU) [3]. Both methods provide significant gain in optimization speed with negligible impact on the end result.

Methods: Spatial RF pulse design for a given gradient pulse can be formulated in a frame of reference where local longitudinal fields are analytically transformed away. For the optimal control pulse design in pTx, the frame transformation corresponds to (i) removing all gradient fields in the Bloch simulation, and instead (ii) multiplying the B_1+ maps $s_1(\boldsymbol{r})$, $s_2(\boldsymbol{r})$, ..., $s_N(\boldsymbol{r})$ by exp[$i\phi(\boldsymbol{r}, t)$] where $\phi(\boldsymbol{r}, t)$ is the integral of the instantaneous Larmor frequency at position $\boldsymbol{r}$. This factor calculates away the effect of the gradient field before RF iteration begins, reducing the numerical workload. Since gradient field is often much stronger than the RF field, this method also allows taking larger time steps in Bloch simulation. We used the cost function of ref. [4] which includes penalties for amplitude and phase of the final excitation, and total RF power. For an inherently refocused gradient pulse, the cost function does not change by frame transformation.

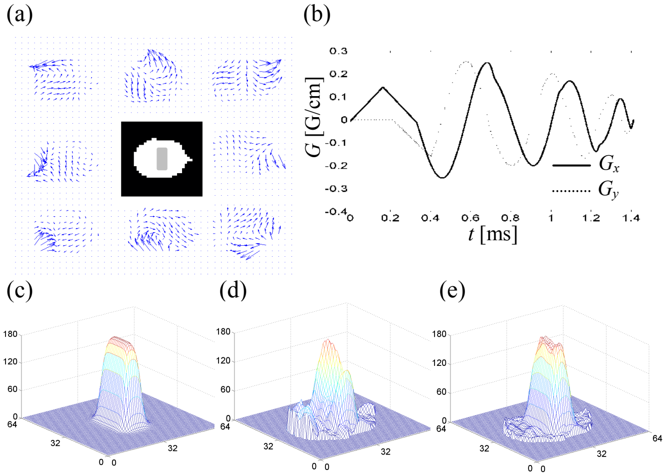

Figure 1. (a) Simulated B_1+ maps for an 8-coil pTx system. (b) Three-turn spiral gradient waveform with a variable slew rate. (c) Target excitation profile. (d) Excitation obtained from non-iterative pTx pulse design. (e) Excitation obtained after iterative correction.

Computationally, the task of repeated Bloch simulations for non-interacting spins is well suited for operation with a GPU, which allocates more memory to arithmetic calculation than CPU. We used a general-purpose graphics card by NVIDIA (Santa Clara, CA) with 512 MB memory for Bloch simulation acceleration within the existing Matlab workflow. In order to assess the calculation speed of the proposed methods, RF pulses were designed for two-dimensional, reduced field-of-view excitation in 8-channel pTx. We compared: CPU calculation of conventional spinor-domain Bloch simulation (method 1), CPU calculation of Bloch simulation in the local rotating frame (method 2), and conventional spinor-domain Bloch simulation performed by a GPU (method 3). All calculation was done with a desktop computer with 3.4 GHz Pentium D CPU and 1 GB RAM.

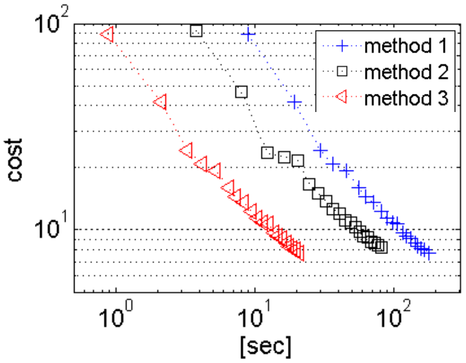

Figure 2. Dynamic behavior of the cost function as RF waveforms are updated iteratively. Three different methods were used to calculate the cost and perform line search in optimal control algorithm.

Results and Discussion: Fig 1(a,b) show the B_1+ maps and the variable-slew-rate gradient [5] waveforms used in this study. Figure 1(c) shows the target profile defined in a 64×64 matrix. Initial RF waveforms were obtained by non-iterative (linear) method, and subsequently corrected by the optimal control algorithm. Figure 1(e) shows the excitation profile after twenty iterative corrections, which greatly improves upon the initial profile Fig. 1(d). All three methods produced the end profile of Fig. 1(e) with negligible difference. Figure 2 shows the decrease of the cost as a function of the calculation time for each of the three methods. GPU provides acceleration of an order of magnitude in reference to the baseline method, and completes optimization in time not much greater than the linear design time (~10 sec) in our example. The analytical frame transform alone gives factor-of-two improvement in speed while running in conventional CPU environment. Judging from the speed gain in an isolated Bloch simulation (factor >3), it is expected that an optimal control algorithm with more refined line-search routine (requiring more cost calculations per optimization direction) will allow greater acceleration for method 2. Combining methods 2 and 3 did not result in proportional speed gain, which indicates that the speed of GPU method is limited more by memory access than arithmetic calculation [3].

Conclusion: We introduced novel analytical and computational methods to accelerate Bloch simulation on uncoupled spins and demonstrated their utility in optimal control-based RF pulse design in parallel transmit. While GPU provides a large gain in speed, limited internal memory of a general purpose card at present limits its application to relatively small matrices. With moderate expansion in hardware and optimization of its memory use, an accelerated Bloch simulator as was explored here could be an important enabler of iterative design as a practical pTx pulse design solution.

References

[1] Xu D et al, MRM 2008;59:547 [2] Grissom WA et al, IEEE Trans. Med. Imaging 2009;28:1548 [3] Lechner SM et al, Proc. ISMRM 2009;2695 [4] Xu D et al, Proc. ISMRM 2008;618 [5] Xu D et al, MRM 2007;58:835