Quantum Finite Element Method with Model Order Reduction

Szabolcs Jóczik

Faculty of Informatics, Eötvös Loránd University, 1117 Budapest, Hungary

December 16, 2020

Abstract

The finite element method is used to approximately solve boundary value problems for differential equations. The method discretises the parameter space and finds an approximate solution by solving a large system of linear equations.

1 Related Works

A quantum algorithm for the finite element method has been introduced by Montanaro and Pallister [1]. Some important notes from their paper:

1. QLE algorithm is indeed applicable to the general FEM, and can achieve substantial speedups over the classical algorithm. However, the quantum speedup obtained is only at most polynomial, if the spatial dimension is fixed and the solution satisfies certain smoothness properties

2. There are two potential sources of error in producing the solution:

   (a) The discretisation process which converts the problem to a system of linear equations.

   (b) Any inaccuracies in solving the system of equations itself and computing the desired function of the solution. (The larger the system of equations produced, the smaller the first type of error is.)

   The QLE algorithm can work with an exponentially larger set of equations in a comparable time to the classical algorithm, so this source of error can be reduced. However, the scaling with accuracy of the QLE algorithm's extraction of a solution from the system of linear equations is substantially worse than the classical algorithm. These two effects can come close to cancelling each other out.

3. The inability of the quantum algorithm to deliver exponential speedups (in some cases) is not a limitation of the algorithm itself, but rather that any quantum algorithm for the FEM will face similar constraints.

(a) Any algorithm which needs to distinguish between two states which are distance ϵ apart must have runtime $\Omega(1/\sqrt{\epsilon})$.

(b) The FEM solving subroutine of any quantum algorithm can likely be replaced with an equivalent classical subroutine with at most a polynomial slowdown (in fixed spatial dimension and when the solution is smooth).

(c) There can be no more than a quadratic speedup if the input to the problem is arbitrary and accessed via queries to a black box or "oracle".

The work [2] and [3] shows that using Euler's method the quantum algorithm can in principle achieve an expotential improvement for approximately computing properties of the solution to the system, if the system of equations is provided implicitly. This approach requires also specifying how the equations are produced and how the property of interest is computed. If the equations are generated by a discretization procedure such as FDM, similar qualitative conclusions to those [1] derived for the FEM seem likely to hold.

2 Summary

The state space representation of a (continuous time-invariant) physical system is given by 2 equations:

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{1}$$
$$y(t) = Cx(t) + Du(t), \tag{2}$$

where

1. $x \in \mathbb{R}^n$ is the state vector
2. $u \in \mathbb{R}^p$ is the input vector
3. $y \in \mathbb{R}^q$ is the output vector
4. $A \in \mathbb{R}^{n \times n}$ is the system matrix
5. $B \in \mathbb{R}^{n \times p}$ is the input matrix
6. $C \in \mathbb{R}^{q \times n}$ is the output matrix
7. $D \in \mathbb{R}^{q \times p}$ is the feedthrough matrix

In a system simulation the challenge is to numerically integrate a very high-dimensional differential-algebraic system of equations problem. Thus the need for model order reduction methods

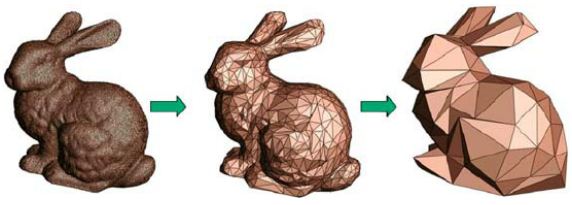

Figure 1: Model Order Reduction illustrated

Projection based MOR methods rely on the approximation of the high dimensional state space vector $x(t) \in \mathbb{R}^n$ by one reduced vector $z(t) \in \mathbb{R}^r$, where $r \ll n$ such that the two systems produce a similar output when excited with the same input. Let's assume that there exists a $V \in \mathbb{R}^{n \times r}$ such that

$$x(t) \approx \widetilde{x}(t) = Vz(t)$$

If we substitute this into Eq.1, we arrive at the following reduced state space representation

$$\begin{aligned} V\dot{z}(t) &= AVz(t) + Bu(t) + r(t) \in \mathbb{R}^n \\ \widetilde{y}(t) &= CVz(t) + Du(t), \end{aligned}$$

Applying the Petrov-Galerkin technique, we demand the residual $(r(t))$ to be orthogonal to some testspace $\mathcal{W}$. Assuming that the columns of a matrix $W \in \mathbb{R}^{n \times r}$ span this testspace, the mathematical formulation of this orthogonality becomes

$$\begin{aligned} W^T V\dot{z}(t) = \dot{z}(t) &= W^T AVz(t) + W^T Bu(t)) + W^T r(t) \\ &= W^T AVz(t) + W^T Bu(t)) \in \mathbb{R}^r \end{aligned}$$

To relate V and W we demand biorthogonality of the spaces V and W spanned by the columns of the two matrices, respectively $W^T V = I_r$. With this, the reduced problem is the projection of the full problem onto $\mathcal{V}$ along $\mathcal{W}$. MOR projection methods are characterised by the way of how to construct the matrices V and W that define the projection.

Going back to state space representation Eq.1 and taking the Laplace transform of the time domain representation of the linear network problem, we obtain the following frequency domain representation:

$$\begin{aligned} sX(s) &= AX(s) + BU(s), \\ Y(s) &= CX(s) + DU(s), \end{aligned}$$

where $X(s), U(s), Y(s)$, are the Laplace transforms of the states, the input and the output, respectively. We assumed zero initial conditions, $x(0) = 0, u(0) = 0$ and $y(0) = 0$. Eliminating the variable $X(s)$ in the frequency domain representation we see that the system's response to the input $U(s)$ in the frequency domain is given by

$$Y(s) = H(s)U(s)$$

with the matrix-valued transfer function

$$H(s) = -C(sI - A)^{-1}B + D \in \mathbb{C}^{q \times p}.$$

2.1 Krylov method

The idea is to construct a reduced order model such that the series expansions of the transfer function H_r of the reduced model and the full problem's transfer function agree up to a certain index of summation. In the following we will restrict to SISO systems, i. e, single-input single-output systems. In this case we have $p = q = 1$,

thus $B = b$, $D = d$ and $C = c^T$ where $b, c \in \mathbb{R}^n$ and $d \in \mathbb{R}$, the (scalar) transfer function becomes:

$$H(s) = -c^T(sI - A)^{-1}b + d \in \mathbb{C}. \tag{3}$$

The goal of model reduction, in this setting, is to produce a much smaller order system $H_r(s)$ with transfer function

$$H_r(s) = -c_r^T(sI_r - A_r)^{-1}b_r + d_r \in \mathbb{C},$$

where $A_r \in \mathbb{R}^{r\times r}$, $b_r, c_r \in \mathbb{R}^r$ and $d \in \mathbb{R}$. Given $H(s)$ as in Eq.3, let the nth-order discrete-time system

$$\hat{H}(z) = -\hat{c}^T(zI - \hat{A})^{-1}\hat{b} + \hat{d}$$

be obtained from H(s) via a bilinear transformation [5], where

$$\begin{aligned} \hat{A} &= (\omega_0 I + A)(\omega_0 I - A)^{-1}, & \hat{b} &= \sqrt{2\omega_0}(\omega_0 I - A)^{-1}b, \\ \hat{c}^T &= \sqrt{2\omega_0}c^T(\omega_0 I - A)^{-1}, & \hat{d} &= d + c^T(\omega_0 I - A)^{-1}b, \end{aligned}$$

where $\omega_0 > 0$. It is well known that(?)

$$H(s) = \hat{H}\left(\frac{\omega_0 + s}{\omega_0 - s}\right) \quad \text{or} \quad H\left(\omega_0\frac{z-1}{z+1}\right) = \hat{H}(z).$$

References

[1] Montanaro, Ashley, and Sam Pallister. "Quantum algorithms and the finite element method." Physical Review A 93.3 (2016): 032324.

[2] Berry, Dominic W. "High-order quantum algorithm for solving linear differential equations." Journal of Physics A: Mathematical and Theoretical 47.10 (2014): 105301.

[3] Leyton, Sarah K., and Tobias J. Osborne. "A quantum algorithm to solve nonlinear differential equations." arXiv preprint arXiv:0812.4423 (2008).

[4] Antoulas, Athanasios C., et al. "Model order reduction: methods, concepts and properties." Coupled multiscale simulation and optimization in nanoelectronics. Springer, Berlin, Heidelberg, 2015. 159-265.

[5] `https://en.wikipedia.org/wiki/Bilinear_transform`

[6] A new frequency weighted Fourier-based method for model order reduction `https://www.sciencedirect.com/science/article/abs/pii/S0005109817305575`