

Adaptive mesh refinement quantum algorithm for Maxwell's equations

Elise Fressart^{1,2}, Michel Nowak¹, and Nicole Spillane²

¹cortAIX Labs, Thales Research and Technology, 91120 Palaiseau, France

²CMAP, CNRS, École polytechnique, Institut Polytechnique de Paris, 91120 Palaiseau, France.

Abstract—Algorithms that promise to leverage resources of quantum computers efficiently to accelerate the finite element method have emerged. However, the finite element method is usually incorporated into a high-level numerical scheme which allows the adaptive refinement of the mesh on which the solution is approximated. In this work, we propose to extend adaptive mesh refinement to the quantum formalism, and apply our method to the resolution of Maxwell's equations. An important step in this procedure is the computation of error estimators, which guide the refinement. By using block-encoding, we propose a way to compute these estimators with quantum circuits. We present first numerical experiments on a 2D geometry.

Index Terms—Quantum computing, Finite element method, Adaptive mesh refinement, Maxwell's equations, Block-encoding

I. INTRODUCTION

Partial differential equations (PDEs) appear in various scientific fields such as fluid dynamics, mechanics, electromagnetism. The finite element method (FEM) [1], [2] is a conventional approach to numerically solve PDEs. After discretizing the domain, the FEM builds a large linear system whose unknowns represent the degrees of freedom of the solution. Solving such systems can be very resource demanding on a classical computer even if they are sparse.

The Harrow-Hassidim-Lloyd (HHL) algorithm [3] is the first quantum algorithm to solve the quantum linear system problem (QLSP). Given a quantum state $|b\rangle$ and a matrix A , the algorithm outputs, up to some precision, a quantum state $|x\rangle$ proportional to the solution x of the linear system $Ax = b$. Under suitable assumptions, the total runtime of HHL depends logarithmically on the size N of the problem. This is an exponential speedup compared to a classical iterative solver like the conjugate gradient method [4]. More recently, variable-time amplitude amplification [5], linear combination of unitaries (LCU) [6], and block-encoding associated with quantum singular value transformation (QSVT) [7] propose to improve the complexity. Recent research has focused on algorithms inspired by, or based on, adiabatic quantum computing [8]–[11]. An optimal complexity, with respect to the precision ϵ and the condition number κ , of the matrix A is reached in [11]. The integration of a QLSP solver into the FEM has been proposed by [12] where a second order elliptic PDE is

used to illustrate the new numerical scheme. An application to electromagnetic simulations is proposed in [13].

Near-term variational algorithms to solve the QLSP have also received growing attention. The work in [14] applies the variational quantum linear solver [15] to the FEM.

However, the integration of a QLSP solver into the FEM might be lacking one important consideration: the mesh on which the solution is discretized is often chosen arbitrarily. In order to design a mesh that captures all the variations of the solution properly, we rely on classical adaptive schemes [16]. They allow to reduce the computational cost by optimizing the placement of the degrees of freedom in order to decrease the error. The adaptive schemes refine iteratively the cells of the mesh where the error is large. This can be done by applying a posteriori error estimation [17], [18], which aims at approximating the error distribution from the lastly computed solution.

In this work, we focus on Maxwell's propagation equations. We suppose that we have access to a quantum linear system solver that efficiently outputs the degrees of freedom of the FEM in a quantum register. From there, we reformulate the local error estimators and propose quantum circuits to compute them directly. We then iterate on the refined meshes and show that global error is indeed reduced compared to a uniform refinement strategy.

The rest of the article is organized as follows. Section II presents the classical adaptive mesh refinement algorithm relying on residual-based a posteriori error estimation for Maxwell's equation. In Section III, we propose a hybrid algorithm for adaptive mesh refinement. Section IV focuses on numerical simulations. In Section V, we summarize our findings and offer up some perspectives for future work.

II. CLASSICAL APPROACH

A. Maxwell's equations

We start by recalling Maxwell's equations in a linear, non-dispersive, isotropic medium with time-independent permittivity ϵ , permeability μ and conductivity σ [19]. These equations introduce the electric field $\mathbf{E}(\mathbf{x}, t)$ and the magnetic field $\mathbf{B}(\mathbf{x}, t)$ for all $\mathbf{x} \in \mathbb{R}^3$ and $t \in \mathbb{R}$. The first equation relates the divergence of the electric field \mathbf{E} with the charge density ρ by

$$\nabla \cdot (\epsilon \mathbf{E}) = \rho. \quad (1)$$

elise.fressart@thalesgroup.com
michel.nowak@thalesgroup.com
nicole.spillane@cmap.polytechnique.fr

The second equation states that the divergence of the magnetic field is null

$$\nabla \cdot \mathbf{B} = 0. \quad (2)$$

The third equation relates the curl of the electric field to the time derivative of the magnetic field as follows

$$\epsilon \frac{\partial \mathbf{B}}{\partial t} + \nabla \times \mathbf{E} = 0. \quad (3)$$

Finally, the last equation is between the curl of the magnetic field, the current density \mathbf{J} , which under Ohm's law with applied current density \mathbf{J}_e equals $\mathbf{J}_e + \sigma \mathbf{E}$, and the time derivative of the electric field

$$\epsilon \frac{\partial \mathbf{E}}{\partial t} - \nabla \times (\mu^{-1} \mathbf{B}) = -\mathbf{J}_e - \sigma \mathbf{E}. \quad (4)$$

Taking the time derivative of (4) and switching the order of the spatial and time derivatives leads to

$$\epsilon \frac{\partial^2 \mathbf{E}}{\partial^2 t} - \nabla \times \left(\mu^{-1} \frac{\partial \mathbf{B}}{\partial t} \right) = -\frac{\partial \mathbf{J}_e}{\partial t} - \sigma \frac{\partial \mathbf{E}}{\partial t}. \quad (5)$$

Using (3), we obtain:

$$\epsilon \frac{\partial^2 \mathbf{E}}{\partial^2 t} + \sigma \frac{\partial \mathbf{E}}{\partial t} + \nabla \times (\mu^{-1} \nabla \times \mathbf{E}) = -\frac{\partial \mathbf{J}_e}{\partial t}. \quad (6)$$

In the time-harmonic setting, the time dependence is periodic with pulsation ω so the fields, the charge density and the current density can be rewritten as $f(x, t) = f(x) \exp(-i\omega t)$ (with generic f denoting any of these variables). A time derivative multiplies by $-i\omega$. By replacing \mathbf{E} with $\epsilon_0^{1/2} \mathbf{E}$ and multiplying by $\epsilon_0^{1/2} \mu_0$, we obtain the final propagation equation:

$$\nabla \times (\mu_r^{-1} \nabla \times \mathbf{E}) - k^2 \epsilon_r \mathbf{E} = ik \mu_0^{1/2} \mathbf{J}_e \quad (7)$$

where $k = \omega \sqrt{\epsilon_0 \mu_0}$ is the wave number, $\epsilon_r = \frac{1}{\epsilon_0} (\epsilon + \frac{i\sigma}{\omega})$ and $\mu_r = \frac{\mu}{\mu_0}$ are normalized by ϵ_0 and μ_0 (respectively the permittivity and permeability of free space).

B. Propagation in a perfect conductor cavity

In a bounded polyhedral domain $\Omega \subset \mathbb{R}^d$ ($d = 2, 3$) with boundary Γ and outward unit normal ν , we seek the electric field \mathbf{E} satisfying the time-harmonic Maxwell's equation and subject to the perfect electric conductor boundary condition:

$$\nabla \times (\mu_r^{-1} \nabla \times \mathbf{E}) - k^2 \epsilon_r \mathbf{E} = \mathbf{F} \quad \text{in } \Omega \quad (8)$$

$$\nu \times \mathbf{E} = 0 \quad \text{on } \Gamma \quad (9)$$

where \mathbf{F} is the source term, k the wave number and ϵ_r and μ_r are the relative permittivity and the relative permeability respectively. We assume that $\nabla \cdot \mathbf{F} = 0$. A possible domain is represented in Fig 1. The solution space of the problem is

$$\mathbf{H}_0(\text{curl}, \Omega) = \left\{ \mathbf{u} \in (L^2(\Omega))^d, \nabla \times \mathbf{u} \in (L^2(\Omega))^d \mid \mathbf{u} \times \nu = 0 \text{ on } \partial\Omega \right\}$$

equipped with the norm

$$\|\mathbf{v}\|_{\mathbf{H}(\text{curl}, \Omega)} = \left(\|\mathbf{v}\|_{(L^2(\Omega))^d}^2 + \|\nabla \times \mathbf{v}\|_{(L^2(\Omega))^d}^2 \right)^{1/2},$$

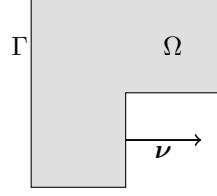


Fig. 1: Example of a two-dimensional domain Ω with boundary Γ and outer unit normal vector ν .

where in turn $\|\mathbf{v}\|_{(L^2(\Omega))^d} = \left(\sum_{i=1}^d \int_{\Omega} |\mathbf{v}_i|^2 \right)^{1/2}$.

The variational formulation is obtained by multiplying (8) by a test function in $\mathbf{H}_0(\text{curl}, \Omega)$ and using integration by parts. The boundary term vanishes because of the boundary condition satisfied by the test function. The variational formulation consists in finding $\mathbf{E} \in \mathbf{H}_0(\text{curl}, \Omega)$ such that

$$\int_{\Omega} \mu_r^{-1} \nabla \times \mathbf{E} \cdot \nabla \times \bar{\mathbf{v}} - \int_{\Omega} k^2 \epsilon_r \mathbf{E} \cdot \bar{\mathbf{v}} = \int_{\Omega} \mathbf{F} \cdot \bar{\mathbf{v}}, \quad (10)$$

for all $\mathbf{v} \in \mathbf{H}_0(\text{curl}, \Omega)$. We assume that k is not a resonant wave number so that (10) has a unique solution for any $\mathbf{F} \in (L^2(\Omega))^d$ [19, Corollary 4.19]. Let \mathbf{E} denote this solution.

C. The Finite Element Method

The Finite Element Method (FEM) [19], [20] is a standard approach to approximate the solution to (10). It consists in the following steps.

1) *Meshing the domain:* The domain is discretized with a set of elements (e.g triangles, tetrahedra) forming a mesh \mathcal{T}_h as in Fig.2.

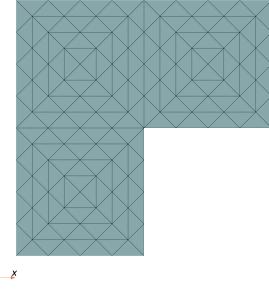


Fig. 2: Illustration of a meshed domain

2) *Choosing the basis functions:* The FEM restricts the weak formulation (10) to a finite-dimensional space denoted V_h leading to the discrete variational problem : find $\mathbf{E}_h \in V_h$ such that

$$\int_{\Omega} \mu_r^{-1} \nabla \times \mathbf{E}_h \cdot \nabla \times \bar{\mathbf{v}}_h - \int_{\Omega} k^2 \epsilon_r \mathbf{E}_h \cdot \bar{\mathbf{v}}_h = \int_{\Omega} \mathbf{F} \cdot \bar{\mathbf{v}}_h, \quad (11)$$

for all $\mathbf{v} \in V_h$. A popular choice of basis functions for electromagnetic simulations are edge basis functions [21], denoted $\{\phi_i, 0 \leq i < N\}$. Edge elements, also called Nedelec

elements, are $\mathbf{H}(\text{curl}, \Omega)$ conforming meaning that the finite element space V_h is a subspace of $\mathbf{H}(\text{curl}, \Omega)$. Fig 3 represents the degrees of freedom for edge elements of the lowest order.

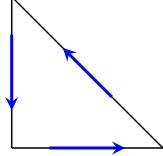


Fig. 3: Degrees of freedom for Nédélec finite elements of order 1. An arrow collinear to an edge indicates that the degree of freedom is the value of the tangential component integrated over this edge.

3) *Assembling the linear system:* By choosing basis functions as test functions and expressing \mathbf{E}_h in the finite element basis in (11), we end up with a linear system:

$$A\mathbf{\underline{E}}_h = \mathbf{\underline{F}}, \quad (12)$$

where $A \in \mathbb{C}^{N \times N}$ and $\mathbf{\underline{F}} \in \mathbb{C}^N$ are defined by

$$\begin{aligned} A &= \left(\int_{\Omega} \mu_r^{-1} \nabla \times \phi_i \cdot \nabla \times \phi_j - \int_{\Omega} k^2 \epsilon_r \phi_i \cdot \phi_j \right)_{1 \leq i, j \leq N}, \\ \mathbf{\underline{F}} &= \left(\int_{\Omega} \mathbf{F} \cdot \phi_i \right)_{1 \leq i \leq N}, \end{aligned}$$

and $\mathbf{\underline{E}}_h = (E_{h1}, \dots, E_{hN})^T \in \mathbb{C}^N$ contains the degrees of freedom of the discrete finite element solution, *i.e.*, $\mathbf{E}_h = \sum_{i=1}^N E_{hi} \phi_i$.

4) *Solving the linear system:* With classical computing resources, the linear system (12) can be solved using iterative solvers like GMRES [4]. Indeed, the matrix A is indefinite and possibly non-Hermitian (in the case where $\epsilon_r \notin \mathbb{R}$).

D. Adaptive mesh refinement

In the previous section, we covered how Maxwell's equations can be rewritten until finding a solution on a discretized mesh with the FEM. However, when little is known about the system at hand, the mesh is usually initialized arbitrarily. In order to have a good convergence of the solution, we need to refine the cells of the mesh specifically where it will improve the precision of the numerical scheme. In order to do so, we rely on adaptive mesh refinement strategies. More precisely, we focus on a refining strategy that uses a posteriori error estimates. We then select the cells to be refined according to a refining criterion and iterate on the mesh.

1) *A posteriori error estimation:* Let's start by looking at how we can estimate locally (in each cell) the error made compared to the exact unknown solution. The difficulty is of course that the exact solution is unknown. For problem (10), [22] proposes the following residual-based error estimator η that can be decomposed into contributions from each element K . Each local term is itself decomposed into 4 terms:

$$\eta = \left\{ \sum_{K \in \mathcal{T}_h} \eta_{1K}^2 + \eta_{2K}^2 + \eta_{3K}^2 + \eta_{4K}^2 \right\}^{1/2}. \quad (13)$$

In order to define η , the following notation is introduced. Let h_K denote the diameter of an element $K \in \mathcal{T}_h$ with outer unit normal ν_K . Its boundary denoted ∂K is composed of faces f (or edges if $d = 2$) with diameter h_f . For a face f located in the interior of Ω shared by two elements K_1 and K_2 (see Fig. 4), the jump operators are defined by

$$[\mathbf{v}]_T = \mathbf{v}|_{K_1} \times \nu_{K_1} + \mathbf{v}|_{K_2} \times \nu_{K_2} \quad (14)$$

$$[\mathbf{v}]_N = \mathbf{v}|_{K_1} \cdot \nu_{K_1} + \mathbf{v}|_{K_2} \cdot \nu_{K_2} \quad (15)$$

where the subscript $|_{K_i}$ for $i = 1, 2$ denotes the restriction to element K_i .

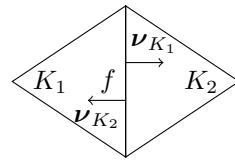


Fig. 4: An interior face f is shared by K_1 and K_2

The first term η_{1K}^2 can be understood as how well the propagation equation is solved:

$$\eta_{1K}^2 := h_K^2 \|\mathbf{F} - \nabla \times \mu_r^{-1} \nabla \times \mathbf{E}_h + k^2 \epsilon_r \mathbf{E}_h\|_{(L^2(K))^d}^2. \quad (16)$$

The second term η_{2K}^2 can be understood as how well the divergence of the propagation equation is solved:

$$\eta_{2K}^2 := h_K^2 \|k^2 \nabla \cdot (\epsilon_r \mathbf{E}_h)\|_{L^2(K)}^2. \quad (17)$$

The third term η_{3K}^2 accounts for the tangential discontinuity of the curl of the field on each interface between the cells:

$$\eta_{3K}^2 := \frac{1}{2} \sum_{f \in \partial K \setminus \Gamma} h_f \| [\mu_r^{-1} \nabla \times \mathbf{E}_h]_T \|_{(L^2(f))^d}^2. \quad (18)$$

The fourth term η_{4K}^2 takes into account the normal discontinuity on each interface between the cells:

$$\eta_{4K}^2 := \frac{1}{2} \sum_{f \in \partial K \setminus \Gamma} h_f k^2 \| [\epsilon_r \mathbf{E}_h]_N \|_{(L^2(f))^d}^2. \quad (19)$$

The factor $\frac{1}{2}$ in η_{3K} and η_{4K} is to split the contribution of an interior face between the two elements sharing this face. This allows to define a local error estimator η_K as the sum of the four contributions $\eta_K^2 = \eta_{1K}^2 + \eta_{2K}^2 + \eta_{3K}^2 + \eta_{4K}^2$.

2) *Justification for the use of a posteriori estimates of the error:* The element estimators η_K do not depend on the unknown exact solution \mathbf{E} for their definition. They still estimate the error $\mathbf{e} = \mathbf{E} - \mathbf{E}_h$. Firstly, \mathbf{e} is bounded from above with respect to η . This property is called reliability.

Theorem 2.1 (Thm 3.3 in [22]): There exists a constant C independent of \mathcal{T}_h such that

$$\|\mathbf{e}\|_{\mathbf{H}(\text{curl}, \Omega)}^2 \leq C \eta^2. \quad (20)$$

Secondly, the error is also bounded from below with respect to the a posteriori estimate.

Theorem 2.2 (Thm 3.4 in [22]): There exists a constant C independent of \mathcal{T}_h such that

$$\eta^2 \leq \|\mathbf{e}\|_{\mathbf{H}(\text{curl}, \Omega)}^2 + \sum_{K \in \mathcal{T}_h} h_K^2 \|\mathbf{F} - \mathcal{P}_K \mathbf{F}\|_{(L^2(K))^d}^2, \quad (21)$$

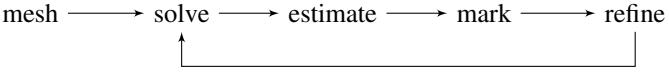
where \mathcal{P}_K is the L^2 projection on $\prod_{K \in \mathcal{T}_h} \mathbb{P}_k^d(K)$ with $\mathbb{P}_k^d(K)$ denoting polynomials of order k .

The second part in the right hand side cancels when \mathbf{F} is a polynomial of order k . Since the element estimators are local, they can be used for adaptive local refinement to mark elements that should be refined. The evaluation cost on a classical computer is cheap thanks to this locality.

3) *Selection criteria:* After meshing the domain, solving the linear system and computing the error estimates, elements to be refined are marked. One possible marking strategy, the Maximum Strategy [16, Section 7], consists in selecting elements in the set

$$\left\{ K \in \mathcal{T}_h \mid \eta_K \geq \theta \max_{K' \in \mathcal{T}_h} \eta_{K'} \right\} \quad (22)$$

where $\theta \in [0, 1]$ is an input parameter. The marked elements, i.e., the ones with the largest error estimators, are refined. All of these steps are iterated until a criterion is reached. The procedure is illustrated in the following diagram.



III. ADAPTIVE MESH REFINEMENT WITH QUANTUM COMPUTERS

Now that we have derived the classical strategy to adaptively refine meshes for the FEM, we turn towards the proposal of a numerical scheme to perform the same operations with quantum computers. We first propose a reformulation of the error estimators so that they can be sampled with quantum circuits. We then discuss how to estimate the norm of the solution depending on the quantum linear system solver.

A. Reformulation of the error estimator

The error estimators can be reformulated as linear combinations of expectation values. For the sake of simplicity, we assume that $\epsilon_r \kappa^2 = 1$ and $\mu_r = 1$. For η_{1K}^2 , we introduce the matrices M_K , S_K and C_K defined by

$$\begin{aligned} M_K &= \left(h_K^2 \int_K \phi_i \cdot \phi_j \right)_{1 \leq i, j \leq N}, \\ S_K &= \left(h_K^2 \int_K \phi_i \cdot \nabla \times \nabla \times \phi_j \right)_{1 \leq i, j \leq N}, \\ C_K &= \left(h_K^2 \int_K \nabla \times \nabla \times \phi_i \cdot \nabla \times \nabla \times \phi_j \right)_{1 \leq i, j \leq N}. \end{aligned}$$

By expanding the norm and using the previous matrices, we get

$$\begin{aligned} \eta_{1K}^2 &= h_K^2 \left[\int_K \mathbf{F} \cdot \bar{\mathbf{F}} + \int_K \nabla \times \nabla \times \mathbf{E}_h \cdot \nabla \times \nabla \times \bar{\mathbf{E}}_h \right. \\ &\quad + \int_K \mathbf{E}_h \cdot \bar{\mathbf{E}}_h - 2 \operatorname{Re} \left(\int_K \mathbf{F} \cdot \nabla \times \nabla \times \bar{\mathbf{E}}_h \right) \\ &\quad \left. + 2 \operatorname{Re} \left(\int_K \mathbf{F} \cdot \bar{\mathbf{E}}_h \right) - 2 \operatorname{Re} \left(\int_K \nabla \times \nabla \times \mathbf{E}_h \cdot \bar{\mathbf{E}}_h \right) \right] \\ &= \underline{E}^\dagger M_K \underline{E} + \underline{E}_h^\dagger C_K \underline{E}_h + \underline{E}_h^\dagger M_K \underline{E}_h \\ &\quad - 2 \operatorname{Re}(\underline{E}^\dagger S_K \underline{E}_h) + 2 \operatorname{Re}(\underline{E}^\dagger M_K \underline{E}_h) \\ &\quad - 2 \operatorname{Re}(\underline{E}_h^\dagger S_K \underline{E}_h). \end{aligned}$$

Similarly, η_{2K} can be formulated as

$$\eta_{2K}^2 = \underline{E}_h^\dagger D_K \underline{E}_h,$$

where

$$D_K = \left(h_K^2 \int_K \nabla \cdot \phi_i \nabla \cdot \phi_j \right)_{1 \leq i, j \leq N}.$$

For η_{3K} , contributions of the neighboring elements appear because of the jump. By definition of the jump, it holds that

$$\eta_{3K}^2 = \frac{1}{2} \sum_{\substack{f \in \partial K \setminus \Gamma \\ K \cap K' = f}} h_f \| (\nabla \times \mathbf{E}_h \times \boldsymbol{\nu})|_K + (\nabla \times \mathbf{E}_h \times \boldsymbol{\nu})|_{K'} \|_{(L^2(f))^d}^2.$$

For two elements K_1 and K_2 sharing a face f , let $C_{K_1 K_2 f}$ be the matrix defined by

$$C_{K_1 K_2 f} = \left(h_f \int_f (\nabla \times \phi_i \times \boldsymbol{\nu})|_{K_1} \cdot (\nabla \times \phi_j \times \boldsymbol{\nu})|_{K_2} \right)_{1 \leq i, j \leq N}.$$

By expanding the norm, η_{3K} can be rewritten as

$$\begin{aligned} \eta_{3K}^2 &= \frac{1}{2} \sum_{\substack{f \in \partial K \setminus \Gamma \\ K \cap K' = f}} \left[\underline{E}_h^\dagger C_{KKf} \underline{E}_h + \underline{E}_h^\dagger C_{K'K'f} \underline{E}_h \right. \\ &\quad \left. + 2 \operatorname{Re}(\underline{E}_h^\dagger C_{KK'f} \underline{E}_h) \right]. \end{aligned}$$

The process is similar for η_{4K} leading to

$$\begin{aligned} \eta_{4K}^2 &= \frac{1}{2} \sum_{\substack{f \in \partial K \setminus \Gamma \\ K \cap K' = f}} \left[\underline{E}_h^\dagger M_{KKf} \underline{E}_h + \underline{E}_h^\dagger M_{K'K'f} \underline{E}_h \right. \\ &\quad \left. + 2 \operatorname{Re}(\underline{E}_h^\dagger M_{KK'f} \underline{E}_h) \right], \end{aligned}$$

where

$$M_{K_1 K_2 f} = \left(h_f \int_f (\phi_i \cdot \boldsymbol{\nu})|_{K_1} (\phi_j \cdot \boldsymbol{\nu})|_{K_2} \right)_{1 \leq i, j \leq N}.$$

Combining these four quantities, one element estimator η_K contains 31 terms for $d = 3$ and 25 terms for $d = 2$.

B. Quantum circuits for error estimation

The terms presented in section III-A can be separated into three categories $\underline{E}_h M \underline{E}_h$ (bilinear), $\underline{F}^\dagger M \underline{F}$ (constant) and $\underline{E}_h^\dagger M F$ (linear). Several points need particular focus to be able to estimate these terms. Quantum computing relies on normalized states and unitary operators. However the vectors \underline{E}_h and \underline{F} do not have euclidian norm equal to one and the matrices are not unitary.

We rely on block-encoding, a technique that allows to embed a subnormalized matrix as the top-left block of a unitary [7].

Definition 3.1 (Block-encoding [7], [23]): Given an n -qubit matrix M , an (α, m, ϵ) -block-encoding of M is an $(m+n)$ -qubit unitary matrix U_M such that

$$\|M - \alpha(\langle 0^m | \otimes I_n) U_M (\langle 0^m | \otimes I_n)\| \leq \epsilon.$$

The parameter α is the subnormalization factor, m is the number of ancilla qubits in the block-encoding and ϵ is the precision. Supposing access to block-encodings of the matrices M_K , S_K , C_K , D_K , $C_{KK'f}$ and $M_{KK'f}$, Hadamard tests can be used to estimate the different expectation values [24]. We suppose access to a quantum linear system solver that outputs $|E_h\rangle$ and to a unitary U_F preparing $|F\rangle$. Given U_M a (α, m, ϵ) -block-encoding of M , the circuit from Fig. 5 allows to estimate $\frac{1}{\alpha} \text{Re}(\langle E_h | M | E_h \rangle)$ or $\frac{1}{\alpha} \text{Re}(\langle F | M | F \rangle)$. The circuit from Fig. 6 can be used to estimate $\frac{1}{\alpha} \text{Re}(\langle E_h | M | F \rangle)$. More precisely, if U_M is a $(\alpha, m, 0)$ -block-encoding, the probabilities of observing 0 when measuring the ancilla qubit are $\frac{1}{2}(1 + \frac{1}{\alpha} \text{Re}(\langle \psi | M | \psi \rangle))$ and $\frac{1}{2}(1 + \frac{1}{\alpha} \text{Re}(\langle \phi | M | \psi \rangle))$.

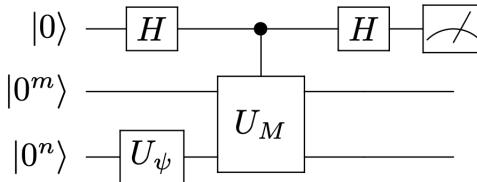


Fig. 5: Circuit for a Hadamard test to estimate $\frac{1}{\alpha}(\text{Re}(\langle \psi | M | \psi \rangle))$. The unitary U_ψ is defined by $U_\psi |0\rangle = |\psi\rangle$.

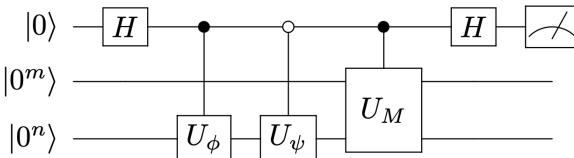


Fig. 6: Circuit for a Hadamard test to estimate $\frac{1}{\alpha} \text{Re}(\langle \phi | M | \psi \rangle)$. The unitaries U_ϕ and U_ψ prepare respectively the states $|\phi\rangle$ and $|\psi\rangle$.

To approximately compute the error estimators using the results from the Hadamard tests, the euclidian norms of \underline{F} and \underline{E}_h as well as the subnormalization factors of the different block-encodings are required. We can assume access to $\|F\|$

since \mathbf{F} is an input of the problem. The norm of \underline{E}_h can be approximately computed. The procedure, which depends on the quantum solver, is detailed in the next section. The values of α depend on the norms of the block-encoded matrices and on the particular construction of the block-encodings.

C. Quantum linear system solver

Both non-variational and variational quantum linear system solvers can be used to approximate $|E_h\rangle$. But the procedure to estimate the norm depends on the solver.

1) *VQLS:* The variational quantum linear solver (VQLS) [15] is a hybrid algorithm. The inputs of VQLS are a circuit U_F preparing $|F\rangle$ and the matrix A given as a linear combination of unitaries. A quantum parametrized circuit, called ansatz, is used to approximate the solution $|E_h\rangle$. We choose the local cost function C_L defined by

$$C_L = \frac{\langle E_h | H_L | E_h \rangle}{\|A | E_h \rangle\|}, \quad (23)$$

associated with the local Hamiltonian

$$H_L = A^\dagger U_F \left(\mathbb{1} - \frac{1}{n} \sum_{j=1}^n |0_j\rangle \langle 0_j| \otimes \mathbb{1}_{\bar{j}} \right) U_F^\dagger A, \quad (24)$$

where $|0_j\rangle$ is the zero state on qubit j and $\mathbb{1}_{\bar{j}}$ is the identity applied to all qubits but j . The parameters are optimized using a classical computer. The euclidian norm of the solution $\|\underline{E}_h\|$ satisfies [25]

$$\|\underline{E}_h\| = \frac{\|F\|}{\langle F | A | E_h \rangle}, \quad (25)$$

in which the quantity $\langle F | A | E_h \rangle$ can be computed in a way similar to the cost functions in [15].

2) *Non-variational quantum linear system solver:* The work [26, Thm 33] proposes an algorithm that outputs a state ϵ -close to the solution $A^{-1}|F\rangle / \|A^{-1}|F\rangle\|$ with a query complexity to a (α, a, δ) -block-encoding of A of $\mathcal{O}(\kappa \text{polylog}(\kappa/\epsilon))$ (with κ the condition number of A). It is also possible to estimate the norm of the solution $\|A^{-1}|F\rangle\|$ with a complexity $\mathcal{O}(\kappa \epsilon^{-1} \text{polylog}(\kappa/\epsilon))$ using variable-time amplitude estimation.

D. Hybrid adaptive mesh refinement loop

A high-level overview of the hybrid adaptive mesh refinement is shown in Fig 7. First, the step “mesh” consists in creating the initial mesh on a classical computer. In “solve”, the linear system (12) is solved using a quantum linear system solver. The quantum step “estimate” outputs classical local error estimators $\tilde{\eta}_K$ approximately computed following the quantum procedure presented in section III-B. These are used to mark elements using the criterion (22) in “mark”. “Refine” splits the tagged elements on a classical computer.

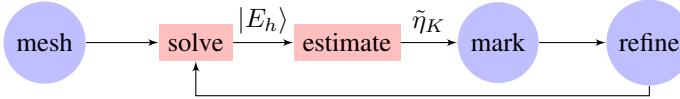


Fig. 7: Hybrid adaptive mesh refinement algorithm. Blue circles symbolise steps performed on a classical computer. Red rectangles refer to steps performed on a quantum computer.

E. Query complexity to a QLSP solver

For a mesh made up of N_{elements} elements, there are N_{elements} local error estimators to approximate. For each one of them, to reach precision ϵ on the squared local estimator, i.e. $|\eta_K^2 - \tilde{\eta}_K^2| \leq \epsilon$, one needs a precision $\frac{\epsilon}{\max \alpha \|\underline{E}_h\|^2}$ on the results of the Hadamard tests. As a consequence $\mathcal{O}\left(\frac{(\max \alpha)^2 \|\underline{E}_h\|^4}{\epsilon^2}\right)$ samples are required. The query complexity to the quantum linear system solver is $\mathcal{O}\left(\frac{N_{\text{elements}} (\max \alpha)^2 \|\underline{E}_h\|^4}{\epsilon^2}\right)$. For edge elements of lowest order, $\mathcal{O}(N_{\text{elements}}) = \mathcal{O}(N)$ (with N the size of the problem) and the query complexity is $\mathcal{O}\left(\frac{N (\max \alpha)^2 \|\underline{E}_h\|^4}{\epsilon^2}\right)$.

IV. NUMERICAL SIMULATIONS

We now turn to numerical experiments that support our theoretical findings. In order to be able to run experiments with reduced emulation resources, we choose to test VQLS as the linear system solver.

A. Software

The Python library DOLFINx [27] is used for the FEM simulations. The procedure detailed in III-B is implemented using the Qiskit framework [28]. The different block-encodings are obtained using the FABLE framework [29]. VQLS simulations are ran using vqls-prototype [30].

B. Error estimators

Simulations are performed in a two-dimensional L-shaped domain $\Omega = [-1, 1]^2 \setminus [0, 1] \times [-1, 0]$. We choose the right hand side and the boundary condition such that $\mathbf{E} = \nabla(r^{2/3} \sin(2\phi/3))$ (in polar coordinates) is the exact solution. This solution, which is depicted in Fig 8, diverges at the reentrant corner where $r = 0$.

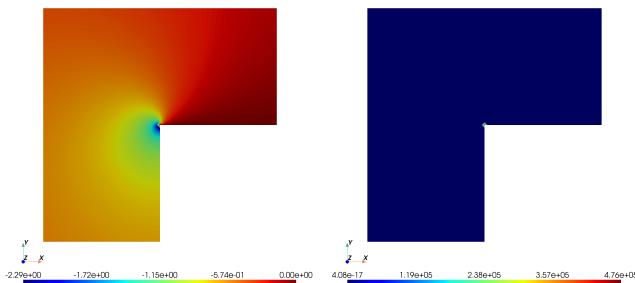


Fig. 8: Components along the x-axis (left) and y-axis (right) of the exact solution. The exact solution is interpolated on the finest uniform mesh using order one Lagrange finite elements.

The discretization uses edge elements of lowest order. We choose $\theta = 0.6$ for the selection criterion (22) and the algorithm stops when ten adaptive iterations have been performed. There are 22 degrees of freedom on the input mesh meaning that $n = 5$ qubits encode the solution. Because $2^5 = 32$, 22 amplitudes encode the degrees of freedom, the rest of the amplitudes are padded with zeros.

The local error estimators η_K are good approximations for the usually unknown local $\mathbf{H}(\text{curl})$ errors

$$\|\mathbf{e}\|_{\mathbf{H}(\text{curl}, K)} = \left(\|\mathbf{e}\|_{(L^2(K))^d}^2 + \|\nabla \times \mathbf{e}\|_{(L^2(K))^d}^2 \right)^{1/2} \quad (26)$$

with $\mathbf{e} = \mathbf{E} - \mathbf{E}_h$ as can be seen in Fig. 9. Both these quantities are large near the re-entrant corner where the solution is singular. This explains the concentration of elements in this region in the final mesh compared to the initial mesh (see Fig. 10).

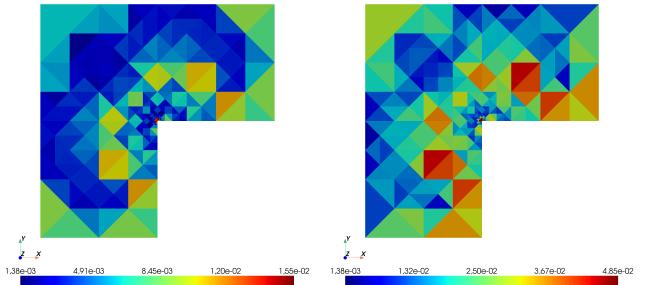


Fig. 9: Local $\mathbf{H}(\text{curl})$ error (left) and local error estimator η_K (right) at iteration 5 in the classical refinement loop.

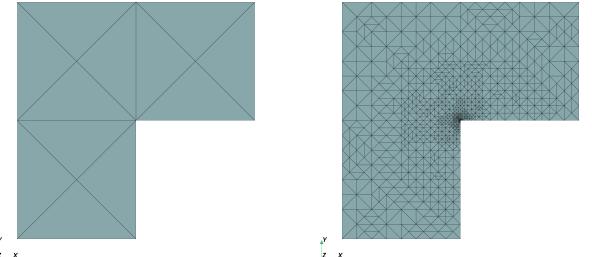


Fig. 10: Initial mesh (left) and mesh at the final iteration (right).

Fig. 11 represents the $\mathbf{H}(\text{curl})$ error and the a posteriori error estimator η as a function of the number of degrees of freedom. The classical convergence rate is improved using adaptive mesh refinement rather than uniform mesh refinement (i.e. all the elements are refined). So this example illustrates the importance of classical adaptive mesh refinement. We are able to compute the local estimators with the quantum circuit for three iterations by encoding the classical finite element solution in a quantum state at each iteration. Without

sampling error, the quantum local estimators are equal to the classical ones. This validates the correctness of the circuits. In preliminary runs taking into account the sampling error, this error dominates the estimators to the point that the result is not a valid approximation of the error.

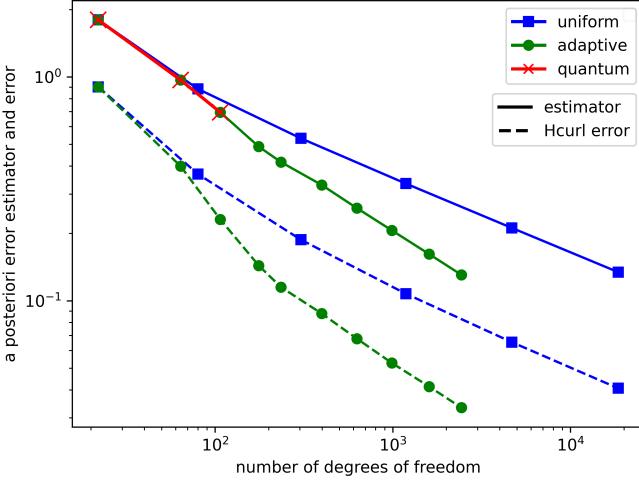


Fig. 11: $\mathbf{H}(\text{curl}, \Omega)$ error (dashed lines) and a posteriori error estimator η (solid lines) as a function of the number of degrees of freedom. The blue squares correspond to classical uniform mesh refinement. The green dots correspond to classical adaptive mesh refinement. The red crosses are obtained from the quantum circuits presented in section III-B. Without sampling error, the classical and quantum error estimators are overlaid.

C. Experiments with VQLS

In order to implement the entire workflow we need to solve the quantum linear system problem focusing on VQLS. Since the solution is real-valued, we are interested in an ansatz that produces a quantum state with real amplitudes. The two studied fixed-structure layered hardware-efficient ansatzes are represented in Fig. 12.

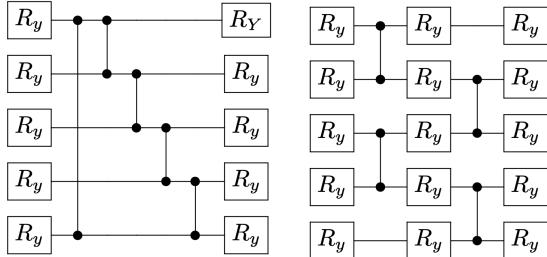


Fig. 12: Linear circular R_y -CZ ansatz [31] (left) and linear alternating R_y -CZ ansatz [15] (right) represented for $n = 5$ qubits and one layer.

We use the constrained optimization by linear approximation (COBYLA) algorithm [32] to optimize the parameters. We denote by $|E_h^{\text{VQLS}}\rangle$ the VQLS output and define

$$|E_h^{\text{FEM}}\rangle = \sum_{i=1}^N \frac{|E_{hi}\rangle}{\|E_h\|} |i\rangle \quad (27)$$

in order to compare both quantities. We investigate the evolution of the fidelity $|\langle E_h^{\text{VQLS}} | E_h^{\text{FEM}} \rangle|^2$ with respect to the number of qubits. To vary the problem size, we extract submatrices and subvectors from the initial 5-qubit linear system. The number of layers in the ansatzes and the maximum number of iterations for the optimizer vary with the number of qubits. Table I details the parameters used for the simulations.

TABLE I: Parameters for the ansatz and the optimizer used in VQLS simulations.

| | number of qubits n | 2 | 3 | 4 | 5 |
|------------------------------|----------------------|------|------|-------|---|
| number of layers | 1 | 2 | 4 | 8 | |
| maximum number of iterations | 1000 | 2000 | 5000 | 10000 | |

The results, without sampling error, are shown in Fig. 13. The fidelity is computed using five simulations. The fidelity is high for two qubits. For both ansatzes, the fidelity decreases when the number of qubits increases and reaches very low values, which shows that VQLS does not perform well on our problem.

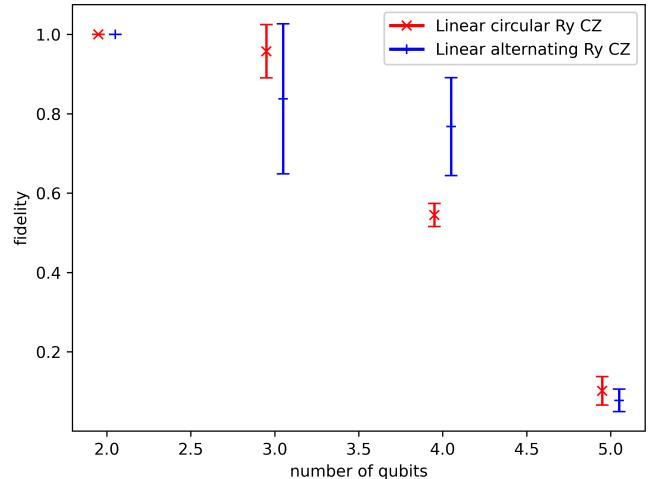


Fig. 13: Mean fidelity $|\langle E_h^{\text{VQLS}} | E_h^{\text{FEM}} \rangle|^2$ computed with 5 simulations for a number of qubits ranging from 2 to 5 for the linear circular R_y -CZ ansatz (red x) and the linear alternating R_y -CZ ansatz (blue +). The 95% confidence intervals are plotted.

V. CONCLUSION

In this work, we have proposed an extension of an adaptive mesh refinement algorithm for time-harmonic Maxwell's

equation to the quantum formalism. In order to do so, we have first derived the propagation equation from Maxwell's equations. Then, we covered the classical mesh refinement loop and studied its formulation in order to adapt it to the quantum formalism. The local error estimators can be extracted as expectation values of quantum circuits. Using block-encoding, localized a posteriori error estimators are approximately computed on quantum computers. Our method has been validated on simulations which do not account for sampling error thus recovering the convergence rate of the classical setup.

Until now, in the hybrid mesh adaptation loop, the quantum sampling error dominates the error coming from the discretization onto the mesh. Future work should aim at improving this and proposing an algorithm that is efficient overall. Preconditioning of the linear system as proposed by [33] is a promising lead.

REFERENCES

- [1] Philippe G. Ciarlet. The finite element method for elliptic problems. Studies in Mathematics and its Applications. Vol. 4. Amsterdam - New York - Oxford: North-Holland Publishing Company. XIX, 580 p. Dfl. 130.00; \$ 56.00 (1978)., 1978.
- [2] Susanne C. Brenner and L. Ridgway Scott. *The mathematical theory of finite element methods*, volume 15 of *Texts Appl. Math.* New York, NY: Springer, 3rd ed. edition, 2008.
- [3] Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd. Quantum algorithm for linear systems of equations. *Phys. Rev. Lett.*, 103:150502, Oct 2009.
- [4] Yousef Saad. *Iterative methods for sparse linear systems*. Philadelphia, PA: SIAM Society for Industrial and Applied Mathematics, 2nd ed. edition, 2003.
- [5] Andris Ambainis. Variable time amplitude amplification and quantum algorithms for linear algebra problems. In Christoph Dürr and Thomas Wilke, editors, *29th International Symposium on Theoretical Aspects of Computer Science (STACS 2012)*, volume 14 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 636–647, Dagstuhl, Germany, 2012. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.
- [6] Andrew M. Childs, Robin Kothari, and Rolando D. Somma. Quantum algorithm for systems of linear equations with exponentially improved dependence on precision. *SIAM Journal on Computing*, 46(6):1920–1950, 2017.
- [7] András Gilyén, Yuan Su, Guang Hao Low, and Nathan Wiebe. Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, STOC ’19, page 193–204. ACM, June 2019.
- [8] Yi git Subaş ı, Rolando D. Somma, and Davide Orsucci. Quantum algorithms for systems of linear equations inspired by adiabatic quantum computing. *Phys. Rev. Lett.*, 122:060504, Feb 2019.
- [9] Dong An and Lin Lin. Quantum linear system solver based on time-optimal adiabatic quantum computing and quantum approximate optimization algorithm. *ACM Transactions on Quantum Computing*, 3(2), March 2022.
- [10] Lin Lin and Yu Tong. Optimal polynomial based quantum eigenstate filtering with application to solving quantum linear systems. *Quantum*, 4:361, November 2020.
- [11] Pedro C.S. Costa, Dong An, Yuval R. Sanders, Yuan Su, Ryan Babbush, and Dominic W. Berry. Optimal scaling quantum linear-systems solver via discrete adiabatic theorem. *PRX Quantum*, 3:040303, Oct 2022.
- [12] Ashley Montanaro and Sam Pallister. Quantum algorithms and the finite element method. *Physical Review A*, 93(3):032324, 2016.
- [13] B. D. Clader, B. C. Jacobs, and C. R. Sprouse. Preconditioned quantum linear system algorithm. *Phys. Rev. Lett.*, 110:250504, Jun 2013.
- [14] Corey Jason Trahan, Mark Loveland, Noah Davis, and Elizabeth Ellison. A variational quantum linear solver application to discrete finite-element methods. *Entropy*, 25(4), 2023.
- [15] Carlos Bravo-Prieto, Ryan LaRose, M. Cerezo, Yigit Subasi, Lukasz Cincio, and Patrick J. Coles. Variational Quantum Linear Solver. *Quantum*, 7:1188, November 2023.
- [16] Ricardo H. Nochetto, Kunibert G. Siebert, and Andreas Veeser. Theory of adaptive finite element methods: An introduction. In *Multiscale, nonlinear and adaptive approximation. Dedicated to Wolfgang Dahmen on the occasion of his 60th birthday*, pages 409–542. Berlin: Springer, 2009.
- [17] I. Babuška and W. C. Rheinboldt. A-posteriori error estimates for the finite element method. *International Journal for Numerical Methods in Engineering*, 12(10):1597–1615, 1978.
- [18] Rüdiger Verfürth. *A review of a posteriori error estimation and adaptive mesh-refinement techniques*. Chichester: John Wiley & Sons; Stuttgart: B. G. Teubner, 1996.
- [19] Peter Monk. *Finite element methods for Maxwell's equations*. Numer. Math. Sci. Comput. Oxford: Oxford University Press, 2003.
- [20] Jian-Ming Jin. *The finite element method in electromagnetics*. Hoboken, NJ: John Wiley & Sons, 3rd ed. edition, 2014.
- [21] Jean-Claude Nedelec. Mixed finite elements in \mathbb{R}^3 . *Numer. Math.*, 35:315–341, 1980.
- [22] Zhiming Chen, Long Wang, and Weiyang Zheng. An adaptive multilevel method for time-harmonic maxwell equations with singularities. *SIAM Journal on Scientific Computing*, 29(1):118–138, 2007.
- [23] Lin Lin. Lecture notes on quantum algorithms for scientific computation, 2022.
- [24] Yu Tong, Dong An, Nathan Wiebe, and Lin Lin. Fast inversion, preconditioned quantum linear system solvers, fast green’s-function computation, and fast evaluation of matrix functions. *Phys. Rev. A*, 104:032422, Sep 2021.
- [25] Abhishek Arora, Benjamin M. Ward, and Caglar Oskay. An implementation of the finite element method in hybrid classical/quantum computers, 2025.
- [26] Shantanav Chakraborty, András Gilyén, and Stacey Jeffery. The power of block-encoded matrix powers: Improved regression techniques via faster hamiltonian simulation. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2019.
- [27] Igor A. Baratta, Joseph P. Dean, Jørgen S. Dokken, Michal Habera, Jack S. Hale, Chris N. Richardson, Marie E. Rognes, Matthew W. Scroggs, Nathan Sime, and Garth N. Wells. DOLFINx: the next generation FEniCS problem solving environment. preprint, 2023.
- [28] Ali Javadi-Abhari, Matthew Treinish, Kevin Krsulich, Christopher J. Wood, Jake Lishman, Julien Gacon, Simon Martiel, Paul D. Nation, Lev S. Bishop, Andrew W. Cross, Blake R. Johnson, and Jay M. Gambetta. Quantum computing with Qiskit, 2024.
- [29] Daan Camps and Roel Van Beeumen. Fable: Fast approximate quantum circuits for block-encodings. In *2022 IEEE International Conference on Quantum Computing and Engineering (QCE)*, pages 104–113, 2022.
- [30] <https://github.com/QuantumApplicationLab/vqls-prototype>.
- [31] Sukin Sim, Peter D. Johnson, and Alán Aspuru-Guzik. Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms. *Advanced Quantum Technologies*, 2(12), October 2019.
- [32] M. J. D. Powell. A direct search optimization method that models the objective and constraint functions by linear interpolation. In *Advances in optimization and numerical analysis. Proceedings of the 6th workshop on optimization and numerical analysis, Oaxaca, Mexico, January 1992*, pages 51–67. Dordrecht: Kluwer Academic Publishers, 1994.
- [33] Matthias Deiml and Daniel Peterseim. Quantum realization of the finite element method. *arXiv preprint arXiv:2403.19512*, 2024.