Solving the Schrödinger Eigenproblem with PUMA

Clelia Albrecht, Marc Alexander Schweitzer

Fraunhofer SCAI

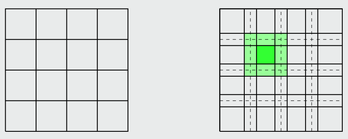
Flat-top Partition of Unity Method: Motivation

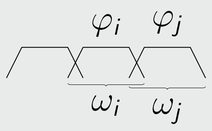
While classical FEM are widely used and well-suited for a large number of applications, they have a few shortcomings:

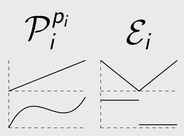
- Grid handling is expensive
- Approximation problems:
  - crack modeling
  - singularities in solution
  - geometrical singularities
- No possibility to insert a-priori knowledge about physics of problem into computation
- In general: no arbitrary approximation spaces
- Methods that address these issues: Generalized FEM, Extended FEM, EFE, PUFEM,...

Flat-Top PUM

The flat-top Partition of Unity Method (PUM) can be understood as a (very) generalized, meshfree variant of FEM.







Partition of Unity Space

- Partition of unity (PU) $\{\varphi_i\}$ with $\omega_i := \text{supp}(\varphi_i)$
- Smooth splicing of local spaces

$$V^{\text{PU}} := \sum_{i=1}^{N} \varphi_i V_i(\omega_i) = \sum_{i=1}^{N} \varphi_i (\mathcal{P}^{p_i} + \mathcal{E}_i).$$

- Approximation by $V_i(\omega_i)$, functions φ_i just "glue".

Enrichments $\mathcal{E}_i$

- Can be any general functions
- Chosen to match physics of the problem

Stablilization

- Restrictions of enrichment functions can cause stability problems on local patches
- Couplings of polynomials and enrichment functions can also be ill-conditioned
- In the flat-top PUM, we overcome these stability issues by constructing a stability transformation
- This is achieved by partial orthogonalization wrt. a chosen inner product.

Variational Mass Lumping

- In the flat-top PUM, we are also able to construct a localized mass matrix and thus have a variational mass lumping approach
- This construction is independent of local spaces (arbitrary enrichments, order)
- We derive a block-diagonal, symetric positive definite matrix $\bar{M}$

PUMA software framework

- C++ implementation of flat-top PUM
- Parallelized using MPI
- Provides possibility to use your own enrichment functions
- Accessible through Python interface

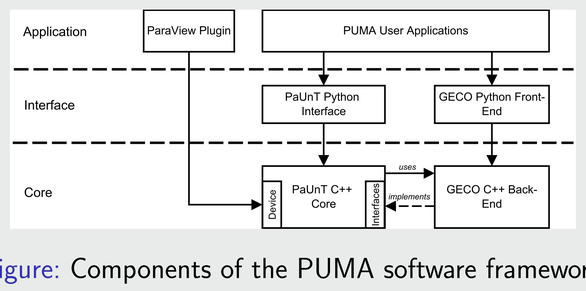


Figure: Components of the PUMA software framework

Bibliography

[1] Clelia Albrecht, Constanze Klaar, John Ernest Pask, Marc Alexander Schweitzer, N. Sukumar, and Albert Ziegenhagel, *Orbital-enriched flat-top partition of unity method for the Schrödinger eigenproblem*, Computer Methods in Applied Mechanics and Engineering **342** (2018), 224 – 239.

[2] Clelia Albrecht, Constanze Klaar, and Marc Alexander Schweitzer, *Stable and Efficient Quantum Mechanical Calculations with PUMA on triclinic lattices*, Ninth International Workshop on Meshfree Methods for Partial Differential Equations, Proceedings in Preparation, 2019.

[3] M. A. Schweitzer, *A parallel multilevel partition of unity method for elliptic partial differential equations*, Lecture Notes in Computational Science and Engineering, vol. 29, Springer, 2003.

Application: The Schrödinger Eigenproblem

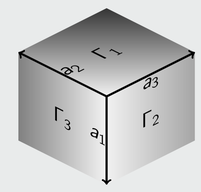
This is a joint work with N. Sukumar (UC Davis) and John Pask (LLNL)

Domain

Consider $\Omega \subset \mathbb{R}^3$ a parallelepiped unit cell with primitive lattice vectors $a_\ell,\ \ell \in \{1,2,3\}$, i.e.

$$\Omega = \text{conv}\{a_1, a_2, a_3\},$$

with six faces with $\bigcup_\ell (\Gamma_\ell \cup (\Gamma_\ell + a_\ell)) = \Gamma,\ \ell \in \{1,2,3\}$



The one-electron Schrödinger equation

$$\begin{aligned} -\tfrac{1}{2}\nabla^2\psi(\boldsymbol{x}) + V(\boldsymbol{x})\psi(\boldsymbol{x}) &= \varepsilon\psi(\boldsymbol{x}) && \text{in } \Omega, \\ \psi(\boldsymbol{x} + a_\ell) &= \exp(i\boldsymbol{k}\cdot a_\ell)\psi(\boldsymbol{x}) && \text{on } \Gamma_\ell, \\ \nabla\psi(\boldsymbol{x} + a_\ell) &= \exp(i\boldsymbol{k}\cdot a_\ell)\nabla\psi(\boldsymbol{x}) && \text{on } \Gamma_\ell \end{aligned}$$

Calculations on triclinic unit cells

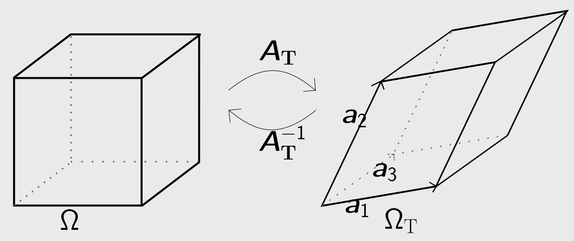


Figure: Schematic representation of the transformation $\boldsymbol{A}_{\mathbf{T}} : \Omega \longrightarrow \Omega_{\mathrm{T}}$. This (linear) transformation can be represented by the matrix $\boldsymbol{A}_{\mathbf{T}} = (a_1, a_2, a_3)$, whose column vectors are the primitive lattice vectors $\boldsymbol{a}_d$, $d = 1, 2, 3$ of Ω_{T}.

Model problem: Gaussian Potential on a triclinic unit cell

Let $\Omega = \text{conv}(a_1, a_2, a_3)$ with

$a_1 = a(1, 0.02, -0.04)$, $a_2 = a(0.06, 1.05, -0.08)$ and $a_3 = a(0.10, -0.12, 1.10)$,

with lattice parameter $a = 5$ and $\boldsymbol{\tau} = \frac{a_1+a_2+a_3}{2}$.
The Gaussian potential is defined as

$$V(\boldsymbol{x}) = \sum_R V_g(|\boldsymbol{x} - \boldsymbol{\tau} - R|),$$

with

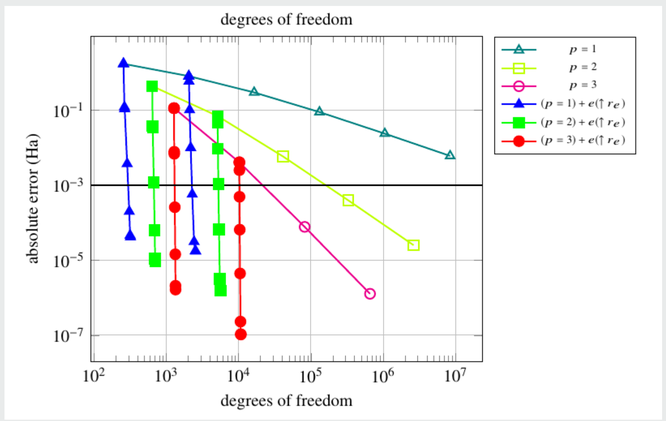
$$V_g(\boldsymbol{r}) = -10 \exp\left(-\frac{\boldsymbol{r}^2}{2.25}\right)$$

and we want to solve

$$-\frac{1}{2}\nabla^2\psi + V\psi = \varepsilon\psi \text{ in } \Omega$$

with Bloch-periodic boundary conditions (wave vector $\boldsymbol{k} = (0.12, 0.23, 0.34)$ in reciprocal lattice coordinates).

Numerical Results



Results were obtained on the Drachenfels cluster at Fraunhofer SCAI, on 4 computational nodes (Intel® Xeon® CPU E5-2650 v2 @ 2.6GHz nodes with 16 cores each).

Figure: Convergence history of the lowest eigenvalue λ_1 for the gaussian oscillator potential attained for different refinement schemes. We consider a purely polynomial approximation ($p = 1, 2, 3$) on a sequence of uniformly refined covers, which shows the expected $2p$-convergence rates (see [1]). Furthermore, we consider a refinement by increasing the enrichment radius with a single enrichment function on a fixed uniform cover ($4 \times 4 \times 4$ and $8 \times 8 \times 8$) that is labeled by $p = 1, 2, 3 + e \uparrow r_e$, where we observe spectral convergence.

Conclusion

- Employing enrichment functions drastically reduces number of necessary DOFs
- We overcome problems of enriched FEM like instability and the need to solve a general EVP

Future work

- Improve parallel integration scheme for enrichment functions, explore use of hybrid parallelization
- Optimize eigenvalue solver
- Compute larger and more realistic quantum mechanical problems (more atoms, full DFT-loop implementation)

Fraunhofer Institute for Algorithms and Scientific Computing SCAI
Schloss Birlinghoven, 53754 Sankt Augustin, Germany
clelia.albrecht@scai.fraunhofer.de