Centre of Excellence

MICROCARD-2

Deliverable 1.1

D1.1 algebraic adaptivity report

PROJECT INFORMATION	
Grant agreement number	101172576
Project acronym	MICROCARD-2
Project title	Numerical modeling of cardiac electrophysiology at the cellular scale
Start date of the project	1 November 2024
Duration of the project	30 months
Call	HORIZON-EUROHPC-JU-2023-COE-03 – Centres Of Excellence For Exascale HPC Applications
Topic	HORIZON-EUROHPC-JU-2023-COE-03-01
Scientific coordinator	Mark Potse (UBx)

DELIVERABLE INFORMATION	
Deliverable number	1.1
Dissemination level	public
Work packages	1
Tasks	1.2, 1.3, 3.4
WP leader	Martin Weiser
Lead beneficiary	ZIB
Authors	Fatemeh Chegini, Julian Schramm, Martin Weiser
Reviewers	Simone Pezzuto, Hartwig Anzt
Due date of Deliverable	30 June 2026
Submission date	30 June 2026

EuroHPC Joint Undertaking

Co-funded by the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EuroHPC. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the EuroHPC Joint Undertaking and its members (including top-up funding by ANR, BMBF, and Ministero dello sviluppo economico).

Contents

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EuroHPC. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the EuroHPC Joint Undertaking and its members (including top-up funding by ANR, BMBF, and Ministero dello sviluppo economico).

Authors Fatemeh Chegini, Julian Schramm, Martin Weiser

History of changes

version	date	change history	authors	organisation
1	30/06/2026	initial version	F. Chegini, J. Schramm, M. Weiser	ZIB

List of beneficiaries

nr	organisation name	short name	country	PI
1	Université de Bordeaux	UBx	France	Mark Potse
2	Université de Strasbourg	UNISTRA	France	Vincent Loechner
3	Simula Research Laboratory AS	SIMULA	Norway	Aslak Tveito
4	Università degli studi di Pavia	UNIPV	Italy	Luca Pavarino
5	Technische Universität München	TUM	Germany	Hartwig Anzt
6	Karlsruhe Institute of Technology	KIT	Germany	Axel Loewe
7	Zuse Institute Berlin	ZIB	Germany	Martin Weiser
8	MEGWARE Computer Vertrieb und Service GMBH	MEGWARE	Germany	Axel Auweter
9	Inria	Inria	France	Olivier Aumage
10	Università di Trento	UTrento	Italy	Simone Pezzuto

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EuroHPC. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the EuroHPC Joint Undertaking and its members (including top-up funding by ANR, BMBF, and Ministero dello sviluppo economico).

PROJECT SUMMARY

Cardiac function is coordinated by an electric system whose disorders are among the most frequent causes of death and disease. Numerical models of this complex system are mature and widely used, but to match observations in aging and diseased hearts they need to move from a continuum approach to a representation of individual cells and their interconnections. This makes the problem more complex, harder to solve, and four orders of magnitude larger, necessitating exascale computers.

The EuroHPC-2019 MICROCARD project developed a simulation platform that can meet this challenge, by a joint effort of HPC experts, numerical scientists, biomedical engineers, and biomedical scientists, from academia and industry. The Centre of Excellence MICROCARD-2 will consolidate and scale up the MICROCARD results enabling digital twins of cardiac tissue.

We will further develop MICROCARD's numerical schemes, moving to second-order spatial discretization. We will develop mixed-precision preconditioners and data compression to reduce communication bandwidth. The highly successful efforts made in MICROCARD towards automated compilation of high-level model descriptions into optimized, energy-efficient system code for different CPUs and GPUs will be extended to upcoming architectures. We will continue efforts to robustify parallel remeshing software and add necessary functionality for parallel mesh partitioning and production of realistic synthetic tissue meshes needed for simulations.

The platform will be benchmarked with realistic test cases and be made accessible for a wide range of users with tailored workflows. It will be adaptable to similar biological systems such as nerves, and several of our products such as improved solvers, preconditioners, remeshers, and partitioners will be reusable in a wide range of applications.

DELIVERABLE SUMMARY

This report describes algebraic adaptivity for both linear and quadratic finite elements, and reports the current implementation and performance status. Algebraic adaptivity is a low-overhead alternative to geometric mesh adaptation, following the algebraic subset-selection strategy developed for integrating cardiac electrophysiology with spectral deferred correction (SDC) methods [1–3]. A fixed mesh is retained, while the hierarchical second-order finite-element space is used to separate the linear/P1 part of the solution from the quadratic edge coefficients. This separation should eventually allow the code to identify regions where first-order finite elements are sufficiently accurate and to restrict ion-current evaluation and linear-solver work to the degrees of freedom that are actually needed. The completed work reported here establishes the main prerequisites for that goal in openCARP [4]: a distributed Extracellular-Membrane-Intracellular (EMI) formulation, its spatial discretization with hierarchical second-order (P2) finite elements, associated assembly and membrane projection/scattering operators, and a hierarchical vertex/edge field-split preconditioner. The report also documents the current SDC implementation, including the Jacobian-free treatment of ionic state variables through

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EuroHPC. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the EuroHPC Joint Undertaking and its members (including top-up funding by ANR, BMBF, and Ministero dello sviluppo economico).

LIMPET. Current work addresses the integration of these validated building blocks into a fully automatic adaptivity algorithm.

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EuroHPC. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the EuroHPC Joint Undertaking and its members (including top-up funding by ANR, BMBF, and Ministero dello sviluppo economico).

1 Introduction

The EMI model of cardiac electrophysiology resolves **e**xtracellular space, cell **m**embranes, and **i**ntracellular space explicitly [5, 6]. This provides a detailed representation of cardiac tissue at the cellular scale, but it also leads to very large coupled finite-element systems. Reducing the cost of these simulations requires improvements at several levels: the spatial discretization, the time integration, the solver strategy, and the algebraic representation of higher-order degrees of freedom. A major milestone was the release of openCARP version 19.0 on April 1, 2026, the first official release to provide EMI model functionality [7]. Building upon the distributed first-order EMI implementation with IMEX time stepping, the prerequisites for higher-order spatial and temporal discretization and, ultimately, algebraic adaptivity for cellular-resolution EMI simulations have been implemented in the openCARP code base.

The first-order implementation uses a piecewise linear finite element discretization on a tetrahedral mesh (P1), a piecewise constant representation of ionic states on membrane faces, semi-implicit time stepping, and a parallel parabolic solve. It defines the mesh processing and operator assembly structure that the higher-order implementation must preserve. The second-order extension introduces a *hierarchical* piecewise quadratic discretization (P2) of the protentials. In this representation, the P1 ansatz space is retained and enriched by piecewise quadratic basis functions associated to mesh edges and providing a higher-order correction. This separation is the structure needed for both two-level preconditioning and adaptive activation of quadratic edge degrees of freedom (DOFs). The P2 extension is introduced to improve the spatial discretization efficiency, and should be complemented by a corresponding reduction of the time discretization error by higher order time stepping schemes. We therefore also consider spectral deferred correction (SDC) methods.

SDC constructs a higher order time integrator from repeated low-order correction sweeps. Within each time step, the solution is represented at several collocation points, and the subinterval problems are solved sequentially during each sweep. This allows the existing semi-implicit EMI update, parabolic solver, and LIMPET ionic-model evaluation to remain the fundamental computational building blocks, rather than replacing them with a single, fully coupled high order solve.

In a combination of P2 space and SDC time discretization, the computational effort can be reduced significantly by simply not performing correction steps when the expected correction size is below the accepted tolerance level. This is enabled by the hierarchical P2 and the SDC discretizations act as corrections.

2 The EMI model and its discretization

2.1 EMI model

The Extracellular–Membrane–Intracellular model represents cardiac tissue as a collection of separate intracellular regions embedded in an extracellular space. Let Ω_i, $i = 0, \ldots, n$ denote

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EuroHPC. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the EuroHPC Joint Undertaking and its members (including top-up funding by ANR, BMBF, and Ministero dello sviluppo economico).

the spatial subdomains of n myocytes ($i \geq 1$) and extracellular space $i = 0$ covering the computational domain Ω, and let Γ denote the membrane and intercellular interfaces separating the intracellular and extracellular domains and adjacent myocytes. The unknowns are the intracellular and extracellular electric potentials in Ω, denoted by u_i, and the ionic states w on Γ. The transmembrane voltage is the jump of the potential across the membrane,

$$v = u|_{\Omega_i} - u|_{\Omega_j} \quad \text{on } \Gamma,$$

with the convention $i > j$ for incident subdomains fixing the sign. In the intracellular and extracellular volume domains, the EMI model describes elliptic electrostatics:

$$-\mathrm{div}(\sigma_i \nabla u_i) = 0 \quad \text{in } \Omega_i.$$

The domains are coupled through the membrane current

$$I_m = C_\mathrm{m} \frac{\partial v}{\partial t} + I_\mathrm{ion}(v, w) - I_\mathrm{stim},$$

comprising capacitive current, ionic current depending on transmembrane voltage and ionic state w, and optional stimulation. The membrane current coincides with the adjacent normal fluxes $n_i^T \sigma_i \nabla u|_{\Omega_i}$. The ionic states evolve according to some membrane model as

$$\frac{\partial w}{\partial t} = f(v, w).$$

Several ionic models exist and can be prescribed on different parts of Γ. In particular, gap junctions on the intercalated discs separating adjacent myocytes are represented by passive linear interface currents, with a conductance parameter multiplying the transmembrane voltage. More complex, nonlinear dynamics is prescribed on myocyte-extracellular interfaces.

2.2 First-order EMI discretization

The first order spatial discretization uses hybrid continuous/discontinuous, piecewise linear finite elements on a simplicial triangulation $\mathcal{T}_h^\Omega$ for the potentials, which are continuous within each myocyte and in the extracellular space but are discontinuous across the membranes. The P1 finite element space for potentials is

$$V_h = \left\{ \varphi \in \bigoplus_{i=0}^{n} H^1(\Omega_i) \,\middle|\, \forall T \in \mathcal{T}_h^\Omega : \varphi|_T \in \mathcal{P}_1 \right\},$$

where $\mathcal{P}_p$ is the space of polynomials up to order p. Note that in contrast to a standard continuous P1 space, the nodal Lagrange basis of V_h contains multiple basis functions on membrane vertices, one for each vertex-subdomain incidence. Let $\mathcal{N}$ denote the vertices of $\mathcal{T}_h^\Omega$. Then, the vertex-subdomain incidence set $\mathcal{I}^{(1)} = \{(x, i) \in \mathcal{N} \times \mathbb{N} \mid x \in \overline{\Omega_i}\}$ indexes the basis functions.

Since more than two potentital values exist at triple points, where at least two myocytes and extracellular space meet, transmembrane voltage and ionic currents are not well-defined at these

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EuroHPC. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the EuroHPC Joint Undertaking and its members (including top-up funding by ANR, BMBF, and Ministero dello sviluppo economico).

vertices. Therefore, transmembrane voltages and ionic currents are calculated on membrane faces, corresponding to a piecewise constant representation.

Let $\mathcal{F}_h$ denote the set of faces in $\mathcal{T}_h^\Omega$. The interface representation

$$\Gamma = \left\{ F \in \mathcal{F}_h \,\middle|\, \left|\{i \in \mathbb{N} \mid F \subset \overline{\Omega_i}\}\right| = 2 \right\}$$

contains all membrane facets and is used for representing the transmembrane voltage v and the ionic state variables as well as for evaluating the ion current. In particular, ionic states are represented in the facewise constant space

$$W_h = \{\psi \in L^2(\Gamma) \mid \forall F \in \Gamma : \psi|_F \in \mathcal{P}_0\}.$$

After multiplying the potential equations with test functions from V_h and integrating by parts, the discrete weak EMI formulation for $u \in V_h$ contains volumetric diffusion terms and a surface membrane current term,

$$\sum_{i=0}^{n} \int_{\Omega_i} \nabla\phi^T \sigma \nabla u \, dx + \int_\Gamma I_m(v) \, dS = 0.$$

The coupling between the potentials $u \in V_h$, the voltage $v \in W_h$, and membrane current $I_m \in W_h$, respectively, is expressed by sparse matrices. The signed interpolation matrix $\mathbf{B}$ maps potentials to the transmembrane voltage. For a triangular membrane face $F \in \Gamma$ with vertices n_1, n_2, n_3 incident to subdomains Ω_i and Ω_j, the L^2-orthogonal projection of the jump to W_h is

$$v_b(F) = \frac{1}{3} \sum_{k=1}^{3} [u|_{\Omega_i} - u|_{\Omega_j}](n_k).$$

Thus, each row of $\mathbf{B}$ contains the weights $+1/3$ for the intracellular side and $-1/3$ for the extracellular side. Similarly, the scattering operator $\mathbf{S}$ maps the computed face currents back to the volume right-hand side with face area scaling and orientation-dependent signs,

$$(\mathbf{S}I_m)|_{\Omega_i}(n_k) = \sum_{F \in \Gamma, n_k \in F} s_F \frac{A_F}{3} I_m(F), \qquad k = 1, 2, 3,$$

where A_F is the face area and $s_F \in \{-1, +1\}$ encodes the orientation of the face contribution.

Time stepping uses an implicit-explicit (IMEX) Euler method with operator splitting. The ionic current is evaluated from the membrane voltage and ionic state variables, whereas the mass matrix defines the capacitive current of the parabolic solve. With time step Δt, the linear system for the implicit treatment of electrostatics can be written as

$$\mathbf{A}\,\delta\mathbf{u}^n = \mathbf{r}^n, \qquad \mathbf{u}^{n+1} = \mathbf{u}^n + \delta\mathbf{u}^n,$$

with

$$\mathbf{A} = \mathbf{K} + \frac{C_\mathrm{m}}{\Delta t}\mathbf{M}.$$

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EuroHPC. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the EuroHPC Joint Undertaking and its members (including top-up funding by ANR, BMBF, and Ministero dello sviluppo economico).

Here, $\mathbf{K}$ is the volumetric stiffness matrix for the potentials defined by

$$(\mathbf{K})_{ab} = \sum_{i=0}^{n} \int_{\Omega_i} (\nabla \varphi_a)^T \sigma_i \nabla \varphi_b \, dx \quad \text{for } a, b \in \mathcal{I}^{(1)}, \tag{1}$$

and $\mathbf{M}$ is the membrane-surface mass matrix obtained by integrating over the membrane interface Γ,

$$(\mathbf{M})_{ab} = \int_{\Gamma} \varphi_a \varphi_b \, dS \qquad \text{for } a, b \in \mathcal{I}^{(1)}. \tag{2}$$

This means that $\frac{C_\mathrm{m}}{\Delta t}\mathbf{M}$ is the capacitive part of the implicit operator. The complete right-hand side contains the scattered ionic and stimulus currents, and the stiffness contribution from the potentials. It is assembled as

$$\mathbf{r}^n = -\left(\mathbf{S} I_b^n + \mathbf{K}\mathbf{u}^n\right) \tag{3}$$

with face current

$$I_b^n = I_\mathrm{ion}(v^n, w^n) - I_\mathrm{stim}^n.$$

In addition to membrane stimulus currents, volumetric source terms can be defined as well by specifying a set $\mathcal{N}_\mathrm{stim} \subset \mathcal{N}$ of vertices to which a current density is applied.

2.3 Hierarchical P2 EMI discretization

The hierarchical second-order discretization of the continuous EMI model keeps the ionic state representation of the first-order method, but replaces the finite element space V_h by a piecewise quadratic space. The discrete space is

$$V_h^2 = \left\{ \varphi \in \bigoplus_{i=1}^{n} H^1(\Omega_i) \,\middle|\, \forall T \in \mathcal{T}_h^\Omega : \varphi|_T \in \mathcal{P}_2(T) \right\}.$$

Instead of a Lagrange basis for V_h^2, we employ a hierarchical basis containing the nodal functions from the P1 discretization and additional quadratic "bubbles" associated to edges, which have a parabolic profile on that edge and whose support is limited to the incident elements, see Fig. 1 for illustration. This less common basis has decisive advantages for the EMI implementation in openCARP. Since the quadratic bubble functions form a pure correction to the P1 discretization contained in the P2 discretization, any quadratic DOFs with small coefficient contribute only a small correction and can simply be ignored. This is frequently the case before and behind the depolarization front, which allows to reduce the number of active DOFs significantly (algebraic adaptivity). Moreover, the splitting $V_h^2 = V_h \oplus V_h^+$ of the P2 ansatz space into the P1 ansatz space and an extension space yields an effective subspace correction method corresponding to a block Jacobi preconditioner [8]. Since for the extension block a diagonal preconditioner is sufficient, this block Jacobi preconditioner can be combined with algebraic adaptivity without significant overhead. Moreover, as ion currents can anyways not be computed directly from each vertex coefficient, the missing Lagrange property of the hierarchical basis introduces no inefficiency at all.

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EuroHPC. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the EuroHPC Joint Undertaking and its members (including top-up funding by ANR, BMBF, and Ministero dello sviluppo economico).

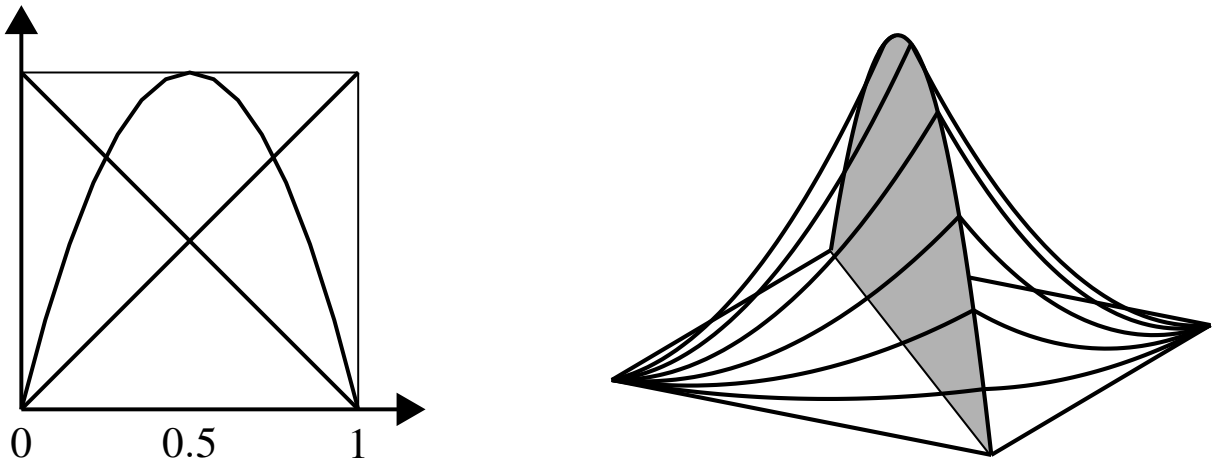

Figure 1: *Left:* Sketch of P2 shape functions in 1D, comprising two linear shape functions and one quadratic bubble restricted to the edge. *Right:* A quadratic bubble function in 2D.

Let $\mathcal{E}$ be the set of edges in $\mathcal{T}_h^\Omega$. Then the quadratic bubble functions are indexed by $\mathcal{I}^+ = \{(e,i) \in \mathcal{E} \times \mathbb{N} \mid e \subset \Omega_i\}$, analogously to the indexing of P1 basis functions. Their explicit separation from the P1 vertex DOFs is used below for field-split preconditioning and algebraic adaptivity. The P2 hierarchical basis functions are then indexed by $\mathcal{I}^{(2)} = \mathcal{I}^{(1)} \cup \mathcal{I}^+$. Note that the coefficients of the bubble functions represent corrections to the embedded P1 function. For example, a spatially constant potential is represented by assigning the same value to all vertex DOFs, while the edge DOF coefficients are zero. This property simplifies setting initial conditions, constant Dirichlet data, and prescribed constant stimulation, as there is no difference to the P1 implementation.

The stiffness and mass matrices are assembled as in (1) and (2), respectively, but using the larger index set $\mathcal{I}^{(2)}$. Of course, the order of the quadrature rule used has to be increased accordingly. The current implementation uses a sub-parametric geometric mapping with affine mapping to the reference element, such that the electric potential is approximated by a hierarchical quadratic expansion, but the element geometry remains piecewise linear. This keeps the P2 discretization compatible with the existing EMI meshes, and constant Jacobian and determinant can be used in the quadrature.

The ionic states are represented as piecewise constant functions from W^h as before. The transmembrane voltage is obtained by averaging the P2 potential over each side of the membrane face and taking the difference between the two averages. Let $\mathcal{S}_F^+$ and $\mathcal{S}_F^-$ denote the indices of P2 basis functions on the two sides of a membrane face F, and define

$$\beta_a(F) = \frac{1}{A_F} \int_F \varphi_a^{(2)} \, dS, \qquad a \in \mathcal{S}_F^\pm,$$

where A_F is the face area. The P2 face voltage is then

$$v(F) = \sum_{a \in \mathcal{S}_F^+} \beta_a(F) U_a^+ - \sum_{a \in \mathcal{S}_F^-} \beta_a(F) U_a^-.$$

Here, U_a^+ and U_a^- are the coefficients of the basis function φ_a on both sides of F, respectively. The resulting $v(F)$ is the face-averaged transmembrane voltage passed to the ionic model for

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EuroHPC. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the EuroHPC Joint Undertaking and its members (including top-up funding by ANR, BMBF, and Ministero dello sviluppo economico).

computing I_{ion} and f. The face average of each of the three linear vertex basis functions and each of the three quadratic edge basis functions is $1/3$. Therefore, if $F = \{1, 2, 3\}$ has edge DOFs e_{12}, e_{23}, and e_{31}, the one-sided P2 face average has the form

$$\bar{u}_F^{(2)} = \frac{1}{3}(U_1 + U_2 + U_3) + \frac{1}{3}(U_{e_{12}} + U_{e_{23}} + U_{e_{31}}).$$

Consequently, nonzero quadratic edge DOF coefficients influence the face-averaged transmembrane voltage passed to the ionic model, but the membrane model is not evaluated separately for these DOFs. The ionic states and membrane current remain stored and computed per membrane face.

The same face-average weights are used to scatter membrane currents back into the right-hand side (3). Testing the spatially constant membrane current density $I_b(F)$ on a membrane face F with P2 shape functions yields the scatter contribution

$$(\mathbf{S}^{(2)} I_b)_a = \sum_{F:\, a \in \mathcal{S}_F} s_F \, A_F \, \beta_a(F) \, I_b(F),$$

where $s_F \in \{-1, +1\}$ is the orientation sign of the face contribution. For a triangular face, the same current value contributes to the six face-supported P2 rows as

$$b_1 \mathrel{+}= s_F A_F \frac{1}{3} I_b(F), \quad b_2 \mathrel{+}= s_F A_F \frac{1}{3} I_b(F), \quad b_3 \mathrel{+}= s_F A_F \frac{1}{3} I_b(F),$$

$$b_{e_{12}} \mathrel{+}= s_F A_F \frac{1}{3} I_b(F), \quad b_{e_{23}} \mathrel{+}= s_F A_F \frac{1}{3} I_b(F), \quad b_{e_{31}} \mathrel{+}= s_F A_F \frac{1}{3} I_b(F).$$

Here, b_1, b_2, b_3 and $b_{e_{12}}, b_{e_{23}}, b_{e_{31}}$ denote the corresponding entries of the assembled P2 volume right-hand-side vector $\mathbf{r}^n$.

As before, the resulting P2 parabolic Euler step system matrix is

$$\mathbf{A}^{(2)} = \mathbf{K}^{(2)} + \frac{C_{\text{m}}}{\Delta t} \mathbf{M}^{(2)}.$$

Finally, visualization and comparison with first-order EMI use a restriction operator from the P2 discretization to the P1 discretization by just extracting the P1-related DOFs,

$$\mathbf{u}_{\text{out}}^{(1)} = \mathbf{u}^{(2)}|_{\mathcal{I}^{(1)}}.$$

For good locality in distributed simulations, the P2 DOFs are ordered globally by MPI rank, and within each rank the P1 DOFs first, followed by the quadratic extension DOFs.

2.4 Stimulation and boundary conditions

The stimulation for starting tissue excitation can be provided in three different ways: as membrane stimulation current as discussed above, as volumetric stimulus, and by prescribing Dirichlet boundary conditions.

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EuroHPC. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the EuroHPC Joint Undertaking and its members (including top-up funding by ANR, BMBF, and Ministero dello sviluppo economico).

openCARP allows to define volumetric stimuli currents in terms of selecting a set $\mathcal{N}_{\text{stim}} \subset \mathcal{N}$ of affected vertices on specific subdomains Ω_i. A piecewise linear stimulus current density in V_h is defined by setting the coefficients of affected vertices to the given value. In a distributed simulation, a stimulated region may cross an MPI partition boundary and an edge may be present on more than one rank. The implementation therefore constructs the support using global vertex identifiers. The thus defined finite element function is translated to the right hand side $\mathbf{r}^n$ by multiplying it with a volumetric mass matrix. In the P2 discretization, all edge DOFs in the stimulus current density are set to zero, retaining the piecewise linear stimulus current density.

Piecewise constant Dirichlet conditions on boundary faces are applied by prescribing the given value to all incident vertex coefficients and the system matrix is modified accordingly. In a P2 discretization, the additional edge coefficients are set to zero, such that the same piecewise linear boundary function is obtained. Still, the matrix modification has to be applied to the corresponding rows and columns.

If the active Dirichlet boundary changes in time, the affected matrix rows and right-hand-side entries are updated using the same procedure as in the first-order formulation, but with the P2 numbering and vectors, and the preconditioner updated accordingly.

2.5 Accuracy of P1 and P2 discretizations

The aim of the higher order discretization is a higher accuracy on the same mesh width than P1 or, equivalently, the same accuracy on a coarser mesh. A growing efficiency benefit of P2 discretization is expected for decreasing mesh width. We present a convergence study for verification of these expected results.

The study uses a synthetic brick mesh containing a fixed $2 \times 2 \times 1$ block of cuboidal myocytes embedded in an extracellular bath and connected through intercalated disk segments. The cells and extracellular region are explicitly tagged, allowing the EMI mesh processing pipeline to reconstruct membrane and gap junction faces. Starting from `brick_2x2x1_h32`, nested uniform refinement produces the levels `base`, `refine1`, `refine2`, and `refine3`, with $h = 32, 16, 8, 4\,\mu$m, respectively. Each refinement splits every tetrahedron into eight elements and every membrane triangle into four faces.

All runs use the same electrophysiological parameters: $\sigma_0 = 2.0\,\text{mS}/\mu\text{m}$ (extracellular), $\sigma_i = 0.4\,\text{mS}/\mu\text{m}$ for $i \geq 1$ (intracellular), the Aliev–Panfilov membrane model, and the Plonsey gap-junction model with $R_m = 0.003$. A transmembrane current of $20\,\mu\text{A}/\text{cm}^2$ is applied for 0.1 ms, with $\Delta t = 10\,\mu$s and a final time of 1 ms. The P1 and P2 runs therefore differ only in their spatial discretizations.

Since no analytical solution is available for this EMI test case, the finest completed P2 run, `refine3`, is used as the reference solution. For each coarser run, the transmembrane voltage is compared over every fifth time frame. Membrane face centers are reconstructed from the output membrane mesh, and each coarse face is matched to the nearest membrane face in the reference mesh. The reported quantity is an unweighted relative discrete space–time ℓ^2 error over saved

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EuroHPC. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the EuroHPC Joint Undertaking and its members (including top-up funding by ANR, BMBF, and Ministero dello sviluppo economico).

level	h [μm]	vertices	P1 DOFs	P2 DOFs	P1 error	P2 error	P1/P2
`base`	32	609	1029	5 983	$1.555 \cdot 10^{-2}$	$5.411 \cdot 10^{-3}$	2.87
`refine1`	16	4 347	5 983	39 257	$9.646 \cdot 10^{-3}$	$1.931 \cdot 10^{-3}$	4.99
`refine2`	8	32 837	39 257	280 637	$4.914 \cdot 10^{-3}$	$4.896 \cdot 10^{-4}$	10.04
`refine3`	4	255 241	280 637	2 113 669	$2.068 \cdot 10^{-3}$	reference	–

Table 1: Empirical accuracy comparison of P1 and P2 EMI discretizations on the nested uniformly refined `brick_2x2x1_h32` mesh hierarchy. Errors are unweighted relative discrete space–time ℓ^2 errors of V_m against the `refine3` P2 run.

time frames and membrane faces,

$$E_h^{(p)} = \frac{\left(\sum_{n=0}^{N_t} \sum_{f \in \Gamma_h} \left| V_{m,h}^{n,p}(f) - V_{m,\mathrm{ref}}^{n}(\Pi f) \right|^2 \right)^{1/2}}{\left(\sum_{n=0}^{N_t} \sum_{f \in \Gamma_h} \left| V_{m,\mathrm{ref}}^{n}(\Pi f) \right|^2 \right)^{1/2}},$$

where $p \in \{1, 2\}$ denotes the spatial order and Πf is the nearest reference membrane face. Since the sums are not weighted by face area, this is a discrete comparison of saved membrane traces rather than a continuous $L^2(0, T; L^2(\Gamma))$ norm. It provides a practical verification metric, but not a formal manufactured-solution convergence test, because the reference is numerical and the membrane coupling remains face averaged.

P2 is more accurate than P1 at every comparable mesh level. This does also hold if the higher number of DOFs for the P2 discretization on each level (factor eight) is taken into account, see Fig. 2 for a comparison. For example, the error of the P2 discretizaion on the base mesh is comparable to that of the P1 discretization on `refine2`, but the P1 discretization contains more than eight times as many DOFs. Thus, the second-order discretization is more efficient than a linear finite element method already on rather coarse meshes. Whether this translates into lower wall-clock cost depends on the efficiency of the P2 solver and preconditioner.

The empirical orders of convergence are 0.97 for P1 and 1.73 for P2. These values are somewhat preliminary due to the small range of refinements investigated, such that pre-asymptotic effects may play a role. The fact that the P2 discretization does not achieve the full order two could also be caused by the use of piecewise constant ionic states $w \in W_h$. Formally, the ion currents affecting the evolution of potential differences on the membrane are fluxes contained in the Sobolev space $H^{-1/2}(\Gamma)$. Compared to the potentials in $H^1(\Omega)$, a difference of two in approximation order can be expected to lead to an order reduction of the complete discretization by 1/2, and a convergence order of 1.5. This and the observed order of 1.75 suggest that the ionic state discretization should be moved to piecewise linear to achieve the full accuracy of the P2 discretization.

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EuroHPC. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the EuroHPC Joint Undertaking and its members (including top-up funding by ANR, BMBF, and Ministero dello sviluppo economico).

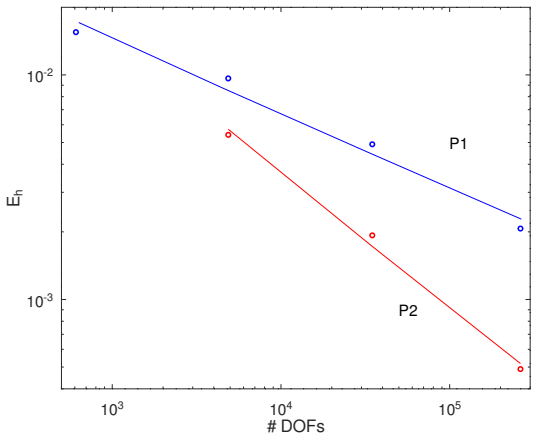

Figure 2: Finite element discretization error versus the number of degrees of freedom for P1 and P2 discretizations. Computed error metric E_h and linear regression with empirical orders 0.97 and 1.73, respectively.

2.6 Two-level preconditioning for hierarchical P2

The hierarchical P2 basis induces a natural 2-by-2 block structure of the parabolic EMI system. The first field contains the vertex (P1) coefficients $\mathbf{u}_l$, while the second field contains the edge coefficients $\mathbf{u}_q$, which represent the higher-order correction. If these coefficients are ordered as

$$\mathbf{u}^{(2)} = \begin{bmatrix} \mathbf{u}_l \\ \mathbf{u}_q \end{bmatrix}, \tag{4}$$

the IMEX Euler system matrix assumes the block form

$$\mathbf{A}^{(2)} = \begin{bmatrix} \mathbf{A}_{ll} & \mathbf{A}_{lq} \\ \mathbf{A}_{ql} & \mathbf{A}_{qq} \end{bmatrix}.$$

This splitting is used to define a two-level preconditioner and implement it for the PETSc backend using the provided field-split functionality.

The openCARP implementation constructs this split from the actual global numbering, where the ordering (4) is local to each MPI rank. When the selected solver options request a field-split preconditioner, the PETSc solver creates two index sets for the P2 discretization: one for the vertex unknowns and one for the edge unknowns. These index sets are registered with PETSc as the fields `l` and `q`, respectively. PETSc then applies `PCFIELDSPLIT` to the already assembled P2 matrix. This way the same matrix assembly path can be used for all P2 simulations, while the preconditioner can exploit the disctinction between P1 coefficients and the hierarchical edge DOFs explicitly.

The implemented additive two-level preconditioner has a block-diagonal structure, applying different preconditioners P_l and P_q to the P1 block A_{ll} and the hierarchical extension block A_{qq}, respectively:

$$P^{-1}_{\text{add}} = \begin{bmatrix} P_l^{-1} & 0 \\ 0 & P_q^{-1} \end{bmatrix}.$$

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EuroHPC. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the EuroHPC Joint Undertaking and its members (including top-up funding by ANR, BMBF, and Ministero dello sviluppo economico).

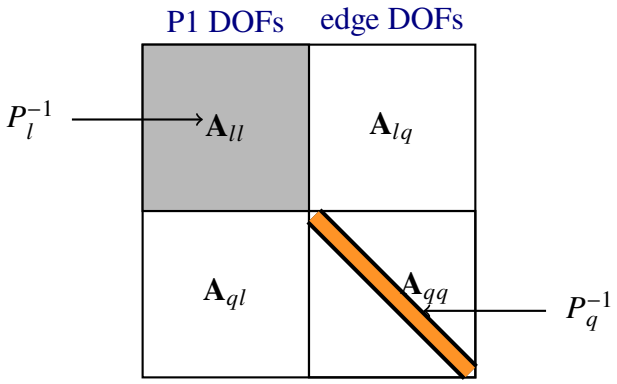

Figure 3: Block interpretation used for the hierarchical P2 PETSc field split. The full matrix remains coupled through $\mathbf{A}_{lq}$ and $\mathbf{A}_{ql}$, while the preconditioner acts separately on the vertex/P1 block $\mathbf{A}_{ll}$ and on the quadratic edge DOF block $\mathbf{A}_{qq}$.

For P_l, any of the effective preconditioners considered for the EMI model before can be used; for simplicity, we restrict the investigation to the algebraic multigrid method GAMG provided by PETSc, which uses aggregation-based coarsening with a Jacobi smoother. Since no low-frequency functions are contained in V_h^+, the condition number of A_{qq} is quite small, such that a Jacobi preconditioner P_q is effective, see Fig. 3. It is well-known that this two-level preconditioner is effective and optimal for elliptic problems, provided that P_l is optimal [8].

We compare the performance of the two-level preconditioner on the P2 discretization to other choices of P_q, namely symmetric Gauß-Seidel (SSOR) and GAMG, and to GAMG on both a pure P1 discretization and the full P2 discretization. The same test problem and meshes as in the convergence study in Sec. 2.5, though with fewer time steps. We use a conjugate gradient solver with a relative tolerance of 10^{-8} for the preconditioned residual. The complete PETSc option strings for setting up the preconditioner are provided in Appendix A.

Tab. 2 reports the number of DOFs, the average Krylov iteration count, the accumulated parabolic-solver time, and the resulting time per Krylov iteration. On the same mesh, P2 is necessarily more expensive because it contains substantially more unknowns. The relevant comparison is therefore at comparable accuracy as provided in Tab. 1. The `base` P2/GAMG run takes 0.15 s, similar to P1/GAMG on `refine1` (0.16 s), while providing an error close to P1/GAMG on `refine2` (0.54 s). Similarly, comparing P2/GAMG on `refine1` with P1/GAMG on `refine3` reveals a run time ratio of more than 4 in favor of P2, while the achieved accuracy is comparable. We also observe that GAMG is less effective on P2 discretizations than on P1 discretizations of the same size, as can be seen from comparing the iteration count of P2/GAMG on `refine2` and P1/GAMG on `refine3`. Despite that, the higher efficiency of P2 provides the observed performance benefit.

Moreover, we observe that within the P2 preconditioners, the two-level splitting takes more iterations than GAMG, but requires much less time per iteration, such that the overall run time is decreased by a factor of more than two. This is more clearly visible on larger meshes, as setup cost dominates on very small meshes.

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EuroHPC. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the EuroHPC Joint Undertaking and its members (including top-up funding by ANR, BMBF, and Ministero dello sviluppo economico).

refine	disc.	DOFs	P_l	P_q	avg. itr	time [s]	time/itr [ms]
base	P1	1029	GAMG	—	39.4	0.06	0.155
	P2	5 983	GAMG		81.6	0.15	0.184
			GAMG	Jacobi	117.0	0.15	0.127
			GAMG	SSOR	117.0	0.15	0.127
			GAMG	GAMG	117.0	0.15	0.132
refine1	P1	5 983	GAMG	—	62.0	0.16	0.248
	P2	39 257	GAMG		156.0	1.14	0.735
			GAMG	Jacobi	190.0	0.59	0.313
			GAMG	SSOR	189.8	0.60	0.314
			GAMG	GAMG	190.0	0.59	0.310
refine2	P1	39 257	GAMG	—	99.6	0.54	0.533
	P2	280 637	GAMG		309.0	14.62	4.706
			GAMG	Jacobi	368.0	6.05	1.649
			GAMG	SSOR	368.4	5.98	1.627
			GAMG	GAMG	368.2	5.95	1.620
refine3	P1	280 637	GAMG	—	153.4	4.70	2.979
	P2	2 113 669	GAMG		619.6	245.14	39.456
			GAMG	Jacobi	751.8	95.33	12.726
			GAMG	SSOR	751.8	94.29	12.588
			GAMG	GAMG	751.8	97.97	13.079

Table 2: Solver comparison for P1 and hierarchical P2 discretization on the uniformly refined `brick_2x2x1_h32` mesh hierarchy. The P2 rows compare the single-block GAMG preconditioner with additive vertex/edge field splits.

Together, the hierarchic P2 discretization enabling the application of the two-level block-diagonal preconditioner yields a speedup of factor eight compared to the baseline P1 discretization.

Surprisingly, Jacobi, SSOR, and GAMG yield nearly identical results on the quadratic edge block, both in iteration count and run time. This indicates that the preconditioner quality is dominated by the GAMG on P1 and the $V_h \oplus V_h^+$ split. The unexpected identical run time, despite GAMG being much more complex than Jacobi, is still under investigation.

2.7 Memory consumption

The P2 discretization has the same number of potential degrees of freedom as a once uniformly refined P1 discretization, but the corresponding system matrices $\mathbf{K}^{(1)}$ and $\mathbf{K}^{(2)}$, respectively, have different sparsity patterns, with $\mathbf{K}^{(2)}$ being slightly more dense by a factor of less than two. On the other hand, the two-level preconditioner from Sec. 2.6 requires much less storage than GAMG on a P1 discretization of the same size. The memory demand of an algebraic multigrid preconditioner such as GAMG is roughly linear in the number of DOFs. Consequently, the two-

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EuroHPC. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the EuroHPC Joint Undertaking and its members (including top-up funding by ANR, BMBF, and Ministero dello sviluppo economico).

level preconditioner occupies just $1/8$ the memory of GAMG, since the Jacobi preconditioner on the $\mathbf{A}_{qq}$ block takes no additional memory at all.

3 Spectral Deferred Correction method

Spectral deferred correction (SDC) methods construct a higher-order time integrator from a sequence of low-order correction sweeps [9, 10]. In other popular high-order one-step methods like Runge-Kutta the solution on a sub time grid is direcly computed. In SDC instead, we solve an equation for a defect and iteratively correct the corresponding approximate solution on a similar time grid in each step. For an abstract ODE $\dot{u} = f(u)$ with solution u^* and current iterate u^k, the defect equation reads

$$\begin{aligned} \delta u^k(t) = (u^* - u^k)(t) &= \int_0^t (f(u^*) - \dot{u}^k)\, d\tau \\ &= \int_0^t \left(f(u^k + \delta u^k) - f(u^k) + f(u^k) - \dot{u}^k\right) d\tau \\ &\approx \int_0^t \Big(\underbrace{f'(u^k)\delta u^k}_{\text{correction}} + \underbrace{f(u^k) - \dot{u}^k}_{\text{residual}} \Big)\, d\tau \end{aligned} \tag{5}$$

and defines a (linearized) correction δu^k. A high-order quadrature rule is used for the residual part which, as u^k is explicitly known, can be evaluated directly at any point in time. The propagation of the correction δu^k on the other hand uses a simpler quadrature, based on an implicit Euler step in the simplest case. This allows to use existing low order time stepping schemes as the basic building blocks. The intended formulation follows the preconditioned SDC ideas used in the earlier algebraic-adaptivity study [2, 11]; implementation details such as the particular Euler, SDIRK, or LU sweep can be varied without changing the EMI spatial operators.

This is attractive for EMI simulations for two reasons. First, SDC provides higher order time discretizations with better efficiency than the first order method implemented in openCARP so far. This is similar to the efficiency gain provided by P2 in comparison to P1 on the spatial side. Second, it allows to reduce the computational cost by not performing corrections in parts of the domain where the correction is sufficiently small. This form of adaptivity works on the algebraic level without incurring the overhead of mesh refinement and re-assembly of system matrices.

3.1 SDC for potentials

Let one time step be written as $[t_n, t_{n+1}]$, with $\Delta t = t_{n+1} - t_n$, and let

$$0 = \tau_0 < \tau_1 < \cdots < \tau_m = \Delta t$$

be the SDC collocation grid. In the current implementation, Radau points are used because if combined with an appropriate initial solution or if the method is based on the LU-trick they

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EuroHPC. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the EuroHPC Joint Undertaking and its members (including top-up funding by ANR, BMBF, and Ministero dello sviluppo economico).

define an L-stable method. L-stability is a necessary property, since the spatial discretized EMI model is a differential-algebraic equation of index 1. For each collocation point, we store a potential approximation $\mathbf{u}_i \approx \mathbf{u}(t_n + \tau_i)$ in Ω and an ionic state approximation $\mathbf{w}_i \approx \mathbf{w}(t_n + \tau_i)$ on Γ.

The semi-discrete EMI system from Sec. 2.1 can be written as

$$C_\mathrm{m}\mathbf{M}\dot{\mathbf{u}} = -\mathbf{K}\mathbf{u} - \mathbf{S}\mathbf{I}_b(\mathbf{v}, \mathbf{w}),$$

$$\dot{\mathbf{w}} = \mathbf{R}(\mathbf{v}, \mathbf{w}), \qquad \mathbf{v} = \mathbf{B}\mathbf{u}.$$

Here, $\mathbf{M}$ is the embedded membrane-surface mass operator, such that $C_\mathrm{m}\mathbf{M}\dot{\mathbf{u}}$ represents the capacitive current term. The matrix $\mathbf{K}$ is the volumetric stiffness matrix, $\mathbf{I}_b$ contains ionic and stimulus membrane currents on Γ, and $\mathbf{R}$ is the ionic-state right-hand side provided by LIMPET.[1]

The collocation time discretization is formulated in terms of a high-order quadrature matrix $\mathbf{S}^{\mathrm{SDC}}$ as

$$C_\mathrm{m}\mathbf{M}(\mathbf{u}_{i+1} - \mathbf{u}_i) = \sum_{j=1}^{m} \mathbf{S}_{ij}^{\mathrm{SDC}}\, \mathbf{F}(\mathbf{u}_j, \mathbf{w}_j), \qquad i = 0, \ldots, m-1,$$

$$\mathbf{w}_{i+1} - \mathbf{w}_i = \sum_{j=1}^{m} \mathbf{S}_{ij}^{\mathrm{SDC}}\, \mathbf{R}(\mathbf{B}\mathbf{u}_j, \mathbf{w}_j), \qquad i = 0, \ldots, m-1,$$

with the right hand side

$$\mathbf{F}(\mathbf{u}, \mathbf{w}) = -\mathbf{K}\mathbf{u} - \mathbf{S}\mathbf{I}_b(\mathbf{B}\mathbf{u}, \mathbf{w}).$$

In principle, this collocation system can be solved by a fixed point iteration, e.g., Newton's method. However, since high-order quadrature matrices $\mathbf{S}^{\mathrm{SDC}}$ are dense, the systems are coupled over all collocation time steps $i = 1, \ldots, m$. Computing the Newton corrections, whose equations are formulated with a full matrix including the Jacobian blocks of the right hand side times, requires a large and coupled linear solve. In SDC, instead we compute corrections $\delta\mathbf{u}$, $\delta\mathbf{w}$ based on a lower triangular approximation $\widehat{\mathbf{S}}$ of $\mathbf{S}^{\mathrm{SDC}}$. The corresponding system can be solved by forward sweeping, where each block elimination step is actually one time step of the low order IMEX Euler method moving from collocation point τ_i to τ_{i+1}. The quadrature matrix $\widehat{\mathbf{S}}$ is fixed by the chosen SDC sweep and collocation rule, and usually does not depend on the iteration index k.

Given a current collocation approximation $(\mathbf{u}_i^k,\ \mathbf{w}_i^k)$ for $i = 1, \ldots, m$ at iteration k, the SDC sweep first computes the residual

$$\mathbf{\Phi}_{u,i}^k = -C_\mathrm{m}\mathbf{M}(\mathbf{u}_{i+1}^k - \mathbf{u}_i^k) + \sum_{j=1}^{m} \mathbf{S}_{ij}^{\mathrm{SDC}}\, \mathbf{F}(\mathbf{u}_j^k, \mathbf{w}_j^k)$$

[1] The code generation by LIMPET from ionic model descriptions in a domain-specific language needed to be extended to provide the right hand side $\mathbf{R}$ explicitly. Before, only updated ionic states had been provided by LIMPET, which precluded the use of time stepping schemes different from those provided directly in the ionic models, see Sec. 3.3.

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EuroHPC. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the EuroHPC Joint Undertaking and its members (including top-up funding by ANR, BMBF, and Ministero dello sviluppo economico).

using the high-order quadrature matrix $\mathbf{S}^{\mathrm{SDC}}$. Integrating the residual with a high-order quadrature rule ensures that the fixed points $(\mathbf{u}_i^*,\ \mathbf{w}_i^*)$ are high-order collocation solutions in the sense that at all τ_i the differential equation is satisfied. Subsequently, the corrections $\delta\mathbf{u}_i^k$ and $\delta\mathbf{w}_i^k$ are computed sequentially for $i = 0, \ldots, m-1$ by solving

$$C_{\mathrm{m}}\mathbf{M}(\delta\mathbf{u}_{i+1}^k - \delta\mathbf{u}_i^k) + \sum_{j=1}^{i+1} \widehat{\mathbf{S}}_{ij}\mathbf{K}\delta\mathbf{u}_j^k = \mathbf{\Phi}_{u,i}^k$$

for $\delta\mathbf{u}_{i+1}^k$, which is a low-order time stepping for the defect equation. After rearranging terms, the matrix of the system to be solved is

$$\mathbf{A}_i^{\mathrm{SDC}} = \mathbf{K} + \frac{C_{\mathrm{m}}}{\widehat{\mathbf{S}}_{i,i+1}}\mathbf{M}.$$

This has exactly the same structure as the first-order EMI matrix $\mathbf{K} + (C_{\mathrm{m}}/\Delta t)\mathbf{M}$, with $\widehat{\mathbf{S}}_{i,i+1}$ acting as the effective implicit SDC sub time step for interval i. For a fixed time step, collocation rule, and sweep type, these coefficients do not change between SDC sweeps. Hence the m system matrices $\mathbf{A}_i^{\mathrm{SDC}}$ for each collocation interval can be assembled once and reused in subsequent sweeps; only the right-hand side changes. The same holds for any preconditioners to be used. The existing linear solver and preconditioner infrastructure can also be reused. Membrane currents are recomputed at the collocation states and enter through the residual, while the capacitance and diffusion terms define the implicit part of the correction. The ionic current is handled explicitly through repeated current evaluations in the SDC residual, without assembling the ionic-current Jacobian $\partial\mathbf{I}_b/\partial\mathbf{u}$.

3.2 SDC for ionic states

For the ionic state variables, the residual is analogously

$$\hat{\mathbf{r}}_{w,i}^k = \mathbf{w}_i^k - \mathbf{w}_{i+1}^k + \sum_{j=1}^{m} S_{ij}^{\mathrm{SDC}}\,\mathbf{R}(\mathbf{B}\mathbf{u}_j^k, \mathbf{w}_j^k).$$

An implicit treatment of stiff ionic equations in the SDC method, similar to the electrostatics of the potentials, would require Jacobians of all ionic models with respect to their state variables. While technically possible to compute, implementing this for all ionic models supported by LIMPET was considered too high an effort.

Instead, we designed a Jacobian-free yet implicit SDC variant that makes use of the implicit time stepping schemes $\mathcal{L}$ hidden in the ionic models and available through LIMPET as updated ionic states

$$\mathcal{L}_{\Delta t}(\mathbf{v}, \mathbf{w}) \approx \mathbf{w} + \int_t^{t+\Delta t} \mathbf{R}(\mathbf{v}, \mathbf{w})\, d\tau.$$

Returning to (5), we let LIMPET evaluate the correction integral. Instead of solving the implicit Euler based SDC equation, we compute

$$\delta\mathbf{w}_{i+1}^k = \mathcal{L}_{\Delta\tau_i}\left(\mathbf{v}_i^k, \mathbf{w}_i^k + \delta\mathbf{w}_i^k + \hat{\mathbf{r}}_{w,i}^k\right) - \mathcal{L}_{\Delta\tau_i}\left(\mathbf{v}_i^k, \mathbf{w}_i^k\right), \qquad \Delta\tau_i = \tau_{i+1} - \tau_i,$$

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EuroHPC. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the EuroHPC Joint Undertaking and its members (including top-up funding by ANR, BMBF, and Ministero dello sviluppo economico).

where also the residual contribution $\hat{\mathbf{r}}_{w,i}^{k}$ is pushed through the implicit LIMPET integrator, which is important for very stiff ionic models. When $\mathcal{L}$ is chosen as the implicit Euler scheme, we thereby compute the defects of the implict Euler based SDC.

3.3 LIMPET updates for SDC

The standard EMI time step treats LIMPET as a black box: the transmembrane voltage is passed to the ionic model, the ionic state $\mathbf{w}$ is advanced once over the fixed global time step Δt to $\mathcal{L}_{\Delta t}(\mathbf{w})$, and the resulting membrane current I_{ion} is returned to the EMI solver. This is not sufficiently flexible for SDC, where the right hand side $\mathbf{R}$ is required for computing the residual, the ionic states must be reset to the initial value at the beginning of each sweep, and varying time step sizes $\Delta\tau_i$ arise due to the nonuniform collocation grid.

In response to these requirements, LIMPET has been extended to (i) explicitly providing the right hand side $\mathbf{R}$ of ionic dynamics that is internally available in all ionic models and (ii) to be able to set and get ionic states $\mathbf{w}$ explicitly. This has been implemented via an abstract, model-agnostic interface, such that heterogeneous simulations employing different ionic models on different parts of Γ are seamlessly supported.

For speeding up ionic model evaluations, lookup tables replacing expensive trigonometric/exponential function calls had been introduced in LIMPET before. These lookup tables depend on the time step Δt. Consequently, LIMPET was extended to support a fixed number of different time step sizes $\Delta\tau_i$ by maintaining a corresponding set of lookup tables.

3.4 SDC time integration results

We test the current SDC implementation in openCARP on a mesh representing a chain of ten cuboid myocytes surrounded by extracellular space. It consists of 2560 tetrahedra and 825 vertices. We apply a stimulation through Dirichlet boundary conditions for the potential on the myocyte at one end of the chain. Fig. 4 shows the error in the FE coefficients of the potential $||\mathbf{u}_\tau - \mathbf{u}_{\text{ref}}||_2$. As expected, the errors of the IMEX Euler scheme and the SDC method with one sweep are both of first order and differ by a constant factor corresponding to the smaller collocation time steps compared to Δt. The second sweep (purple) increases the empirical convergence order to two, as expected. The third sweep appears not to increase the order further, but, at least for larger step sizes, error reduction by a constant factor is observed. We are currently investigating this unexpected behavior.

Next we compare the basic IMEX Euler scheme and the SDC method with two collocation points and three sweeps for $\tau \in [10^{-1}, 10^{1}]$ in view of computational effort. An SDC step is more accurate than one step of IMEX Euler, but also more expensive. As a measure of the cost for the simulation of a given time interval with one or the other method, we count the number of large linear solves by n_{imex} and n_{sdc} respectively. One time step of IMEX Euler requires one linear solve, whereas a step of this SDC variant requires six linear solves. From figure (4) we can read the time step sizes τ_{imex} and τ_{sdc} needed to achive the errors listed in table (3). As we

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EuroHPC. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the EuroHPC Joint Undertaking and its members (including top-up funding by ANR, BMBF, and Ministero dello sviluppo economico).

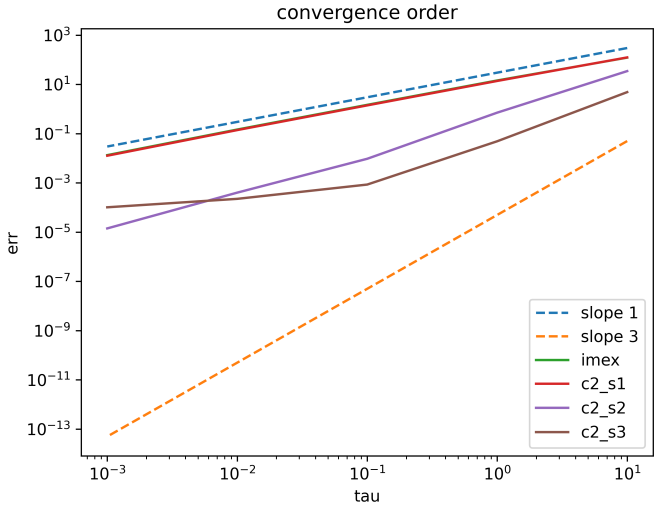

Figure 4: Error in the potential u versus time step size τ. Displayed is the first order IMEX scheme and the SDC method with two Radau IIa collocation points with one to three sweeps.

err [mV]	0.01	0.1	1.0
τ_{imex} $[\mu s]$	0.00069	0.00694	0.06944
τ_{sdc} $[\mu s]$	0.42920	1.46080	4.97180
$n_{\text{imex}}/n_{\text{sdc}}$	103.7	35.1	11.9
`speedup`	86.4	29.2	9.9

Table 3: SDC with two Radau collocation points and three sweeps vs. IMEX Euler

see the time step size of the SDC method can be choosen much larger than the one of the IMEX Euler scheme. Therefore we can simulate a given time interval in less steps. The ratio $n_{\text{imex}}/n_{\text{sdc}}$ gives us the theoretical speedup of the SDC over the IMEX scheme. This ratio only accounts for the number of large linear solves. In practice also compute steps of the ionic models and other vector operations have to be considered. We measured a speedup that is slightly lower than this theoretical ratio.

The SDC implementation support described above is available as a prototype, but the final EMI convergence table is not yet included in this report. The numerical validation still has to establish the observed order for the coupled EMI–LIMPET system and to separate possible error sources from output precision, ionic-state correction, and the parabolic EMI correction equation. The required study should compare the baseline semi-implicit method with SDC runs at several time step sizes, report the error against a sufficiently accurate reference solution, and measure the additional runtime per SDC sweep.

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EuroHPC. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the EuroHPC Joint Undertaking and its members (including top-up funding by ANR, BMBF, and Ministero dello sviluppo economico).

# myocytes	# DOFs	AA	cascadic AA
44	394,106	4.526	5.342
86	628,586	4.753	5.973
208	1,664,019	5.432	8.252
415	3,286,868	5.863	8.847

Table 4: Speedup factors for EMI simulations with algebraic adptivity on meshes of different size compared to simulations with full SDC corrections.

4 Algebraic Adaptivity

Adaptivity uses the fact that the correction computed by later SDC sweeps is often localized near the depolarization and repolarization fronts. If the local correction magnitude is already below the requested tolerance far from the fronts, the accurate computation of the correction can be restricted to the vicinity of the fronts.

One way to achieve this is to use established local mesh refinement and coarsening. Numerical experience has shown, however, that this is highly effective in view of reducing the problem size, but does not provide speedup due to the overhead of mesh modifications, system matrix re-assembly, and reconstruction of preconditioners.

Within MICROCARD-1, we had developed an adaptivity approach that works on the level of algebraic DOFs. The idea is to restrict later SDC correction solves to a selected subset of DOFs corresponding to the spatial region where the correction is expected to be significantly large. The mesh is never refined nor coarsened, such that the main overhead is due to submatrix extraction and preconditioner updating.

4.1 Algebraic adaptivity for P1 discretizations

For a first-order EMI or monodomain SDC correction, the selection rule can be written as

$$\mathcal{I}_{k+1}^{(1)} = \left\{ i \in \mathcal{I}_k^{(1)} \mid |\delta \mathbf{u}_i^k| \geq \mathrm{TOL}_{\mathrm{drop}} \right\},$$

where $\delta \mathbf{u}^k$ is the SDC correction from sweep k and $\mathcal{I}(1)_k$ is the active index set used in that sweep, with $\mathcal{I}_0^{(1)} = \mathcal{I}^{(1)}$. We start on the whole problem and discard DOFs where the previous correction was sufficiently small. This reduces the problem size for later SDC sweeps substantially in both monodomain and EMI examples, because the relevant correction is localized around the propagating front [2, 3]. Considerable speedup factors are achieved on several meshes, see Tab. 4. In addition, a cascadic approach, where the number of collocation points used is increased up to the final value m during the SDC iteration, yields further speedup.

For EMI, also the coupling between potentials in the volume and ionic states on the membranes must be respected. Selecting active potential DOFs therefore also determines which ionic DOFs must remain active. A face (ionic state DOF) should stay active whenever one of its incident vertices (potential DOF) is active, and $\mathcal{I}_k^{(1)}$ should include the directly coupled neighboring DOFs entering the transmembrane voltage. DOFs outside the active set are kept fixed during

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EuroHPC. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the EuroHPC Joint Undertaking and its members (including top-up funding by ANR, BMBF, and Ministero dello sviluppo economico).

that correction step, which is equivalent to a homogeneous Dirichlet boundary condition for the correction on the artificial algebraic boundary.

A challenge for this approach is that the system matrices change over the SDC iteration, getting smaller by discarding DOFs. This requires to adapt the preconditioners as well. When using BDDC, a subsetwise selection of DOFs can be used to restrict the necessary modifications to a refactorization of the global coarse level system.

4.2 Algebraic adaptivity for P2 discretizations

The hierarchical P2 basis provides a second form of algebraic adaptivity. The P2 DOFs contain the P1 DOFs as well as coefficients for quadratic bubble functions, which act as higher order correction to the P1 discretization. Keeping the P1 DOFs globally active while retaining edge DOFs only where needed for representing the SDC correction to sufficient accuracy. In this strategy, the mesh, the full P1 discretization, and the ionic state representation remain unchanged. Adaptivity acts only on the quadratic edge space V_h^+, which is about seven times larger than the P1 subspace.

An adaptive hierarchical P2 index set in SDC iteration k is $\mathcal{I}_k^{(2)} = \mathcal{I}^{(1)} \cup \mathcal{I}_k^+$, such that the P1 subspace is always included. A selection of the active edge index set $\mathcal{I}_k^+$ still needs to be implemented and tested. One candidate is the conservative choice $\mathcal{I}_0^+ = \mathcal{I}^+$ and dropping indices when the SDC correction magnitude drops below $\mathrm{TOL}_{\mathrm{drop}}$, as in the P1 algebraic adaptivity. The limitation is that the first sweep is always performed on the full problem, which limits the possible speedup to $8s/(7+s) \in [1.8, 8]$ if s is the number of SDC sweeps, again similar to P1 algebraic adaptivity. Alternatively, $\mathcal{I}_k^+$ can be chosen based on the magnitude of the residual coefficients, already for the first sweep. This has the potential of increasing the speedup factor bound to eight even if only few SDC iterations are performed, but requires careful a priori error analysis. For a consistent computation of transmembrane voltages, quadratic bubble functions associated to the same edge on both sides of a membrane need to be included or excluded jointly.

Further refinements are conceivable: Due to the minimal overhead of modifying the system for the two-level preconditioner, the adaptation can in principle also be performed independently for each collocation point. The different mechanisms of algebraic adaptivity for P1 and P2 discretizations can also be combined for further reduction of the computational cost. These strategies are subject of ongoing research and development.

5 Conclusion

This report documents the main building blocks needed for algebraic adaptivity in distributed EMI simulations with hierarchical P2 finite elements. The first-order EMI formulation provides the production baseline, while the hierarchical P2 extension improves spatial accuracy by adding quadratic edge DOFs without changing the piecewise constant discretization of ionic states. The accuracy study confirms the benefit of the P2 discretization of potentials on a simplified geometry.

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EuroHPC. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the EuroHPC Joint Undertaking and its members (including top-up funding by ANR, BMBF, and Ministero dello sviluppo economico).

The solver study shows that a vertex/edge field-split preconditioner can be substantially faster than applying one GAMG preconditioner to the complete P2 system, even though the split requires more solver iterations. In total, a speedup of factor 8 has been observed on relevant mesh widths.

The SDC method yields the higher-order time integration needed to improve efficiency by taking much larger time steps, and provides the framework for algebraic adaptivity by working on DOF subsets instead of mesh modifications. The nested region selection approach of algebraic adaptivity developed previously in MICROCARD-1 achieved a speedup of factors 5-8 on monodomain and EMI simulations, but requires to modify preconditioners in every SDC sweep, such as factorizing the coarse problem in BDDC anew. In contrast, the hierarchic P2 discretization combined with the two-level preconditioner allows an overhead-free order reduction in regions of small correction magnitude due to the simplicity of the Jacobi preconditioner for the $\mathbf{A}_{qq}$ block.

References

[1] Martin Weiser, Fatemeh Chegini. Adaptive multirate integration of cardiac electrophysiology with spectral deferred correction methods. In P. Nithiarasu, C. Vergara, editors, CMBE22 – 7th International Conference on Computational and Mathematical Biomedical Engineering, pages 528–531, 2022.

[2] Fatemeh Chegini, Thomas Steinke, Martin Weiser. Efficient adaptivity for simulating cardiac electrophysiology with spectral deferred correction methods. arXiv:2311.07206, submitted, 2022.

[3] F. Chegini, A. Froehly, N.M. Huynh, L. Pavarino, M. Potse, S. Scacchi, M. Weiser. Efficient numerical methods for simulating cardiac electrophysiology with cellular resolution. In 10th Int. Conf. Computational Methods for Coupled Problems in Science and Engineering 2023, 2023.

[4] Gernot Plank, Axel Loewe, Aurel Neic, Christoph Augustin, Yung-Lin Huang, Matthias Gsell, Elias Karabelas, Mark Nothstein, Jorge Sánchez, Anton Prassl, Gunnar Seemann, Edward Vigmond. The openCARP simulation environment for cardiac electrophysiology. Computer Methods and Programs in Biomedicine, 2021;208:106223. doi:10.1016/j.cmpb.2021.106223.

[5] K. H. Jæger, A. G. Edwards, W. R. Giles, A. Tveito. From millimeters to micrometers; re-introducing myocytes in models of cardiac electrophysiology. Frontiers in Physiology, 2021;12:763584. doi:10.3389/fphys.2021.763584.

[6] Karoline Horgmo Jæger, Aslak Tveito. Derivation of a cell-based mathematical model of excitable cells. In Aslak Tveito, Kent-Andre Mardal, Marie E. Rognes, editors, Modeling Excitable Tissue: The EMI Framework, pages 1–13. Springer International Publishing, Cham, 2021. doi:10.1007/978-3-030-61157-6_1.

[7] openCARP consortium. openCARP releases. https://git.opencarp.org/openCARP/openCARP/-/releases, 2026. Accessed June 7, 2026.

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EuroHPC. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the EuroHPC Joint Undertaking and its members (including top-up funding by ANR, BMBF, and Ministero dello sviluppo economico).

[8] P. Deuflhard, M. Weiser. Adaptive numerical solution of PDEs. de Gruyter, 2012.

[9] Alok Dutt, Leslie Greengard, Vladimir Rokhlin. Spectral deferred correction methods for ordinary differential equations. BIT Numerical Mathematics, 2000;40(2):241–266. doi:10.1023/A:1022338906936.

[10] Michael L. Minion. Semi-implicit spectral deferred correction methods for ordinary differential equations. Communications in Mathematical Sciences, 2003;1(3):471–500.

[11] Martin Weiser. Faster SDC convergence on non-equidistant grids by DIRK sweeps. BIT Numerical Mathematics, 2015;55(4):1219–1241. doi:10.1007/s10543-014-0540-y.

A PETSc options for the hierarchical P2 solver

The following options are generated internally when no external solver-options file is supplied.

```
-ksp_type cg
-ksp_rtol 1e-8
-pc_type gamg
-pc_gamg_type agg
-mg_levels_ksp_type richardson
-mg_levels_ksp_richardson_scale 0.75
-mg_levels_pc_type jacobi
-options_left
```

```
-ksp_type cg
-ksp_rtol 1e-8
-pc_type fieldsplit
-pc_fieldsplit_type additive
-fieldsplit_l_ksp_type preonly
-fieldsplit_l_pc_type gamg
-fieldsplit_l_pc_gamg_type agg
-fieldsplit_l_mg_levels_ksp_type richardson
-fieldsplit_l_mg_levels_ksp_richardson_scale 0.75
-fieldsplit_l_mg_levels_pc_type jacobi
-fieldsplit_q_ksp_type preonly
-fieldsplit_q_pc_type jacobi
-options_left
```

```
-ksp_type cg
-ksp_rtol 1e-8
-pc_type fieldsplit
-pc_fieldsplit_type additive
-fieldsplit_l_ksp_type preonly
-fieldsplit_l_pc_type gamg
-fieldsplit_l_pc_gamg_type agg
-fieldsplit_l_mg_levels_ksp_type richardson
-fieldsplit_l_mg_levels_ksp_richardson_scale 0.75
-fieldsplit_l_mg_levels_pc_type jacobi
-fieldsplit_q_ksp_type preonly
-fieldsplit_q_pc_type sor
-fieldsplit_q_pc_sor_symmetric
-options_left
```

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EuroHPC. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the EuroHPC Joint Undertaking and its members (including top-up funding by ANR, BMBF, and Ministero dello sviluppo economico).

```
-ksp_type cg
-ksp_rtol 1e-8
-pc_type fieldsplit
-pc_fieldsplit_type additive
-fieldsplit_l_ksp_type preonly
-fieldsplit_l_pc_type gamg
-fieldsplit_l_pc_gamg_type agg
-fieldsplit_l_mg_levels_ksp_type richardson
-fieldsplit_l_mg_levels_ksp_richardson_scale 0.75
-fieldsplit_l_mg_levels_pc_type jacobi
-fieldsplit_q_ksp_type preonly
-fieldsplit_q_pc_type gamg
-fieldsplit_q_pc_gamg_type agg
-fieldsplit_q_mg_levels_ksp_type richardson
-fieldsplit_q_mg_levels_ksp_richardson_scale 0.75
-fieldsplit_q_mg_levels_pc_type jacobi
-options_left
```

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EuroHPC. Neither the European Union nor the granting authority can be held responsible for them. The project is supported by the EuroHPC Joint Undertaking and its members (including top-up funding by ANR, BMBF, and Ministero dello sviluppo economico).