Received: 30 June 2023 | Revised: 2 August 2023 | Accepted: 23 August 2023

DOI: 10.1002/pamm.202300157

RESEARCH ARTICLE

Targeting a faster time-to-solution of mortar-based contact problems

Christopher Steimer[1] | **Matthias Mayr**[1,2] | **Alexander Popp**[1]

[1]Institute for Mathematics and Computer-Based Simulation, University of the Bundeswehr Munich, Neubiberg, Germany

[2]Data Science & Computing Lab, University of the Bundeswehr Munich, Neubiberg, Germany

Correspondence
Christopher Steimer, Institute for Mathematics and Computer-Based Simulation, University of the Bundeswehr Munich, 85577 Neubiberg, Germany.
Email: christopher.steimer@unibw.de

Abstract

While mortar methods are undoubtedly among the most prominent and accurate approaches in computational contact mechanics, they come with considerable cost. The required numerical effort originates not only from the evaluation of the mortar integrals but also from the solution of the system of linear equations within Newton-type solvers. While each of these cost factors has been analyzed and addressed individually in the past, this contribution outlines approaches and first results to comprehensively reduce the time to solution by combining dynamic load balancing techniques with algebraic multigrid preconditioners. We present a proof of concept for a three-dimensional unilateral contact simulation with an evolving contact zone.

1 | INTRODUCTION

For some time, mortar finite element methods (FEM) have been particularly popular in contact mechanics undergoing large deformations [1–5], as they provide high accuracy and variational consistency. They have also been extended to contact formulations using isogeometric analysis [6] or to various multiphysics phenomena [7, 8].

The numerical evaluation of the mortar terms requires mesh projections and intersections to define integration cells for the subsequent quadrature. Due to the locality and curse of dimensionality of the contact surface, the associated workload is usually distributed to only a subset of available hardware units, unfortunately, such that special measures are necessary to recover scalability of the contact evaluation on parallel computing clusters [9].

When using Lagrange multipliers to enforce the contact constraints, the straightforward use of standard shape functions results in a generalized saddle-point system, which can pose additional challenges for iterative linear solvers and their preconditioners. To deal with this particular block structure of the linear system on unstructured finite element meshes, we have recently proposed an algebraic multigrid (AMG) preconditioner for contact problems [10].

Despite their popularity, the efficiency of mortar-related computations in contact problems has only recently been targeted and many opportunities for improvement remain. This paper will discuss possible computational savings through the combined application of load-balancing strategies [9] and AMG preconditioners [10] for contact problems and will demonstrate a proof of concept, which hints at the design of extensive performance analysis in future work.

2 | UNILATERAL CONTACT OF SOLID BODIES

2.1 | Governing equations

For the sake of simplicity, we limit the presentation to a frictionless two-body contact problem with hyperelastic bodies $\Omega_0^{(i)}, i \in \{1,2\}$, and the contact interface $\Gamma_{\text{co}} = \partial\Omega^{(1)} \cap \partial\Omega^{(2)}$. As usual, both bodies can be subject to Dirichlet and Neumann boundary conditions on their boundaries $\Gamma_u^{(i)}$ and $\Gamma_\sigma^{(i)}$, respectively. Additionally, contact conditions are imposed via the Hertz–Signorini–Moreau conditions for frictionless contact, reading

$$g_n \geq 0, \quad p_n \leq 0, \quad g_n p_n = 0 \qquad \text{on } \Gamma_{\text{co}}, \tag{1a}$$

$$\mathbf{t}_\tau = 0 \qquad \text{on } \Gamma_{\text{co}}, \tag{1b}$$

where the first inequality of (1a) acts on the gap function g_n to ensure the nonpenetration of the bodies, the second inequality disallows adhesive stresses in the contact zone by restricting the contact pressure p_n in the normal direction, and the third expression, often called the complementarity condition, limits contact traction to states where the two bodies are in actual contact. Condition (1b) enforces a frictionless response on the contact boundary by setting the tangential contact traction $\mathbf{t}_\tau$ to zero.

After introducing a Lagrange multiplier field $\boldsymbol{\lambda}$ to enforce the contact constraints, the virtual work is given as

$$-\delta\mathcal{W}_{\text{kin}} - \delta\mathcal{W}_{\text{int,ext}} + \int_{\Gamma_{\text{co}}^{(\text{sl})}} \boldsymbol{\lambda}\big(\mathbf{u}^{(1)} - \mathbf{u}^{(2)}\big)\,\mathrm{dA} = 0, \tag{2a}$$

$$\int_{\Gamma_{\text{co}}^{(\text{sl})}} \big(\delta\boldsymbol{\lambda}^{(1)} - \delta\boldsymbol{\lambda}^{(2)}\big)\mathbf{g}_n \geq 0, \tag{2b}$$

where $\delta\mathcal{W}_{\text{kin}}$ and $\delta\mathcal{W}_{\text{int,ext}}$ denote the kinetic virtual work and the virtual work of internal and external forces, respectively. The integral in (2a) represents the virtual work $\delta\mathcal{W}_{\text{co}}$ of the contact forces, while (2b) resembles a variational inequality formulation of the contact constraints.

2.2 | Discretization

For spatial discretization of the solid bodies, we employ the FEM. A standard or dual mortar method is applied for discretization of the Lagrange multiplier field on the slave side of the contact interface. Then, the discretization of $\delta\mathcal{W}_{\text{co}}$ results in the mortar integrals

$$-\delta\mathcal{W}_{\text{co}} \approx \sum_{j=1}^{m^{(1)}} \sum_{k=1}^{n^{(1)}} \boldsymbol{\lambda}_j^{\text{T}} \underbrace{\left[\int_{\Gamma_{*,\text{h}}^{\text{sl}}} \Phi_j N_k^{(1)}\,\mathrm{d}\Gamma\right]}_{\mathfrak{D}[j,k]} \delta\mathbf{u}_k^{(1)} - \sum_{j=1}^{m^{(1)}} \sum_{\ell=1}^{n^{(2)}} \boldsymbol{\lambda}_j^{\text{T}} \underbrace{\left[\int_{\Gamma_{*,\text{h}}^{\text{sl}}} \Phi_j \left(N_\ell^{(2)} \circ \chi_{\text{h}}\right)\mathrm{d}\Gamma\right]}_{\mathfrak{M}[j,\ell]} \delta\mathbf{u}_\ell^{(2)}, \tag{3}$$

to be evaluated in each nonlinear iteration [2]. Thereby, N_j and Φ_j denote the shape function of the displacement and Lagrange multiplier field at node j, respectively. The number of solid nodes on the slave surface is denoted with $n^{(1)}$, while $m^{(1)}$ slave nodes carry a Lagrange multiplier unknown. On the master side, $n^{(2)}$ denotes the number of solid nodes. To deal with non-matching meshes, the discrete operator $\chi : \Gamma_{*,\text{h}}^{\text{ma}} \to \Gamma_{*,\text{h}}^{\text{sl}}$ maps data from the master side to the slave side of the contact interface. As depicted in Figure 1, the evaluation of the integrals in (3) requires projecting and imprinting of both surface meshes as well as subsequent quadrature, all together causing high numerical costs. The mortar matrices $\mathfrak{D}$ and $\mathfrak{M}$ are finally assembled from the nodal blocks $\mathfrak{D}[j,k]$ and $\mathfrak{M}[j,\ell]$, respectively.

2.3 | Algebraic formulation of linear systems

In our work, the inherent nonlinearities of the underlying contact problem due to both the nonlinear behavior of the solid bulk domains as well as nonlinearities arising from the contact constraints are simultaneously tackled via a semismooth

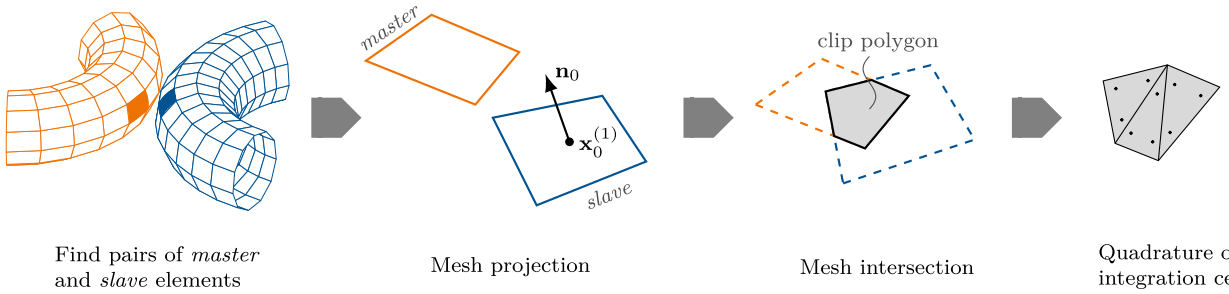

FIGURE 1 Evaluation of mortar terms to be performed in every nonlinear iteration (adopted from [9]).

Newton scheme. After a consistent linearization of all contact-related terms [2], one can assemble the linearized system to be solved in every nonlinear iteration.

When using the Lagrange multiplier method for constraint enforcement and allowing both standard or dual shape functions for the discretization of the Lagrange multiplier field, the arising linear system exhibits saddle-point characteristics. After all bulk and mortar terms have been evaluated and assembled, the resulting linear system reads [10]:

$$\left(\begin{array}{ccccc:cc} K_{\mathcal{N}_1\mathcal{N}_1} & K_{\mathcal{N}_1\mathcal{M}} & 0 & 0 & 0 & 0 & 0 \\ K_{\mathcal{M}\mathcal{N}_1} & K_{\mathcal{M}\mathcal{M}} & K_{\mathcal{M}\mathcal{I}} & K_{\mathcal{M}\mathcal{A}} & 0 & -\mathsf{M}_{\mathcal{I}}^T & -\mathsf{M}_{\mathcal{A}}^T \\ 0 & K_{\mathcal{I}\mathcal{M}} & K_{\mathcal{I}\mathcal{I}} & K_{\mathcal{I}\mathcal{A}} & K_{\mathcal{I}\mathcal{N}_2} & \mathsf{D}_{\mathcal{I}\mathcal{I}}^T & \mathsf{D}_{\mathcal{I}\mathcal{A}}^T \\ 0 & K_{\mathcal{A}\mathcal{M}} & K_{\mathcal{A}\mathcal{I}} & K_{\mathcal{A}\mathcal{A}} & K_{\mathcal{A}\mathcal{N}_2} & \mathsf{D}_{\mathcal{A}\mathcal{I}}^T & \mathsf{D}_{\mathcal{A}\mathcal{A}}^T \\ 0 & 0 & K_{\mathcal{N}_2\mathcal{I}} & K_{\mathcal{N}_2\mathcal{A}} & K_{\mathcal{N}_2\mathcal{N}_2} & 0 & 0 \\ \hdashline 0 & 0 & 0 & 0 & 0 & \mathsf{I} & 0 \\ 0 & \mathsf{N}_{\mathcal{M}} & \mathsf{N}_{\mathcal{I}} & \mathsf{N}_{\mathcal{A}} & 0 & 0 & 0 \\ 0 & 0 & \mathsf{F}_{\mathcal{I}} & \mathsf{F}_{\mathcal{A}} & 0 & 0 & \mathsf{T}_{\mathcal{A}} \end{array}\right) \left(\begin{array}{c} \Delta\mathbf{d}_{\mathcal{N}_1} \\ \Delta\mathbf{d}_{\mathcal{M}} \\ \Delta\mathbf{d}_{\mathcal{I}} \\ \Delta\mathbf{d}_{\mathcal{A}} \\ \Delta\mathbf{d}_{\mathcal{N}_2} \\ \Delta\lambda_{\mathcal{I}} \\ \Delta\lambda_{\mathcal{A}} \end{array}\right) = - \left(\begin{array}{c} \mathbf{r}_{\mathcal{N}_1} \\ \mathbf{r}_{\mathcal{M}} \\ \mathbf{r}_{\mathcal{I}} \\ \mathbf{r}_{\mathcal{A}} \\ \mathbf{r}_{\mathcal{N}_2} \\ \mathbf{r}_{\mathcal{I}}^{\lambda} \\ \mathbf{r}_{\mathcal{A}}^{\lambda,n} \\ \mathbf{r}_{\mathcal{A}}^{\lambda,\tau} \end{array}\right) \tag{4}$$

The dashed lines group all matrix blocks according to their physical meaning. The equations related to the balance of linear momentum of both solid bodies are collected above the horizontal dashed line, while the contact constraints are imposed via the three block rows below the horizontal dashed line. One can find matrices to be multiplied with the solid's displacement unknowns left of the vertical dashed line, while matrices to the right require multiplication with the Lagrange multiplier unknowns. At first glance, the upper left block is built from the stiffness matrices of both solid bodies, while a closer look reveals the presence of additional terms, for example, $K_{\mathcal{I}\mathcal{M}}$, $K_{\mathcal{M}\mathcal{I}}$, $K_{\mathcal{A}\mathcal{M}}$, and $K_{\mathcal{M}\mathcal{A}}$, due to linearizations of the solid residuals w.r.t. the Lagrange multiplier unknowns, which we will revisit later in Section 3.2 for the setup of a multigrid preconditioner.

The specific block structure of the matrix in (4) and in particular its saddle-point characteristics prevent the application of out-of-the-box preconditioners for iterative linear solvers. However, this block structure can be exploited in the design of a physics-based AMG preconditioner, which will be sketched out in Section 3.2.

3 | COMPUTATIONAL CHALLENGES

When going to large problems or very fine meshes, today's simulation toolchains resort to *parallel computing* to distribute the computational work to be executed in parallel on multiple computing cores. However, the evaluation of mortar integrals (cf. Section 2.2) and the solution of the linear system of equations (cf. Section 2.3) pose additional challenges in a parallel, distributed memory environment.

3.1 | Scalable evaluation of mortar integrals through dynamic load balancing

Evaluating the mortar integrals from (3) on a distributed memory hardware needs to overcome two obstacles: To begin with, since the interface information is distributed to multiple processes, contact search, mesh projection, and quadrature might need access to data from another parallel process. While one could store all interface data redundantly on all

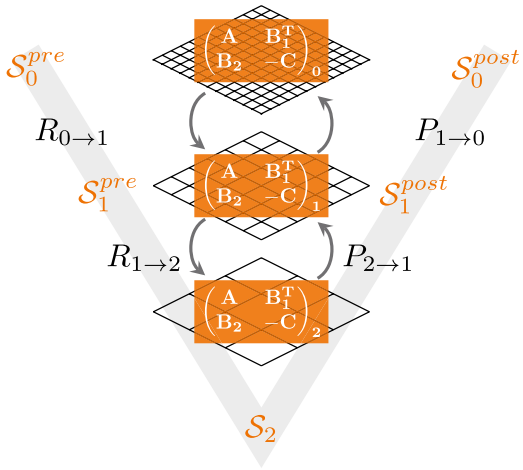

FIGURE 2 V-cycle of a three-level AMG hierarchy with transfer operators $P_{\ell+1\to\ell}$ and $R_{\ell\to\ell+1}$ and level smoothers $\mathcal{S}_\ell$ for a 2×2 block matrix arising in contact mechanics.

processes, we have proposed a geometrically motivated binning approach to determine the actual data to be ghosted, which ensures availability of all necessary data while limiting the amount of data to be copied to other processes [9]. Additionally, the contact interface is a $(d-1)$-dimensional manifold in the case of a d-dimensional problem, $d \in \{2,3\}$. While a well-balanced distribution of the bulk mesh via overlapping domain decomposition (ODD) to all available processes is standard in today's FEM solvers, the interface may not adopt the bulk distribution in order to avoid idling parallel processes. This curse of dimensionality can be overcome through an independent ODD of the slave interface. Moreover, changes in the contact zone or contact topology affect the balance of the interface ODD, possibly requiring multiple rebalancing steps throughout the course of a simulation. Therefore, we have recently proposed a dynamic load-balancing strategy for contact problems, where a rebalancing of the interface ODD is triggered dynamically as soon as an unacceptable imbalance among all participating processes has been detected [9]. At its core, the imbalance of the computational work among all participating parallel processes is measured. Since application engineers are interested in running their simulations as fast as possible, a practically meaningful and also straightforward to implement quantity for the imbalance is the time $t_{\text{eval},p}$ spent in the mortar evaluation on process p. If the ratio η_t of the measured imbalance between the "slowest" and "fastest" process exceeds a user-prescribed tolerance $\hat{\eta}_\text{t}$, reading

$$\eta_\text{t} = \frac{\max_p \left(t_{\text{eval},p}\right)}{\min_p \left(t_{\text{eval},p}\right)} > \hat{\eta}_\text{t},$$

the simulation triggers a rebalancing of the interface ODD dynamically at run time. Ultimately, this approach enables weak and strong scalability of the mortar evaluation and, thus, optimal use of all available hardware resources as shown in [9].

3.2 | Scalable iterative solvers through AMG preconditioning

When solving large systems of linear equations $Ax = b$ (such as in (4))—for example in the context of a semismooth Newton scheme to solve nonlinear contact problems—preconditioned Krylov solvers with multigrid preconditioning are among the most efficient solution methods (cf. [11]).

Multigrid methods use a hierarchy of coarse representations of the original fine-level problem and reconstruct the fine-level solution x from information from coarser levels (cf. Figure 2). Due to their flexibility to work with matrices arising from unstructured meshes, we employ aggregation-based AMG methods, which form coarse representations in a purely algebraic fashion based on the fine-level matrix A_0.

However, AMG cannot be applied out-of-the-box to the linear system in (4) due to the particular block structure of the matrix. Furthermore, classical block relaxation strategies, for example, block Gauss–Seidel, are inadmissible. In this

contribution, we follow our prior work [10] and carry the saddle-point characteristics of (4) through all levels of an AMG hierarchy. In order to make the AMG coarse grid correction as meaningful as possible, it must respect the underlying mechanics of contact, where both bodies can slide along each other along the contact interface. Relying on the greedy aggregation algorithms in Trilinos/MueLu [12], we need to filter the matrix to drop entries in the matrix graph that link the slave and master body in order to avoid aggregates that glue both bodies together on multigrid levels $\ell > 0$. The filtered matrix reads

$$\begin{pmatrix} K_{\mathcal{N}_1\mathcal{N}_1} & K_{\mathcal{N}_1\mathcal{M}} & 0 & 0 & 0 & 0 & 0 \\ K_{\mathcal{M}\mathcal{N}_1} & K_{\mathcal{M}\mathcal{M}} & \square & \square & 0 & -\mathsf{M}_{\mathcal{I}}^T & -\mathsf{M}_{\mathcal{A}}^T \\ 0 & \square & K_{\mathcal{I}\mathcal{I}} & K_{\mathcal{I}\mathcal{A}} & K_{\mathcal{I}\mathcal{N}_2} & \mathsf{D}_{\mathcal{I}\mathcal{I}}^T & \mathsf{D}_{\mathcal{I}\mathcal{A}}^T \\ 0 & \square & K_{\mathcal{A}\mathcal{I}} & K_{\mathcal{A}\mathcal{A}} & K_{\mathcal{A}\mathcal{N}_2} & \mathsf{D}_{\mathcal{A}\mathcal{I}}^T & \mathsf{D}_{\mathcal{A}\mathcal{A}}^T \\ 0 & 0 & K_{\mathcal{N}_2\mathcal{I}} & K_{\mathcal{N}_2\mathcal{A}} & K_{\mathcal{N}_2\mathcal{N}_2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathsf{I} & 0 \\ 0 & \mathsf{N}_{\mathcal{M}} & \mathsf{N}_{\mathcal{I}} & \mathsf{N}_{\mathcal{A}} & 0 & 0 & 0 \\ 0 & 0 & \mathsf{F}_{\mathcal{I}} & \mathsf{F}_{\mathcal{A}} & 0 & 0 & \mathsf{T}_{\mathcal{A}} \end{pmatrix} \begin{pmatrix} \Delta\mathbf{d}_{\mathcal{N}_1} \\ \Delta\mathbf{d}_{\mathcal{M}} \\ \Delta\mathbf{d}_{\mathcal{I}} \\ \Delta\mathbf{d}_{\mathcal{A}} \\ \Delta\mathbf{d}_{\mathcal{N}_2} \\ \Delta\lambda_{\mathcal{I}} \\ \Delta\lambda_{\mathcal{A}} \end{pmatrix} = - \begin{pmatrix} \mathbf{r}_{\mathcal{N}_1} \\ \mathbf{r}_{\mathcal{M}} \\ \mathbf{r}_{\mathcal{I}} \\ \mathbf{r}_{\mathcal{A}} \\ \mathbf{r}_{\mathcal{N}_2} \\ \mathbf{r}_{\mathcal{I}}^{\lambda} \\ \mathbf{r}_{\mathcal{A}}^{\lambda,n} \\ \mathbf{r}_{\mathcal{A}}^{\lambda,\tau} \end{pmatrix}, \tag{5}$$

where we have marked dropped blocks by squares. Then, we need to define aggregates of the Lagrange multiplier unknowns. Due to the saddle-point characteristics, the bottom right block in the matrix cannot be used as input to the aggregation process, but we rather construct aggregates for the Lagrange multipliers that match those of the underlying slave body along the contact interface. Through segregated transfer operators, we can preserve the matrix structure throughout all multigrid levels. For smoothing, we resort to a SIMPLE-type block smoother, originally proposed in the context of fluid dynamics [13].

3.2.1 | A holistic approach to reducing the time to solution

In our prior work [9, 10], we have addressed the computational efficiency of mortar-based contact problems by trying to isolate individual aspects of the numerical effort through the approaches outlined in the previous Sections 3.1 and 3.2. While each approach individually enables parallel scalability for a part of a contact simulation, optimal performance can only be achieved through the combination and simultaneous application of both approaches. Challenges arise from the changes in the ODD due to dynamic load balancing, which requires to carefully track and update the parallel data layout of the contact interface for the AMG preconditioner.

While we believe that such a combined approach ultimately will be key to reduce the total time to solution, we are currently working on enabling the simultaneous use of dynamic load balancing and AMG preconditioning for contact problems with the long-term goal of reducing the total time to solution.

4 | NUMERICAL EXAMPLE

As a proof of concept, we study the rolling motion of a cylinder on a flat table, mimicking technical systems such as tire/road contact or cylindrical roller bearings. Due to the rolling motion, the contact zone changes constantly and moves along the circumference of the cylinder. Without loss of generality, we assume frictionless contact.

The motion of the cylinder is imposed via a Dirichlet boundary condition on the inner surface of the hollow cylinder. Both bodies are modeled with a compressible Neo-Hooke material with Young's modulus $E = 1$, Poisson's ratio $\nu = 0.3$, and density $\rho = 10^{-6}$. We use purely displacement-based first-order hexahedral finite elements to discretize the solid domain, resulting in approximately 27 000 primal degrees of freedom (DOF). The outer surface of the hollow cylinder is chosen as the slave side $\Gamma_{*,\mathrm{h}}^{\mathrm{sl}}$ of the contact interface, while the top surface of the block takes the role of the master surface. Implicit time integration is performed with the generalized-α method [14] with a spectral radius $\rho_\infty = 0.8$ and a time step size of $\Delta t = 0.02$.

The simulation starts with a gap $g_n > 0$ between the slave and master interface. Then, the cylinder is pushed onto the initially flat table and, after contact has been established, the rotating motion is prescribed on the inner side of the cylinder. The cylinder executes at least a half rotation to ensure a significant change in the contact zone and demonstrate

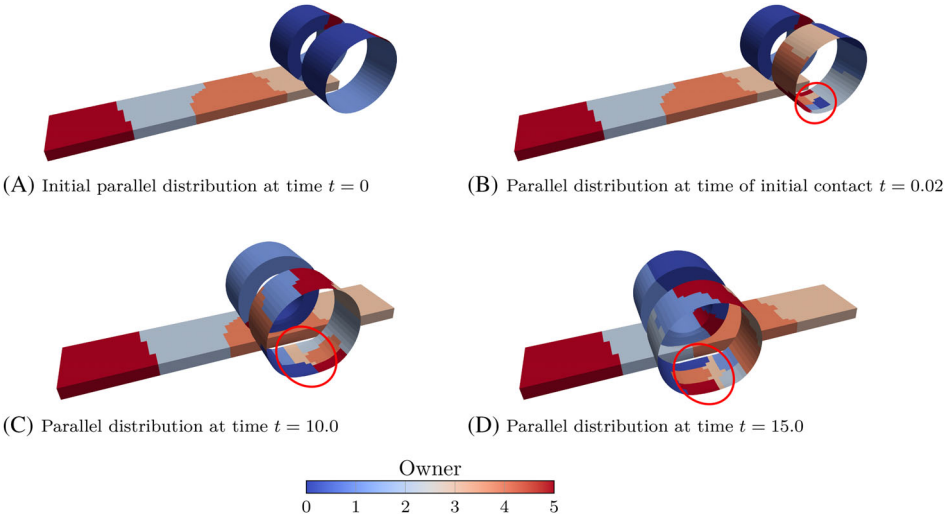

FIGURE 3 Visualization of the distribution of MPI ranks of the bulk and interface discretization.

the dynamic load-balancing methods from Section 3.1. The parallel redistribution of the contact interface is triggered, if the threshold $\hat{\eta}_t = 1.8$ for the imbalance in evaluation time is exceeded.

The stopping criteria for the nonlinear solver within each time step is reached, if the length-scaled 2-norm of the residual of both the primal and the dual variable fulfills

$$\frac{\left\|\mathbf{r}_i^u\right\|_2}{\sqrt{n_u^{\text{dof}}}} \leq 10^{-6} \wedge \frac{\left\|\mathbf{r}_i^\lambda\right\|_2}{\sqrt{n_\lambda^{\text{dof}}}} \leq 10^{-6} \tag{6}$$

in nonlinear iteration step i, where n_u^{dof} and n_λ^{dof} denote the number of displacement and Lagrange multiplier unknowns. Within each nonlinear step i, the linear system is solved in saddle-point form with a preconditioned GMRES solver [15]. The convergence criterion for the iterative linear system is chosen as

$$\frac{\left\|\mathbf{r}^k\right\|_2}{\left\|\mathbf{r}^0\right\|_2} \leq 10^{-6} \tag{7}$$

for the full residual vector ${\mathbf{r}^k}^T = \left[\mathbf{r}^u, \mathbf{r}^\lambda\right]$ in the linear iteration step k. Here, the subscript i for the nonlinear Newton iteration has been dropped.

The GMRES solver is preconditioned with the AMG preconditioner outlined in Section 3.2. The segregated transfer operators use smoothed-aggregation AMG for the solid DOFs and plain-aggregation AMG for the Lagrange multiplier unknowns. As a level smoother for the block matrix, we apply a single sweep of CheapSIMPLEC with a damping parameter $\omega_S = 0.8$, which internally applies three iterations of a damped symmetric Gauss–Seidel ($\omega = 0.8$) to both the prediction and the correction step. The same level smoother layout is used on all levels except the coarsest, for which a distributed-memory sparse direct solver is applied. Even though the problem size does not necessarily require a three-level V-cycle multigrid design, we deliberately set the minimum number of multigrid levels to three to demonstrate the framework's support of a true multigrid preconditioner.

Using the message passing interface (MPI) to address parallelism in the distributed memory paradigm, Figure 3 shows the time course of the deformation of the system as well as the distribution of the ODD of the bulk and interface domains for the case of 6 MPI ranks. While the initial interface ODD is adopted from the ODD of the underlying bulk field (cf. Figure 3A), the onset of contact triggers a first rebalancing of the interface ODD to distribute the work of mortar evaluation to all available MPI ranks (cf. Figure 3B). The recurring rebalancing due to the rolling motion becomes obvious from Figure 3C to Figure 3D.

A representative time step is chosen to showcase the performance of the preconditioned GMRES solver within each semismooth Newton iteration. Table 1 exemplarily shows the number of linear iterations needed for each of the three nonlinear iterations until convergence in time step 500. The number of nonlinear iterations throughout the simulation

TABLE 1 Number of GMRES iterations in each nonlinear iteration of time step 500 ($t = 10.0$; see Figure 3C).

Number of nonlinear iteration	Number of GMRES iterations
1	10
2	12
3	10

stays smaller than five, with the GMRES solver continuously converging within a maximum of 14 iterations. The AMG preconditioner constructs three levels, with 29 085 rows on the fine level, 2424 rows on level 1, and 162 rows on the coarse level for the third nonlinear iteration in time step 500, for example.

5 | CONCLUDING REMARKS

In this contribution, we have outlined computational challenges of computational contact mechanics on parallel computing clusters. While the two most pressing bottlenecks—evaluation of mortar terms and solution of the linear system—have already been addressed individually [9, 10], we now demonstrate the possibility of their combined application with the goal of an overall speed-up of the computation. A proof of concept shows the ability to apply both approaches simultaneously, yet a detailed performance analysis of the combined approach is left for future work.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the computing time provided by the Data Science & Computing Lab at the University of the Bundeswehr Munich.

Open access funding enabled and organized by Projekt DEAL.

ORCID

Christopher Steimer https://orcid.org/0009-0000-1848-657X
Matthias Mayr https://orcid.org/0000-0002-2780-1233
Alexander Popp https://orcid.org/0000-0002-8820-466X

REFERENCES

1. Fischer, K. A., & Wriggers, P. (2006). Mortar based frictional contact formulation for higher order interpolations using the moving friction cone. *Computer Methods in Applied Mechanics and Engineering*, *195*(37), 5020–5036.
2. Popp, A., Gitterle, M., Gee, M. W., & Wall, W. A. (2010). A dual mortar approach for 3D finite deformation contact with consistent linearization. *International Journal for Numerical Methods in Engineering*, *83*(11), 1428–1465.
3. Popp, A., Wohlmuth, B. I., Gee, M. W., & Wall, W. A. (2012). Dual quadratic mortar finite element methods for 3D finite deformation contact. *SIAM Journal on Scientific Computing*, *34*(4), B421–B446.
4. Puso, M. A., & Laursen, T. A. (2004). A mortar segment-to-segment contact method for large deformation solid mechanics. *Computer Methods in Applied Mechanics and Engineering*, *193*(6–8), 601–629.
5. Wohlmuth, B. (2011). Variationally consistent discretization schemes and numerical algorithms for contact problems. *Acta Numerica*, *20*, 569–734.
6. Lorenzis, L. D., Wriggers, P., & Hughes, T. J. (2014). Isogeometric contact: A review. *GAMM-Mitteilungen*, *37*(1), 85–123.
7. Mayr, M., Klöppel, T., Wall, W. A., & Gee, M. W. (2015). A temporal consistent monolithic approach to fluid-structure interaction enabling single field predictors. *SIAM Journal on Scientific Computing*, *37*(1), B30–B59.
8. Fang, R., Farah, P., Popp, A., & Wall, W. A. (2018). A monolithic, mortar-based interface coupling and solution scheme for finite element simulations of lithium-ion cells. *International Journal for Numerical Methods in Engineering*, *114*(13), 1411–1437.
9. Mayr, M., & Popp, A. (2023). Scalable computational kernels for mortar finite element methods. *Engineering with Computers*. Advance online publication. https://doi.org/10.1007/s00366-022-01779-3
10. Wiesner, T. A., Mayr, M., Popp, A., Gee, M. W., & Wall, W. A. (2021). Algebraic multigrid methods for saddle point systems arising from mortar contact formulations. *International Journal for Numerical Methods in Engineering*, *122*(15), 3749–3779.
11. Saad, Y. (2003). *Iterative methods for sparse linear Systems*. SIAM.
12. Berger-Vergiat, L., Glusa, C. A., Harper, G., Hu, J. J., Mayr, M., Prokopenko, A., Siefert, C. M., Tuminaro, R. S., & Wiesner, T. A. (2023). *MueLu user's guide* (Tech. Rep. SAND2023-12265). Sandia National Laboratories.

13. Patankar, S., & Spalding, D. (1972). A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. *International Journal of Heat and Mass Transfer*, *15*(10), 1787–1806.
14. Chung, J., & Hulbert, G. (1993). A time integration algorithm for structural dynamics with improved numerical dissipation: The generalized-α method. *Journal of Applied Mechanics*, *60*(2), 371–375.
15. Saad, Y., & Schultz, M. H. (1986). GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, *7*(3), 856–869.

How to cite this article: Steimer, C., Mayr, M., & Popp, A. (2023). Targeting a faster time-to-solution of mortar-based contact problems. *Proceedings in Applied Mathematics and Mechanics*, *23,* e202300157. https://doi.org/10.1002/pamm.202300157