NONLINEAR FETI-DP AND BDDC METHODS∗
AXEL KLAWONN†, MARTIN LANSER†, AND OLIVER RHEINBACH‡

Abstract. New nonlinear FETI-DP (dual-primal finite element tearing and interconnecting) and BDDC (balancing domain decomposition by constraints) domain decomposition methods are introduced. In all these methods, in each iteration, local nonlinear problems are solved on the subdomains. The new approaches can significantly reduce communication and show a significantly improved performance, especially for problems with localized nonlinearities, compared to a standard Newton–Krylov–FETI-DP or BDDC approach. Moreover, the coarse space of the nonlinear FETI-DP methods can be used to accelerate the Newton convergence. It is also found that the new nonlinear FETI-DP and nonlinear BDDC methods are not as closely related as in the linear context. Numerical results for the p-Laplace operator are presented.

Key words. nonlinear FETI-DP, nonlinear BDDC, Newton–Krylov, nonlinear domain decomposition, p-Laplace

AMS subject classifications. 65N55, 65F08, 65F10, 65Y05

DOI. 10.1137/130920563

1. Introduction. The traditional domain decomposition approach for the solution of nonlinear problems resulting from the discretization of nonlinear partial differential equations, can be characterized by a geometric decomposition after linearization. There, we solve a given discrete nonlinear problem

\[ A(u) = 0 \]

by using a Newton-type method \( u^{(k+1)} = u^{(k)} - \alpha^{(k)} \delta u^{(k)} \) with a suitable step length \( \alpha^{(k)} \). Newton methods are often used because of their fast convergence in the neighborhood of the solution. Of course, in principle, the nonlinear methods in this paper can also be formulated using any other nonlinear solution method. In our Newton methods, in each iteration, we have to solve the linearized system \( DA(u^{(k)}) \delta u^{(k)} = A(u^{(k)}) \), where \( DA(u^{(k)}) \) is the Jacobian or tangential matrix of \( A \) in the \( k \)th iteration step.

In the context of domain decomposition methods, this can be done by overlapping or nonoverlapping algorithms, e.g., finite element tearing and interconnecting (FETI-1), dual-primal finite element tearing and interconnecting (FETI-DP), balancing domain decomposition (DD) by constraints (BDDC), or overlapping Schwarz. Such approaches are typically named Newton–Krylov (NK) DD (NK-DD), e.g., NK-FETI-DP or NK-Schwarz.

∗Submitted to the journal’s Methods and Algorithms for Scientific Computing section May 10, 2013; accepted for publication (in revised form) January 6, 2014; published electronically April 15, 2014. This work was supported in part by the German Research Foundation (DFG) through the Priority Programme 1648 Software for Exascale Computing (SPPEXA). A preliminary version has been already presented in On an adaptive coarse space and on nonlinear domain decomposition in Proceedings of the 21st International Conference on Domain Decomposition Methods in Science and Engineering, Rennes, France, 2013 [26]. The present article has been completely revised and significantly extended.

†Mathematisches Institut, Universität zu Köln, Weyertal 86-90, 50931 Köln, Germany (axel.klawonn@uni-koeln.de, martin.lanser@uni-koeln.de).

‡Fakultät für Mathematik und Informatik, Institut für Numerische Mathematik und Optimierung, Technische Universität Bergakademie Freiberg, 09596 Freiberg, Germany (oliver.rheinbach@math.tu-freiberg.de).
Alternative approaches to the traditional domain decomposition approach can be characterized by linearization after a geometric decomposition (here denoted as DD-NK, i.e., FETI-DP-NK). Such methods can be interpreted also in the context of nonlinear preconditioning, as in the ASPIN (additive Schwarz preconditioned inexact Newton) approach; see Cai and Keyes [4] and Knoll and Keyes [36], where a survey on Jacobian-free Newton methods is given. In ASPIN, instead of solving (1.1) directly by a Newton method, a nonlinear equation $G(A(u)) = 0$ is solved. The nonlinear preconditioner $G$ is constructed from a nonlinear additive Schwarz (AS) method. The ASPIN approach can be classified as an AS-NK method and has been shown to be more robust than the traditional NK-AS approach and highly scalable, even for flow problems with high Reynolds numbers. The ASPIN method can be equipped with a linear or a nonlinear coarse problem [5, 24, 25]. Recently, the ASPIN approach has successfully been applied to problems from nonlinear structural mechanics; see Gross and Krause [23]. For a recursive trust region globalization framework applied to a nonlinear Schwarz DD; see [22]. Nonlinear Schwarz methods as a solver, i.e., not as a preconditioner, have already been considered much earlier; see, e.g., [3, 13].

In the following, we will present new approaches for nonoverlapping, nonlinear DD methods, i.e., versions of nonlinear FETI-DP and BDDC methods. Some preliminary results were already presented in a proceedings paper [26]. Nonlinear nonoverlapping DD methods have already successfully been used, for the special case of two subdomains, in multiphysics coupling, e.g., in fluid-structure interaction; see Deparis et al. [9, 10] and Deparis [8]. See also the discussion and references in Fernandez et al. [20] on “linearize first” versus “decompose first” in strongly coupled, partitioned fluid-structure-interaction algorithms.

As in FETI-DP [15, 16, 30, 31, 34, 35] and BDDC [7, 11, 37, 40, 41] DD methods for linear or linearized problems, the coarse spaces for our nonlinear methods are constructed from partial assembly of the finite elements. Let us note here that the coarse problem of the nonlinear FETI-DP method is incorporated into the nonlinear operator and not into the preconditioner. We will see later that the choice of the coarse space will therefore directly affect the Newton iteration rather than only the Krylov space iteration.

The standard FETI-DP and BDDC methods were derived from the earlier FETI-1 [1, 17, 18, 19, 33] and balancing Neumann–Neumann [14, 33, 39, 48] iterative substructuring methods; see also Toselli and Widlund [48]. Recently, a Nonlinear-FETI-1 DD approach was introduced by Pebrel, Rey, and Gosselet [45]. As in linear FETI-1 methods the coarse problem of this method is formed by a projection and thus is linear. The coarse problem of our nonlinear FETI-DP methods will be nonlinear. A nonlinear Neumann–Neumann method was then introduced by Bordas, Boucard, and Gosselet [2].

One important building block of our nonlinear FETI-DP and BDDC methods is the solution of (weakly coupled) nonlinear problems on the subdomains. Such an algorithmic step has been denoted nonlinear localization or nonlinear relocalization and is, of course, also a building block of nonlinear FETI-1 methods [45] and nonlinear Neumann–Neumann methods [2]. Nonlinear localization has also been proposed as an enhancement step for NK methods; see Cresta et al. [6]. A simple linear/nonlinear strategy, which may be seen as nonlinear localization, was used in [38] for brittle materials with strong localized nonlinearities.

We will discuss two different strategies of nonlinear dual primal FETI methods, named the Nonlinear-FETI-DP-1 method (linearization first) and Nonlinear-FETI-DP-2 method (elimination first).
The remainder of the article is organized as follows. First, we present the general idea of nonlinear DD in section 2 and after that we introduce our two nonlinear FETI-DP methods in section 3. We also describe how to implement a coarse space by transformation of basis in the case of the Nonlinear-FETI-DP methods. In section 4, we present our nonlinear methods in an algorithmic way and analyze typical computational cost. In section 5, we define a nonlinear BDDC method and, finally, in section 6, we present some numerical results for problems in two dimensions involving the p-Laplace operator. Here, we compare our three algorithms to the classical NK-FETI-DP approach.

2. Nonlinear domain decomposition. Let $\Omega_i, i = 1, \ldots, N$, be a decomposition of the domain $\Omega \subset \mathbb{R}^d$, $d = 2, 3$, into nonoverlapping subdomains. Each subdomain is a union of finite elements. We denote the associated local finite element spaces by $W_i$ and the product space by $W = W_1 \times \cdots \times W_N$, and we define $\bar{W} \subset W$ as the subspace of functions from $W$, which are continuous in all interface variables between subdomains. We denote by $R_i : V^h \rightarrow W_i$, $i = 1, \ldots, N$, local restriction operators where $V^h = V^h(\Omega)$ is the space of globally assembled finite element functions. Note that $V^h$ and $\bar{W}$ are isomorphic. These notations are standard in the theory of linear FETI-DP methods; see, e.g., [27, 34, 35, 48]. In our notation we will not distinguish between finite element functions and the corresponding vectors of nodal values. The same holds for the discrete operators and their corresponding matrix representations.

Instead of the solution of the discrete nonlinear problem (1.1), we consider the minimization of the related nonlinear energy functional $J : V^h \rightarrow \mathbb{R}$,

\[
\min_{\bar{u} \in V^h} \ J(\bar{u}).
\]

We make the following assumption.

Assumption 1. There exist local energy functionals $J_i : W_i \rightarrow \mathbb{R}$, $i = 1, \ldots, N$, such that for $\bar{u} \in V^h$, the global energy functional can be represented as a sum of local energies

\[
J(\bar{u}) = \sum_{i=1}^{N} J_i(u_i),
\]

where $u_i := R_i \bar{u}$.

Assumption 1 is satisfied for the p-Laplace problem that we will consider in this paper as a model problem as well as for relevant nonlinear problems such as standard or incompressible hyperelasticity. In general, Assumption 1 is satisfied for standard problems discretized by finite elements. Assumption 1 then simply follows from the additivity of the integral. It may not be satisfied for problems with nonlocal phenomena such as nonlocal damage models in structural mechanics.

Next, we introduce a linear, discrete jump operator $B = (B_1, \ldots, B_N)$ which enforces continuity across the interface $\Gamma$ given as the union of the interior subdomain boundaries. Here, for $u = (u_1^T, \ldots, u_N^T)^T$ with $u_i \in W_i$, $Bu = 0$ corresponds to $u \in \bar{W}$.

Using Assumption 1 and the notation $u = (u_1^T, \ldots, u_N^T)^T \in W$, we obtain

\[
\min_{\bar{u} \in V^h} J(\bar{u}) = \min_{u_i \in V^h} \sum_{i=1}^{N} J_i(u_i) = \min_{B\bar{u} = 0} \sum_{i=1}^{N} J_i(u_i).
\]
We introduce the space of Lagrange multipliers $V := \text{range}(B)$. Now, it is possible to derive a nonlinear saddle point problem.

We have to compute the stationary points of the Lagrange function

$$L : W \times V \rightarrow \mathbb{R},$$

which are the solutions of the equation

$$\begin{pmatrix}
\sum_{i=1}^{N} J_i'(u_i)(v_i) + (Bv)^{T}\lambda \\
(Bu)^{T}\mu
\end{pmatrix} =
\begin{pmatrix}
0 \\
0
\end{pmatrix} \quad \forall v \in W, \forall \mu \in V.$$

For each $i = 1, \ldots, N$, we denote by $\varphi_{i,j}$, $j = 1, \ldots, N$, the nodal finite element basis functions for the local finite element space $W_i$. We assume that for each $i = 1, \ldots, N$, we have the following representation $J_i'(u_i)(\varphi_{i,j}) = (K_i(u_i) - f_i)_j$. Here, $f_i$ is independent of $u_i$. With the notation

$$K(u) := \begin{pmatrix}
K_1(u_1) \\
\vdots \\
K_N(u_N)
\end{pmatrix}, \quad f := \begin{pmatrix}
f_1 \\
\vdots \\
f_N
\end{pmatrix}, \quad \text{and} \quad u := \begin{pmatrix}
u_1 \\
\vdots \\
u_N
\end{pmatrix},$$

we obtain from (2.4) the discrete nonlinear system of equations

$$K(u) + B^{T}\lambda = f,$$

$$Bu = 0,$$

which can be seen as a nonlinear analog of the linear FETI master system.

We formulate our algorithms based on Newton linearization to obtain fast convergence close to the solution. In our numerical experiments, we often use a line search with a step length satisfying the Wolfe conditions; see [43]. For a minimizing problem $\min_{x \in \mathbb{R}^n} J(x)$ and a search direction $\delta x$ the Wolfe conditions read

$$J(x + \alpha \delta x) \leq J(x) + c_1 \alpha \delta x^T \nabla J(x)$$

and

$$\delta x^T \nabla J(x + \alpha \delta x) \geq c_2 \delta x^T \nabla J(x)$$

with $0 < c_1 < c_2 < 1$.

3. Nonlinear FETI-DP methods. After partitioning all interface variables in a set of primal variables $\Pi$ and a set of dual variables $\Delta$, we can define the standard FETI-DP partial assembly operator $R^{T}_{\Pi}$ (see [30] for the notation) and the FETI-DP jump operator $B$ in the standard form. Note that here, as opposed to (2.6), $B$ only enforces continuity in the dual variables $\Delta$.

To construct our nonlinear FETI-DP methods we use partial assembly of $K_i(u_i)$ in the primal variables. All of our nonlinear FETI-DP methods are then based on the following nonlinear FETI-DP master system

$$R^{T}_{\Pi} K(R_{\Pi} \hat{u}) + B^{T}\lambda - R^{T}_{\Pi} f = 0,$$

$$B\hat{u} = 0,$$
where \( \tilde{u} \in \tilde{W} \), and the Lagrange multipliers \( \lambda \in V \). We denote the space of finite element functions in \( W \), which are continuous in all primal variables, by \( \tilde{W} \). We will also write (3.1) in compact form
\[
\tilde{K}(\tilde{u}) + B^T \lambda - \tilde{f} = 0, \\
B\tilde{u} = 0,
\]
where
\[
\tilde{K}(\tilde{u}) := R_H^T K(R_H\tilde{u})
\]
and \( \tilde{f} := R_H^T f \). From the chain rule, we have
\[
D\tilde{K}(\tilde{u}) := D(\tilde{K}(\tilde{u})) = D(R_H^T K(R_H\tilde{u})) = R_H^T DK(R_H\tilde{u}) R_H,
\]
i.e., the derivative of the partially assembled nonlinear operator \( \tilde{K} \) can be computed by assembling \( DK \), i.e., by assembling the derivatives of the local subdomain operators \( K_i \).

We can proceed in two different ways in order to solve (3.1). We may linearize first and then reduce the result to Lagrange multipliers (Nonlinear-FETI-DP-1), or, by the help of the inverse function theorem, we can perform a nonlinear elimination and then linearize the reduced nonlinear system (Nonlinear-FETI-DP-2). Both variants will be discussed in the remainder of this paper in sections 3.2 and 3.4.

3.1. Additional assumptions. First, let us note, that our nonlinear FETI-DP methods are identical to standard FETI-DP methods if they are applied to linear problems. Moreover, nonlinear FETI-DP methods using a transformation of basis are also identical to standard FETI-DP methods using a transformation of basis if applied to linear problems.

Second, as in the ASPIN method [4], the equivalence of the original problem (2.1) to the nonlinear FETI-DP formulations (3.1) and (3.14) relies on certain assumptions which we very briefly discuss now.

To ensure the equivalence of the original problem (2.1) and the saddle point system (2.6), Assumption 1 has to be satisfied. From the saddle point system (2.6) we will obtain the saddle point system (3.1) by replacing some of the dual constraints by primal constraints implemented by partial assembly. This does not essentially change the solution. Let us now assume that (3.1) has a solution \( (\tilde{u}^*, \lambda^*) \). Then, from the first set of equations of (3.1), we have \( \tilde{K}(\tilde{u}^*) = \tilde{f} - B^T \lambda^* \). We make the following assumption.

**Assumption 2.** Let \( U \) be an open neighborhood of \( \tilde{u}^* \). The function \( \tilde{K} \) is continuously differentiable in \( U \). The derivative \( DK(\tilde{u}^*) \) of \( K \) in \( \tilde{u}^* \) is a regular matrix.

Under Assumption 2 and using the inverse function theorem, there exist some neighborhoods \( U_\alpha \subset U \) with \( \tilde{u}^* \in U_\alpha \) and \( V_\alpha \) with \( \tilde{f} - B^T \lambda^* \in V_\alpha \), such that \( \tilde{K} : U_\alpha \rightarrow V_\alpha \) is a diffeomorphism. Under these assumptions, \( \tilde{u}^* \) is a point of attraction of the Newton method for \( \tilde{K}(\tilde{u}^*) + B^T \lambda^* - \tilde{f} = 0 \); see, e.g., Ortega and Rheinboldt [44, section 10.2.2]. An analogous result holds for the Nonlinear-FETI-DP-1 system (3.2) and the Nonlinear-FETI-DP-2 system (3.14) if, in addition, the saddle point system on the left-hand side of (3.5) is nonsingular for \( (\tilde{u}^*, \lambda^*) \); this is satisfied if \( B \) has full rank.

3.2. Nonlinear-FETI-DP-1 method (linearize first). We now consider the first approach, denoted by Nonlinear-FETI-DP-1, in which we first linearize the system (3.1) and then reduce the resulting linear system to the space of Lagrange multipliers.
Newton linearization with respect to \((\tilde{u}, \lambda)\) of the saddle point problem (3.2) results in the linear system

\[
\begin{pmatrix}
    D\tilde{K}(\tilde{u}^{(k)}) & B^T \\
    B & 0
\end{pmatrix}
\begin{pmatrix}
    \delta\tilde{u}^{(k)} \\
    \delta\lambda^{(k)}
\end{pmatrix} =
\begin{pmatrix}
    \tilde{K}(\tilde{u}^{(k)}) + B^T\lambda^{(k)} - \tilde{f} \\
    B\tilde{u}^{(k)}
\end{pmatrix}.
\]

Here, \(\tilde{u}^{(0)} \in \widetilde{W}\) and \(\lambda^{(0)} \in V\) are initial values, and our Newton iteration to solve problem (3.1) is defined by

\[
\begin{pmatrix}
    \tilde{u}^{(k+1)} \\
    \lambda^{(k+1)}
\end{pmatrix} =
\begin{pmatrix}
    \tilde{u}^{(k)} \\
    \lambda^{(k)}
\end{pmatrix} - \alpha^{(k)}
\begin{pmatrix}
    \delta\tilde{u}^{(k)} \\
    \delta\lambda^{(k)}
\end{pmatrix}
\]

with a suitable step length \(\alpha^{(k)}\). The linear system (3.5) can be solved as in the standard (linear) FETI-DP framework, i.e., we can eliminate the variables \(\delta\tilde{u}^{(k)}\) by a step of block Gauss elimination under the assumption that \(D\tilde{K}\) is regular. This yields

\[
\begin{pmatrix}
    D\tilde{K}(\tilde{u}^{(k)}) & B^T \\
    0 & -B(D\tilde{K}(\tilde{u}^{(k)}))^{-1}B^T
\end{pmatrix}
\begin{pmatrix}
    \delta\tilde{u}^{(k)} \\
    \delta\lambda^{(k)}
\end{pmatrix} =
\begin{pmatrix}
    f_{NL1}(\tilde{u}^{(k)}, \lambda^{(k)}) \\
    d_{NL1}(\tilde{u}^{(k)}, \lambda^{(k)})
\end{pmatrix},
\]

where

\[
\begin{align*}
    f_{NL1}(\tilde{u}, \lambda) &= \tilde{K}(\tilde{u}) + B^T\lambda - \tilde{f}, \\
    d_{NL1}(\tilde{u}, \lambda) &= -B\tilde{u} + B(D\tilde{K}(\tilde{u}))^{-1}(\tilde{K}(\tilde{u}) + B^T\lambda - \tilde{f}).
\end{align*}
\]

The second row in (3.7) yields the FETI-DP system. As in the case of standard linear FETI-DP, we have reduced the problem of computing the update in (3.6) to the solution of a linear system operating on the Lagrange multipliers, and it remains to solve

\[
F_{NL1}(\tilde{u}^{(k)})\delta\lambda^{(k)} = d_{NL1}(\tilde{u}^{(k)}, \lambda^{(k)}),
\]

where

\[
F_{NL1}(\tilde{u}) := B(D\tilde{K}(\tilde{u})^{-1})B^T.
\]

This system can be solved by a preconditioned Krylov iteration using one of the standard FETI-DP preconditioners \(M^{-1}\), e.g., the Dirichlet preconditioner [48]. Note that here continuity of the solution may not be reached until convergence of the Newton iteration. This is different from the NK-FETI-DP method where each Newton iterate is continuous.

### 3.3. Choosing initial values for the Nonlinear-FETI-DP-1 method.

A suitable initial value \(\tilde{u}^{(0)}\) for Nonlinear-FETI-DP-1 has to be continuous in all primal variables and should provide a good local approximation of the given problem. But it may be discontinuous in the dual variables. A possible choice of an initial value \(\tilde{u}^{(0)}\) can be obtained from solving the nonlinear problem

\[
\tilde{K}(\tilde{u}^{(0)}) = \tilde{f} - B^T\lambda^{(0)}
\]

by some Newton-type iteration for some given initial value \(\lambda^{(0)}\). Throughout this paper we use \(\lambda^{(0)} = 0\). This step can be seen as a nonlinear localization step. It
can thus be performed not only in the initialization but also during the iteration. Nonlinear localization was already used in [6] and, of course, in [2, 45]. Note that in (3.12) we solve local nonlinear subdomain problems which are only coupled in the primal unknowns. This step thus requires only communication in the primal variables and is otherwise completely parallel. More sophisticated choices for \(\lambda^{(0)}\), especially for hard problems as, e.g., incompressible hyperelasticity and problems with strong local nonlinearities, are also possible.

3.4. Nonlinear-FETI-DP-2 method (eliminate first). Let us now consider the second approach denoted by Nonlinear-FETI-DP-2. Instead of linearizing the nonlinear saddle point problem (3.1), we may first perform a nonlinear elimination of the variables \(\tilde{u}\). From (3.3), we recall

\[
\tilde{K}(\tilde{u}) = R_N K(R_N \tilde{u}).
\]

Assuming that \(\tilde{K}(\tilde{u})\) is locally invertible (see section 3.1 for more details) the first equation of (3.1) can be written as

\[
\tilde{u} = \tilde{K}^{-1}(\tilde{f} - B^T \lambda),
\]

where \(\tilde{K}^{-1}\) is the inverse operator of \(\tilde{K}\). Inserting (3.13) into the continuity condition in (3.1), we obtain

\[
F(\lambda) := B\tilde{K}^{-1}(\tilde{f} - B^T \lambda) = 0.
\]

To be able to apply the inverse function theorem, we assume that \(\tilde{K}(\cdot)\) is continuously differentiable in a neighborhood of \(\tilde{u}^*\) where \((\tilde{u}^*, \lambda^*)\) is a solution of (3.1). For more details, see Assumption 2 in section 3.1. Again, we use a Newton-type iteration to solve (3.14), and obtain the iteration

\[
\lambda^{(k+1)} = \lambda^{(k)} - \alpha^{(k)} (D_{\lambda} F(\lambda^{(k)}))^{-1} F(\lambda^{(k)}).
\]

We compute \(D_{\lambda} F(\lambda)\) using the chain rule, the inverse function theorem, and (3.13):

\[
D_{\lambda} F(\lambda) = D_{\lambda}(B\tilde{K}^{-1}(\tilde{f} - B^T \lambda)) = -B(D\tilde{K}^{-1}(\tilde{f} - B^T \lambda))B^T
\]

\[
= -B(D\tilde{K}(\tilde{u}))^{-1}B^T = -B(R_N^T(DK(R_N \tilde{u}))R_N)^{-1}B^T.
\]

Note that, since the coarse space is included in the nonlinear operator \(F\), and choosing different primal conditions leads to different nonlinear operators \(F\), the choice of the coarse space not only affects the Krylov subspace iteration but also the convergence of the Newton scheme. The coarse space can thus be used to improve the convergence of the Newton iteration for nonlinear problems. The nonlinear coarse problem introduces a coupling of the local nonlinear problems on the subdomains. A good coarse problem should be small but at the same time the resulting coupling should lead to a good approximation of the original global nonlinear problem. We also present some numerical results in Tables 3 and 4, which show the effect of different coarse spaces on the number of Newton and Krylov iterations. A detailed description of the associated model problem and the two different coarse spaces can be found in section 5.2.3.

Remark, that the pair \((\tilde{u}, \lambda)\) in (3.16) has to satisfy (3.13), such that we can use the inverse function theorem. We may now define

\[
F_{NL2}(\tilde{u}) := B(D\tilde{K}(\tilde{u}))^{-1}B^T.
\]

We have \(F_{NL2}(\tilde{u}) = -D_{\lambda} F(\lambda)\) and \(F(\lambda) = B\tilde{u}\), if the pair \((\tilde{u}, \lambda)\) fulfills (3.13).
In each Newton step, to compute the right-hand side, we first have to solve the nonlinear system

\[(3.18) \quad \tilde{K}(\tilde{u}^{(k)}) = \tilde{f} - B^T\lambda^{(k)},\]

e.g., by some Newton-type iteration. Note that the form of the systems (3.18) and (3.12) is identical. Subsequently, we obtain our Newton update \(\delta \lambda^{(k)}\) by solving

\[(3.19) \quad F_{NL2}(\tilde{u}^{(k)})\delta \lambda^{(k)} = B\tilde{u}^{(k)}\]
or, equivalently,

\[(3.20) \quad D_\lambda F(\lambda^{(k)})\delta \lambda^{(k)} = F(\lambda^{(k)})\]

with some Krylov iteration and update

\[\lambda^{(k+1)} = \lambda^{(k)} - \alpha^{(k)}\delta \lambda^{(k)}\]

with a suitable step length \(\alpha^{(k)}\).

Note that the system matrix in (3.19) for the Krylov iteration is the same as in NL-FETI-DP-1 and NK-FETI-DP if the linearization is performed at the same point, i.e., we have \(F_{NL1}(\tilde{u}) = F_{NL2}(\tilde{u}) = F_{NK}(\tilde{u})\), where \(F_{NK}(\tilde{u})\) is defined in section 3.5. Only the right-hand side differs. To obtain the right-hand side and to calculate the matrix \(F_{NL2}(\tilde{u}^{(k)})\), we have to solve local nonlinear subdomain problems which are only coupled in the primal unknowns.

### 3.5. Newton–Krylov–FETI-DP

To compare the Nonlinear-FETI-DP methods with the standard NK-FETI-DP approach, we will review the NK-FETI-DP algorithm using the same notation as above. Let us now consider the fully assembled problem operating on \(V_h\),

\[(3.21) \quad R^T K(Ru) - R^T f = 0,\]

where \(A(u) = R^T K(Ru) - R^T f\); see (1.1), and \(R = (R_1^T, \ldots, R_N^T)^T\). Next, we linearize first and obtain the Newton iteration

\[(3.22) \quad u^{(k+1)} = u^{(k)} - \alpha^{(k)}\delta u^{(k)}\]

To obtain the update \(\delta u^{(k)}\), we have to solve the fully assembled linearized system

\[R^T DK(Ru^{(k)}) R\delta u^{(k)} = R^T K(Ru^{(k)}) - R^T f.\]

In order to solve the linearized problem by a FETI-DP method we again introduce a set of primal variables \(\Pi\) and the corresponding jump operator \(B\), acting on the remaining dual interface variables. We then solve

\[(3.23) \quad \begin{pmatrix} R_\Pi^T DK(R_\Pi u^{(k)}) R_\Pi & B^T \\ 0 & \end{pmatrix} \begin{pmatrix} \delta \tilde{u}^{(k)} \\ \lambda \end{pmatrix} = \begin{pmatrix} R_\Pi^T K(R_\Pi u^{(k)}) - \tilde{f} \\ 0 \end{pmatrix},\]

where \(R_\Pi^T DK(R_\Pi u) R_\Pi\) is the partially assembled form of the local tangential matrices. Note that, using (3.3) and (3.4), we can write (3.23) also as

\[(3.24) \quad \begin{pmatrix} D\tilde{K}(\tilde{u}^{(k)}) & B^T \\ B & \end{pmatrix} \begin{pmatrix} \delta \tilde{u}^{(k)} \\ \lambda \end{pmatrix} = \begin{pmatrix} \tilde{K}(\tilde{u}^{(k)}) - \tilde{f} \\ 0 \end{pmatrix}.\]
The partially assembled variables \( \tilde{u} \) and the assembled variables \( u \) are related to each other in the usual canonical way. Formally, we can introduce corresponding operators, i.e., we can write \( \tilde{u}^{(k)} := R_{H,D}^T R u^{(k)} \) and, after solving, \( \delta \tilde{u}^{(k)} := R_{H}^T R_{H} \delta \tilde{u}^{(k)} \). The matrix \( R_{H,D}^T \) is then a scaled version of \( R_{H}^T \), i.e., the rows in \( R_{H}^T \) which correspond to a primal variable are scaled with the inverse of the multiplicity of the primal variable. The matrix \( R_{D}^T \) is a scaled version of the matrix \( R_{D}^T \), where the rows are scaled with the inverse of the multiplicity of the corresponding global degrees of freedom.

By one step of block elimination, we obtain, from (3.24),

\[
\begin{pmatrix}
D \bar{K}(\tilde{u}^{(k)}) & B^T \\
0 & -B(D \bar{K}(\tilde{u}^{(k)}))^{-1} B^T
\end{pmatrix}
\begin{pmatrix}
\delta \tilde{u}^{(k)} \\
\lambda
\end{pmatrix}
= \begin{pmatrix}
-B(D \bar{K}(\tilde{u}^{(k)}))^{-1}(\bar{K}(\tilde{u}^{(k)}) - \bar{f})
\end{pmatrix}.
\]

It remains to solve the reduced system for the Lagrange multiplier variables

\[
F_{NK}(\tilde{u}^{(k)}) \lambda = d_{NK}(\tilde{u}^{(k)}),
\]

where

\[
F_{NK}(\tilde{u}^{(k)}) = B(D \bar{K}(\tilde{u}^{(k)}))^{-1} B^T, \quad d_{NK} = B(D \bar{K}(\tilde{u}^{(k)}))^{-1}(\bar{K}(\tilde{u}^{(k)}) - \bar{f}).
\]

This can again be solved by some preconditioned Krylov iteration using one of the standard FETI-DP preconditioners \( M^{-1} \), e.g., the Dirichlet preconditioner [48].

It can be easily seen that, if the linearization is performed at the same \( \tilde{u} \), then \( F_{NL1}(\tilde{u}) = F_{NK}(\tilde{u}) \). This does not imply that the methods are identical. For the same \( \tilde{u} \), the Nonlinear-FETI-DP-1 approach differs from the standard NK-FETI-DP by using a different right-hand side, i.e., \( d_{NK}(\tilde{u}) = d_{NL1}(\tilde{u}, 0) - B \tilde{u} \); cf., also, (3.24) and (3.5). Note that, as a result of

\[
B \delta \tilde{u}^{(k)} = B \tilde{u}^{(k)},
\]

the jumps in the Newton update will be present only if the initial value \( \tilde{u}^{(0)} \) has jumps. This special jump condition (3.28) gives us the possibility of choosing initial values \( \tilde{u}^{(0)} \) for Nonlinear-FETI-DP-1 (see section 3.3) with nonzero jumps in all nonprimal interface variables.

3.6. Algorithms and cost comparison. In this section, we give an algorithmic description of our nonlinear FETI-DP methods and of NK-FETI-DP. We also discuss the computational cost of the algorithms. Note that the Newton direction is a descent direction if the Jacobi matrix is symmetric positive definite.


NK-FETI-DP:

\textbf{Init:} \( \tilde{u}^{(0)} \in \hat{W} \)

\textbf{for} \( k = 0, \ldots, \text{convergence} \)

\textbf{build:} \( K(\tilde{u}^{(k)}) \) and \( D \bar{K}(\tilde{u}^{(k)}) \)

\textbf{solve:} \( B(D \bar{K}(\tilde{u}^{(k)}))^{-1} B^T \lambda = B(D \bar{K}(\tilde{u}^{(k)}))^{-1}(\bar{K}(\tilde{u}^{(k)}) - \bar{f}) \) \quad \text{// See (3.26)}

\textbf{update:} \( \delta \tilde{u}^{(k)} = D \bar{K}(\tilde{u}^{(k)})^{-1}(\bar{K}(\tilde{u}^{(k)}) - \bar{f} - B^T \lambda) \) \quad \text{// Compute \( \delta \tilde{u} \) from \( \lambda \)}

\textbf{compute steplength } \alpha^{(k)}

\textbf{update:} \( \tilde{u}^{(k+1)} := \tilde{u}^{(k)} - \alpha^{(k)} \delta \tilde{u}^{(k)} \)

\textbf{end}
3.6.2. Typical work and cost. As opposed to a standard NK-FETI-DP approach in our nonlinear FETI-DP methods, weakly coupled nonlinear local problems are solved; see (3.12) and (3.18). By increasing the local computational work we aim to reduce the amount of communication and the need for synchronization.

For a rough cost comparison, in order to estimate the local computational work, we will count the number of factorizations of $DK$, denoted $\#Factor$. and, for the global communication, we will count the number of Krylov iterations, denoted $\#KrylovIt$. The factorization of $DK$ also includes some communication but only in the primal variables.

In all of the three algorithms, NK-FETI-DP, Nonlinear-FETI-DP-1, and Nonlinear-FETI-DP-2, we have to solve two different types of linear systems. The first type of
system is of the form

\[(3.29) \quad D\tilde{K}(\tilde{u})\delta\tilde{u} = rhs_1\]

with some right-hand side \(rhs_1\). This system appears in each Newton step in the computation of the initial value in Nonlinear-FETI-DP-1 (see (3.12)) and in each Newton step on the right-hand side of Nonlinear-FETI-DP-2 (see (3.18)). It also appears in each Krylov iteration of the NK-FETI-DP approach as part of the system matrix; see (3.20).

Linear systems as in (3.29) are typically solved directly in exact FETI-DP methods and thus the solution requires one factorization of \(D\tilde{K}\).

The second type is of the form

\[(3.30) \quad B(D\tilde{K}(\tilde{u}))^{-1}B^T \delta\lambda = rhs_2\]

with some right-hand side \(rhs_2\). This system appears on the left-hand side in Nonlinear-FETI-DP-1 (see (3.10)), Nonlinear-FETI-DP-2 (see (3.19)), as well as NK-FETI-DP (see (3.26)). We solve this system iteratively and matrix-free with a preconditioned Krylov method. The solution of this system requires the computation of the factorization of \(D\tilde{K}\) for the setup and, in addition, a number of Krylov iterations.

In our cost estimate we thus consider the solution of the linear system (3.30) as requiring one factorization of \(D\tilde{K}\) and a certain number of Krylov iterations.

3.7. Transformation of basis in the nonlinear FETI-DP methods. A well known and effective way to implement weighted or arithmetic edge averages in the FETI-DP coarse problem is the local transformation into a new basis, which can be performed independently for all edges and subdomains. Of course other known approaches to implement coarse problems in FETI-DP methods are also possible here.

Here, for robustness, we use orthogonal transformations constructed by a local Gram-Schmidt process on the edge as in [30]. In the new basis, the edge averages are represented by one nodal constraint for each former edge constraint. We mostly retain the sparsity of the tangential matrices (see [29]) and the form of \(R_{\Omega}\) and \(B\): we only have to augment \(R_{\Omega}\) with one new nodal constraint for each edge constraint and remove the new primal variables from our dual space and also from \(B\). With given local and orthogonal transformations \(T_i\), operating on \(W_i\), we can write a \(\tilde{\tilde{u}}_T\), where the index \(T\) marks the description in the transformed basis, in the form \(\tilde{\tilde{u}}_T = R_{\Omega}^{T_i}B^T T_i^T\tilde{\tilde{u}}_T\). Here \(\tilde{\tilde{u}}\) is described in the nodal basis and we have the matrix \(T := \text{diag}(T_1, \ldots, T_N)\). Now we reformulate the saddle point problem (3.1) in the new basis by

\[(3.31) \quad R_{\Omega}^{TT}K(TR_{\Omega}\tilde{\tilde{u}}_T) + B^T\lambda - R_{\Omega}^{TT}f = 0, \quad B\tilde{\tilde{u}}_T = 0,\]

or, in a shorter notation, by

\[(3.32) \quad \tilde{\tilde{K}}_T(\tilde{\tilde{u}}_T) + B^T\lambda - R_{\Omega}^{TT}f = 0, \quad B\tilde{\tilde{u}}_T = 0,\]

where \(\tilde{\tilde{K}}_T(\tilde{\tilde{u}}_T):= R_{\Omega}^{TT}K(TR_{\Omega}\tilde{\tilde{u}}_T).\)

Let us first consider the initial value $\tilde{u}_T$ for Nonlinear-FETI-DP-1. Let us consider (3.12). In the transformed basis, we now have to solve

$$K_T(\tilde{u}_T) + B^T \lambda^{(0)} - R_{TT}^T f = 0$$

by a Newton iteration of the form

$$\tilde{u}_T^{(k+1)} = \tilde{u}_T^{(k)} - \alpha^{(k)} \delta \tilde{u}_T^{(k)}.$$

We can actually perform the complete iteration in the new basis by starting with $\tilde{u}_T^{(0)} := R_{TT}^T R_{TT}^{(0)}$ and with the Newton updates defined by

$$R_{TT}^T DK(T R_{TT} \tilde{u}_T^{(k)}) T R_{TT} \delta \tilde{u}_T^{(k)} = K_T(\tilde{u}_T^{(k)}) + B^T \lambda^{(0)} - R_{TT}^T f.$$

For a shorter notation we define $D \tilde{K}_T(\tilde{u}_T) := R_{TT}^T DK(T R_{TT} \tilde{u}_T) T R_{TT}$.

We obtain the following iteration for our Nonlinear-FETI-DP-1 method,

$$\begin{pmatrix} \tilde{u}_T^{(k+1)} \\ \lambda^{(k+1)} \end{pmatrix} = \begin{pmatrix} \tilde{u}_T^{(k)} \\ \lambda^{(k)} \end{pmatrix} - \alpha^{(k)} \begin{pmatrix} \delta \tilde{u}_T^{(k)} \\ \delta \lambda^{(k)} \end{pmatrix},$$

where the update is obtained by solving

$$\begin{pmatrix} D \tilde{K}_T(\tilde{u}_T) \\ B \\ 0 \end{pmatrix} \begin{pmatrix} \delta \tilde{u}_T^{(k)} \\ \delta \lambda^{(k)} \end{pmatrix} = \begin{pmatrix} \tilde{K}_T(\tilde{u}_T^{(k)}) + B^T \lambda^{(k)} - R_{TT}^T f \\ B \tilde{u}_T^{(k)} \end{pmatrix}.$$

We can proceed as in the case without a transformation of basis. After convergence, we obtain the solution in the nodal basis by applying the transformation $T$.

In our implementation we perform all Newton iterations in the original nodal basis and only solve the linearized systems in the new basis. This is convenient, because discretized functions, gradients, and tangential matrices are typically represented in the nodal basis, as well as the stopping criteria. Therefore, we remark that

$$\tilde{u}^{(k+1)} := R_{TT}^T R_{TT} \tilde{u}_T^{(k+1)} = R_{TT}^T R_{TT} \left( \tilde{u}_T^{(0)} + \sum_{i=1}^{k} \delta \tilde{u}_T^{(i)} \right)$$

$$= \tilde{u}^{(0)} + \sum_{i=1}^{k} R_{TT}^T R_{TT} \delta \tilde{u}_T^{(i)}.$$

The iteration for the computation of the initial value (3.34) becomes

$$\tilde{u}^{(k+1)} = \tilde{u}^{(k)} - \alpha^{(k)} R_{TT}^T R_{TT} \delta \tilde{u}_T^{(k)},$$

where the update is obtained by solving

$$R_{TT}^T DK(\tilde{u}^{(k)}) T R_{TT} \delta \tilde{u}_T^{(k)} = R_{TT}^T DK(\tilde{u}^{(k)}) T R_{TT} \delta \tilde{u}_T^{(k)} = R_{TT}^T DK(\tilde{u}^{(k)}) T R_{TT} \delta \tilde{u}_T^{(k)} = R_{TT}^T DK(\tilde{u}^{(k)}) T R_{TT} \delta \tilde{u}_T^{(k)}.$$

Now, we will also formulate (3.36) and (3.37) in the nodal basis and obtain

$$\begin{pmatrix} \tilde{u}_T^{(k+1)} \\ \lambda^{(k+1)} \end{pmatrix} = \begin{pmatrix} \tilde{u}_T^{(k)} \\ \lambda^{(k)} \end{pmatrix} - \alpha^{(k)} \begin{pmatrix} R_{TT}^T R_{TT} \delta \tilde{u}_T^{(k)} \\ \delta \lambda^{(k)} \end{pmatrix},$$
where the update is obtained by solving
\[
\begin{pmatrix}
R_{\Omega}^T T^T D K(\tilde{u}^{(k)}) T R_{\Omega} & B^T \\
0 & B
\end{pmatrix}
\begin{pmatrix}
\delta \tilde{u}^{(k)} \\
\delta \lambda^{(k)}
\end{pmatrix}
= \begin{pmatrix}
R_{\Omega}^T T^T K(\tilde{u}^{(k)}) + B^T \lambda^{(k)} - R_{\Omega}^T T^T f \\
BR_{\Omega, D}^T T^T R_{\Omega} \tilde{u}^{(k)}
\end{pmatrix}.
\]

In this approach we have the same linearized systems because we evaluate the discrete operators in the same points, but all solutions and iterations are formulated in the nodal basis. In all cases we only have local multiplications of $T$ or $T^T$ with vectors $R_{\Omega} \tilde{u}$ or discrete operators $K(\cdot)$ and tangential matrices $DK(\cdot)$.

### 3.7.2. Transformation of basis in the Nonlinear-FETI-DP-2 method.

Let us recall
\[
\bar{K}_T(\tilde{u}_T) = R_{\Omega}^T T^T K(T R_{\Omega} \tilde{u}_T).
\]

By performing a nonlinear elimination of $\tilde{u}_T$ in (3.31), we obtain
\[
F_T(\lambda) = B \bar{K}_T^{-1}(R_\Omega^T T^T f - B^T \lambda)
\]
and
\[
D_A F_T(\lambda) = -B(D \bar{K}_T(\tilde{u}_T))^{-1} B^T.
\]

Here, the pair $(\lambda, \tilde{u}_T)$ has to satisfy the equation
\[
\bar{K}_T(\tilde{u}_T) + B^T \lambda - R_{\Omega}^T T^T f = 0.
\]

Now we have all the ingredients to formulate Nonlinear-FETI-DP-2 completely in the new basis. The computation of a $\tilde{u}_T$, which satisfies (3.41) for the iterate $\lambda^{(k)}$, can be carried out analogously to (3.34) with $\lambda^{(k)}$ replacing $\lambda^{(0)}$ in (3.35). For the dual Newton iteration, we have
\[
\lambda^{(k+1)} = \lambda^{(k)} - \alpha^{(k)} \delta \lambda^{(k)}
\]
and the update is obtained by solving
\[
B(D \bar{K}_T(\tilde{u}_T))^{-1} B^T \delta \lambda^{(k)} = -B \tilde{u}_T.
\]

As in Nonlinear-FETI-DP-1 we can perform all iterations in the nodal basis. Therefore we compute a $\tilde{u}$ such that $R_{\Omega, D}^T T^T R_{\Omega} \tilde{u}$ satisfies (3.41) for the iterate $\lambda^{(k)}$ analogously to (3.38). The dual update $\delta \lambda^{(k)}$ can be obtained by solving
\[
B(R_{\Omega}^T T^T D K(\tilde{u}) T R_{\Omega})^{-1} B^T \delta \lambda^{(k)} = -BR_{\Omega, D}^T T^T R_{\Omega} \tilde{u}.
\]

Again, we only perform local multiplications with $T$ or $T^T$.

### 3.8. Improving the convergence of Nonlinear-FETI-DP-2.

Since we generate a fully nonlinear problem $F(\lambda^*) = 0$ in the dual variables by nonlinear elimination of the variables $\tilde{u}$, we will collect some remarks on the convergence of the resulting Newton iteration (3.15). We can hope for quadratic convergence of the Newton method if we start in a sufficiently small neighborhood of the solution $\lambda^*$. Of course, in general, e.g., in nonlinear continuum mechanics, sufficient conditions for local quadratic convergence of the Newton method may not hold. The radius of convergence can be enlarged if a suitable globalization strategy is used. Here, we only consider line search methods. Trust region methods can of course also be used for the subdomain problems as well as for the coarse problem. In nonlinear multilevel methods for elasticity, globalization using recursive trust region methods has been a successful approach; see [21].
3.8.1. Choosing the initial value and choice of the coarse space. For an initial value $\lambda^{(0)}$ we have

$$F(\lambda^{(0)}) = B\tilde{u}$$

for a $\tilde{u}$ which satisfies the equation

$$\tilde{K}(\tilde{u}) + B^T\lambda^{(0)} - \tilde{f} = 0.$$ 

Thus, we may search a $\lambda^{(0)}$ such that the jump on the interface of the corresponding $\tilde{u}$ is sufficiently small. Alternatively, for a given $\lambda^{(0)}$, we can manipulate our primal constraints, such that a $\tilde{u}$ with sufficiently small jump in the dual variables, satisfies

$$\tilde{K}(\tilde{u}) = -B^T\lambda^{(0)} + \tilde{f}.$$ 

Recently developed strategies of adaptive coarse space selection for standard FETI-DP and BDDC methods (see, e.g., [12, 26, 28, 42, 46, 47]), may be especially valuable in this nonlinear context since, here, they can help to improve the convergence of the Newton method.


$$\lambda^{(k+1)} = \lambda^{(k)} - \alpha^{(k)}(D_\lambda F(\lambda^{(k)}))^{-1}F(\lambda^{(k)}),$$

we still need to define a suitable step length. In our tests, for our model problems, for a bad choice of the coarse space, a good step length is sometimes necessary for convergence. In this special case, for Armijo or Wolfe conditions, the expression $F(\lambda^{(k)} - \alpha \delta \lambda^{(k)})$ has to be computed for several $\alpha$. In the context of Nonlinear-FETI-DP-2, for each $\alpha$, a weakly coupled nonlinear system

$$(3.42) \quad F(\lambda^{(k)} - \alpha \delta \lambda^{(k)}) = B\tilde{u} = B(\tilde{K}^{-1}(\tilde{f} - B^T(\lambda^{(k)} - \alpha \delta \lambda^{(k)}))) = 0$$

has to be solved, e.g., using a Newton iteration. We note that the computational work is mostly local. At this point we should remember that we have to compute

$$(3.43) \quad \tilde{u} = \tilde{K}^{-1}(\tilde{f} - B^T\lambda^{(k+1)})$$

in every step of our algorithm, in order to compute the right-hand side; see (3.18).

Here, we describe our strategy to save computational cost. First, it is often not necessary to solve (3.42) until convergence. Second, let us assume a backtracking strategy for $\alpha$, e.g., $\alpha^{(0)} := 1$ and then $\alpha^{(i)} = \rho \alpha^{(i-1)}$, $i = 1, 2, \ldots$, where $0 < \rho < 1$. In this case, the $\tilde{u}$ resulting from (3.42) can be reused as a start value for the iteration for the next value of $\alpha$.

Note that, in our FETI-DP method, we want to minimize the jump, e.g., the norm $\|B\tilde{u}\| = \|F(\lambda^{(k)} - \alpha \delta \lambda^{(k)})\|$. We thus solve (3.42) until, e.g., stagnation of $\|B\tilde{u}\|$.

Finally the value of $\tilde{u}$ resulting from this minimization can also be reused when computing (3.43). Once, we have found the final step length $\alpha$, we choose the stopping criterion defined in section 5.2.

In cases where our simple and less expensive step length strategy is not sufficient for convergence, we suggest to verify the Wolfe condition (cf. (2.7) and (2.8)) or even the strong Wolfe condition (see [43]). This can be combined with the computation of $\tilde{u}$ in (3.43) in a similar way, but will be more expensive.
4. **Nonlinear BDDC method.** In the spirit of our nonlinear FETI-DP methods we can also formulate a nonlinear BDDC method. A nonlinear Neumann-Neumann or BDD method was already introduced by Bordeu, Boucard, and Gosselet in [2]. In our nonlinear BDDC method we will eliminate all interior variables and build a nonlinear Schur complement, which then can be linearized. This approach is thus similar to Nonlinear-FETI-DP-2.

4.1. **Newton–Krylov–BDDC.** We briefly review the classical use of the linear BDDC preconditioner within an NK approach. Here, we want so solve the fully assembled system operating on $V^h$,

\[ R^T K(Ru) - R^T f = 0, \]

where $K$ is defined in (2.5), by a Newton iteration

\[ u^{(k+1)} = u^{(k)} - \alpha^{(k)} \delta u^{(k)} \]

with the update $\delta u^{(k)}$ obtained by solving

\[ R^T DK(Ru^{(k)}) R \delta u^{(k)} = R^T K(Ru^{(k)}) - R^T f. \]

Let us recall the partition of the tangential matrix $DK$ from (4.6),

\[ DK(Ru^{(k)}) = \begin{pmatrix} DK(Ru^{(k)})_{II} & DK(Ru^{(k)})_{II} \\ DK(Ru^{(k)})_{I} & DK(Ru^{(k)})_{II} \end{pmatrix}, \]

and also the partition of the right-hand side from (4.5),

\[ K(Ru^{(k)}) - f = \begin{pmatrix} (K(Ru^{(k)}) - f)_I \\ (K(Ru^{(k)}) - f)_I \end{pmatrix} \]

Here, as before, $I$ denotes the set of interior variables and $\Gamma$ the set of interface variables. Using the BDDC algorithm to solve the linearized system (4.2) we first eliminate the interior variables and solve for the assembled interface variables

\[ M_{BDDC}^{-1}(Ru^{(k)}) S_g(Ru^{(k)}) \delta u_g^{(k)} = M_{BDDC}^{-1}(Ru^{(k)}) g_g(Ru^{(k)}) \]

by some Krylov iteration. Here, we have the assembled Schur complement $S_g(u)$ defined by

\[ S_g(u) := R^T I S_I(RI) R_I = R^T I (DK(u)_{II} - DK(u)_{II} (DK(u)_{II})^{-1} DK(u)_{II}) R_I. \]

The BDDC preconditioner is defined by

\[ M_{BDDC}^{-1}(u) := R^T D (R_{II} D (DK(u)_{II})^{-1} R_D \]

with

\[ R_D := \begin{pmatrix} R^T \Delta D R^T \Delta & 0 \\ 0 & I_{II} \end{pmatrix}, \]

and $R^T \Delta D R_D$ is the weighted restriction from the set $[I, \Delta]$ to $\Delta$. The right-hand side $g_g$ is defined by

\[ g_g(u) := R^T I ((K(u) - f)_I - DK(u)_{II} (DK(u)_{II})^{-1} (K(u) - f)_I). \]

Finally, we get the full Newton update by

\[ \delta u^{(k)} := \begin{pmatrix} (DK(Ru^{(k)})_{II})^{-1} ((K(Ru^{(k)}) - f)_I - DK(Ru^{(k)})_{II}) \delta u_g^{(k)} \end{pmatrix}. \]
4.2. Nonlinear Schur complement. To formulate a nonlinear BDDC algorithm let us first introduce the nonlinear Schur complement system. For the moment, let us assume that $K(u) = f$ is the discretization of our nonlinear problem on a single domain, e.g., this could be a subdomain. As usual, we partition the variables into interior variables denoted by $u_I$ and boundary variables denoted by $u_T$. This leads us to the following partition of the operator $K(u)$ and the right-hand side $f$:

\begin{equation}
K(u) - f = \begin{pmatrix} K_I(u_I, u_T) \\ K_T(u_I, u_T) \end{pmatrix} - \begin{pmatrix} f_I \\ f_T \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.
\end{equation}

We also partition the tangential matrix $DK(u)$ in the same way and obtain

\begin{equation}
DK(u) = \begin{pmatrix} D_{u_I}K_I(u_I, u_T) & D_{u_T}K_I(u_I, u_T) \\ D_{u_I}K_T(u_I, u_T) & D_{u_T}K_T(u_I, u_T) \end{pmatrix} = \begin{pmatrix} DK(u_I, u_T)_{II} & DK(u_I, u_T)_{IT} \\ DK(u_T, u_T)_{TI} & DK(u_T, u_T)_{TT} \end{pmatrix}.
\end{equation}

Under the sufficient assumptions that $K_I$ is continuously differentiable, that there exists a $(u_I, u_T)$ which satisfies $K_I(u_I, u_T) - f_I = 0$, and that $D_{u_I}K_I(u_I, u_T)$ is invertible, there exists an implicit function $h(u_T) = u_I$ and in a neighborhood of $u_T$ we have

$$0 = K_I(u_I, u_T) - f_I = K_I(h(u_T), u_T) - f_I.$$ 

From the implicit function theorem we then obtain the derivative of $h$,

\begin{equation}
Dh(u_T) = -(DK(h(u_T), u_T)_{II})^{-1}DK(h(u_T), u_T)_{IT}.
\end{equation}

Now, we can define the nonlinear Schur complement by

\begin{equation}
S(u_T) := K_T(h(u_T), u_T) - f_T
\end{equation}

and, using the chain rule in combination with (4.7), the derivative of $S$ is given by

$$DS(u_T) = D_{u_T}K_T(h(u_T), u_T)Dh(u_T) + D_{u_T}K_T(h(u_T), u_T)$$
$$= -DK(h(u_T), u_T)_{TI}(DK(h(u_T), u_T)_{II})^{-1}DK(h(u_T), u_T)_{IT} + DK(h(u_T), u_T)_{TT}.$$ 

Let us note that the derivative of the nonlinear Schur complement $DS$ is the same as the Schur complement $S_I$ of the derivative of the nonlinear operator $K$; see also (4.3) in section 4.1.

If we now consider more than one subdomain, then $S(u_T)$ has a block form. Let us reintroduce the index $i = 1, \ldots, N$ and define the nonlinear Schur complement on a subdomain $\Omega_i$ by $S_i(u_{N_i})$. Then, we have

\begin{equation}
S(u_T) := (S_1(u_{1_T})^T, \ldots, S_N(u_{N_T})^T)^T,
\end{equation}

where $u_T := (u_{1_T}^T, \ldots, u_{N_T}^T)^T$.

4.3. Nonlinear BDDC formulation. Our nonlinear BDDC method is obtained by linearizing the nonlinear Schur complement and using a BDDC preconditioner to solve the linearized problem. As in the Nonlinear-FETI-DP-2 method we have to solve local nonlinear problems on the right-hand side. We introduce the assembly operator on the interface $R_I^T := (R_I^{(1)T} \ldots R_I^{(N_T)T})^T$, where $R_I^{(i)}$ is the restriction
from the global interface to the local interface on subdomain $\Omega_i$, $i = 1, \ldots, N$. Next, we solve the assembled nonlinear Schur complement system (cf. (4.9))

$$R_I^T S(R_I \bar{u}_I) = 0$$

by a Newton-type iteration of the form

$$\bar{u}_I^{(k+1)} = \bar{u}_I^{(k)} - \alpha^{(k)} \delta \bar{u}_I^{(k)}$$

for a given initial value $\bar{u}_I^{(0)} := [0, R_I^T u^{(0)}]$, $u^{(0)} \in W$. We obtain the update $\delta \bar{u}_I^{(k)}$ by solving the linear system

$$(4.10) \quad R_I^T D S(R_I \bar{u}_I^{(k)}) R_I \delta \bar{u}_I^{(k)} = R_I^T S(R_I \bar{u}_I^{(k)})$$

or, already using the notation from (4.3),

$$(4.11) \quad S_g(h(R_I \bar{u}_I^{(k)}), R_I \bar{u}_I^{(k)}) \delta \bar{u}_I^{(k)} = R_I^T S(R_I \bar{u}_I^{(k)}).$$

To obtain the right-hand side of (4.11), we have to solve local nonlinear systems. From (4.8) we have

$$R_I^T S(R_I \bar{u}_I^{(k)}) = R_I^T (K_I (h(R_I \bar{u}_I^{(k)}), R_I \bar{u}_I^{(k)}) - f_I).$$

We thus need to compute the nonlinear extension $u_I^* := h(R_I \bar{u}_I^{(k)})$ of the values on the interface by solving

$$(4.12) \quad K_I(u_I^*, R_I \bar{u}_I^{(k)}) - f_I = 0.$$ 

We obtain $u_I^*$ by carrying out the Newton iteration $u_I^{(*+1)} = u_I^{(*)} - \alpha^{(*)} \delta u_I^{(*)}$ with

$$(4.13) \quad DK(u_I^{(*)}, R_I \bar{u}_I^{(k)}) I_{II} \delta u_I^{(*)} = K_I(u_I^{(*)}, R_I \bar{u}_I^{(k)}) - f_I.$$ 

We remark that the problems in (4.13) are uncoupled and therefore completely local. In the last step we apply the linear BDDC preconditioner $M_{BDCC}^{DDC}(u_I^*, R_I \bar{u}_I^{(k)})$ to (4.11), details can be found in the next subsection (cf. also (4.4)), and solve with some Krylov iteration. We also need $h(R_I \bar{u}_I^{(k)}) = u_I^*$ to obtain $S_g(h(R_I \bar{u}_I^{(k)}), R_I \bar{u}_I^{(k)})$ in (4.11).

With the equivalence in (4.11) and $F_{NL1}(\bar{u}) = F_{NL2}(\bar{u}) = F_{NK}(\bar{u})$ for Nonlinear-FETI-DP, and since we use the standard linear preconditioner, we also have the equality of the eigenvalues of the preconditioned system matrices of the Nonlinear-FETI-DP and Nonlinear-BDDC methods except for the eigenvalues zero and one. Of course this is only true if the tangential matrix is built at the same point for all methods. But as a result of the different right-hand sides the nonlinear methods are indeed very different from each other and can show a drastically different convergence behavior. Also note that in our nonlinear FETI-DP methods the coarse problem can improve the Newton convergence whereas in Nonlinear-BDDC this is not the case.

4.4. Comparison of the nonlinear FETI-DP methods and Nonlinear-BDDC. Since the coarse problem is included in the operator in the Nonlinear-FETI-DP methods, the coarse space has a direct influence on the Newton iteration, in addition to the Krylov subspace iteration. This is not the case for the Nonlinear-BDDC method. Since the primal assembly is only part of the linear preconditioner in
Nonlinear-BDDC, the choice of the coarse space has no effect on the Newton iterations but only on the convergence of the Krylov space method. In the nonlinear formulations that we choose for FETI-DP and BDDC, both algorithms therefore have a quite different convergence behavior.

Note that the local nonlinear problems in the Nonlinear-BDDC method are completely decoupled whereas in the Nonlinear-FETI-DP methods the local nonlinear problems are coupled in the primal variables. This increases the communication cost but improves the convergence. Note that an additional coarse space can be applied to the linearized problems, e.g., using deflation; see, e.g., [32].

The coarse space of the Nonlinear-FETI-DP methods is nonlinear whereas in the Nonlinear-FETI-1 approach by Pebrel, Rey, and Gosselet [45] the coarse space is linear. From a theoretical viewpoint, Nonlinear-BDDC relies on assumptions similar to those of the nonlinear FETI-DP methods; see section 3.1.

We may also construct nonlinear FETI-DP and BDDC methods by applying a nonlinear preconditioner to the dual or primal nonlinear Schur complement systems. In this case two nonlinear systems would have to be solved on the right-hand side of each global Newton iteration. We do not discuss such methods here.

5. Nonlinear model problem and numerical results. In this section, we present numerical results for different model problems based on the p-Laplace operator with homogeneous Dirichlet boundary conditions. Results are shown for the two new nonlinear FETI-DP methods, for NK-FETI-DP, and for the new nonlinear BDDC algorithm.

5.1. The p-Laplace. Let us define the p-Laplace operator for $p \geq 2$ by

$$\Delta_p u := \text{div}(|\nabla u|^{p-2} \nabla u).$$

A solution of the partial differential equation

$$-\Delta_p u = b \quad \text{in } \Omega,$$

$$u = 0 \quad \text{on } \partial \Omega,$$

minimizes the energy

$$J(u) := \int_{\Omega} \frac{1}{p} |\nabla u|^p - bu \, dx.$$ 

Using the decomposition of $\Omega$ into nonoverlapping subdomains $\Omega_i$, $i = 1, \ldots, N$, we define

$$J_i(u_i) := \int_{\Omega_i} \frac{1}{p} |\nabla u_i|^p - bu_i \, dx , i = 1, \ldots, N.$$

This decomposition satisfies Assumption 1; see section 2. We have to compute the derivative of $J_i(u_i)$ (see section 2), and obtain

$$J_i'(u_i)(v_i) = \int_{\Omega_i} |\nabla u_i|^{p-2} \nabla u_i^T \nabla v_i - bv_i \, dx.$$ 

Therefore, given a finite element basis $\{\phi_1, \ldots, \phi_N\}$, we have

$$K_i(u_i) := \left( \int_{\Omega_i} |\nabla u_i|^{p-2} \nabla u_i^T \nabla \phi_1 \, dx, \ldots, \int_{\Omega_i} |\nabla u_i|^{p-2} \nabla u_i^T \nabla \phi_N \, dx \right)^T.$$
and the right-hand sides

\[ f_i := \left( \int_{\Omega_i} b\phi_1 dx, \ldots, \int_{\Omega_i} b\phi_N dx \right)^T. \]

For the tangential matrices \( DK_i(u_i) \), we obtain

\[ (DK_i(u_i))_{j,k} := \int_{\Omega_i} |\nabla u_i|^{p-2} \nabla \phi_j \nabla \phi_k dx + (p-2) \int_{\Omega_i} |\nabla u_i|^{p-4} (\nabla u_i^T \nabla \phi_j) \nabla \phi_k dx \]

directly by computing the derivative of \( J'_i(u_i)(v_i) \) in the direction \( w_i \) and by using the basis representations of \( v_i \) and \( w_i \).

5.2. Numerical results for Nonlinear FETI-DP. We test our algorithms for four different nonlinear model problems based on variants of the p-Laplace equation. First, we consider the standard p-Laplace equation (5.1). The second model problem has nonlinearities which are local with respect to the subdomains. In the third model problem, we have local nonlinearities which are not confined to the subdomains. The fourth model problem is nonlinear on the complete domain but has large discontinuities in the coefficients.

In all our experiments, we consider the unit square \( \Omega := [0,1] \times [0,1] \) and decompose it into square subdomains \( \Omega_i, i = 1, \ldots, N \). As finite elements, we have chosen piecewise linear finite elements. In all our tests all vertices are primal and, additionally, all edges are primal, i.e., edge averages are used as primal edge constraints in our linear and nonlinear FETI-DP methods. We carry out experiments using both standard arithmetic edge averages and weighted averages. For a given edge \( E \), we consider edge averages \( \bar{e} \) of the form

\[ \bar{e} := \frac{\sum_{i=1}^{N_E} \rho_i u_i}{\sum_{i=1}^{N_E} \rho_i}, \]

where \( N_E \) is the number of nodes \( x \) on the edge \( E \), \( \rho_i \) is a weight associated with such a node, and \( u_i = u(x) \) is the nodal evaluation of the function to be averaged. For \( \rho_i = 1, i = 1, \ldots, N_E \), we have standard arithmetic edge averages. Otherwise, the weights \( \rho_i \) are defined by using the coefficients of the differential equations; see the discussion below. We always use the following stopping criteria. Newton iterations, which solve systems of the form \( \tilde{K}(\tilde{u}) + B^T \lambda - f = 0 \) (see (3.12) and (3.18)) are stopped when \( ||(\tilde{K}(\tilde{u}) + B^T \lambda - f)||_{L_2} < 1e^{-12} \). Furthermore, in all our FETI-DP algorithms, we stop the global Newton iteration when the fully assembled residual satisfies \( ||R^T K(R_K \tilde{u}^{(k)}) - R^T f||_{L_2} < 1e^{-12} \). Here, \( R^T \) is the global assembly operator on the interface from (3.21). Our stopping criterion is thus based on the variable \( u \) and not on \( \lambda \) for NL-FETI-DP-1 as well as NL-FETI-DP-2. We thus have identical stopping criteria for all three FETI-DP methods.

As initial values we always use \( \lambda^{(0)} = 0 \) and \( u^{(0)}(x_1, x_2) = x_1(1-x_1)x_2(1-x_2) \) which satisfies our homogeneous Dirichlet boundary condition. We cannot use \( u^{(0)} = 0 \) since the tangential matrix is then singular. In all our experiments, we choose the conjugate gradient method (CG) as a Krylov space method. The stopping criterion for the preconditioned Krylov iterations is the relative reduction of the preconditioned residual of \( 1e^{-10} \). Note that we always use recycling of Lagrange multipliers as start.
### Table 1
Numerical scalability for the different methods. The p-Laplace problem described in (5.1) and the p-Laplace inclusions' problem is described in (5.4). For “p-Laplace inclusions” see also Figure 1 (left). Here we used vertex constraints and arithmetic edge averages as primal constraints. In both problems we have $H/h = 16$; $N$ is the number of subdomains; all tests are done without any step length optimization; #Krylov It. gives the sum of all Krylov iterations; #Factor. gives the total number of factorizations of $D\tilde{K}$ required (see also section 3.6.2); min./max. cond. gives the minimal and maximal condition number of the FETI-DP systems.

<table>
<thead>
<tr>
<th>$N$</th>
<th>Solver</th>
<th>p-Laplace inclusions</th>
<th>p-Laplace</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>#Krylov It.</td>
<td>#Factor</td>
<td>Max. cond.</td>
</tr>
<tr>
<td>4</td>
<td>NK-FETI-DP</td>
<td>33</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>5</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>5</td>
<td>15</td>
</tr>
<tr>
<td>16</td>
<td>NK-FETI-DP</td>
<td>105</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>21</td>
<td>18</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>28</td>
<td>18</td>
</tr>
<tr>
<td>64</td>
<td>NK-FETI-DP</td>
<td>164</td>
<td>17</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>30</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>40</td>
<td>20</td>
</tr>
<tr>
<td>256</td>
<td>NK-FETI-DP</td>
<td>190</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>31</td>
<td>22</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>42</td>
<td>22</td>
</tr>
<tr>
<td>1024</td>
<td>NK-FETI-DP</td>
<td>299</td>
<td>21</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>31</td>
<td>24</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>43</td>
<td>24</td>
</tr>
<tr>
<td>4096</td>
<td>NK-FETI-DP</td>
<td>215</td>
<td>23</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>19</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>41</td>
<td>26</td>
</tr>
</tbody>
</table>

values, i.e., we use the Lagrange multiplier of the previous iteration as a start value for the Krylov subspace iteration. If an absolute convergence criterion is used for the CG iteration this simple strategy can lead to a substantial reduction of the number of Krylov iterations.

If a relative convergence criterion is used for the CG iteration, the absolute accuracy of the Krylov solution increases during the Newton iteration, since the residual norm of the initial value decreases by using a recycling of $\lambda$. Therefore, a relative reduction of $1 - 10$ is sufficient for the Krylov iteration. In all our computations the Dirichlet preconditioner was used in the CG iterations.

#### 5.2.1. First model problem
Our first model problem is the p-Laplace equation (5.1) for $p = 4$ with a constant right-hand side $b = 1$. We present some numerical scalability results for the two nonlinear FETI-DP methods and for NK-FETI-DP in Table 1 (right) and Table 2 (right). Here, we have $H/h = 16$, and we increase the number of subdomains. In the usual linear setting, a DD method is considered numerically scalable, if, for an increasing number of subdomains, the number of iterations is asymptotically bounded. Generally, numerical scalability of a DD method cannot be obtained without incorporating a coarse problem to accelerate convergence, by ensuring a global transport of information. Numerical scalability is usually regarded as necessary to obtain weak parallel scalability of a DD method.

In our nonlinear FETI-DP methods the coarse problem is constructed from local constraints as in the case of the linear method and will accelerate the Krylov iteration. But an important property of the nonlinear FETI-DP methods is the possibility of accelerating also the Newton convergence by the choice of the coarse problem.
### Table 2

Numerical scalability. Same set of experiments as in Table 1, but Wolfe step length optimization has been used additionally.

<table>
<thead>
<tr>
<th>N</th>
<th>Solver</th>
<th>p-Laplace inclusions</th>
<th>p-Laplace</th>
<th>#Krylov It.</th>
<th>#Factor</th>
<th>Max. cond.</th>
<th>Min. cond.</th>
<th>#Krylov It.</th>
<th>#Factor</th>
<th>Max. cond.</th>
<th>Min. cond.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>NK-FETI-DP</td>
<td>16</td>
<td>5</td>
<td>16</td>
<td>8</td>
<td>1.0000</td>
<td>1.0000</td>
<td>32</td>
<td>8</td>
<td>1.1243</td>
<td>1.0649</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>5</td>
<td>8</td>
<td>1.2819</td>
<td>1.0000</td>
<td>8</td>
<td>12</td>
<td>1.0644</td>
<td>1.0694</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>9</td>
<td>9</td>
<td>1.2809</td>
<td>1.0000</td>
<td>10</td>
<td>10</td>
<td>1.0644</td>
<td>1.0694</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>NK-FETI-DP</td>
<td>77</td>
<td>21</td>
<td>21</td>
<td>14</td>
<td>1.4719</td>
<td>1.4166</td>
<td>96</td>
<td>12</td>
<td>1.4605</td>
<td>1.3919</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>28</td>
<td>14</td>
<td>1.4240</td>
<td>1.2435</td>
<td>32</td>
<td>33</td>
<td>1.4208</td>
<td>1.4012</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>28</td>
<td>14</td>
<td>1.4240</td>
<td>1.2431</td>
<td>40</td>
<td>40</td>
<td>1.4208</td>
<td>1.4050</td>
<td></td>
<td></td>
</tr>
<tr>
<td>64</td>
<td>NK-FETI-DP</td>
<td>117</td>
<td>30</td>
<td>30</td>
<td>17</td>
<td>1.5680</td>
<td>1.5079</td>
<td>122</td>
<td>12</td>
<td>1.5186</td>
<td>1.4895</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>30</td>
<td>17</td>
<td>1.5256</td>
<td>1.5197</td>
<td>52</td>
<td>52</td>
<td>2.1258</td>
<td>2.1410</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>40</td>
<td>16</td>
<td>1.5254</td>
<td>1.5197</td>
<td>52</td>
<td>52</td>
<td>2.1258</td>
<td>2.1400</td>
<td></td>
<td></td>
</tr>
<tr>
<td>256</td>
<td>NK-FETI-DP</td>
<td>109</td>
<td>30</td>
<td>30</td>
<td>15</td>
<td>1.5608</td>
<td>1.5162</td>
<td>143</td>
<td>13</td>
<td>1.5521</td>
<td>1.5306</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>42</td>
<td>15</td>
<td>1.5520</td>
<td>1.5515</td>
<td>45</td>
<td>45</td>
<td>2.1905</td>
<td>2.1499</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>42</td>
<td>15</td>
<td>1.5653</td>
<td>1.5405</td>
<td>56</td>
<td>56</td>
<td>2.1905</td>
<td>2.1470</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1024</td>
<td>NK-FETI-DP</td>
<td>118</td>
<td>30</td>
<td>30</td>
<td>17</td>
<td>1.5428</td>
<td>1.5407</td>
<td>45</td>
<td>25</td>
<td>2.2061</td>
<td>1.5657</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>42</td>
<td>16</td>
<td>1.5428</td>
<td>1.5407</td>
<td>45</td>
<td>45</td>
<td>2.2061</td>
<td>1.5657</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>42</td>
<td>16</td>
<td>1.5653</td>
<td>1.5407</td>
<td>45</td>
<td>45</td>
<td>2.2061</td>
<td>1.5657</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4096</td>
<td>NK-FETI-DP</td>
<td>128</td>
<td>18</td>
<td>18</td>
<td>17</td>
<td>1.5792</td>
<td>1.5464</td>
<td>167</td>
<td>16</td>
<td>2.4030</td>
<td>1.5626</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>40</td>
<td>17</td>
<td>1.5467</td>
<td>1.5467</td>
<td>42</td>
<td>42</td>
<td>2.2138</td>
<td>2.1640</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>40</td>
<td>17</td>
<td>1.5932</td>
<td>1.5467</td>
<td>44</td>
<td>44</td>
<td>2.2138</td>
<td>2.1671</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

From the results in Table 1 (right) we see that for all methods we obtain Newton convergence using our start value $u^{(0)}$. Moreover, the performance of the two nonlinear FETI-DP methods is similar. Note that for our nonlinear FETI-DP methods, we seem to have numerical scalability in terms of the global number of Krylov iterations. This is a strong statement since this requires the scalability of the Krylov iteration as well as of the Newton iteration.

The most important result is that using the nonlinear FETI-DP methods the number of the global Krylov iterations is reduced by a factor of between 4 and 9 compared to the NK-FETI-DP approach. In a parallel environment the global communication in the Krylov method can thus be significantly reduced. At the same time the number of factorizations is always higher in the nonlinear methods, i.e., by a factor of up to 1.5. This represents an increment of local computational work on the subdomains and is expected.

Of course, in general, Newton’s method will not converge without some kind of load stepping or globalization strategy. We therefore also report on results using the Wolfe step optimization in all methods; see Table 2 (right). Indeed, the NK-FETI-DP approach profits most from the optimized step length. Nevertheless, using the nonlinear methods the number of global Krylov space iterations is still reduced by a factor of between 2 and 4.

#### 5.2.2 Second model problem.

By design, we expect our new methods to perform best for problems with strong localized nonlinearities. In order to analyze problems with such local nonlinearities, as a second model problem, we consider the p-Laplace equation with different values of $p$ in different parts of the computational domain. Note that the p-Laplace operator for $p = 2$ is the standard, linear Laplacian. More precisely, we consider a matrix material, where in each subdomain an inclusion is embedded; see Figure 1 (left). Each inclusion is surrounded by a hull of width $\eta$ of the matrix material. We consider the p-Laplace operator with $p = 2$ in the hull and with $p = 4$ in the inclusions.
Let us denote the hull inside each subdomain $\Omega_i$ by
$$\Omega_i,\eta := \{x \in \Omega_i : \text{dist}(x, \partial \Omega_i) < \eta\}$$
and the inclusion by $\Omega_i,I := \Omega_i \setminus \Omega_i,\eta$. Furthermore we define $\Omega_I := \bigcup_{i=1}^N \Omega_i,I$ and $\Omega_\eta := \bigcup_{i=1}^N \Omega_i,\eta$. We then consider
$$-\alpha \Delta p u - \beta \Delta u = 1 \quad \text{in } \Omega,$$
$$u = 0 \quad \text{on } \partial \Omega,$$
where $\alpha, \beta : \Omega \to \mathbb{R}$ are coefficient functions given by

$$\alpha(x) = \begin{cases} 1 & \text{if } x \in \Omega_I, \\ 0 & \text{elsewhere}, \end{cases} \quad \beta(x) = \begin{cases} 0 & \text{if } x \in \Omega_I, \\ 1 & \text{elsewhere}. \end{cases}$$

We again carry out numerical experiments to analyze the numerical scalability of the different methods; see Table 1 (left) and Table 2 (left). Again, significantly fewer Krylov iterations are needed in comparison to NK-FETI-DP, indicating a considerable reduction of the communication in a parallel environment. If the Wolfe step strategy is used in all methods the number of Krylov iterations is reduced by a factor of between 3 and 4 compared to classical NK-FETI-DP; see Table 2 (left).

For comparison, we also provide the results for the case when no globalization technique is used. In this case the number of Krylov iterations can be reduced by a factor of 4 to 11 when using the nonlinear methods; see Table 1 (left). Again, the number of factorizations is higher in the new methods, as expected.

**5.2.3. Third model problem.** As a third model problem, we consider localized nonlinearities which are not confined to the subdomains. Instead of inclusions, we now consider channels of width $\frac{H}{2}$ which intersect rows of subdomains from the left boundary to the right boundary of $\Omega$. All channels are parallel and each subdomain is intersected by exactly one channel; see Figure 1 (middle and right). We have $p = 2$ in the matrix material outside of the channels and $p = 4$ in the channels. In addition, we scale the $p$-Laplace operator in the channels with a large coefficient $\alpha$; see (5.6).

Let us denote by $\Omega_C$ the union of the channels $\Omega_{i,C}$, $i = 1, \ldots, N^* = N^2$, and the rest of the domain by $\Omega_R := \Omega \setminus \Omega_C$; see Figure 1 (left and middle). We consider problem (5.4) with coefficient functions $\alpha, \beta : \Omega \to \mathbb{R}$:

$$\alpha(x) = \begin{cases} \tilde{\alpha} & \text{if } x \in \Omega_C, \\ 0 & \text{elsewhere}, \end{cases} \quad \beta(x) = \begin{cases} 0 & \text{if } x \in \Omega_C, \\ 1 & \text{elsewhere}, \end{cases}$$
TABLE 3

Numerical scalability for the different algorithms for problems with a high contrast in coefficients. Results for (5.4) with α and β defined as in (5.6). We have p = 4 and a scaling α ∈ \{10^3, 10^6\} in the channels. The matrix material is linear since here we have p = 2. We have \(H/h = 16\). The primal set consists of vertex constraints and arithmetic averages on all edges; N is the number of subdomains; #Krylov It. gives the sum of all Krylov iterations; #Factor. gives the total number of factorizations of \(DK\) required (see also section 3.6.2); min./max. cond. gives the minimal and maximal condition number of the FETI-DP systems.

<table>
<thead>
<tr>
<th>N</th>
<th>Solver</th>
<th>(\alpha = 10^3)</th>
<th></th>
<th>(\alpha = 10^6)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>#Krylov It.</td>
<td>#Factor</td>
<td>Max. cond.</td>
<td>Min. cond.</td>
</tr>
<tr>
<td>4</td>
<td>NK-FETI-DP</td>
<td>37</td>
<td>8</td>
<td>1.0226</td>
<td>1.0020</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>10</td>
<td>9</td>
<td>1.0228</td>
<td>1.0184</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>15</td>
<td>10</td>
<td>1.0365</td>
<td>1.0184</td>
</tr>
<tr>
<td>16</td>
<td>NK-FETI-DP</td>
<td>65</td>
<td>8</td>
<td>1.9053</td>
<td>1.9820</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>32</td>
<td>14</td>
<td>1.2433</td>
<td>1.2399</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>32</td>
<td>11</td>
<td>1.2429</td>
<td>1.2182</td>
</tr>
<tr>
<td>64</td>
<td>NK-FETI-DP</td>
<td>77</td>
<td>8</td>
<td>1.0477</td>
<td>1.0458</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>35</td>
<td>14</td>
<td>1.4603</td>
<td>1.3811</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>34</td>
<td>11</td>
<td>1.4671</td>
<td>1.3811</td>
</tr>
<tr>
<td>256</td>
<td>NK-FETI-DP</td>
<td>101</td>
<td>9</td>
<td>4.3759</td>
<td>4.6399</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>40</td>
<td>14</td>
<td>1.6355</td>
<td>1.5248</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>40</td>
<td>11</td>
<td>1.6469</td>
<td>1.5248</td>
</tr>
<tr>
<td>1024</td>
<td>NK-FETI-DP</td>
<td>111</td>
<td>9</td>
<td>2.6490</td>
<td>2.7986</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>70</td>
<td>13</td>
<td>1.8105</td>
<td>1.6541</td>
</tr>
</tbody>
</table>

where \(\tilde{\alpha} \in \{10^3, 10^6\}\). Again, we test for numerical scalability. In a first set of experiments with \(H/h = 16\), only standard arithmetic edge averages are used for the primal edges; see Table 3. Again, we see the typical behavior, i.e., in the nonlinear methods the number of Krylov iterations is reduced but the number of factorizations, i.e., the local work, is increased. We also see that increasing the coefficient \(\tilde{\alpha}\) from \(10^3\) to \(10^6\) affects all methods, i.e., the number of Krylov iterations grows. For the first time, we also see significant differences in the maximal condition numbers of the linearized systems. In the NK-FETI-DP approach, for \(\alpha = 10^6\), systems with very high condition numbers appear whereas in the nonlinear methods the condition numbers remain low.

Next, we consider a coarse space with weighted edge averages. The scaling is inspired by the weights used in edge averages of linear FETI-DP methods for problems with jumps not aligned with the interface; see \[30\].

Here, the weights in the weighted edge averages (see (5.3)) are defined as

\[
\rho_i := \frac{\int_E (\alpha(x) + \beta(x)) \varphi_i(x) \, dx}{\int_E \varphi_i(x) \, dx}
\]

in each degree of freedom \(x_i\) on the edge \(E\) with corresponding basis function \(\varphi_i\). The numerical scalability results are presented in Table 4. Again, the nonlinear methods reduce the number of Krylov iterations significantly, i.e., by a factor of between 2 and 7. The local work, as measured by the number of factorizations, is increased by a factor of up to 1.5. Comparing Table 3 with Table 4, we see that the scaling improves the results of all methods. The results from Table 4 also indicate that for the nonlinear methods, i.e., Nonlinear-FETI-DP-1 and Nonlinear-FETI-DP-2, we have robustness of the number of Krylov iterations with respect to the coefficient \(\tilde{\alpha}\). This does not seem to be the case for NK-FETI-DP; here the total number of Krylov
Table 4
Numerical scalability. Same set of experiments as in Table 3 but we used weighted edge averages instead of arithmetic ones.

<table>
<thead>
<tr>
<th>N</th>
<th>Solver</th>
<th>(\tilde{\alpha} = 10^3)</th>
<th>(\tilde{\alpha} = 10^6)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>#Krylov Iter</td>
<td>#Factor Max. cond.</td>
<td>Min. cond.</td>
</tr>
<tr>
<td>4</td>
<td>NK-FETI-DP</td>
<td>46 8 1.1915 1.1580</td>
<td>88 17 1.1161 1.1063</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>12 9 1.1915 1.1580</td>
<td>18 19 1.1116 1.1116</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>18 10 1.1916 1.1904</td>
<td>18 19 1.1201 1.1114</td>
</tr>
<tr>
<td>16</td>
<td>NK-FETI-DP</td>
<td>63 8 1.2588 1.1693</td>
<td>115 16 1.1562 1.1541</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>24 12 1.2588 1.2114</td>
<td>21 18 1.1539 1.1541</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>32 11 1.2582 1.2114</td>
<td>21 18 1.1531 1.1533</td>
</tr>
<tr>
<td>64</td>
<td>NK-FETI-DP</td>
<td>70 8 1.3414 1.2161</td>
<td>141 19 1.1669 1.1597</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>27 12 1.3414 1.3397</td>
<td>21 21 1.1591 1.1578</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>35 11 1.3526 1.2728</td>
<td>21 21 1.1601 1.1578</td>
</tr>
<tr>
<td>256</td>
<td>NK-FETI-DP</td>
<td>81 9 1.4586 1.3532</td>
<td>152 19 1.1716 1.1692</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>36 13 1.3977 1.3506</td>
<td>24 21 1.1716 1.1704</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>36 11 1.4426 1.3503</td>
<td>24 21 1.1738 1.1704</td>
</tr>
<tr>
<td>1024</td>
<td>NK-FETI-DP</td>
<td>75 8 1.7873 1.3626</td>
<td>144 18 1.1990 1.1990</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>42 14 2.4643 1.5127</td>
<td>24 21 1.2794 1.1907</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>40 11 1.5151 1.5000</td>
<td>22 21 1.1910 1.1819</td>
</tr>
<tr>
<td>4096</td>
<td>NK-FETI-DP</td>
<td>102 10 2.0980 1.4107</td>
<td>142 18 1.2317 1.1598</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>52 19 3.6975 1.6316</td>
<td>25 20 1.3335 1.2036</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>43 14 1.6402 1.5998</td>
<td>23 20 1.2227 1.1704</td>
</tr>
</tbody>
</table>

iterations increases. Note that at the same time the maximal condition number does not increase. The higher number of Krylov iterations is thus a result of slower Newton convergence in NK-FETI-DP.

5.2.4. Fourth model problem. Finally, in our fourth model problem, we present results for a nonlinear problem where some methods do not converge without a line search.

We have \(p = 4\) in the whole domain but we have high contrasts in the coefficient \(\alpha\) between the channels and the matrix material. We have

\[
\alpha(x) = \begin{cases} 
\tilde{\alpha} & \text{if } x \in \Omega_C, \\
1 & \text{elsewhere,}
\end{cases} \quad \beta(x) = 0 \text{ in } \Omega
\]

with \(\tilde{\alpha} \in \{10^3, 10^6\}\). We present numerical scalability results in Tables 5 and 6. Again we see a reduction of the number of Krylov iterations in the case of a well-chosen coarse space (see Table 6) where weighted edge averages are used; cf. (5.7). It can also be seen that NL-FETI-DP-2 needs a good coarse space to converge or, alternatively, an optimization of the step length as introduced in section 3.6.2. The latter increases the computational cost by some additional factorizations but ensures convergence and reduces the number of Krylov iterations; see Table 5.

5.2.5. Summary of the numerical results. From all our experiments with nonlinear FETI-DP, we can conclude that the nonlinear algorithms always reduce the number of Krylov iterations in comparison to NK-FETI-DP. The reduction can be substantial, i.e., up to 80% if we use Wolfe step length optimization and up to 90% if no globalization technique is used. This is an important advantage in highly nonlinear parallel simulations since a reduction of the number of Krylov iterations will also reduce the amount of communication between processors and nodes. Notably, the new methods show superior performance with respect to the total number of Krylov
### Table 5

Numerical scalability results for (5.4) with $\alpha$ and $\beta$ defined as in (5.8). We have $p = 4$ in the whole domain and in the channels a coefficient $\tilde{\alpha} = 10^6$; “without LS” means that no step length optimization is used and “with LS” means that we use Wolfe conditions in all Newton-type iterations, except the interface problem of Nonlinear-FETI-DP-2, where we use the algorithm from section 3.8.2. We have $N = 16$. The coarse space consists of vertex constraints and arithmetic edge averages; $N$ is the number of subdomains; #Krylov It. gives the sum of all Krylov iterations; #Factor. gives the total number of factorizations of $DK$ required (see also section 3.6.2); min./max. cond. gives the minimal and maximal condition number of the FETI-DP systems.

<table>
<thead>
<tr>
<th>$N$</th>
<th>Solver</th>
<th>Without LS</th>
<th>With LS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>#Krylov It.</td>
<td>#Factor.</td>
<td>Max. cond.</td>
</tr>
<tr>
<td>4</td>
<td>NK-FETI-DP</td>
<td>117</td>
<td>18</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>18</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>18</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>no conv.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>141</td>
<td>29</td>
</tr>
<tr>
<td>64</td>
<td>NK-FETI-DP</td>
<td>1317</td>
<td>22</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>no conv.</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>517</td>
<td>35</td>
</tr>
</tbody>
</table>

### Table 6

Numerical scalability. Same set of experiments as in Table 5, but using weighted edge averages instead of arithmetic ones; “without LS” means that no step length optimization is used and “with LS” means that we use Wolfe conditions in all Newton-type iterations, except in the interface problem of Nonlinear-FETI-DP-2, where we do not use any globalization strategy this time.

<table>
<thead>
<tr>
<th>$N$</th>
<th>Solver</th>
<th>Without LS</th>
<th>With LS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>#Krylov It.</td>
<td>#Factor.</td>
<td>Max. cond.</td>
</tr>
<tr>
<td>4</td>
<td>NK-FETI-DP</td>
<td>121</td>
<td>18</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>18</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>18</td>
<td>20</td>
</tr>
<tr>
<td>16</td>
<td>NK-FETI-DP</td>
<td>210</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>18</td>
<td>22</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>18</td>
<td>22</td>
</tr>
<tr>
<td>64</td>
<td>NK-FETI-DP</td>
<td>523</td>
<td>22</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-2</td>
<td>18</td>
<td>24</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-FETI-DP-1</td>
<td>20</td>
<td>24</td>
</tr>
</tbody>
</table>

iterations, not only for problems with localized nonlinearities but also for the standard p-Laplace problem.

For some problems, see the third and fourth model problem above, a sufficiently good coarse space has to be chosen. Then, we can also save Newton iterations and factorizations. It should be pointed out that in some cases, without a carefully chosen coarse space, the NL-FETI-DP-2 method does not converge without using an appropriate line search method. This is not a drawback since, in general, suitable globalization techniques are necessary for Newton-based iterative algorithms when applied to nonlinear problems. It also merely reflects the additional flexibility to influence the Newton convergence by the nonlinear FETI-DP coarse space. The choice of a good coarse space is thus even more vital than in linear problems. Our results suggest that known strategies for the construction of problem specific coarse spaces may carry over from the linear case. Finally, let us point out that in some cases the nonlinear algorithms clearly have smaller condition numbers than NK-FETI-DP.
Comparison of nonlinear FETI-DP and nonlinear BDDC methods. Same problem setting as in Table 1.

<table>
<thead>
<tr>
<th>N</th>
<th>Solver</th>
<th>p-Laplace inclusions</th>
<th>p-Laplace</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>#Krylov It.</td>
<td>#Factor</td>
</tr>
<tr>
<td>4</td>
<td>Nonlinear-FETI-DP-2</td>
<td>5</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-BDDC</td>
<td>16</td>
<td>23</td>
</tr>
<tr>
<td>16</td>
<td>Nonlinear-FETI-DP-2</td>
<td>21</td>
<td>18</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-BDDC</td>
<td>40</td>
<td>28</td>
</tr>
<tr>
<td>64</td>
<td>Nonlinear-FETI-DP-2</td>
<td>30</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-BDDC</td>
<td>44</td>
<td>30</td>
</tr>
<tr>
<td>256</td>
<td>Nonlinear-FETI-DP-2</td>
<td>31</td>
<td>22</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-BDDC</td>
<td>43</td>
<td>32</td>
</tr>
<tr>
<td>1024</td>
<td>Nonlinear-FETI-DP-2</td>
<td>31</td>
<td>24</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-BDDC</td>
<td>41</td>
<td>34</td>
</tr>
<tr>
<td>4096</td>
<td>Nonlinear-FETI-DP-2</td>
<td>19</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>Nonlinear-BDDC</td>
<td>36</td>
<td>36</td>
</tr>
</tbody>
</table>

5.3. Numerical results for the Nonlinear-BDDC method. Here, we present numerical results for the nonlinear BDDC method applied to some of the model problems from the previous subsection. We restrict ourselves to two of our model problems.

First, in Table 7, we present some results for the first model problem; see section 5.2.1. The setting is the same as in the experiments for Table 1. It can be seen that in Nonlinear-BDDC a higher number of Krylov iterations and factorizations is needed compared with Nonlinear-FETI-DP-2. Note that the solutions of the local nonlinear problems (4.12) that appear on the right-hand side of (4.11) are completely local to the subdomains and thus have a lower computational cost than the problems (3.12) and (3.18) that appear in nonlinear FETI-DP methods. Let us also note that the nonlinear BDDC and the nonlinear FETI-DP algorithms have a difference in the coarse space. That of the nonlinear FETI-DP algorithms is nonlinear whereas that of the nonlinear BDDC methods is linear. This might explain the difference in the number of Krylov iterations of the two methods.

In Table 8 we also present numerical results for Nonlinear-BDDC applied to the third model problem; see section 5.2.3. The setting is the same as in the experiments for Table 4. Again, a higher number of Krylov iterations and factorizations can be seen for Nonlinear-BDDC in comparison to Nonlinear-FETI-DP-2.

6. Conclusion. In this paper we have introduced new approaches to nonoverlapping domain decomposition, i.e., nonlinear FETI-DP and nonlinear BDDC methods. Whereas the algorithmic building blocks of these methods are largely identical to the standard, linear FETI-DP and BDDC methods, these building blocks are now arranged in a different way. In the new methods local nonlinear problems are solved. This additional iteration increases the local computational work but, at the same time, can lead to a significantly faster convergence of the global iteration, especially, but not only, for problems with strong localized nonlinearities. As a result, in the new algorithms, the need for global synchronization is reduced. Increasing the portion of local work will also help to contain soft errors, e.g., memory errors from radiation, locally, and may facilitate algorithmic level fault checking and fault resilience on future supercomputers. In all FETI-DP and BDDC methods, the communication is limited to the boundary of subdomains. Still, on future supercomputers at the exascale,
we expect that the communication cost occurring in the Krylov subspace iteration will eventually limit the performance of standard NK-FETI-DP or NK-BDDC approaches. In our numerical examples, the number of global Krylov subspace iterations is reduced significantly, i.e., by up to an order of magnitude, for our new methods.

REFERENCES


NONLINEAR FETI-DP AND BDDC METHODS


