The FETI Family of Domain Decomposition Methods for Inequality-Constrained Quadratic Programming: Application to Contact Problems with Conforming and Nonconforming Interfaces

Philip Avery† and Charbel Farhat†

† Department of Aeronautics and Astronautics
and Department of Mechanical Engineering
and Institute for Computational and Mathematical Engineering
Building 500, Room 501G, 488 Escondido Mall, Mailcode 3035
Stanford University, Stanford, CA 94305, USA
Tel: +1 650 723 3840; fax: +1 725 3525; e-mail: cfarhat@stanford.edu

Abstract

Two domain decomposition methods with Lagrange multipliers for solving iteratively quadratic programming problems with inequality constraints are presented. These methods are based on the FETI and FETI-DP substructuring algorithms. In the case of linear constraints, they do not perform any Newton-like iteration. Instead, they solve a constrained problem by an active set strategy and a generalized conjugate gradient based descent method equipped with controls to guarantee convergence monotonicity. Both methods possess the desirable feature of minimizing numerical oscillations during the iterative solution process. Performance results and comparisons are reported for several numerical simulations that suggest that both methods are numerically scalable with respect to both the problem size and the number of subdomains. Their parallel scalability is also illustrated on a Linux cluster for a complex 1.4 million degree of freedom multibody problem with frictionless contact and nonconforming discrete interfaces.

1 Introduction

Contact problems occur frequently in structural analysis, particularly when studying the assembly of substructures. They are characterized by inequality constraints such as non-penetration conditions, and an active area of contact that is unknown *a priori*.

Several approaches exist for solving frictionless contact problems [29, 45, 43] as well as contact problems with friction [7]. In most of these, the numerical algorithms that are employed for enforcing the contact constraints can be grouped into penalty and Lagrange multiplier methods [44].

Penalty methods [28, 2] are closely related to the regularization of the contact constraints. They are usually formulated in terms of the displacement variables and therefore are primal methods. They allow treating contact as a material behavior, as exemplified by the method of joint finite elements [1]. Penalty methods can experience various numerical difficulties, especially ill-conditioning, when a too large or too small penalty parameter is introduced.

Lagrange multiplier methods are dual methods where the multipliers, which represent the contact reaction forces, are introduced in order to enforce exactly the non-penetration conditions. These methods can be further classified into (a) trial and error approaches for determining the active zone of contact, (b) mathematical programming techniques [30, 26], (c) search gradient methods [36, 13, 38, 39], and (d) a combination of all of these approaches. Augmented Lagrange multiplier methods [3, 31, 37, 8] result in mixed formulations involving both displacement and force unknowns. The numerical solution schemes underlying both the Lagrange multiplier and augmented Lagrange multiplier methods are often related to the Uzawa algorithm [4, 42, 10].

Recently, various forms of domain decomposition have been introduced in the formulation of large-scale frictionless contact problems in order to speed up their solution on both serial and parallel computing platforms [9, 14, 6, 7, 15]. In particular, a domain decomposition method for the solution of frictionless contact problems with Lagrange multipliers based on the FETI (Finite Element Tearing and Interconnecting) [17, 23, 22, 34, 35, 21, 18] method was presented in [15]. This method – named FETI-C (C for contact) – employs a single iterative procedure in which both contact and equilibrium conditions are updated. A similar approach, but based on the FETI-DP (DP for Dual-Primal) domain decomposition method [20, 24] and therefore named FETI-DPC, was recently presented in [5].

In this paper an alternative optimization-style formulation of the FETI-C and FETI-DPC methods for frictionless contact problems with nonconforming subdomain interfaces is presented. The presentation also includes an important modification to the original algorithms that guarantees convergence monotonicity. To this effect, the paper begins by describing in Section 2 the FETI and FETI-DP methods for unconstrained and strictly convex quadratic minimization problems in order to keep this paper as self-contained as possible. Then, two FETI and FETI-DP based methods for the solution of convex quadratic programming problems with linear inequality constraints are presented in Section 3. Their performance is assessed in Section 4 for some frictionless contact problems. Their parallel scalability is also demonstrated on a Linux cluster for a complex 1.4 million degree of freedom contact problem with nonconforming discrete interfaces. Finally, Section 5 summarizes and concludes this paper.

2 Unconstrained minimization – static problems

The FETI [17, 23, 25, 21, 18] and FETI-DP methods [20] were originally developed for solving efficiently on parallel processors a large-scale linear or linearized system of equations arising from the finite element discretization of a structural mechanics problem defined on a *global* domain Ω_g. Such a system of equations can be written as

$$\boldsymbol{K_g u_g} = \boldsymbol{f_g} \tag{1}$$

where for a large class of applications, $\boldsymbol{K_g} \in \mathbb{R}^{n_g}$ is a symmetric positive definite generalized stiffness matrix constructed by direct assembly of element-level generalized stiffness matrices, $\boldsymbol{u_g} \in \mathbb{R}^{n_g}$ is a vector of generalized displacements, and $\boldsymbol{f_g} \in \mathbb{R}^{n_g}$ is a vector of prescribed generalized forces. The subscript g is used here to denote a global quantity or variable.

Problem (1) can alternatively be posed as the following unconstrained minimization quadratic programming (QP) problem

$$\boldsymbol{u_g} = \arg\min_{\boldsymbol{v_g} \in \mathbb{R}^{n_g}} f_g(\boldsymbol{v_g}) \tag{2}$$

where $f_g(\boldsymbol{v_g}) = \frac{1}{2}\boldsymbol{v_g}^{\mathrm{T}}\boldsymbol{K_g v_g} - \boldsymbol{v_g}^{\mathrm{T}}\boldsymbol{f_g}$ is the objective function, $\boldsymbol{v_g}$ is the optimization variable, and $\boldsymbol{u_g}$ is the optimal point.

2.1 The FETI method

Consider a given set of finite elements representing a discretization of a global domain Ω_g with degrees of freedom (dofs) numbered globally from 1 to n_g. This global finite element set is *decomposed* into N_s disjoint subsets of elements, each representing a discretization of a subdomain Ω^s with dofs numbered locally from 1 to n^s.

The FETI method essentially reformulates the initial unconstrained minimization problem (2) as an equivalent constrained minimization problem expressed in terms of the local generalized displacement unknowns $u = (\boldsymbol{u}^1, \ldots, \boldsymbol{u}^{N_s})$, local variables, and local knowns defined at the subdomain level. This equivalent problem takes the equality-constrained QP form

$$u = \arg\min_{v \in B} f(v), \quad B = \left\{ v \in \prod_{s=1}^{s=N_s} \mathbb{R}^{n^s} : \sum_{s=1}^{s=N_s} \boldsymbol{B}^s \boldsymbol{v}^s = \boldsymbol{0} \right\} \tag{3}$$

where

$$f(v) = \sum_{s=1}^{s=N_s} \left(\frac{1}{2} \boldsymbol{v}^{s^{\mathrm{T}}} \boldsymbol{K}^s \boldsymbol{v}^s - \boldsymbol{f}^{s^{\mathrm{T}}} \boldsymbol{v}^s \right) \tag{4}$$

$\boldsymbol{B}^s$ is the signed Boolean matrix that extracts from a subdomain vector such as $\boldsymbol{v}^s$ its signed ($\pm$) restriction to the subdomain interface boundary, $\boldsymbol{K}^s \in \mathbb{R}^{n^s}$ is the local symmetric positive semidefinite generalized stiffness matrix of subdomain Ω^s constructed by direct assembly of the local and element-level generalized stiffness matrices, and $\boldsymbol{f}^s$

is a local vector of prescribed generalized forces constructed by direct assembly of local and element-level vectors of prescribed generalized forces (for example, body forces or surface tractions) and/or extraction of the local subdomain contribution to the global vector of discrete prescribed generalized forces.

Problem (3) admits a dual formulation that is described in Section 2.1.1 and for which an iterative solution method is overviewed in Section 2.1.2.

2.1.1 Dualization

The Lagrangian function associated with problem (3) is formed by augmenting the objective function with a weighted sum of the constraint functions as follows

$$L(v,\boldsymbol{\mu}) = f(v) + \boldsymbol{\mu}^{\mathrm{T}} \sum_{s=1}^{s=N_s} \boldsymbol{B}^s \boldsymbol{v}^s \tag{5}$$

$$= \sum_{s=1}^{s=N_s} \left(\frac{1}{2} \boldsymbol{v}^{s^{\mathrm{T}}} \boldsymbol{K}^s \boldsymbol{v}^s - \boldsymbol{f}^{s^{\mathrm{T}}} \boldsymbol{v}^s \right) + \boldsymbol{\mu}^{\mathrm{T}} \sum_{s=1}^{s=N_s} \boldsymbol{B}^s \boldsymbol{v}^s \tag{6}$$

μ_i is referred to as the Lagrange multiplier associated with the i^{th} constraint and $\boldsymbol{\mu}$ as the l-long vector of l Lagrange multipliers or dual variables.

The Lagrange dual function is defined as the minimum value of the Lagrangian function over v

$$h(\boldsymbol{\mu}) = \inf_{v} L(v,\boldsymbol{\mu}) \tag{7}$$

which is

(a) by definition $-\infty$ when the Lagrangian function L is unbounded from below in v; observing that L has a quadratic form in v with a symmetric positive semidefinite coefficient matrix, it follows that L is bounded from below in v if and only if $\boldsymbol{R}^{s^{\mathrm{T}}}(\boldsymbol{f}^s - \boldsymbol{B}^{s^{\mathrm{T}}}\boldsymbol{\mu}) = \boldsymbol{0}$ for all subdomains, where the columns of $\boldsymbol{R}^s \in \mathbb{R}^{n^s \times q^s}$ span the left null space of $\boldsymbol{K}^s$ and represent the rigid body modes of subdomain Ω^s,

(b) otherwise attained when the gradient of the Lagrangian function L with respect to v is zero, specifically when $\boldsymbol{v}^s = \boldsymbol{K}^{s^+}(\boldsymbol{f}^s - \boldsymbol{B}^{s^{\mathrm{T}}}\boldsymbol{\mu}) + \boldsymbol{R}^s\boldsymbol{\beta}^s$ for all subdomains, where the vector $\boldsymbol{\beta}^s \in \mathbb{R}^{q^s}$ represents the amplitudes of the rigid body modes of subdomain Ω^s and the superscript $+$ denotes the generalized inverse of a matrix.

After some algebraic manipulations, the Lagrange dual function can be re-written as

$$h(\boldsymbol{\mu}) = \begin{cases} -\frac{1}{2}\boldsymbol{\mu}^{\mathrm{T}}\boldsymbol{F}\boldsymbol{\mu} + \boldsymbol{d}^{\mathrm{T}}\boldsymbol{\mu} - c & \text{if } \boldsymbol{G}^{\mathrm{T}}\boldsymbol{\mu} = \boldsymbol{e} \\ -\infty & \text{otherwise} \end{cases} \tag{8}$$

where

$$
\begin{aligned}
\boldsymbol{F} &= \sum_{s=1}^{s=N_s} \boldsymbol{B}^s \boldsymbol{K}^{s^+} \boldsymbol{B}^{s^{\mathrm{T}}} \\
\boldsymbol{d} &= \sum_{s=1}^{s=N_s} \boldsymbol{B}^s \boldsymbol{K}^{s^+} \boldsymbol{f}^s \\
c &= \frac{1}{2} \sum_{s=1}^{s=N_s} \boldsymbol{f}^{s^{\mathrm{T}}} \boldsymbol{K}^{s^+} \boldsymbol{f}^s \\
\boldsymbol{G} &= \begin{bmatrix} \boldsymbol{B}^1 \boldsymbol{R}^1 & \cdots & \boldsymbol{B}^{N_s} \boldsymbol{R}^{N_s} \end{bmatrix} \\
\boldsymbol{e} &= \begin{bmatrix} \boldsymbol{f}^{1^{\mathrm{T}}} \boldsymbol{R}^1 & \cdots & \boldsymbol{f}^{N_s{}^{\mathrm{T}}} \boldsymbol{R}^{N_s} \end{bmatrix}^{\mathrm{T}}
\end{aligned}
$$

Consider now the optimization problem associated with the following question: what is the vector of Lagrange multipliers for which the dual function gives the best lower bound on the optimal value $f(u)$? This is the so-called Lagrange dual problem which can be written as

$$\boldsymbol{\lambda} = \arg\max_{\boldsymbol{\mu} \in \mathbb{R}^l} h(\boldsymbol{\mu}) \tag{9}$$

where $h(\boldsymbol{\mu})$ is the dual objective function, $\boldsymbol{\mu}$ is the dual optimization variable and $\boldsymbol{\lambda}$ is the dual optimal point. In this context and to avoid confusion, (3) is referred to as the primal problem, $f(v)$ as the primal objective function, v as the primal optimization variable, u as the primal optimal point, and B as the primal feasible set.

For optimization problems of the form given in (3), it can be established that points at which the dual function is $-\infty$ will not be dual optimal. The Lagrange dual problem may therefore be equivalently stated in the equality-constrained QP form

$$\boldsymbol{\lambda} = \arg\min_{\boldsymbol{\mu} \in G} g(\boldsymbol{\mu}), \quad G = \left\{ \boldsymbol{\mu} \in \mathbb{R}^l : \boldsymbol{G}^{\mathrm{T}} \boldsymbol{\mu} = \boldsymbol{e} \right\} \tag{10}$$

where $g(\boldsymbol{\mu}) = \boldsymbol{\mu}^{\mathrm{T}} \boldsymbol{F} \boldsymbol{\mu} - \boldsymbol{d}^{\mathrm{T}} \boldsymbol{\mu}$ and is referred to – by slightly abusing the terminology – as the dual objective function, and G is referred to as the dual feasible set.

2.1.2 Solving the dual problem

Let $\boldsymbol{\mu} = \boldsymbol{\lambda}^0 + \boldsymbol{P} \boldsymbol{M}^{-\frac{1}{2}} \bar{\boldsymbol{\mu}}$ where $\boldsymbol{\lambda}^0$ is any dual feasible point, $\boldsymbol{P} = \boldsymbol{I} - \boldsymbol{G}(\boldsymbol{G}^{\mathrm{T}}\boldsymbol{G})^{-1}\boldsymbol{G}^{\mathrm{T}}$ be the matrix defined so that $\boldsymbol{P}\boldsymbol{x}^{\boldsymbol{0}}$ is the orthogonal projection of $\boldsymbol{x}^{\boldsymbol{0}}$ on to the left null space of $\boldsymbol{G}$ – that is, the subspace $\{\boldsymbol{x} \in \mathbb{R}^l : \boldsymbol{G}^{\mathrm{T}}\boldsymbol{x} = \boldsymbol{0}\}$ – and $\boldsymbol{M}^{-1} \in \mathbb{R}^{l \times l}$ a symmetric positive definite preconditioner chosen such that $\boldsymbol{M}^{-1} \approxeq (\boldsymbol{P}^{\mathrm{T}}\boldsymbol{F}\boldsymbol{P})^{-1}$. Using this change of variable, the constrained minimization QP problem (10) is transformed into the following two independent and simpler problems:

(a) the convex feasibility problem (CFP) $\boldsymbol{G}^{\mathrm{T}}\boldsymbol{\lambda} = \boldsymbol{e}$, which in this case may be solved simply as $\boldsymbol{\lambda}^0 = \boldsymbol{G}(\boldsymbol{G}^{\mathrm{T}}\boldsymbol{G})^{-1}\boldsymbol{e}$, and

(b) the unconstrained QP problem

$$\bar{\boldsymbol{\lambda}} = \underset{\bar{\boldsymbol{\mu}} \in \mathbb{R}^l}{\arg\min}\, \bar{g}(\bar{\boldsymbol{\mu}}) \tag{11}$$

where $\bar{g}(\bar{\boldsymbol{\mu}}) = \frac{1}{2}\bar{\boldsymbol{\mu}}^{\mathrm{T}}\bar{\boldsymbol{F}}\bar{\boldsymbol{\mu}} - \bar{\boldsymbol{d}}^{\mathrm{T}}\bar{\boldsymbol{\mu}} + \bar{c}$, and

$$\begin{aligned} \bar{\boldsymbol{F}} &= \boldsymbol{M}^{-\frac{1}{2}^{\mathrm{T}}}\boldsymbol{P}^{\mathrm{T}}\boldsymbol{F}\boldsymbol{P}\boldsymbol{M}^{-\frac{1}{2}} \\ \bar{\boldsymbol{d}} &= \boldsymbol{M}^{-\frac{1}{2}^{\mathrm{T}}}\boldsymbol{P}^{\mathrm{T}}(\boldsymbol{d} - \boldsymbol{F}\boldsymbol{\lambda}^0) \\ \bar{c} &= \frac{1}{2}\boldsymbol{\lambda}^{0^{\mathrm{T}}}\boldsymbol{F}\boldsymbol{\lambda}^0 - \boldsymbol{d}^{\mathrm{T}}\boldsymbol{\lambda}^0 \end{aligned}$$

that can be solved by the well-known Conjugate Gradient (CG) algorithm.

The solution of problem (10) can then be obtained directly from the solutions of the above two subproblems as $\boldsymbol{\lambda} = \boldsymbol{P}\boldsymbol{M}^{-\frac{1}{2}}\bar{\boldsymbol{\lambda}} + \boldsymbol{\lambda}^0$. This solution approach can also be written in the form of a Preconditioned Conjugate Projected Gradient (PCPG) algorithm and is known as the FETI method.

2.2 The FETI-DP method

The FETI-DP method is a variant of the FETI method in which continuity of some or all components of the displacement field at some designated "corner" interface nodes [20] is exactly satisfied at each CG iteration. The global and local vectors of generalized displacements can be written as column block vectors

$$\boldsymbol{v_g} = \begin{bmatrix} \boldsymbol{v}_{g_r} \\ \boldsymbol{v}_{g_c} \end{bmatrix} \tag{12}$$

$$\boldsymbol{v}^s = \begin{bmatrix} \boldsymbol{v}_r^s \\ \boldsymbol{v}_c^s \end{bmatrix} \tag{13}$$

where the subscripts c, r denote the corner and non-corner – or "remainder" – sub-vectors, respectively. Then, the local generalized stiffness matrices can be written as block matrices as follows

$$\boldsymbol{K}^s = \begin{bmatrix} \boldsymbol{K}_{rr}^s & \boldsymbol{K}_{rc}^s \\ \boldsymbol{K}_{rc}^{s^{\mathrm{T}}} & \boldsymbol{K}_{cc}^s \end{bmatrix} \tag{14}$$

Each sub-matrix $\boldsymbol{K}_{rr}^s$ is guaranteed to be symmetric positive definite by selecting sufficient and appropriate corners [33].

The problem of interest can now be reformulated in terms of the local subdomain remainder unknowns $v_r = (\boldsymbol{v}_r^1, \ldots, \boldsymbol{v}_r^{N_s})$ and global corner unknowns $\boldsymbol{v}_{g_c}$, and in the form of an equality-constrained QP problem as follows

$$\begin{aligned} &(u_r, \boldsymbol{u}_{g_c}) = \underset{(v_r, \boldsymbol{v}_{g_c}) \in B^*}{\arg\min}\; f^*(v_r, \boldsymbol{v}_{g_c}), \\ &B^* = \left\{ (v_r, \boldsymbol{v}_{g_c}) \in \prod_{s=1}^{s=N_s} \mathbb{R}^{n_r^s} \times \mathbb{R}^{n_{cg}} : \sum_{s=1}^{s=N_s} \boldsymbol{B}_{rr}^s \boldsymbol{v}_r^s = \boldsymbol{0} \right\} \end{aligned} \tag{15}$$

where

$$f^*(v_r, \boldsymbol{v}_{\boldsymbol{g}_c}) = \sum_{s=1}^{s=N_s} \left(\frac{1}{2} \begin{bmatrix} \boldsymbol{v}_r^{s\mathrm{T}} & \boldsymbol{v}_{\boldsymbol{g}_c}^{\mathrm{T}} \boldsymbol{L}_{cc}^{s\mathrm{T}} \end{bmatrix} \begin{bmatrix} \boldsymbol{K}_{rr}^s & \boldsymbol{K}_{rc}^s \\ \boldsymbol{K}_{rc}^{s\mathrm{T}} & \boldsymbol{K}_{cc}^s \end{bmatrix} \begin{bmatrix} \boldsymbol{v}_r^s \\ \boldsymbol{L}_{cc}^s \boldsymbol{v}_{\boldsymbol{g}_c} \end{bmatrix} - \begin{bmatrix} \boldsymbol{v}_r^{s\mathrm{T}} & \boldsymbol{v}_{\boldsymbol{g}_c}^{\mathrm{T}} \boldsymbol{L}_{cc}^{s\mathrm{T}} \end{bmatrix} \begin{bmatrix} \boldsymbol{f}_r^s \\ \boldsymbol{f}_c^s \end{bmatrix} \right)$$

$\boldsymbol{L}_{cc}^s$ is the restriction to the corner dofs of the Boolean global to local map of subdomain Ω^s defined such that $\boldsymbol{v}_c^s = \boldsymbol{L}_{cc}^s \boldsymbol{v}_{\boldsymbol{g}_c}$, and $\boldsymbol{B}_{rr}^s$ is the restriction to the remainder dofs of $\boldsymbol{B}^s$.

2.2.1 Dualization

The Lagrangian and Lagrange dual functions associated with problem (15) are defined as

$$L^*(v_r, \boldsymbol{v}_{\boldsymbol{g}_c}, \boldsymbol{\mu}_r) = f^*(v_r, \boldsymbol{v}_{\boldsymbol{g}_c}) + \boldsymbol{\mu}_r^{\mathrm{T}} \sum_{s=1}^{s=N_s} \boldsymbol{B}_{rr}^s \boldsymbol{v}_r^s$$

and

$$h^*(\boldsymbol{\mu}_r) = \inf_{(v_r, \boldsymbol{v}_{\boldsymbol{g}_c})} L^*(v_r, \boldsymbol{v}_{\boldsymbol{g}_c}, \boldsymbol{\mu}_r) \tag{16}$$

respectively. As in the formulation of the FETI dual problem (Section 2.1.1), the Lagrange function has a quadratic form in the primal variable – in this case $(v_r, \boldsymbol{v}_{\boldsymbol{g}_c})$ – but now with a symmetric *positive definite* coefficient matrix rather than positive semidefinite. The infimum of the Lagrangian function is therefore simply attained when the gradient with respect to $(v_r, \boldsymbol{v}_{\boldsymbol{g}_c})$ is zero. Hence, the Lagrange dual function h^* can be written as

$$h^*(\boldsymbol{\mu}_r) = -\frac{1}{2} \boldsymbol{\mu}_r^{\mathrm{T}} \boldsymbol{F}_{rr}^* \boldsymbol{\mu}_r + \boldsymbol{\mu}_r^{\mathrm{T}} \boldsymbol{d}_r^* - c_r^* \tag{17}$$

where

$$\begin{aligned}
\boldsymbol{F}_{rr}^* &= \sum_{s=1}^{s=N_s} \boldsymbol{B}_{rr}^s \boldsymbol{K}_{rr}^{s^{-1}} \left(\boldsymbol{I} - \boldsymbol{K}_{rc}^s \boldsymbol{L}_{cc}^s \boldsymbol{K}_{cc}^{*^{-1}} \sum_{s=1}^{s=N_s} \boldsymbol{L}_{cc}^{s\mathrm{T}} \boldsymbol{K}_{rc}^{s\mathrm{T}} \boldsymbol{K}_{rr}^{s^{-1}} \right) \boldsymbol{B}_{rr}^{s\mathrm{T}} \\
\boldsymbol{d}_r^* &= \sum_{s=1}^{s=N_s} \boldsymbol{B}_{rr}^s \boldsymbol{K}_{rr}^{s^{-1}} \left(\boldsymbol{f}_r^s - \boldsymbol{K}_{rc}^s \boldsymbol{L}_{cc}^s \boldsymbol{K}_{cc}^{*^{-1}} \sum_{s=1}^{s=N_s} \boldsymbol{L}_{cc}^{s\mathrm{T}} \left(\boldsymbol{f}_c^s - \boldsymbol{K}_{rc}^{s\mathrm{T}} \boldsymbol{K}_{rr}^{s^{-1}} \boldsymbol{f}_r^s \right) \right) \\
c_r^* &= \sum_{s=1}^{s=N_s} \left(\frac{1}{2} \boldsymbol{f}_r^{s\mathrm{T}} \boldsymbol{K}_{rr}^s \boldsymbol{f}_r^s + \boldsymbol{f}_r^{s\mathrm{T}} \boldsymbol{K}_{rc}^s \boldsymbol{L}_{cc}^{s\mathrm{T}} \boldsymbol{f}_c^s + \frac{1}{2} \boldsymbol{f}_c^{s\mathrm{T}} \boldsymbol{K}_{cc}^s \boldsymbol{f}_c^s \right)
\end{aligned}$$

and

$$\boldsymbol{K}_{cc}^* = \sum_{s=1}^{s=N_s} \boldsymbol{L}_{cc}^{s\mathrm{T}} \left(\boldsymbol{K}_{cc}^s - \boldsymbol{K}_{rc}^{s\mathrm{T}} \boldsymbol{K}_{rr}^{s^{-1}} \boldsymbol{K}_{rc}^s \right) \boldsymbol{L}_{cc}^s$$

The FETI-DP dual problem takes then the unconstrained QP form

$$\boldsymbol{\lambda}_r = \arg\max_{\boldsymbol{\mu}_r \in \mathbb{R}^{l_r}} h^*(\boldsymbol{\mu}_r) \tag{18}$$

or, equivalently, the minimization problem

$$\boldsymbol{\lambda}_r = \underset{\boldsymbol{\mu}_r \in \mathbb{R}^{l_r}}{\arg\min}\, g^*(\boldsymbol{\mu}_r) \tag{19}$$

where $g^*(\boldsymbol{\mu}_r) = \frac{1}{2}\boldsymbol{\mu}_r^{\mathrm{T}} \boldsymbol{F}^*_{rr} \boldsymbol{\mu}_r - \boldsymbol{\mu}_r^{\mathrm{T}} \boldsymbol{d}^*_r$.

2.2.2 Solving the dual problem

In the FETI-DP method, the solution of the dual problem (19) is obtained by a Preconditioned Conjugate Gradient (PCG) algorithm. The preconditioning step can be viewed as resulting from the change of variable $\boldsymbol{\mu}_r = \boldsymbol{M}_{rr}^{-\frac{1}{2}} \bar{\boldsymbol{\mu}}_r$ where $\boldsymbol{M}_{rr}^{-1} \in \mathbb{R}^{l_r \times l_r}$ is a symmetric positive definite preconditioning matrix chosen such that $\boldsymbol{M}_{rr}^{-1} \approxeq \boldsymbol{F}_{rr}^{*^{-1}}$.

3 Constrained minimization – contact problems

As described in Section 2, the classical FETI and FETI-DP methods can be interpreted as numerical algorithms for solving unconstrained minimization problems. In this section, these two methods are extended for

(a) constrained minimization QP problems with linear inequality constraints of the form

$$\boldsymbol{u_g} = \underset{\boldsymbol{v_g} \in \boldsymbol{C_g}}{\arg\min}\, f_g(\boldsymbol{v_g}), \quad C_g = \{\boldsymbol{v_g} \in \mathbb{R}^{n_g} : \boldsymbol{C_g v_g} \leq \boldsymbol{a}\} \tag{20}$$

where $\boldsymbol{C_g} \in \mathbb{R}^{m \times n_g}$ and $\boldsymbol{a} \in \mathbb{R}^m$,

(b) symmetric positive semidefinite stiffness matrices $\boldsymbol{K_g}$.

Throughout the remainder of this paper, it is assumed that at least one feasible point exists, and f_g is bounded from below on the feasible set.

The following subsections describe a numerical algorithm based on the FETI method for solving the domain decomposed minimization problem formulated above.

3.1 The re-designed FETI-C method

An equivalent problem to (20) can be formulated in terms of the local subdomain knowns and unknowns as follows

$$u = \underset{v \in \widehat{B}}{\arg\min}\, f(v), \quad \widehat{B} = \left\{ v \in \prod_{s=1}^{s=N_s} \mathbb{R}^{n^s} : \sum_{s=1}^{s=N_s} \boldsymbol{B}^s \boldsymbol{v}^s = \boldsymbol{0},\ \sum_{s=1}^{s=N_s} \boldsymbol{C}^s \boldsymbol{v}^s \leq \boldsymbol{a} \right\} \tag{21}$$

where $\boldsymbol{C}^s \in \mathbb{R}^{m \times n^s}$ is constructed by extracting the local subdomain contributions to the global constraint matrix $\boldsymbol{C_g}$.

3.1.1 Dualization

The Lagrangian and Lagrange dual functions associated with problem (21) are defined as

$$\hat{L}(v, \boldsymbol{\mu}, \boldsymbol{\xi}) = f(v) + \boldsymbol{\mu}^{\mathrm{T}} \sum_{s=1}^{s=N_s} \boldsymbol{B}^s \boldsymbol{v}^s + \boldsymbol{\xi}^{\mathrm{T}} \left(\sum_{s=1}^{s=N_s} \boldsymbol{C}^s \boldsymbol{v}^s - \boldsymbol{a} \right)$$

and

$$\hat{h}(\boldsymbol{\mu}, \boldsymbol{\xi}) = \inf_{v} \hat{L}(v, \boldsymbol{\mu}, \boldsymbol{\xi})$$

respectively. The infimum of the Lagrangian function is

(a) by definition $-\infty$ when the Lagrangian function $\hat{L}$ is unbounded from below in v; observing that $\hat{L}$ has a quadratic form in v with a symmetric positive semidefinite coefficient matrix, it follows that $\hat{L}$ is bounded from below in v if and only if $\boldsymbol{R}^{s^{\mathrm{T}}}(\boldsymbol{f}^s - \boldsymbol{B}^{s^{\mathrm{T}}}\boldsymbol{\mu} - \boldsymbol{C}^{s^{\mathrm{T}}}\boldsymbol{\xi}) = \boldsymbol{0}$ for all subdomains,

(b) otherwise attained when the gradient of the Lagrangian function $\hat{L}$ with respect to v is zero – that is, when $\boldsymbol{v}^s = \boldsymbol{K}^{s^+}(\boldsymbol{f}^s - \boldsymbol{B}^{s^{\mathrm{T}}}\boldsymbol{\mu} - \boldsymbol{C}^{s^{\mathrm{T}}}\boldsymbol{\xi}) + \boldsymbol{R}^s\boldsymbol{\beta}^s$ for all subdomains.

Hence, the Lagrange dual function can be re-written as

$$\hat{h}(\boldsymbol{\mu}, \boldsymbol{\xi}) = \begin{cases} -\frac{1}{2}\hat{\boldsymbol{\mu}}^{\mathrm{T}}\widehat{\boldsymbol{F}}\hat{\boldsymbol{\mu}} + \hat{\boldsymbol{d}}^{\mathrm{T}}\hat{\boldsymbol{\mu}} - c & \text{if } \widehat{\boldsymbol{G}}^{\mathrm{T}}\hat{\boldsymbol{\mu}} = \boldsymbol{e} \\ -\infty & \text{otherwise} \end{cases} \tag{22}$$

where $\hat{\boldsymbol{\mu}} = [\boldsymbol{\mu}^{\mathrm{T}}\ \boldsymbol{\xi}^{\mathrm{T}}]^{\mathrm{T}}$, $\widehat{\boldsymbol{B}}^s = [\boldsymbol{B}^{s^{\mathrm{T}}}\ \boldsymbol{C}^{s^{\mathrm{T}}}]^{\mathrm{T}}$, and

$$\begin{aligned} \widehat{\boldsymbol{F}} &= \sum_{s=1}^{s=N_s} \widehat{\boldsymbol{B}}^s \boldsymbol{K}^{s^+} \widehat{\boldsymbol{B}}^{s^{\mathrm{T}}} \\ \hat{\boldsymbol{d}} &= \sum_{s=1}^{s=N_s} \widehat{\boldsymbol{B}}^s \boldsymbol{K}^{s^+} \boldsymbol{f}^s - \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{a} \end{bmatrix} \\ \widehat{\boldsymbol{G}} &= \begin{bmatrix} \widehat{\boldsymbol{B}}^1 \boldsymbol{R}^1 & \cdots & \widehat{\boldsymbol{B}}^{N_s} \boldsymbol{R}^{N_s} \end{bmatrix} \end{aligned}$$

The above dual problem has the form

$$(\boldsymbol{\lambda}, \boldsymbol{\nu}) = \arg\max_{(\boldsymbol{\mu}, \boldsymbol{\xi}) \in H} \hat{h}(\boldsymbol{\mu}, \boldsymbol{\xi}), \quad H = \left\{ (\boldsymbol{\mu}, \boldsymbol{\xi}) \in \mathbb{R}^l \times \mathbb{R}^m : \xi_i \geq 0, i = 1, \ldots, m \right\} \tag{23}$$

The points for which the dual function is $-\infty$ are not dual optimal. Therefore, the Lagrange dual problem can be equivalently stated as a QP problem with both equality and inequality constraints, the latter having the particular form of simple lower bounds on a subset of the optimization variables, as follows

$$\hat{\boldsymbol{\lambda}} = \arg\min_{\hat{\boldsymbol{\mu}} \in \widehat{G}} \hat{g}(\hat{\boldsymbol{\mu}}), \quad \widehat{G} = \left\{ \hat{\boldsymbol{\mu}} \in \mathbb{R}^{l+m} : \widehat{\boldsymbol{G}}^{\mathrm{T}}\hat{\boldsymbol{\mu}} = \boldsymbol{e}, \hat{\mu}_{l+i} \geq 0, i = 1, \ldots, m \right\} \tag{24}$$

where $\hat{g}(\hat{\boldsymbol{\mu}}) = \frac{1}{2}\hat{\boldsymbol{\mu}}^{\mathrm{T}}\widehat{\boldsymbol{F}}\hat{\boldsymbol{\mu}} - \hat{\boldsymbol{d}}^{\mathrm{T}}\hat{\boldsymbol{\mu}}$, and $\hat{\boldsymbol{\lambda}} = [\boldsymbol{\lambda}^{\mathrm{T}}\ \boldsymbol{\nu}^{\mathrm{T}}]^{\mathrm{T}}$.

3.1.2 Solving the dual problem

The Karush-Kuhn-Tucker conditions [27, 32], which are necessary and sufficient conditions for the existence of a solution of problem (24), can be written as

$$\widehat{\boldsymbol{F}}\hat{\boldsymbol{\lambda}} + \widehat{\boldsymbol{G}}\boldsymbol{\alpha} + \begin{bmatrix}\mathbf{0}\\ \boldsymbol{\gamma}\end{bmatrix} = \hat{\boldsymbol{d}} \tag{25a}$$

$$\widehat{\boldsymbol{G}}^{\mathrm{T}}\hat{\boldsymbol{\lambda}} = \boldsymbol{e} \tag{25b}$$

$$\hat{\lambda}_{l+i} \geq 0,\ i = 1, \ldots, m \tag{25c}$$

$$\gamma_i \hat{\lambda}_{l+i} = 0,\ i = 1, \ldots, m \tag{25d}$$

$$\gamma_i \leq 0,\ i = 1, \ldots, m \tag{25e}$$

where $\boldsymbol{\alpha} \in \mathbb{R}^q$ and $\boldsymbol{\gamma} \in \mathbb{R}^m$ are the vectors of Lagrange multipliers associated with the *dual* equality and inequality constraints at the optimal point.

In this section, an iterative descent method utilizing an active-set strategy for computing a solution satisfying the optimality conditions of the Lagrange dual problem (24) is presented.

A dual constraint is said here to be active if its equality form is satisfied for a given dual feasible point. If the statuses of the dual constraints at the optimal point was known a priori, problem (24) would simplify to an equality-constrained QP problem and could be solved by a PCPG algorithm. Since this is not usually the case, an active-set strategy is incorporated within the PCPG algorithm. It begins by predicting which constraints are active at the optimal point then proceeds to minimize the resulting equality-constrained QP problem using PCPG iterations until

(a) an infeasible iterate is generated,

(b) the solution of the equality-constrained QP problem is determined by some metric to be converging toward a nonoptimal point, or

(c) the optimality conditions are satisfied.

In cases (a) and (b), a new prediction of the active set is made and the iterative process is repeated. This prediction is henceforth called the "working set".

Furthermore, as will be explained in Section 3.1.4, the iterative descent method proposed for the solution of (25) is designed so that the sequence of values taken by the dual function at successive iterates is monotone descending – that is, $\hat{g}(\hat{\boldsymbol{\lambda}})^{k+1} \leq \hat{g}(\hat{\boldsymbol{\lambda}})^k$ – which guarantees convergence in a finite number of iterations.

3.1.3 Preconditioned conjugate projected gradient iterations

Let $\Gamma_C = \{l+1, l+2, \ldots, l+m\}$ denote the index set for the Lagrange multipliers corresponding to the primal inequality constraints, and $\tilde{\Gamma}_C^k = \{M \in \Gamma_C : \hat{\lambda}_M^k = 0\}$ the dual active index subset at the point $\hat{\boldsymbol{\lambda}}^k$. The working set $\tilde{\tilde{\Gamma}}_C^k$ at the point $\hat{\boldsymbol{\lambda}}^k$ will be defined in Sections 3.1.4 and 3.1.5 according to a specific strategy; for now it is noted that it must be a subset the active index set $\tilde{\Gamma}_C^k$.

For a given working set $\tilde{\tilde{\Gamma}}_C^k$, the PCPG sequence is generated by introducing in (24) the change of variable $\hat{\boldsymbol{\mu}} = \hat{\boldsymbol{\lambda}}^0 + \boldsymbol{P_A}\widehat{\boldsymbol{M}}^{-1}\hat{\boldsymbol{\mu}}$, as described in Section 2.1 for unconstrained

problems. However in this case, $\hat{\boldsymbol{\lambda}}^0$ is a feasible point and $P_A(\boldsymbol{x^0}) = \boldsymbol{P_A x^0}$ is the orthogonal projection of $\boldsymbol{x^0}$ onto the subspace

$$A = \{\boldsymbol{x} \in \mathbb{R}^{l+m} : \widehat{\boldsymbol{G}}^{\mathrm{T}} \boldsymbol{x} = \boldsymbol{0}\} \cap C \tag{26}$$

where

$$C = \{\boldsymbol{x} \in \mathbb{R}^{l+m} : x_M = 0 \text{ for all } M \in \tilde{\tilde{\Gamma}}_C^k\} \tag{27}$$

The projection matrix $\boldsymbol{P_A}$ can be written in the form

$$\boldsymbol{P_A} = \boldsymbol{P_C}\left(\mathbf{I} - \widehat{\boldsymbol{G}}(\widehat{\boldsymbol{G}}^{\mathrm{T}}\boldsymbol{P_C}\widehat{\boldsymbol{G}})^{+}\widehat{\boldsymbol{G}}^{\mathrm{T}}\right)\boldsymbol{P_C} \tag{28}$$

where $P_C(\boldsymbol{x^0}) = \boldsymbol{P_C x^0}$ is the orthogonal projection of $\boldsymbol{x^0}$ onto the subspace C. The diagonal projection matrix $\boldsymbol{P_C}$ can be defined component-wise as

$$[\boldsymbol{P_C}]_{(M,M)} = \begin{cases} 0 & \text{for all } M \in \tilde{\tilde{\Gamma}}_C^k \\ 1 & \text{otherwise} \end{cases} \tag{29}$$

Also, $\widehat{\boldsymbol{M}}^{-1} \in \mathbb{R}^{(l+m)\times(l+m)}$ is the symmetric positive definite preconditioning matrix chosen such that $\widehat{\boldsymbol{M}}^{-1} \approxeq (\boldsymbol{P_A}^{\mathrm{T}}\widehat{\boldsymbol{F}}\boldsymbol{P_A})^{-1}$. In this work, $\widehat{\boldsymbol{M}}^{-1}$ is chosen as the following extension of the optimal Dirichlet preconditioner [22] equipped with the topological scaling [40]

$$\widehat{\boldsymbol{M}}^{-1} = \boldsymbol{W}\sum_{s=1}^{s=N_s} \boldsymbol{P_C}\widehat{\boldsymbol{B}}^s \begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{S}_{bb}^s \end{bmatrix} \widehat{\boldsymbol{B}}^{s^{\mathrm{T}}} \boldsymbol{P_C}^{\mathrm{T}} \boldsymbol{W} \tag{30}$$

where $\boldsymbol{S}_{bb}^s = \boldsymbol{K}_{bb}^s - \boldsymbol{K}_{ib}^{s^{\mathrm{T}}} \boldsymbol{K}_{ii}^{s^{-1}} \boldsymbol{K}_{ib}$ and the subscripts i and b designate the restrictions to the zero and nonzero columns, respectively, of $\boldsymbol{P_C}\widehat{\boldsymbol{B}}^s$ and

$$\boldsymbol{W} = \left(\sum_{s=1}^{s=N_s} \boldsymbol{P_C}\widehat{\boldsymbol{B}}^s \widehat{\boldsymbol{B}}^{s^{\mathrm{T}}} \boldsymbol{P_C}^{\mathrm{T}}\right)^{+} \tag{31}$$

3.1.4 Working set expansion and dual planing

Let $\boldsymbol{p}^k$ and η_{CG}^k denote the search direction and step length, respectively, generated at some k^{th} iteration by the PCPG algorithm for the equality constrained minimization on the current working set $\tilde{\tilde{\Gamma}}_C^k$. If the corresponding update $\hat{\boldsymbol{\lambda}}^{k-1} + \eta^{k,0}\boldsymbol{p}^k$ is feasible then it is accepted. Otherwise, an auxiliary feasibility problem that may be posed as follows arises: find a nearby point $\hat{\boldsymbol{\lambda}}^k \in \widehat{G}$ such that $\hat{g}(\hat{\boldsymbol{\lambda}}^k) \leq \hat{g}(\hat{\boldsymbol{\lambda}}^{k-1})$. A direct algorithm for solving this problem is simply to reduce the step length to the maximum feasible step length in the direction $\boldsymbol{p}^k$

$$\eta^k = \min_{\{M \in \Gamma_C : \boldsymbol{p}_M^k < 0\}} \left(\eta^{k,0}, \frac{-\hat{\boldsymbol{\lambda}}_M^{k-1}}{\boldsymbol{p}_M^k}\right) \tag{32}$$

However, this algorithm permits very few (typically only one) constraints to be added to or removed from the active set at each iteration, resulting in a potentially very slow convergence of the overall solution method when the number of inequality constraints is large. An alternative iterative algorithm permitting a large number of constraints to be added to or removed from the active set at each iteration is the projected line search used in the well-known gradient projection scheme

$$\eta^k = \arg\max_{\theta>0} \hat{g}\left(P_{\widehat{G}}(\hat{\boldsymbol{\lambda}}^{k-1} + \theta \boldsymbol{r}^k)\right) \tag{33}$$

where $P_{\widehat{G}}$ is the orthogonal projection operator onto the dual feasible set and $\boldsymbol{r}^k$ is the negative gradient of the dual objective function. However, this algorithm tends to cause undesirable oscillations in the active set and does not exploit the potential benefits of using the conjugate gradient method. For these reasons, the following variation of the standard line search algorithm is proposed

$$\eta^k = \arg\max_{\theta>0} \hat{g}\left(P_D(\hat{\boldsymbol{\lambda}}^{k-1} + \theta \boldsymbol{p}^k)\right) \tag{34}$$

This variant line search algorithm uses the PCPG search direction $\boldsymbol{p}^k$ instead of the negative gradient, and orthogonal projections onto a *subset* of the feasible set,

$$D = \{\boldsymbol{x} \in \mathbb{R}^{l+m} : \widehat{\boldsymbol{G}}^{\mathrm{T}} \boldsymbol{x} = \boldsymbol{e}\} \cap E \tag{35}$$

where

$$E = \left\{ \boldsymbol{x} \in \mathbb{R}^{l+m} : \begin{array}{ll} x_M = 0 & \text{for all } M \in \tilde{\tilde{\Gamma}}_C^{k-1} \\ x_M \geq 0 & \text{for all } M \in \Gamma_C \setminus \tilde{\tilde{\Gamma}}_C^{k-1} \end{array} \right\} \tag{36}$$

Since the working set $\tilde{\tilde{\Gamma}}_C^{k-1}$ is defined as a subset of the active set at $\hat{\boldsymbol{\lambda}}^{k-1}$ (i.e., $\tilde{\tilde{\Gamma}}_C^{k-1} \subseteq \tilde{\tilde{\Gamma}}_C^{k-1}$), projecting onto D instead of $\widehat{G}$ effectively guarantees that all of the constraints in the working set remain active, while any other inequality constraint may change status.

An inexact backtracking projected line search algorithm for approximating the solution of problem (34) and ensuring the monotonicity of the overall contact solution strategy is presented in Table 1. A corresponding iterative algorithm for computing an orthogonal projection onto D, referred to here as the *dual planing* procedure because it has the effect of removing as if with a carpenter's plane the infeasible part of the dual variables, is given in Table 2. In this table, P_E is the orthogonal projection operator onto E, defined component-wise as

$$[P_E(\boldsymbol{x})]_M = \begin{cases} 0 & \text{if } M \in \tilde{\tilde{\Gamma}}_C^{k-1} \\ \langle x_M \rangle_+ & \text{if } M \in \Gamma_C \setminus \tilde{\tilde{\Gamma}}_C^{k-1} \\ x_M & \text{otherwise} \end{cases} \tag{37}$$

ε_D is a small convergence tolerance, and

$$[\langle \boldsymbol{x} \rangle_+]_i = \begin{cases} 0 & \text{if } x_i < 0 \\ x_i & \text{otherwise} \end{cases}$$

Initialize

$\eta^{k,0} = \eta^k_{CG}$

Iterate $j = 0, 1, \ldots$ until convergence

1. Project	$\hat{\boldsymbol{\lambda}}^{k,j}$	$= P_D(\hat{\boldsymbol{\lambda}}^{k-1} + \eta^{k,j}\boldsymbol{p}^k)$
2. Test monotonicity	if	$g(\hat{\boldsymbol{\lambda}}^{k,j}) \leq g(\hat{\boldsymbol{\lambda}}^{k-1})$ then $\eta^k = \eta^{k,j}$, end
3. Backtrack	$\eta^{k,j+1}$	$= \tau\eta^{k,j}$ where $0 < \tau \leq 1$

Table 1: Backtracking projected line search algorithm for computing η^k

Initialize

$\bar{\boldsymbol{\lambda}}^0 = \boldsymbol{x}^0$

Iterate $n = 1, 2, \ldots$ until convergence

1. Enforce positiveness	$\tilde{\boldsymbol{\lambda}}^n$	$= P_E(\bar{\boldsymbol{\lambda}}^{n-1})$
2. Update working set	$\tilde{\tilde{\Gamma}}^k_C$	$= \{M \in \Gamma^k_C : \tilde{\lambda}^n_M = 0\}$
3. Test self-equilibrium	if	$\|\|\widehat{\boldsymbol{G}}^{\mathrm{T}}\tilde{\boldsymbol{\lambda}}^n - \boldsymbol{e}\|\| \leq \varepsilon_D$ then $P_D(\boldsymbol{x}^0) = \tilde{\boldsymbol{\lambda}}^n$, end.
4. Update multipliers	$\bar{\boldsymbol{\lambda}}^n$	$= \bar{\boldsymbol{\lambda}}^{n-1} + \widehat{\boldsymbol{G}}(\widehat{\boldsymbol{G}}^{\mathrm{T}}\boldsymbol{P}_{\boldsymbol{C}}\widehat{\boldsymbol{G}})^+(\boldsymbol{e} - \widehat{\boldsymbol{G}}^{\mathrm{T}}\tilde{\boldsymbol{\lambda}}^n)$

Table 2: Dual planing procedure for computing $P_D(\boldsymbol{x}^0)$ and expanding the working set

To establish that the dual planing procedure described above – which, it should be noted, is subtly different from previous versions published in [15, 5] – is indeed a projection algorithm, it is first observed that the problem of finding the projection onto the closed convex set D can be defined as the QP problem

$$P_D(\boldsymbol{x}^0) = \underset{\boldsymbol{x}\in D}{\arg\min}\{\frac{1}{2}||\boldsymbol{x}-\boldsymbol{x}^0||^2\} \tag{38}$$

whose optimality conditions may be expressed in the form [12]

$$P_D(\boldsymbol{x}^0) = P_E(\boldsymbol{x}^0 - \widehat{\boldsymbol{G}}\boldsymbol{\phi}) \tag{39a}$$

$$\widehat{\boldsymbol{G}}^{\mathrm{T}} P_E(\boldsymbol{x}^0 - \widehat{\boldsymbol{G}}\boldsymbol{\phi}) = \boldsymbol{e} \tag{39b}$$

The optimal point can therefore be computed by first solving the nonlinear equation (39b) for the parameter vector $\boldsymbol{\phi}$, then substituting this solution into (39a) to compute $P_D(\boldsymbol{x}^0)$. Newton's method is used here for computing the solution of (39b), which requires the availability of the Fréchet derivative of the projection operator P_E at each point $\bar{\boldsymbol{\lambda}}^{n-1} = \boldsymbol{x}^0 - \widehat{\boldsymbol{G}}\boldsymbol{\phi}^{n-1}$ corresponding to some current approximation $\boldsymbol{\phi}^{n-1}$. This derivative is simply the projection operator onto the subspace C defined in (29) since $\tilde{\tilde{\Gamma}}_C^k$ is by definition (see Table 2, step 2) the index set of constraints in E that are active at the point $\tilde{\boldsymbol{\lambda}}^n = P_E(\bar{\boldsymbol{\lambda}}^{n-1})$. Given some initial point $\boldsymbol{\phi}^0$, the Newton iterations for solving (39b) can therefore be written in the form

$$\boldsymbol{\phi}^n = \boldsymbol{\phi}^{n-1} - (\widehat{\boldsymbol{G}}^{\mathrm{T}}\boldsymbol{P}_{\boldsymbol{C}}\widehat{\boldsymbol{G}})^+ \left(\boldsymbol{e} - \widehat{\boldsymbol{G}}^{\mathrm{T}} P_E(\boldsymbol{x}^0 - \widehat{\boldsymbol{G}}\boldsymbol{\phi}^{n-1})\right) \tag{40}$$

Defining $\bar{\boldsymbol{\lambda}}^n = \boldsymbol{x}^0 - \widehat{\boldsymbol{G}}\boldsymbol{\phi}^n$, it follows that

$$\bar{\boldsymbol{\lambda}}^n - \bar{\boldsymbol{\lambda}}^{n-1} = \left(\boldsymbol{x}^0 - \widehat{\boldsymbol{G}}\boldsymbol{\phi}^n\right) - \left(\boldsymbol{x}^0 - \widehat{\boldsymbol{G}}\boldsymbol{\phi}^{n-1}\right) \tag{41}$$

$$= -\widehat{\boldsymbol{G}}(\boldsymbol{\phi}^n - \boldsymbol{\phi}^{n-1}) \tag{42}$$

From (40) an expression for $\boldsymbol{\phi}^n - \boldsymbol{\phi}^{n-1}$ can be obtained

$$\boldsymbol{\phi}^n - \boldsymbol{\phi}^{n-1} = -(\widehat{\boldsymbol{G}}^{\mathrm{T}}\boldsymbol{P}_{\boldsymbol{C}}\widehat{\boldsymbol{G}})^+ \left(\boldsymbol{e} - \widehat{\boldsymbol{G}}^{\mathrm{T}} P_E(\boldsymbol{x}^0 - \widehat{\boldsymbol{G}}\boldsymbol{\phi}^{n-1})\right) \tag{43}$$

$$= -(\widehat{\boldsymbol{G}}^{\mathrm{T}}\boldsymbol{P}_{\boldsymbol{C}}\widehat{\boldsymbol{G}})^+ \left(\boldsymbol{e} - \widehat{\boldsymbol{G}}^{\mathrm{T}} P_E(\bar{\boldsymbol{\lambda}}^{n-1})\right) \tag{44}$$

Finally, substituting (44) into (42) and rearranging gives an alternative expression for the Newton iterations in terms of $\bar{\boldsymbol{\lambda}}$ instead of $\boldsymbol{\phi}$, which can be written in the form

$$\bar{\boldsymbol{\lambda}}^n = \bar{\boldsymbol{\lambda}}^{n-1} + \widehat{\boldsymbol{G}}(\widehat{\boldsymbol{G}}^{\mathrm{T}}\boldsymbol{P}_{\boldsymbol{C}}\widehat{\boldsymbol{G}})^+ \left(\boldsymbol{e} - \widehat{\boldsymbol{G}}^{\mathrm{T}} P_E(\bar{\boldsymbol{\lambda}}^{n-1})\right) \tag{45}$$

Hence, the dual planing procedure iterates defined in Table 2 are identical to those of (40) with the initial point $\boldsymbol{\phi}^0 = \boldsymbol{0}$.

3.1.5 Primal planing and working set contraction

Since the algorithm presented in Section 3.1.4 permits only the addition of constraints to the active set, the overall minimization method may not converge to a solution that satisfies all of the optimality conditions, specifically the condition (25e) which states that the Lagrange multipliers associated with the *dual* inequality constraints must themselves be non-positive. Hence, a mechanism for releasing constraints from the active set is needed. A logical choice is to remove from the working set those constraints whose dual Lagrange multipliers would otherwise violate (25e), by projecting the negative gradient $\boldsymbol{r}^k$ at some k^{th} iteration onto the so-called *tangent cone* K – that is, the closure of the set of all feasible directions – and then redefining the working set as the subset of constraints defining the tangent cone which are active at $P_K(\boldsymbol{r}^k)$. The tangent cone at the feasible point $\hat{\boldsymbol{\lambda}}^k$ is the closed convex set

$$K = \{\boldsymbol{x} \in \mathbb{R}^{l+m} : \boldsymbol{G}^{\mathrm{T}}\boldsymbol{x} = \boldsymbol{0}\} \cap J \tag{46}$$

where

$$J = \{\boldsymbol{x} \in \mathbb{R}^{l+m} : x_M \geq 0 \text{ for all } M \in \tilde{\Gamma}^k\} \tag{47}$$

It is well known that the optimality conditions for problem (24) are satisfied when the norm of the negative gradient projected onto the tangent cone is zero at a feasible point. Therefore, this criterion is also used to test the convergence of the overall minimization method as follows

$$\boxed{||P_K(\boldsymbol{r}^k)|| \leq \varepsilon ||P_K(\boldsymbol{r}^0)||} \tag{48}$$

where ε is a small convergence tolerance and $||\cdot||$ denotes the Euclidean norm.

Furthermore, to minimize the occurrence of undesirable oscillations, it is chosen to perform the tangent cone projection only when the dual feasibility error corresponding to the current working set exceeds a certain threshold. The computation of the projected gradient using either P_A or P_K can be implemented as a projection algorithm and is referred to here as the primal planing procedure. It is presented in Table 3 where P_J denotes the orthogonal projection operator onto J and is defined component-wise as

$$[P_J(\boldsymbol{x})]_M = \begin{cases} \langle x_M \rangle_+ & \text{if } M \in \tilde{\Gamma}_C^k \\ x_M & \text{otherwise} \end{cases} \tag{49}$$

ε_P is a small convergence tolerance, and $\Delta \geq 0$ is a given constant. The latter is typically set to 1, in which case the magnitude of the violation of the fifth optimality condition (25e) is not allowed to exceed that of the first optimality condition (25a).

3.1.6 Summary

In this paper, the FETI-C method is re-defined as the overall solution method described in subsections 3.1.1–3.1.5. It is essentially a re-designed version of the FETI-C method first proposed in [15] that is equipped with controls to guarantee monotonicity and therefore convergence. It is summarized in algorithmic notation in Table 4.

Initialize

$$\begin{aligned}\bar{\boldsymbol{w}}^0 &= \boldsymbol{x}^0 - \widehat{\boldsymbol{G}}(\widehat{\boldsymbol{G}}^{\mathrm{T}}\boldsymbol{P}_{\boldsymbol{C}}\widehat{\boldsymbol{G}})^{+}\widehat{\boldsymbol{G}}^{\mathrm{T}}\boldsymbol{P}_{\boldsymbol{C}}(\boldsymbol{x}^0)\\ \text{if}\quad &||\langle\bar{\boldsymbol{w}}^0 - \boldsymbol{P}_{\boldsymbol{C}}\bar{\boldsymbol{w}}^0\rangle_{+}|| \le \Delta||\boldsymbol{P}_{\boldsymbol{C}}\bar{\boldsymbol{w}}^0|| \text{ then } P_P(\boldsymbol{x}^0) = P_A(\boldsymbol{x}^0) = \boldsymbol{P}_{\boldsymbol{C}}\bar{\boldsymbol{w}}^0, \text{ end.}\end{aligned}$$

Iterate $n = 1, 2, \ldots$ until convergence

$$\begin{aligned}\text{1. Enforce positiveness}\quad &\tilde{\boldsymbol{w}}^n = P_J(\bar{\boldsymbol{w}}^{n-1})\\ \text{2. Update working set}\quad &\tilde{\tilde{\Gamma}}_C^k = \{M \in \tilde{\Gamma}_C^k : \tilde{w}_M^n = 0\}\\ \text{3. Test feasibility}\quad &\text{if}\quad ||\widehat{\boldsymbol{G}}^{\mathrm{T}}\tilde{\boldsymbol{w}}^n|| \le \varepsilon_P \text{ then } P_P(\boldsymbol{x}^0) = P_K(\boldsymbol{x}^0) = \tilde{\boldsymbol{w}}^n, \text{ end.}\\ \text{4. Update } \bar{\boldsymbol{w}}\quad &\bar{\boldsymbol{w}}^n = \bar{\boldsymbol{w}}^{n-1} - \widehat{\boldsymbol{G}}(\widehat{\boldsymbol{G}}^{\mathrm{T}}\boldsymbol{P}_{\boldsymbol{C}}\widehat{\boldsymbol{G}})^{+}\widehat{\boldsymbol{G}}^{\mathrm{T}}\tilde{\boldsymbol{w}}^n\end{aligned}$$

Table 3: Primal planing procedure for computing $P_P(\boldsymbol{x}^0)$ and contracting the working set

3.2 The re-designed FETI-DPC method

Another equivalent problem to (20) can be constructed in terms of the subdomain knowns, subdomain remainder unknowns, and global corner unknowns as follows

$$\begin{aligned}(u_r, \boldsymbol{u}_{\boldsymbol{g_c}}) &= \underset{(v_r, \boldsymbol{v}_{\boldsymbol{g}_c})\in\widehat{B}^*}{\arg\min}\ f^*(v_r, \boldsymbol{v}_{\boldsymbol{g}_c}),\\ \widehat{B}^* &= \left\{(v_r, \boldsymbol{v}_{\boldsymbol{g}_c}) \in \prod_{s=1}^{s=N_s}\mathbb{R}^{n_{r^s}} \times \mathbb{R}^{n_{g_c}} : \sum_{s=1}^{s=N_s}\boldsymbol{B}_{rr}^s\boldsymbol{v}_r^s = \boldsymbol{0},\quad \sum_{s=1}^{s=N_s}\boldsymbol{C}^s\begin{bmatrix}\boldsymbol{v}_r^s\\ \boldsymbol{L}_{cc}^s\boldsymbol{v}_{\boldsymbol{g}_c}\end{bmatrix} \le \boldsymbol{a}\right\}\end{aligned} \tag{50}$$

The following subsections describe a numerical algorithm based on the FETI-DP method for solving the domain decomposed minimization problem formulated above.

3.2.1 Dualization

The Lagrangian and Lagrange dual functions associated with problem (50) are given by

$$\hat{L}^*(v_r, \boldsymbol{v}_{\boldsymbol{g}_c}, \boldsymbol{\mu}_r, \boldsymbol{\xi}) = \hat{f}^*(v_r, \boldsymbol{v}_{\boldsymbol{g}_c}) + \boldsymbol{\mu}_r^{\mathrm{T}}\sum_{s=1}^{s=N_s}\boldsymbol{B}_{rr}^s\boldsymbol{v}_r^s + \boldsymbol{\xi}^{\mathrm{T}}\left(\sum_{s=1}^{s=N_s}\boldsymbol{C}^s\begin{bmatrix}\boldsymbol{v}_r^s\\ \boldsymbol{L}_{cc}^s\boldsymbol{v}_{\boldsymbol{g}_c}\end{bmatrix} - \boldsymbol{a}\right)$$

and

$$\hat{h}^*(\boldsymbol{\mu}_r, \boldsymbol{\xi}) = \inf_{(v_r, \boldsymbol{v}_{\boldsymbol{g}_c})}\hat{L}^*(v_r, \boldsymbol{v}_{\boldsymbol{g}_c}, \boldsymbol{\mu}_r, \boldsymbol{\xi})$$

respectively. The infimum of the Lagrangian function is

(a) by definition $-\infty$ when $\hat{L}^*$ is unbounded from below in $(v_r, \boldsymbol{v}_{\boldsymbol{g}_c})$,

(b) otherwise attained when the gradient of the Lagrangian function $\hat{L}^*$ with respect to $(v_r, \boldsymbol{v}_{\boldsymbol{g}_c})$ is zero.

Initialize

$$
\begin{aligned}
\tilde{\tilde{\Gamma}}_C^0 &= \emptyset \\
\hat{\boldsymbol{\lambda}}^0 &= P_D(\mathbf{0}) \\
\boldsymbol{r}^0 &= \hat{\boldsymbol{d}} - \widehat{\boldsymbol{F}}\hat{\boldsymbol{\lambda}}^0 \\
\boldsymbol{w}^0 &= P_P(\boldsymbol{r}^0)
\end{aligned}
$$

Iterate $k = 1, 2, \ldots$ until convergence

$$
\begin{aligned}
\text{Precondition} \quad \boldsymbol{z}^{k-1} &= \widehat{\boldsymbol{M}}^{-1}\boldsymbol{w}^{k-1} \\
\text{Project} \quad \boldsymbol{y}^{k-1} &= \boldsymbol{P_A}\boldsymbol{z}^{k-1} \\
\text{Conjugate} \quad \zeta^k &= \frac{\boldsymbol{y}^{k-1^{\mathrm{T}}}\boldsymbol{w}^{k-1}}{\boldsymbol{y}^{k-2^{\mathrm{T}}}\boldsymbol{w}^{k-2}} \quad (\zeta^k = 0 \text{ if } k = 1 \text{ or status change}) \\
\boldsymbol{p}^k &= \boldsymbol{y}^{k-1} + \zeta^k\boldsymbol{p}^{k-1} \quad (\boldsymbol{p}^1 = \boldsymbol{y}^{k-1}) \\
\text{Minimize} \quad \eta_{CG}^k &= \frac{\boldsymbol{p}^{k^{\mathrm{T}}}\boldsymbol{w}^{k-1}}{\boldsymbol{p}^{k^{\mathrm{T}}}\widehat{\boldsymbol{F}}\boldsymbol{p}^k} \\
\eta^k &= \arg\max_{\theta>0} \hat{g}(P_D(\hat{\boldsymbol{\lambda}}^{k-1} + \theta\boldsymbol{p}^k)) \\
\text{Update} \quad \hat{\boldsymbol{\lambda}}^k &= P_D(\hat{\boldsymbol{\lambda}}^{k-1} + \eta^k\boldsymbol{p}^k) \\
\text{Evaluate residual} \quad \boldsymbol{r}^k &= \begin{cases} \boldsymbol{r}^{k-1} - \eta^k\widehat{\boldsymbol{F}}\boldsymbol{p}^k & \text{if dual status change} \\ \hat{\boldsymbol{d}} - \widehat{\boldsymbol{F}}\hat{\boldsymbol{\lambda}}^k & \text{otherwise} \end{cases} \\
\text{Project} \quad \boldsymbol{w}^k &= P_P(\boldsymbol{r}^k)
\end{aligned}
$$

Table 4: Nonlinear FETI-C solution algorithm

Hence, the Lagrange dual function can be re-written as

$$\hat{h}^*(\boldsymbol{\mu},\boldsymbol{\xi}) = \begin{cases} -\frac{1}{2}\hat{\boldsymbol{\mu}}_r^{\mathrm{T}}\widehat{\boldsymbol{F}}_{rr}^*\hat{\boldsymbol{\mu}}_r + \hat{\boldsymbol{d}}^{*^{\mathrm{T}}}\hat{\boldsymbol{\mu}}_r - c_r^* & \text{if } \widehat{\boldsymbol{G}}_r^{*^{\mathrm{T}}}\hat{\boldsymbol{\mu}}_r = \boldsymbol{e}^* \\ -\infty & \text{otherwise} \end{cases} \tag{51}$$

where $\hat{\boldsymbol{\mu}}_r = [\boldsymbol{\mu}_r^{\mathrm{T}}\ \boldsymbol{\xi}^{\mathrm{T}}]^{\mathrm{T}}$, $\widehat{\boldsymbol{B}}_{rr}^s = [\boldsymbol{B}_{r*}^{s^{\mathrm{T}}}\ \boldsymbol{C}^{s^{\mathrm{T}}}]^{\mathrm{T}}$,

$$\begin{aligned}
\widehat{\boldsymbol{F}}_{rr}^* &= \widehat{\boldsymbol{F}}_{rr} - \widehat{\boldsymbol{F}}_{rc}\boldsymbol{K}_{cc}^{*^+}\widehat{\boldsymbol{F}}_{rc}^{\mathrm{T}} \\
\hat{\boldsymbol{d}}_r^* &= \hat{\boldsymbol{d}}_r + \widehat{\boldsymbol{F}}_{rc}\boldsymbol{K}_{cc}^{*^+}\boldsymbol{f}_c^* \\
\widehat{\boldsymbol{F}}_{rr} &= \sum_{s=1}^{s=N_s} \widehat{\boldsymbol{B}}_{rr}^s \boldsymbol{K}_{rr}^{s^{-1}} \widehat{\boldsymbol{B}}_{rr}^{s^{\mathrm{T}}} \\
\widehat{\boldsymbol{F}}_{rc} &= \sum_{s=1}^{s=N_s} \left(\widehat{\boldsymbol{B}}_{rr}^s \boldsymbol{K}_{rr}^{s^{-1}} \boldsymbol{K}_{rc}^s + \widehat{\boldsymbol{B}}_{rc}\right) \boldsymbol{L}_{cc}^s \\
\hat{\boldsymbol{d}}_r &= \sum_{s=1}^{s=N_s} \widehat{\boldsymbol{B}}_{rr}^s \boldsymbol{K}_{rr}^{s^{-1}} \boldsymbol{f}_r^s - \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{a}_r \end{bmatrix} \\
\boldsymbol{f}_c^* &= \sum_{s=1}^{N_s} \boldsymbol{L}_{cc}^{s^T} \left(\boldsymbol{f}_c^s - \boldsymbol{K}_{rc}^{s^T} \boldsymbol{K}_{rr}^{s^{-1}} \boldsymbol{f}_r^s\right) \\
\widehat{\boldsymbol{G}}_r^* &= \begin{bmatrix} \boldsymbol{C}^1 \boldsymbol{R}^{*^1} & \cdots & \boldsymbol{C}^{N_s} \boldsymbol{R}^{*^{N_b}} \end{bmatrix} \\
\boldsymbol{e}^* &= \begin{bmatrix} \boldsymbol{f}^{1^{\mathrm{T}}} \boldsymbol{R}^{*^1} & \cdots & \boldsymbol{f}^{N_b{}^{\mathrm{T}}} \boldsymbol{R}^{*^{N_b}} \end{bmatrix}^{\mathrm{T}}
\end{aligned}$$

the columns of $\boldsymbol{R}^{*^b} \in \mathbb{R}^{n^b \times q^{*^b}}$ span the left null space of the generalized body stiffness matrix $\boldsymbol{K}^b$ and represent the rigid body modes of body b, and N_b denotes the number of bodies.

The above dual problem is of the form

$$(\boldsymbol{\lambda}_r, \boldsymbol{\nu}) = \underset{(\boldsymbol{\mu}_r,\boldsymbol{\xi})\in H^*}{\arg\max}\ g^*(\boldsymbol{\mu}_r,\boldsymbol{\xi}),\ H^* = \left\{(\boldsymbol{\mu}_r,\boldsymbol{\xi}) \in \mathbb{R}^{l_r} \times \mathbb{R}^m : \xi_i \geq 0, i = 1,\ldots,m\right\} \tag{52}$$

or equivalently

$$\hat{\boldsymbol{\lambda}}_r = \underset{\hat{\boldsymbol{\mu}}_r\in\widehat{G}^*}{\arg\min}\ \hat{g}^*(\hat{\boldsymbol{\mu}}_r),\ \widehat{G}_r^* = \left\{\hat{\boldsymbol{\lambda}}_r \in \mathbb{R}^{l_r+m} : \widehat{\boldsymbol{G}}_r^{*^{\mathrm{T}}}\hat{\boldsymbol{\lambda}}_{\boldsymbol{r}} = \boldsymbol{e}^*, \hat{\lambda}_{r_{l+i}} \geq 0, i = 1,\ldots,m\right\} \tag{53}$$

where $\hat{g}^*(\hat{\boldsymbol{\mu}}_r) = -\frac{1}{2}\hat{\boldsymbol{\mu}}_r^{\mathrm{T}}\widehat{\boldsymbol{F}}_{rr}^*\hat{\boldsymbol{\mu}}_r + \hat{\boldsymbol{d}}^{*^{\mathrm{T}}}\hat{\boldsymbol{\mu}}_r$ and $\hat{\boldsymbol{\lambda}}_r = [\boldsymbol{\lambda}_r^{\mathrm{T}}\ \boldsymbol{\nu}^{\mathrm{T}}]^{\mathrm{T}}$.

3.2.2 Solving the dual problem

In this case, the necessary and sufficient Karush-Kuhn-Tucker conditions lead to

$$\widehat{\boldsymbol{F}}^*_{rr}\hat{\boldsymbol{\lambda}}_r + \widehat{\boldsymbol{G}}^*_r\boldsymbol{\alpha}^* + \begin{bmatrix}\mathbf{0}\\ \boldsymbol{\gamma}\end{bmatrix} = \hat{\boldsymbol{d}}^*_r \tag{54a}$$

$$\widehat{\boldsymbol{G}}^{*^{\mathrm{T}}}_r \hat{\boldsymbol{\lambda}}_r = \boldsymbol{e}^* \tag{54b}$$

$$\hat{\lambda}_{r_{l+i}} \geq 0,\ i = 1, \ldots, m \tag{54c}$$

$$\gamma_i \hat{\lambda}_{r_{l+i}} = 0,\ i = 1, \ldots, m \tag{54d}$$

$$\gamma_i \leq 0,\ i = 1, \ldots, m \tag{54e}$$

The optimal point can be found using an adaptation of the re-designed FETI-C algorithm summarized in Table 4. This adaptation consists of merely replacing the FETI variables and operators by their FETI-DP counterparts. For example, replacing $\widehat{\boldsymbol{F}}$ by $\widehat{\boldsymbol{F}}^*_{rr}$, $\widehat{G}$ by $\widehat{G}^*_r$, and $\hat{\boldsymbol{\lambda}}$ by $\hat{\boldsymbol{\lambda}}_r$. The number of dual equality constraints q^* is in this case zero for a large class of problems, and in any case much smaller than that obtained with the FETI-C method.

3.2.3 Summary

In this paper, the FETI-DPC method is re-defined as the overall solution method described in sections 3.2.1–3.2.2. It can be viewed as an adaptation to the FETI-DP formalism of the re-designed FETI-C method. It is also a re-designed version of the FETI-DPC method first proposed in [5] that is equipped with controls to guarantee monotonicity and therefore convergence using the same criterion (48) as in the re-designed FETI-C method.

4 Applications

In this section, the performance of the re-designed FETI-C and FETI-DPC methods is assessed and their numerical and parallel scalability properties are evaluated. Additional performance results for an instance of the FETI-DPC method equipped with the augmentation coarse problem proposed in [20] and referred to here as the augmented FETI-DPC method are also presented. To this effect, a six-body contact problem of the academic type but with global rigid body modes (which makes it more challenging) is first considered, then followed by a complex 1.4 million degree of freedom contact problem of the industrial type. In both cases, contact is assumed to be frictionless. All computations are performed in double precision arithmetic on a Dell Linux cluster with 216 nodes, each comprising two quad-core Intel Xeon processors and equipped with 16 gigabytes of memory.

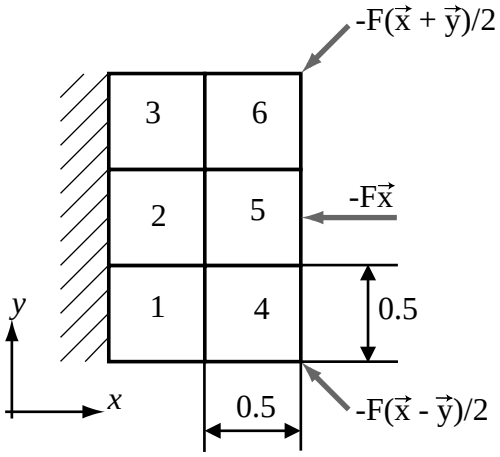

Figure 1: A six-block equilibrium problem with seven contact surfaces

4.1 A two-dimensional six-body frictionless contact problem

First, the problem graphically depicted in Figure 1 is considered with $F = 10^4$ N. In this problem, six blocks sharing seven contact interfaces are pushed against each other by a set of external forces, and the displacement field in the y direction is restrained in block 5. Hence, this contact problem has a priori five global rigid body modes. The reader can observe in Figure 1 that the applied loads are self-equilibrated in the y direction. All blocks are assumed to be made of the same linear elastic material characterized by a Young's modulus $E = 2.05\ 10^3$ MPa, and a Poisson's ratio $\nu = 0.3$. These blocks are discretized by four-noded plane stress elements. Small meshes with 2,178 dofs as well as larger ones with up to 4,930,572 dofs are generated and partitioned into six to 384 subdomains.

Let H and h denote the sizes of a subdomain and an element, respectively. More specifically, four different domain decompositions with six ($H = 1/2$), 24 ($H = 1/4$), 96 ($H = 1/8$), and 384 ($H = 1/16$) subdomains are considered. In each subdomain, a suite of $(1/h) \times (1/h)$ uniform meshes with $10 \leq H/h \leq 80$ is generated. For each domain decomposition and each mesh size, the performance results of the FETI-C, FETI-DPC and augmented FETI-DPC methods equipped with the topologically scaled Dirichlet preconditioner are reported in Tables 5, 6 and 7, respectively. For this problem, convergence of the nonlinear solution algorithm is defined by criterion (48) with $\varepsilon = 10^{-10}$, and convergence of the dual and primal planing procedures are monitored using $\varepsilon_D = \varepsilon_P = 10^{-20}$. These performance results consist of the number of iterations for convergence, N_{itr}, the number of iterations that undergo working set expansions which is referred to here as the number of dual status changes, N_{dsc}, the total number of dual planing subiterations N_{dpl}, the number of iterations that undergo working set contractions which is also referred to here as the number of primal status changes, N_{psc}, the total number of primal planing subiterations, N_{ppl}, the total number

H	H/h	N_s	n_g	m	N_{itr}	N_{dsc}	N_{dpl}	N_{psc}	N_{ppl}	N_{ls}	Solver CPU time (s)
1/2	10	6	1,452	75	12	3	2	1	0	0	0.011
1/2	20	6	5,292	145	8	1	0	1	0	0	0.015
1/2	40	6	20,172	285	9	1	0	1	0	0	0.69
1/2	80	6	78,732	565	9	1	0	1	0	0	0.50
1/4	10	24	5,292	145	22	1	2	1	1	0	0.040
1/4	20	24	20,172	285	32	2	3	2	2	0	0.13
1/4	40	24	78,732	565	36	3	4	2	2	0	0.65
1/4	80	24	311,052	1,125	39	3	4	2	2	0	3.8
1/8	10	96	20,172	285	34	1	3	2	2	0	0.22
1/8	20	96	78,732	565	40	2	4	2	3	0	0.74
1/8	40	96	311,052	1,125	47	3	5	2	2	0	3.2
1/8	80	96	1,236,492	2,245	53	4	6	2	2	0	18.3
1/16	10	384	78,732	565	42	3	5	2	3	0	1.3
1/16	20	384	311,052	1,125	54	4	10	3	4	0	4.1
1/16	40	384	1,236,492	2,245	56	4	8	2	2	0	15.2
1/16	80	384	4,930,572	4,485	65	9	14	2	2	0	88.1

Table 5: Two-dimensional six-block contact problem: performance results of the FETI-C method on six cores

of projected line search subiterations, N_{ls}, and the total elapsed "solver" CPU time in seconds (excluding pre- and post-processing of the results) on six nodes of the cluster using one core per node. Furthermore, the elapsed CPU time of all three solvers for the same problem but on six, 12, 24 and 48 nodes using one core per node is reported in Table 8. The reader is reminded that N_s denotes the number of subdomains, n_g the total number of dofs, and m the number of inequality constraints.

The performance results summarized in Table 5, Table 6, Table 7 and Table 8 show that for this problem:

- The FETI-C and FETI-DPC methods are numerically scalable with respect to the problem size for a given number of subdomains, and reasonably numerically scalable with respect to the number of subdomains for a given problem size.
- Both methods exhibit very good to excellent parallel scalability for a fixed problem size. For example, Table 8 shows that for the considered problem, increasing the number of CPUs by a factor eight, from six to 48, reduces the CPU time of the FETI-C and FETI-DPC methods by a factor of 5.5 and 7.4, respectively.
- Both methods exhibit excellent scalability of the time-to-solution. For example, when the problem size is increased by a factor equal to 15.9, from $n_g = 311,052$ to $n_g = 4,930,572$, and N_s is increased from 24 to 384, the FETI-C CPU time is increased by a factor equal to 23.2 only, from 3.8 seconds to 88.1 seconds.
- The FETI-C and augmented FETI-DPC methods perform comparably.

H	H/h	N_s	n_g	m	N_{itr}	N_{dsc}	N_{dpl}	N_{psc}	N_{ppl}	N_{ls}	Solver CPU time (s)
1/2	10	6	1,452	75	12	3	2	1	0	0	0.013
1/2	20	6	5,292	145	12	2	1	2	1	0	0.022
1/2	40	6	20,172	285	12	2	1	2	1	0	0.089
1/2	80	6	78,732	565	13	4	3	2	1	0	0.69
1/4	10	24	5,292	145	31	2	0	2	0	0	0.041
1/4	20	24	20,172	285	31	2	1	1	1	0	0.11
1/4	40	24	78,732	565	43	2	1	2	1	0	0.72
1/4	80	24	311,052	1,125	52	7	2	2	2	0	5.1
1/8	10	96	20,172	285	45	3	2	2	3	0	0.15
1/8	20	96	78,732	565	51	3	2	2	3	0	0.65
1/8	40	96	311,052	1,125	75	7	4	3	3	0	4.4
1/8	80	96	1,236,492	2,245	97	13	11	3	3	0	32.6
1/16	10	384	78,732	565	70	5	3	3	2	0	0.89
1/16	20	384	311,052	1,125	104	11	9	4	4	0	4.8
1/16	40	384	1,236,492	2,245	124	23	16	4	3	0	27.9
1/16	80	384	4,930,572	4,485	155	38	28	4	4	0	200.2

Table 6: Two-dimensional six-block contact problem: performance results of the FETI-DPC method on six cores

H	H/h	N_s	n_g	m	N_{itr}	N_{dsc}	N_{dpl}	N_{psc}	N_{ppl}	N_{ls}	Solver CPU time (s)
1/2	10	6	1,452	75	12	3	2	1	0	0	0.013
1/2	20	6	5,292	145	12	2	1	2	1	0	0.022
1/2	40	6	20,172	285	12	2	1	2	1	0	0.089
1/2	80	6	78,732	565	13	4	3	2	1	0	0.69
1/4	10	24	5,292	145	17	1	0	1	0	0	0.047
1/4	20	24	20,172	285	19	1	0	1	0	0	0.10
1/4	40	24	78,732	565	27	2	0	2	0	0	0.56
1/4	80	24	311,052	1,125	32	2	1	2	1	0	3.8
1/8	10	96	20,172	285	28	2	0	2	0	0	0.17
1/8	20	96	78,732	565	32	2	0	2	0	0	0.52
1/8	40	96	311,052	1,125	38	4	0	2	0	0	2.8
1/8	80	96	1,236,492	2,245	42	5	2	2	1	0	18.5
1/16	10	384	78,732	565	40	3	2	3	2	0	0.68
1/16	20	384	311,052	1,125	51	4	2	3	2	0	2.8
1/16	40	384	1,236,492	2,245	50	9	6	3	3	0	13.2
1/16	80	384	4,930,572	4,485	67	12	7	3	2	0	102.5

Table 7: Two-dimensional six-block contact problem: performance results of the augmented FETI-DPC method on six cores

Number of cores	FETI-C Solver CPU time (s)	FETI-DPC Solver CPU time (s)	Augmented FETI-DPC Solver CPU time (s)
6	88.1	200.2	102.5
12	46.6	101.6	51.7
24	26.1	51.9	26.4
48	15.9	27.1	13.8

Table 8: Two-dimensional six-block contact problem: performance results of the FETI-C, FETI-DPC and augmented FETI-DPC methods on six to 48 cores for $H/h = 80$ and $N_s = 384$

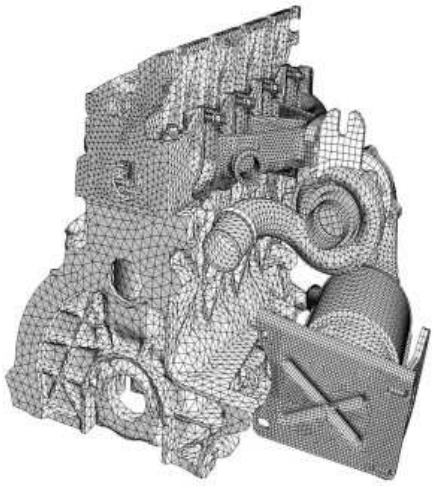

Figure 2: FE model of an assembled car engine

- The FETI-C and FETI-DPC methods perform a relatively small number of dual planing subiterations, primal planing subiterations, and projected line search subiterations.

4.2 A complex 1.4 million degree of freedom car engine assembly problem

The problem discussed in the previous section is of the academic type. It has however the merit of being simple to reproduce by the interested reader. Here, a more realistic problem is considered in order to assess the validity of the conclusions for more complex applications. This realistic problem is associated with the assembly of the car engine shown in Figure 2. This engine contains $N_b = 23$ bodies, some of which are floating and generate a total of $q^* = 130$ rigid body modes. The overall structure is very heterogeneous: some bodies are made of steel, others from elastomer-like materials. The overall structure is discretized by a variety of finite elements including four-noded tetrahedral solid elements, ten-noded tetrahedral solid elements, eight-noded brick elements, three-noded triangular shell elements, two-noded beam elements, and various springs. The overall finite element model contains 275,770 elements, 465,630 nodes and 1,416,829 dofs. It also includes $m = 6,358$ constraint equations, 231 of which are associated with the mortar method applied at the nonconforming discrete interfaces between the various bodies. The augmented FETI-DPC method is used to predict the static structural response of the assembled engine to a specified thermal load. Tables 9 and 10 report, for different domain decompositions of the overall problem, the obtained performance results. For this problem, convergence of the nonlinear solution algorithm is defined by criterion (48) equipped with $\varepsilon = 10^{-8}$, and convergence of the dual and primal planing procedures are monitored using $\varepsilon_D = \varepsilon_P = 10^{-16}$.

The performance results reported in Table 9 reveal that the augmented FETI-DPC

N_s	N_{itr}	N_{dsc}	N_{dpl}	N_{psc}	N_{ppl}	N_{ls}	Solver CPU time (s)
192	936	221	215	9	24	0	164.4
384	931	213	161	11	37	0	119.0
576	978	210	154	9	30	0	126.6
768	857	190	204	8	34	0	105.1
960	874	198	161	8	33	0	95.3
1,152	941	213	192	7	24	0	122.2

Table 9: Car engine assembly problem: performance results of the augmented FETI-DPC method on 48 cores

Number of cores	Solver CPU time (s)
6	542.6
12	289.8
24	150.8
48	95.3

Table 10: Assembled car engine problem: performance results of the augmented FETI-DPC method on six to 48 cores for $N_s = 768$

method is numerically scalable with respect to the number of subdomains. Table 10 confirms the parallel scalability of this method. More specifically, it shows that when the number of CPUs is increased by a factor 8, from six to 48, the solver time is reduced by a factor of 6.5.

5 Closure

FETI-C and FETI-DPC are two dual domain decomposition based iterative methods with Lagrange multipliers for solving quadratic programming (QP) optimization problems with linear inequality constraints. Hence, their scope of applications include contact problems. These two methods are are based on the FETI and FETI-DP methods, respectively. In each of them, the dual Lagrange multiplier problem is solved iteratively by an appropriate preconditioned Krylov method equipped with an active set strategy and controls to reduce oscillations and guarantee convergence monotonicity. An advantage of this approach is that, as long as the deformations of the contact surfaces are small, Newton-like iterations are not required for solving the otherwise nonlinear contact problem. Numerical experiments performed for two- and three-dimensional large-scale frictionless contact problems suggest that both the FETI-C and FETI-DPC methods are numerically scalable with respect to both the number of subdomains and the size of the problem, and that their current parallel implementations exhibit good parallel scalability.

6 Acknowledgments

This research was supported by the Sandia National Laboratories under Contract No. 361632-9.

References

[1] P. Alart and A. Curnier. A mixed formulation for frictional contact problems prone to Newton like solution methods. *Computer Methods in Applied Mechanics and Engineering*, 92:253–375, 1991.

[2] F. Armero and E. Petöcz. Formulation and analysis of conserving algorithms for dynamic contact / impact problems. *Computer Methods in Applied Mechanics and Engineering*, 158:269–300, 1998.

[3] J. S. Arora, A. I. Chahande, and J. K. Paeng. Multiplier methods for engineering optimization. *International Journal for Numerical Methods in Engineering*, 32:1485–1525, 1991.

[4] K. J. Arrow, L. Hurwicz, and H. Uzawa. *Studies in nonlinear programming.* University Press, Stanford, CA, 1958.

[5] P. Avery, G. Rebel, M. Lesoinne, and C. Farhat. A numerically scalable dual-primal substructuring method for the solution of contact problems - part I: the frictionless case. *Computer Methods in Applied Mechanics and Engineering*, 190:2403–2426, 2004.

[6] M. Barboteu, P. Alart, and F. Lebon. Une méthode de décomposition de domaine pour une structure multicontact. *4e Colloque National en Calcul des Structures*, 1:159–164, 1999.

[7] M. Barboteu, P. Alart, and M. Vidrascu. A domain decomposition strategy for nonclassical frictional multi-contact problems. *Computer Methods in Applied Mechanics and Engineering*, 190:4785–4803, 2001.

[8] P. Chabrand, F. Dubois, and M. Raous. Various numerical methods for solving unilateral contact problems with friction. In Rodin and Shillor [41], pages 97–108.

[9] L. Champaney, J.-Y. Cognard, D. Dureisseix, and P. Ladevèze. Large scale applications on parallel computers of a mixed domain decomposition method. *Computational Mechanics*, 19:253–263, 1997.

[10] L. Champaney, J.-Y. Cognard, and P. Ladevèze. Modular analysis of assemblages of three-dimensional structures with unilateral contact conditions. *Computers & Structures*, 73:249–266, 1999.

[11] A. Curnier, editor. *Proceedings of the Contact Mechanics International Symposium.* Presses Polytechniques et Universitaires Romandes, 1992.

[12] S. Dharanipragada and K. S. Arun A quadratically convergent algorithm for convex-set constrained signal recovery. *IEEE Transactions on Signal Processing*, 44(2):248–266, 1996.

[13] G. Dilintas, P. Laurent-Gengoux, and D. Trystram. A conjugate projected gradient method with preconditioning for unilateral contact problems. *Computers & Structures*, 29(4):675–680, 1988.

[14] Z. Dostal, A. Friedlander, and S. Santos. Solution of coercive and semicoercive contact problems by FETI domain decomposition. *Contemporary Mathematics*, 218:82–93, 1998.

[15] D. Dureisseix and C. Farhat. A numerically scalable domain decomposition method for the solution of frictionless contact problems. *International Journal for Numerical Methods in Engineering*, 50:2643–2666, 2001.

[16] R. L. Dykstra. An algorithm for restricted least-squares regression. *Journal of the American Statistical Association*, 78(384):837-842, 1983.

[17] C. Farhat. A Lagrange multiplier based divide and conquer finite element algorithm. *Journal of Computing Systems in Engineering*, 2:149–156, 1991.

[18] C. Farhat, P. S. Chen, J. Mandel, and F.-X. Roux. The two-level FETI method - Part II: extension to shell problems, parallel implementation and performance results. *Computer Methods in Applied Mechanics and Engineering*, 155:153–180, 1998.

[19] C. Farhat, P. S. Chen, F. Risler, and F.-X. Roux. A unified framework for accelerating the convergence of iterative substructuring methods with lagrange multipliers. *International Journal for Numerical Methods in Engineering*, 42:257–288, 1998.

[20] C. Farhat, M. Lesoinne, and K. Pierson. A scalable dual-primal domain decomposition method. *Numerical Linear Algebra with Applications*, 7:687–714, 2000.

[21] C. Farhat and J. Mandel. The two-level FETI method for static and dynamic plate problems - Part I: an optimal iterative solver for biharmonic systems. *Computer Methods in Applied Mechanics and Engineering*, 155:129–152, 1998.

[22] C. Farhat, J. Mandel, and F.-X. Roux. Optimal convergence properties of the FETI domain decomposition method. *Computer Methods in Applied Mechanics and Engineering*, 115:367–388, 1994.

[23] C. Farhat and F.-X. Roux. A method of finite element tearing and interconnecting and its parallel solution algorithm. *International Journal for Numerical Methods in Engineering*, 32:1205–1227, 1991.

[24] C. Farhat, M. Lesoinne, P. LeTallec, K. Pierson and D. Rixen. Feti-DP: A dual-primal unified feti method - Part I: a faster alternative to the two-level FETI method. *International Journal for Numerical Methods in Engineering*, 50:1523–1544, 2000.

[25] C. Farhat and F.-X. Roux. Implicit parallel processing in structural mechanics. In J. T. Oden, editor, *Computational Mechanics Advances*, volume 2. North-Holland, June 1994.

[26] N. D. Hung and G. De Saxce. Frictionless contact of elastic bodies by finite element method and mathematical programming technique. *Computers & Structures*, 11:55–67, 1980.

[27] W. Karush. Minima of functions of several variables with inequalities as side constraints. M.Sc. Dissertation. Dept. of Mathematics, Univ. of Chicago, Chicago, Illinois, 1939

[28] N. Kikuchi. Penalty/finite element approximations of a class of unilateral contact problems. In *Penalty method and finite element method*. ASME, New York, 1982.

[29] N. Kikuchi and J. T. Oden, editors. *Contact problems in elasticity: a study of variational inequalities and finite element methods*. SIAM, Philadelphia, 1988.

[30] A. Klarbring. A mathematical programming approach to three-dimensional contact problems with friction. *Computer Methods in Applied Mechanics and Engineering*, 58:175–200, 1986.

[31] A. Klarbring. Mathematical programming and augmented lagrangian methods for frictional contact problems. In Curnier [11], pages 409–422.

[32] H. W. Kuhn and A. W. Tucker. Nonlinear programming. Proceedings of 2nd Berkeley Symposium: 481–492, Berkeley: University of California Press, 1951.

[33] M. Lesoinne. A FETI-DP corner selection algorithm for three-dimensional problems. *Proceedings of the 2002 Domain Decomposition Conference*, Coyoco, Mexico, 2002

[34] J. Mandel, R. Tezaur and C. Farhat. A scalable substructuring method by Lagrange multipliers for plate bending problems. *SIAM Journal on Scientific and Statistical Computing*, 36:1370–1391, 1999.

[35] J. Mandel and R. Tezaur. Convergence of a substructuring method with Lagrange multipliers. *Numerische Mathematik*, 73:473–487, 1996.

[36] H.-O. May. The conjugate gradient method for unilateral problems. *Computers & Structures*, 12(4):595–598, 1986.

[37] B. Radi, O. A. Baba, and J. C. Gelin. Treatment of the frictional contact via a Lagrangian formulation. In Rodin and Shillor [41], pages 407–412.

[38] M. Raous and S. Barbarin. Conjugate gradient for frictional contact. In Curnier [11].

[39] M. Raous, M. Jean, and J.-J. Moreau, editors. *Proceedings of the Second Contact Mechanics International Symposium*, New York, 1995. Plenum Press.

[40] D. Rixen and C. Farhat. A simple and efficient extension of a class of substructure based preconditioners to heterogeneous structural mechanics problems. *International Journal for Numerical Methods in Engineering*, 46:501–534, 1998.

[41] E. Y. Rodin and M. Shillor, editors. *Mathematical and computer modelling*. Number 28. Pergamon Press, 1998.

[42] J. Simo and T. A. Laursen. An augmented Lagrangian treatment of contact problem involving friction. *Computers & Structures*, 42:97–116, 1992.

[43] P. Wriggers. Finite element algorithms for contact problems. *Archives of Computational Methods in Engineering*, 2:1–49, 1995.

[44] P. Wriggers *et al.* Penalty and augmented Lagrangian formulations for contact problems. In *Proc. NUMETA Conf.*, Swansea, 1985.

[45] Z. Zhong and J. Mackerle. Static contact problems — a review. *Engineering Computations*, 9:3–37, 1992.