InterPore JOURNAL

Original Research

A LINE SEARCH ALGORITHM FOR MULTIPHYSICS PROBLEMS WITH FRACTURE DEFORMATION

Ivar Stefansson

Center for Modeling of Coupled Subsurface Dynamics, Department of Mathematics, University of Bergen, Bergen, Norway

Correspondence to:
Ivar Stefansson,
ivar.stefansson@uib.no

How to Cite:
Stefansson, I. (2024). A Line Search Algorithm for Multiphysics Problems with Fracture Deformation. *InterPore Journal*, 1(3), IPJ271124–7.
https://doi.org/10.69631/ipj.v1i3nr33

RECEIVED: 1 July 2024
ACCEPTED: 3 Oct. 2024
PUBLISHED: 27 Nov. 2024

ABSTRACT

Models for multiphysics problems often involve significant nonlinearities. When fracture contact mechanics are incorporated, discontinuous derivatives arise at the interfaces between open and closed fractures, or between sliding and sticking fractures. The resulting system of equations is highly challenging to solve. The naïve choice of Newton's method frequently fails to converge, calling for more refined solution techniques such as line search methods.

When dealing with strong nonlinearities and discontinuous derivatives, a global line search based on the magnitude of the residual of all equations is at best costly to evaluate and at worst fails to converge. We therefore suggest a cheap and reliable approach tailored to the discontinuities. Utilizing adaptive variable scaling, the algorithm uses a line search to identify the transition between contact states for each nonlinear iteration. Then, a solution update weight is chosen to ensure that fracture cells which change state do not move far beyond the transition point.

We demonstrate the algorithm on a series of test cases for poromechanics and thermoporomechanics in fractured porous media. We consider both single- and multifracture cases, and study the importance of proper scaling of variables and equations.

KEYWORDS

Line search, Fracture deformation, Contact mechanics, Multiphysics, Porous media

@2024 The Author

This is an open access article published by InterPore under the terms of the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License (CC BY-NC-ND 4.0) (https://creativecommons.org/licenses/by-nc-nd/4.0/).

1. INTRODUCTION

This paper concerns solution strategies for numerically solving strongly nonlinear and non-smooth equation systems. The primary motivation is multiphysics problems involving fracture contact mechanics in porous media. The developed methods may, however, be relevant for other problems involving friction and contact constraints.

The Newton-Raphson method is widely favored for solving systems of nonlinear equations due to its quadratic convergence. However, this convergence is only locally guaranteed, which becomes particularly restrictive in non-smooth contexts. This limitation motivates the use of globalization techniques to ensure convergence across a broader range of initial guesses. Developing robust solution strategies for such cases is an emerging and highly relevant research area (18).

Globalization schemes are well-studied in the field of optimization, including applications to problems arising from partial differential equations (PDEs). Following Nocedal and Wright (11), these schemes can be categorized into two main families of methods. Trust region methods define a local region around the current iterate within which the update is sought, often using the descent direction from Newton's method. In contrast, line search methods first determine the update direction and then search along this direction for an optimal solution based on a chosen metric, such as minimizing the residual. Because line search methods tend to integrate more seamlessly with existing solution algorithms, they are the approach adopted in this paper.

Inequality conditions such as those arising in contact mechanics lead to constrained optimization problems, requiring modification of the globalization scheme. A straightforward approach is to replace the objective function, which represents the system of equations, with a merit function that additionally incorporates constraint information. This is achieved using additional variables that penalize violation of the constraints. As discussed by Hiermeier (6), the penalty approach presents an inherent challenge: selecting and adapting penalty parameters which balance the two goals of minimizing the objective function and honoring the constraints.

We expect the non-smooth constraints to be the main source of difficulty for the Newton solver. Indeed, in their review of numerical solution of contact problems, Acary et al. (1) point out that the standard methods may need additional criteria to terminate the line search. Moreover, in multiphysics problems, valuable information contained in the constraint functions may be obscured by the rest of the residual.

Together with the cost of performing a line search on the global residual, these considerations motivate a more targeted way of searching along the Newton direction. Herein, we pursue a line search exploiting knowledge of the problem's irregularities. The search is based on the discontinuous derivatives of the contact mechanics relations and is evaluated for the fracture cells only. It prevents updates from passing far beyond a singular point of the constraints, as suggested in the context of multiphase flow by Khebzegga et al. (9) and Moyner (10). We combine this line search with adaptive scaling of the constraint conditions to achieve an algorithm which is both efficient and dependable.

The rest of the paper is structured as follows: Section 2 outlines the mathematical model for frictional contact mechanics and mixed-dimensional thermoporomechanics, highlighting prominent nonlinearities and non-smoothness. The new solution strategy is described in Section 3 and illustrated and tested by the simulation results of Section 4. Finally, we offer concluding remarks in Section 5.

2. MATHEMATICAL MODEL

We consider a mixed-dimensional discrete fracture-matrix model as described by Boon et al. (4), where the domain is partitioned into subdomains Ω_i of dimension d_i. Each pair of neighbouring subdomains separated by one dimension is connected through an interface Γ_j. We detail only the equations of particular relevance to the suggested algorithm herein, referring to the paper by Stefansson et al. (17) for the full mixed-dimensional models: the quasi-static poromechanical model consists of Equations 1, 4-10, 32 and 34, ignoring the temperature terms. We obtain the thermoporomechanical by adding Equations 2 and 33. We close the models by including the relevant boundary conditions and constitutive laws as defined in the paper's Sections 3.4 and 3.5, replacing Equation 28 by **Equation 2** as defined below.

The fracture contact mechanics being of particular importance in the present context, we repeat the central expressions here. Following Hüeber et al (7), we formulate the fracture deformation constraints as non-smooth complementarity functions (**Eq. 1**):

$$C_\perp(\boldsymbol{\sigma}, [\![\boldsymbol{u}]\!]) = -\sigma_\perp - \max\{0,\ -\sigma_\perp - c([\![\boldsymbol{u}]\!]_\perp - g)\} \tag{1}$$

$$C_\parallel(\boldsymbol{\sigma}, [\![\boldsymbol{u}]\!]) = \chi_o \boldsymbol{\sigma}_\parallel - (1-\chi_0)\left[\boldsymbol{\sigma}_\parallel \max\left\{b, \left\|\boldsymbol{\sigma}_\parallel + c[\![\dot{\boldsymbol{u}}]\!]_\parallel\right\|\right\} - b(\boldsymbol{\sigma}_\parallel + c[\![\dot{\boldsymbol{u}}]\!]_\parallel)\right]$$

Here, $\boldsymbol{\sigma}$ denotes the contact traction, which equals the difference between the trace of the effective poromechanical traction onto the fracture surfaces and the fracture pressure. $[\![\boldsymbol{u}]\!]$ denotes the displacement jump between the two sides of the fracture, with the dot indicating increment between successive (time) states. $b(\sigma_\perp) := -F\sigma_\perp$ is the friction bound. The friction coefficient F is assumed to be constant for simplicity. The characteristic function $\chi_o := b \leq 0$ indicates whether the fracture is open, while $c > 0$ is a numerical parameter. Subscripts $\perp$ and $\parallel$ denote the normal and tangential direction of the fracture, respectively. Finally, g is the gap function, i.e., the distance between the fracture surfaces when in mechanical contact. g accounts for shear dilation according to **Equation 2**:

$$g = \tan\phi \left\|[\![\boldsymbol{u}]\!]_\parallel\right\| \tag{2}$$

The dilation angle ϕ determines the strength of the coupling between tangential and normal displacement and by extension the effect on the hydraulic aperture $a := a_{res} + [\![\boldsymbol{u}]\!]_\perp$, with a_{res} denoting the residual hydraulic aperture. To emphasize the strength of the nonlinear coupling, we specify that the fracture permeability is related to a by the cubic law.

3. SOLUTION STRATEGY

The spatial discretisation uses multi-point finite volume methods for stress and diffusive fluxes (12) and a first-order upwind scheme for advective fluxes. We use the two-point flux approximation scheme in the fractures. With this discretisation scheme, the nonlinearity in the fracture permeability may be treated fully implicitly as described in Stefansson & Keilegavlen (16). This removes a source of deterioration of the nonlinear convergence which could otherwise obscure the results, thus allowing us to focus on the effect of the contact mechanics solution strategy.

The implementation is provided in the PorePy simulation toolbox for fractured porous media, which is described by Keilegavlen et al. (8) and Stefansson et al. (17), which also contains a description of the parts of the solution strategy and discretization not detailed herein.

Denoting the solution vector by $x \in \mathbb{R}^n$, we write the full system of discretised nonlinear equations using **Equation 3**:

$$r(x) = 0 \tag{3}$$

The strongly nonlinear and tightly coupled nature of r causes significant difficulties in solving the system, which we do using the semi-smooth Newton method as described in (3, 7, 15). The following sections present an algorithm centred around the fracture deformation equations. It consists of an adaptive scaling and a line search local to the fractures.

3.1. Scaling

As for most solution algorithms, the present one relies on judgements on the magnitude of (updates) of the variables and equations. Wishing to avoid the error-prone approach of combinations of relative and absolute tolerances throughout the algorithm, we attempt to scale $\boldsymbol{\sigma}$ and $[\![\boldsymbol{u}]\!]$ towards unity.

An educated a priori guess about the expected magnitude of the unscaled variables can be based on the driving forces (boundary conditions and source and sink terms). The simulations in Section 4 all have a Dirichlet boundary condition for displacement as a primary driving force. In this case, we can define the characteristic displacement u_c to equal the boundary value. Furthermore, since the examples are defined on unit cube domains, we set the characteristic traction to $\sigma_c = Eu_c$, with E being Young's

modulus in the matrix. Note that, in cases of elastic normal deformation of the fracture, as described, for example, by Bandis et al. (2), the fracture's normal stiffness may be a more appropriate parameter than E. Given a choice of σ_c and u_c, we can replace $\boldsymbol{\sigma}$ in **Equation 1** by the scaled traction variable $\tilde{\boldsymbol{\sigma}} = \boldsymbol{\sigma}/\sigma_c$. Instead of using a scaled displacement, we choose $c = 1/u_c$, thus scaling g as well as $\boldsymbol{u}$. The rest of the model remain unchanged by this scaling.

We stress that, however experienced the practitioner, this guess will contain high uncertainty, typically several orders of magnitude, for complex, real world applications. Therefore, we suggest an adaptive scaling to be used in combination with the line search as described in the subsequent sections. We first define a scaling estimate s^ν, with the superscript denoting cell ν. Aiming to emphasize high values without ignoring lower ones, we then employ the p-mean with $p = 5$ to the local estimate for all fracture cells (**Eq. 4**):

$$s^\nu = \|\tilde{\sigma}^\nu\| + \|c([\![\boldsymbol{u}]\!]^\nu - \boldsymbol{n}g^\nu)\| \tag{4}$$

$$s = \left(\frac{\sum (s^\nu)^p}{\#n_f}\right)^{1/p}$$

Here, ν denotes individual cells, $\#n_f$ the total number of fracture cells and $\boldsymbol{n}$ is the normal vector of the fracture. While p is heuristically chosen, our experience is that the algorithm is only slightly sensitive to its value. The scale s is fixed to its value at the previous iteration. For numerical robustness, we recommend capping s, herein between 10^{-8} and 10^{8}.

3.2. Line search

To pose the problem in optimisation terminology, we define an objective function f, which is the standard choice using the Euclidian norm of r. Following Nocedal and Wright (11), we consider the standard choice $f = 1/2\,\|r\|^2$ and seek the minimizer x^* satisfying $\nabla f(x^*) = 0$. Using Newton's method, the update at iteration k is $p^k = -J^{-1}(x^k)r(x^k)$, with $J(x)$ denoting the Jacobian of r. A line search method determines a beneficial step length $0 < \alpha^k \leq 1$ and updates the iterate according to $x^{k+1} = x^k + \alpha^k p^k$.

3.2.1. Global, residual-based line search

The basic line search algorithms are based on minimizing f along p^k subject to some conditions ensuring convergence. These approaches inevitably add significantly to the computational cost of each iteration. This is especially true for non-smooth problems, which may require dense sampling of trial values of f. Moreover, standard line search algorithms involve criteria (e.g. Wolfe or Goldstein conditions) which require evaluation of gradients (11). This is problematic due to the lack of differentiability of **Equation 1** at the transition between contact states.

However, we include a simple residual-based algorithm minimising f as a comparison to the approach described in the subsequent section. To avoid differentiability issues, we resort to sampling values of f and using a bisection algorithm. To alleviate computational cost, we sample at a limited number of points and bisect based on an interpolation using the monotone cubic spline scheme proposed by Fritsch and Carlson (5), as suggested by Moyner (10) in a similar context involving discontinuities.

3.2.2. Local, constraint-based line search

To design a more targeted line search, we borrow concepts used in the context of transport problems by Moyner (10) and Pour et al. (13). Since the discontinuities are due to the maximum functions, we design an algorithm based on the relative values of their arguments: If the computed Newton update leads to a reversal of which of the arguments of a maximum function is bigger, we seek a weight corresponding to a damped update just beyond the transition point.

To facilitate efficient weight computation, we introduce state indicator functions related to $\max(\theta, \varphi)$ which are linear in the arguments θ and φ and change sign at the discontinuity. For comparison, we define both a constantly and adaptively scaled version (**Eq. 5**):

$$i_c = (\theta - \varphi) \tag{5}$$

$$i_a := \frac{i_c}{s} = (\theta - \varphi)/s$$

Note that the magnitude range of the arguments should be similar to ensure robustness of the root seeking. Scaling i to approximate unity implies the tolerance introduced below is independent of the problem specifics. For the normal constraint of **Equation 1**, we compose $i_{c\perp}$ from **Equation 6**:

$$\theta_\perp = -\tilde{\sigma}_\perp \tag{6}$$

$$\varphi_\perp = c(u_\perp - g)$$

while in the tangential case, we have **Equation 7**:

$$\theta_\parallel = \left|\tilde{\sigma}_\parallel - c_\parallel \dot{\boldsymbol{u}}_\parallel\right| \tag{7}$$

$$\varphi_\parallel = b(\tilde{\sigma})$$

The tangential constraint is inactive for open cells. Thus, to avoid artificial restriction of open cells, we multiply by the Heaviside function $H(i_{c\perp}) = i_\perp > 0$, obtaining $i_{c\parallel} = (\theta_\parallel - \varphi_\parallel)H(i_{c\perp})$.

Denoting a trial weight by α_t^k and using square brackets as shorthand notation $i[\cdot] := i\big(\theta(\cdot), \varphi(\cdot)\big)$, we define the transition indicator as shown in **Equation 8**:

$$t(\alpha_t^k) := -\mathrm{sgn}\big\{i\big(x^k\big) \cdot i\big[x^k + \alpha_t^k p^k\big]\big\} \cdot \left|i\big[x^k + \alpha_t^k p^k\big]\right| \tag{8}$$

The first factor identifies cells transitioning between contact states, while the second factor limits the degree to which they transition into the new state. By introducing a constraint violation tolerance, $\delta > 0$, we compute cell-wise weights, α_t^ν, for the cells where $t^\nu > \delta$, as defined in **Equation 9**:

$$i\big[x^k + \alpha_c^{\nu,k} p^k\big] + \delta \mathrm{sgn}\big(i[x^k]\big) = 0 \tag{9}$$

This is achieved using the interpolation line search described above. By design of **Equation 9**, the weight α_t^ν ensures that the solution in cell ν does not move too far beyond the transition point. The global trial weight α_t is taken as the minimum among all α_t^ν.

The tolerance δ may allow multiple fracture cells to transition within one iteration. While significantly speeding up convergence, this can in rare cases lead to loss of convergence if too many cells transition. Therefore, we recursively impose a tightening of the tolerance whenever the number of transitioning cells of fracture i, $\#t_i = \sum_{\nu \in \Omega_i} t^\nu > 0$, is high relative to the fracture's number of cells, $\#\nu_i$, i.e. (**Eq. 10**), with the cutoff 1 intended for poorly resolved meshes and γ denoting the relative tolerance.

$$\#t_i > \max(1, \gamma \cdot \#\nu_i) \tag{10}$$

That is, we halve δ and recompute α_t^ν according to **Equation 9** as long as **Equation 10** is satisfied. The weight computation procedure is performed for both the normal constraints using $i_\perp$ and their tangential counterpart using $i_\parallel$. Finally, the minimum of the two resulting trial weights is assigned to all degrees of freedom.

4. SIMULATIONS

This section demonstrates the algorithm's efficiency and reliability for a range of test cases, which differ along five dimensions as detailed in the subsections. We compare the suggested constraint based line search with adaptive scaling (CLS i_a) to three approaches: no line search (No LS), a residual-based line search (RLS), and a constraint line search with constant scaling using i_c from **Equation 5** (CLS i_c). The tolerances of the constraint-based algorithm are set to $\delta = 0.3$ and $\gamma = 0.2$ throughout and use a single time step of length 10^6 s in all simulations.

 https://doi.org/10.69631/ipj.v1i3nr33

We compare the results in terms of number of nonlinear iterations, marking runs which either did not converge within one hundred iterations or diverged (i.e., the residual containing infinite entries or similar) in grey and labelling them with NC or Div, respectively. While we do not report run times due to a non-optimised implementation, we note that, unlike the local line search, the global line search adds perceptively to overall computational cost.

4.1. Single fracture

In the first suite of test cases, we include both poromechanics and thermoporomechanics, thus demonstrating applicability to different physics. We vary the dilation angle $\phi \in \{0.1, 0.2\}$, which influences the coupling strength from deformation to fracture flow, cf. **Equation 2**. We prescribe two different mesh sizes $h \in \{1/6, 1/12\}$, thus testing efficiency with respect to the number of fracture cells. Finally, we vary the characteristic displacement scaling employed in **Equation 1** to investigate the robustness in cases where this quantity is difficult to assess a priori. Since using differently scaled variables affects the residual magnitude, we employ a convergence criterion based on the normalised L^2 norm of the nonlinear increment, $\|p^k\|/\sqrt{n} < 10^{-10}$, thus allowing comparison across cases.

This example is defined on a unit cube domain with a single throughgoing fracture as shown in **Figure 1**. There is inflow on the left and outflow on the right fracture boundary, and no-flow conditions elsewhere. We prescribe zero and heterogeneous displacement values on the bottom and top boundary, respectively, resulting in fracture deformation containing both sticking, sliding and open fracture cells as shown in **Figure 1.** The remaining parameters are listed in **Table 1**. While these do not represent any particular physical setting, some characteristics, such as relatively low values for permeability and high ones for stiffness, contribute to retaining the relevant relative importance of the different terms and couplings in the equation system.

Taking the undamped method (No LS) as the baseline, the results reported in **Figure 2** show that the thermoporomechanical problem is more challenging than the poromechanical. Similarly, increasing the number of cells or the dilation angle adds to the difficulty. Both the residual-based and the constantly scaled constraint-based line search (RLS and CLS i_c) obtain convergence in some of the cases where the reference method does not. However, neither is robust with respect to the characteristic displacement scaling, with increased iteration counts or failure to converge in several cases. In contrast, the adaptively scaled constraint-based method (CLS i_a) reliably converges with an iteration count which is constant with respect to the characteristic displacement. The number of iterations is also consistently low and

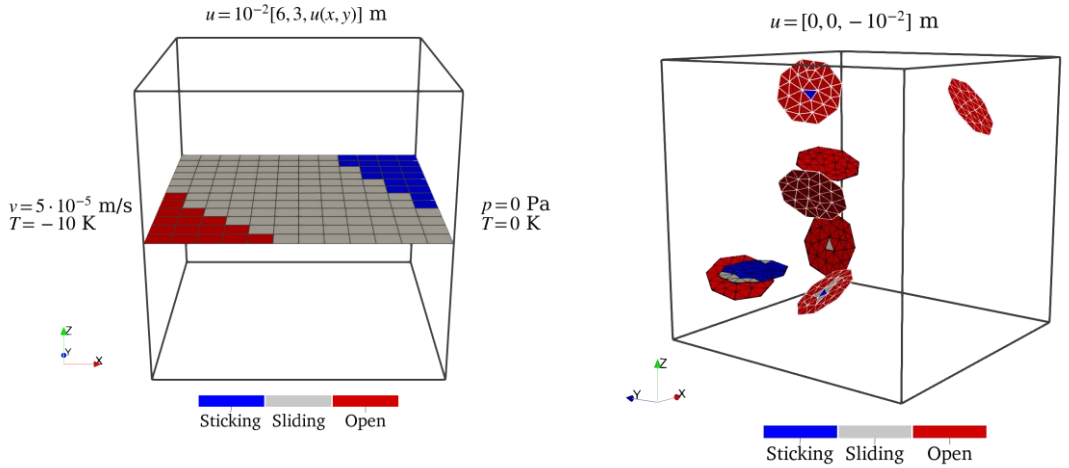

Figure 1: Geometry, selected boundary conditions and fracture deformation solution for the poromechanics $\phi = 0.1$ cases. On the left, we show a Section 4.1 case, with v denoting specific fluid discharge. In the Section 4.2 case on the right, fractures included in the four-fracture case are shown with black edge lines and those unique to the eight-fracture case with white lines. The mostly hidden fracture is fully open.

 https://doi.org/10.69631/ipj.v1i3nr33

Table 1: Simulation parameters.

FLUID PARAMETERS		
Compressibility	$1.0 \cdot 10^{-6}$	1/Pa
Density	1.0	kg/ m^3
Normal thermal conductivity	1.0	W/m/K
Reference pressure	0.0	Pa
Specific heat capacity	100.0	J/kg/K
Reference temperature	0.0	K
Thermal conductivity	1.0	W/m/K
Thermal expansion	0.01	1/K
Viscosity	0.1	Pa s
Inlet/outlet pressure	$1.5 \cdot 10^5 / -1.0 \cdot 10^5$	Pa
Inlet/outlet temperature	$-10.0/0.0$	K
SOLID PARAMETERS		
Biot coefficient	0.8	-
Density	1.0	kg/m^3
Dilation angle	$0.1, 0.2$	-
Friction coefficient	1.0	-
First Lamé parameter	$2.0 \cdot 10^6$	Pa
Normal permeability	$1.0 \cdot 10^{-6}$	m^2
Permeability	$1.0 \cdot 10^{-8}$	m^2
Porosity	0.01	-
Residual aperture	$1.0 \cdot 10^{-3}$	m
Shear modulus	$2.0 \cdot 10^6$	Pa
Specific heat capacity	100.0	J/kg/K
Reference temperature	0.0	K
Thermal conductivity	1.0	W/m/K
Thermal expansion	$1.0 \cdot 10^{-3}$	1/K

competitive with all other converging algorithms. Moreover, the iteration count hardly increases with the number of fracture cells, which equals $1/h^2$.

4.2. Multiple fractures

The second suite also considers two physical models and $\phi \in \{0.1, 0.2\}$, but fixes $u_c = 0.01$. Additionally, we run with both four and eight fully immersed, randomly oriented fractures in the unit cube domain (see Supplementary Material for their vertex coordinates, available online). We prescribe zero fluid and energy flow at all external boundaries. For the momentum balance, we use zero and a constant compressive displacement at the bottom and top, respectively, and zero traction elsewhere. In the centremost cell of each fracture, we prescribe a constant pressure and temperature value, which we pick from two values (which may be interpreted as mimicking injection and production wells). This results in deformation regimes for individual fractures ranging from mostly open through sliding to sticking, depending on fracture orientation and assigned pressure. The remaining parameters are as in the previous section. **Figure 3** shows the fracture geometry, as well as illustrating the contact mechanical state of the poromechanical simulation with $\phi = 0.1$. Since the equations are scaled consistently for all cases, we use a residual-based convergence criterion $\|r^k\|/\sqrt{n} < 10^{-10}$.

The iteration counts shown in **Figure 3** demonstrate trends in convergence behaviour with respect to physical models and ϕ similar to the previous section. Unsurprisingly, increasing the number of fractures also reduces the likelihood of convergence. Again, the residual-based search offers significant but unsatisfactory improvement over the undamped Newton algorithm. The two constraint-based methods consistently converge with similar iteration count, which indicates a quite accurate estimate of u_c. The number of iterations scales very modestly with the number of fractures. This indicates that employing a permissive tolerance δ allows multiple fractures to partly transition within the same iteration.

We emphasize that no convergence issues arose with the relatively lenient tolerances in any test case. Since reducing the tolerances will at some point increase the number of iterations, we recommend this more aggressive choice.

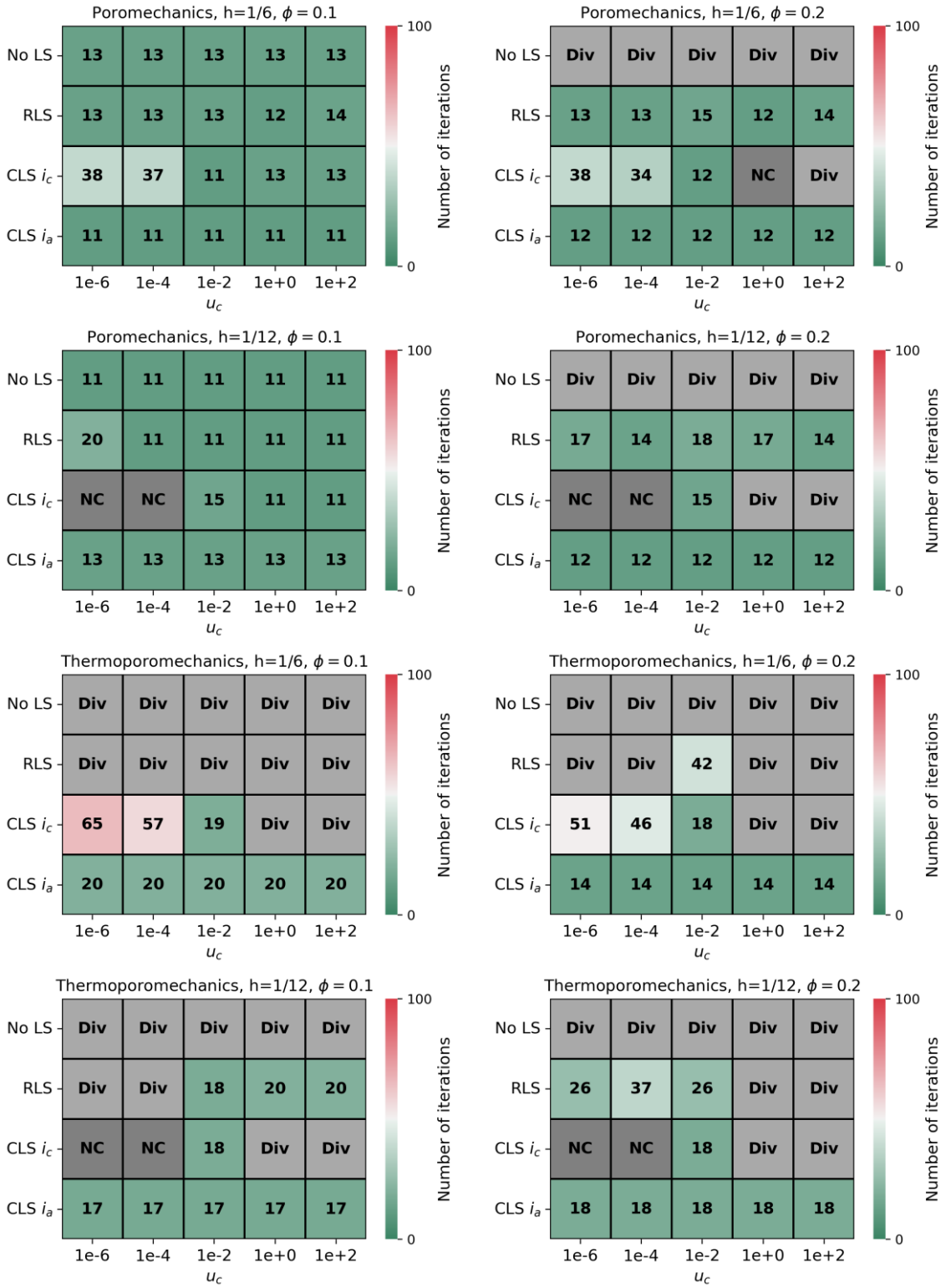

Figure 2: Iteration counts for the examples of Section 4.1. Simulations labeled as 'NC' did not converge within one hundred iterations, while those marked 'Div' indicate cases where the simulation diverged. No LS: undamped method; RLS: residual-based method; CLS i_c: constantly scaled constraint-based method; CLS i_a: adaptively scaled constraint-based method.

 https://doi.org/10.69631/ipj.v1i3nr33

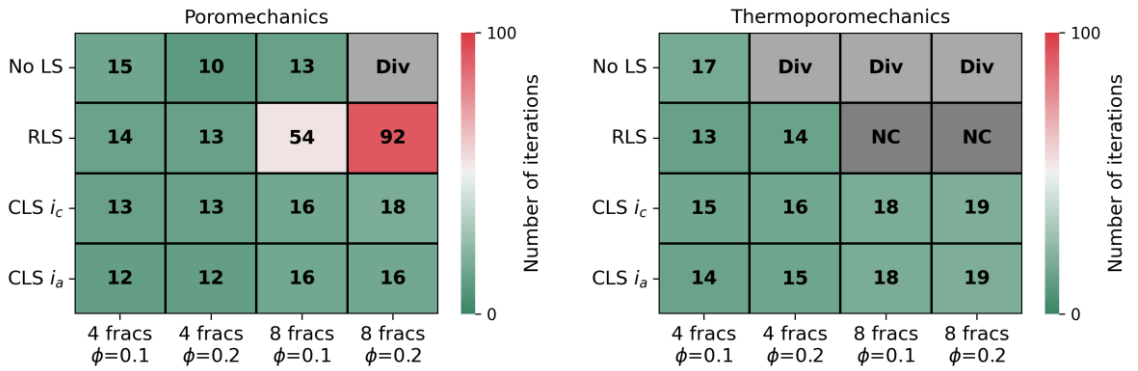

Figure 3: Iteration counts for the Section 4.2 test case with multiple fractures. Simulations labeled as 'NC' did not converge within one hundred iterations, while those marked 'Div' indicate cases where the simulation diverged. No LS: undamped method; RLS: residual-based method; CLS i_c: constantly scaled constraint-based method; CLS i_a: adaptively scaled constraint-based method.

5. CONCLUSION

We have presented a line search algorithm for solving multiphysics problems involving fracture contact mechanics. The nonlinear and non-smooth nature of the equations is addressed in a targeted manner producing a simple, efficient and reliable algorithm. The algorithm consists of adaptive variable scaling and a line search based on the non-smooth part of the fracture contact mechanics equations.

Numerical simulations demonstrate applicability to both poromechanical and thermoporomechanical problems and ability to deal with variables of challenging scales. The suggested approach consistently converges, as opposed to both the standard Newton method and less targeted line search approaches.

We also assess efficiency in terms of number of nonlinear iterations. The results show competitiveness with the alternative approaches for the cases where the latter converge, indicating that the line search hardly introduces any reduction in convergence rate. Moreover, we obtain very favourable scaling with respect to both number of fracture cells and the number of fractures.

STATEMENTS AND DECLARATIONS

Supplementary Material

The supplementary material contains the coordinates defining the fractures used in Section 4.2. This material can be downloaded here.

Conflicts of Interest

There are no conflicts of interest to declare.

Data, Code & Protocol Availability

The simulations were run using PorePy (8) and the run scripts are provided in a GitHub repository at https://github.com/IvarStefansson/A-Line-Search-Algorithm-for-Multiphysics-Problems-with-Fracture-Deformation/. The repository contains a setup for running with a development container based on Visual Studio Code for convenient reproduction code inspection. The full environment is available at Zenodo as a Docker image (14).

Funding Received

This work was funded by the VISTA program, The Norwegian Academy of Science and Letters and Equinor.

ORCID IDS

Ivar Stefansson https://orcid.org/0000-0001-6370-496X

REFERENCES

1. Acary, V., Brémond, M., Huber, O. (2018). On solving contact problems with Coulomb friction: formulations and numerical comparisons. Springer International Publishing. *Advanced Topics in Nonsmooth Dynamics - Transactions of the European Network for Nonsmooth Dynamics*, pp.375-457, 9783319759715. 10.1007/978-3-319-75972-2_10. hal-01878539. https://inria.hal.science/hal-01878539v1
2. Bandis, S. C., Lumsden, A. C., & Barton, N. R. (1983). Fundamentals of rock joint deformation. *International Journal of Rock Mechanics and Mining Sciences & Geomechanics Abstracts*, 20(6), 249–268. https://doi.org/10.1016/0148-9062(83)90595-8
3. Berge, R. L., Berre, I., Keilegavlen, E., Nordbotten, J. M., & Wohlmuth, B. (2020). Finite volume discretization for poroelastic media with fractures modeled by contact mechanics. *International Journal for Numerical Methods in Engineering*, 121(4), 644–663. https://doi.org/10.1002/nme.6238
4. Boon, W. M., Nordbotten, J. M., & Vatne, J. E. (2021). Functional analysis and exterior calculus on mixed-dimensional geometries. *Annali Di Matematica Pura Ed Applicata* (1923 -), 200(2), 757–789. https://doi.org/10.1007/s10231-020-01013-1
5. Fritsch, F. N., & Carlson, R. E. (1980). Monotone piecewise cubic interpolation. *SIAM Journal on Numerical Analysis*, 17(2), 238–246. https://doi.org/10.1137/0717021
6. Hiermeier, M. (2020). Advanced Non-Linear Solution Techniques for Computational Contact Mechanics [Dr. Ing.] Technische Universität München. https://mediatum.ub.tum.de/doc/1521915/1521915.pdf
7. Hüeber, S., Stadler, G., & Wohlmuth, B. I. (2008). A primal-dual active set algorithm for three-dimensional contact problems with coulomb friction. *SIAM Journal on Scientific Computing*, 30(2), 572–596. https://doi.org/10.1137/060671061
8. Keilegavlen, E., Berge, R., Fumagalli, A., Starnoni, M., Stefansson, I., Varela, J., & Berre, I. (2021). PorePy: An open-source software for simulation of multiphysics processes in fractured porous media. *Computational Geosciences*, 25(1), 243–265. https://doi.org/10.1007/s10596-020-10002-5
9. Khebzegga, O., Iranshahr, A., & Tchelepi, H. (2021). A nonlinear solver with phase boundary detection for compositional reservoir simulation. *Transport in Porous Media*, 137(3), 707–737. https://doi.org/10.1007/s11242-021-01584-4
10. Møyner, O. (2017). Nonlinear solver for three-phase transport problems based on approximate trust regions. *Computational Geosciences*, 21(5–6), 999–1021. https://doi.org/10.1007/s10596-017-9660-1
11. Nocedal, J., & Wright, S. J. (1999). Numerical optimization. Springer.
12. Nordbotten, J.M., Keilegavlen, E. (2021). An Introduction to Multi-point Flux (MPFA) and Stress (MPSA) Finite Volume Methods for Thermo-poroelasticity. In: Di Pietro, D.A., Formaggia, L., Masson, R. (eds). *Polyhedral Methods in Geosciences*. SEMA SIMAI Springer Series, vol 27. Springer, Cham. https://doi.org/10.1007/978-3-030-69363-3_4
13. Pour, K. M., Voskov, D., & Bruhn, D. (2023). Nonlinear solver based on trust region approximation for CO 2 utilization and storage in subsurface reservoir. *Geoenergy Science and Engineering*, 225, 211698. https://doi.org/10.1016/j.geoen.2023.211698
14. Stefansson, I. (2024). A line search algorithm for multiphysics problems with fracture deformation. arXiv. https://doi.org/10.48550/ARXIV.2407.01184
15. Stefansson, I., Berre, I., & Keilegavlen, E. (2021). A fully coupled numerical model of thermo-hydro-mechanical processes and fracture contact mechanics in porous media. *Computer Methods in Applied Mechanics and Engineering*, 386, 114122. https://doi.org/10.1016/j.cma.2021.114122
16. Stefansson, I., & Keilegavlen, E. (2023). Numerical treatment of state-dependent permeability in multiphysics problems. Water Resources Research, 59(8), e2023WR034686. https://doi.org/10.1029/2023WR034686
17. Stefansson, I., Varela, J., Keilegavlen, E., & Berre, I. (2024). Flexible and rigorous numerical modelling of multiphysics processes in fractured porous media using PorePy. *Results in Applied Mathematics*, 21, 100428. https://doi.org/10.1016/j.rinam.2023.100428
18. White, J. A., Castelletto, N., Klevtsov, S., Bui, Q. M., Osei-Kuffuor, D., & Tchelepi, H. A. (2019). A two-stage preconditioner for multiphase poromechanics in reservoir simulation. *Computer Methods in Applied Mechanics and Engineering*, 357, 112575. https://doi.org/10.1016/j.cma.2019.112575