Chapter 2
Methodology

The model described in this book computes fluid flow and deformation in discrete fracture networks. As an input, the model requires a realization of the preexisting fracture network. The model has the ability to describe propagation of new fractures, but the potential locations of new fractures must be specified in advance.

Because the boundary element method is used and fluid flow in the matrix surrounding the fractures is assumed negligible, it is not necessary to discretize the volume around the fractures, significantly reducing the number of elements in the discretization.

In the following sections, the modeling methodology is given in detail, including the governing and constitutive equations, numerical methods of solution, methods of discretization, methods of generating realizations of the preexisting (and potentially forming) discrete fracture network, and several special simulation techniques used for realism and efficiency.

2.1 Governing and Constitutive Equations

In the model, fluid flow is single-phase and isothermal. The unsteady-state fluid mass balance equation in a fracture is (adapted from Aziz and Settari 1979):

$$\frac{\partial(\rho E)}{\partial t} = \nabla \cdot (q_{flux} e) + s_a, \tag{2.1}$$

where s_a is a source term (mass per time for a unit of area of the fracture), t is time, E is void aperture (the pore volume per unit area of the fracture), ρ is fluid density, q_{flux} is mass flux (mass flow rate per cross-sectional area of flow), and e is the hydraulic aperture (the effective aperture for flow in the fracture).

Darcy flow is assumed, in which mass flux in a direction x_i is (Aziz and Settari 1979):

M. W. McClure and R. N. Horne, *Discrete Fracture Network Modeling of Hydraulic Stimulation*, SpringerBriefs in Earth Sciences,
DOI: 10.1007/978-3-319-00383-2_2, © The Author(s) 2013

$$q_{flux,i} = \frac{k\rho}{\mu_l}\frac{\partial P}{\partial x_i}, \tag{2.2}$$

where P is fluid pressure, μ_l is fluid viscosity, and k is permeability.

Fluid density and viscosity are functions of pressure (because the simulations were isothermal) and are interpolated from a polynomial curve fit based on values generated using the Matlab script XSteam 2.6 by Holmgren (2007) assuming a constant temperature of 200 °C. A high temperature is used because of the application to geothermal energy. The choice of higher temperature causes fluid density to be modestly lower and viscosity to be nearly a factor of ten lower, relative to room temperature.

The cubic law for fracture transmissivity (the product of permeability and hydraulic aperture) is (Jaeger et al. 2007):

$$T = ke = \frac{e^3}{12}. \tag{2.3}$$

Hydraulic aperture is equal to void aperture between two smooth plates but is lower than void aperture for rough surfaces such as rock fractures (Liu 2005). A "fracture" in a DFN model may represent a crack, but it may also represent a more complex feature such as a fault zone. In the latter case, the void aperture may be much larger than the hydraulic aperture. The model allows e and E to be different.

For single-phase flow in a one-dimensional fracture, the mass flow rate q is:

$$q = \frac{Th\rho}{\mu_l}\frac{\partial P}{\partial x}, \tag{2.4}$$

where h is the "out-of-plane" dimension of the flowing fracture (for example, the height of a vertical fracture that is viewed as a one-dimensional fracture in plan view).

Fluid flow boundary conditions (representing the wellbore) are either constant rate or constant pressure wellbore boundary conditions (Sect. 2.3.10). The boundaries of the spatial domain are impermeable to flow.

Stresses induced by deformation are calculated according to the equations of quasistatic equilibrium in a continuum assuming that body forces are equal to zero. These stresses are given by the vector equation (Jaeger et al. 2007):

$$\nabla^T \mathrm{T}_s = 0, \tag{2.5}$$

where T_s is the stress tensor.

Linear elasticity in an isotropic, homogeneous body is assumed, giving the following relationship between stress and strain according to Hooke's law (Jaeger et al. 2007):

$$\mathrm{T}_s = \frac{2Gv_p}{1-2v_p} trace(\varepsilon) I + 2G\varepsilon, \tag{2.6}$$

where I is the unit matrix, ε is the strain tensor, v_p is Poisson's ratio, and G is the shear modulus.

The shear cumulative displacement discontinuity at any point, D, is equal to the time integral of sliding velocity, v:

$$D = \int v dt. \tag{2.7}$$

A distinction is made between mechanically open and closed fractures. An open fracture is in tension such that the walls are physically separated and out of contact. A closed fracture bears compressive stress, and its walls are in contact.

For a closed fracture, Coulomb's law requires that shear stress be less than or equal to the frictional resistance to slip. We include an additional term, $v\eta$ (the radiation damping term), to approximate the effect of inertia during sliding at high velocities (Rice 1993; Segall 2010). The radiation damping coefficient, η, is defined to be equal to $G/(2v_s)$, where v_s is the shear wave velocity (Rice 1993; Segall 2010). The radiation damping term is on the order of several MPa, which means that the radiation damping term is small unless sliding velocity is at least centimeters per second. The Coulomb failure criterion with a radiation damping term is (Jaeger et al. 2007; Segall 2010):

$$|\tau - \eta v| = \mu_f \sigma_n' + S_0, \tag{2.8}$$

where μ_f is the coefficient of friction, S_0 is fracture cohesion, and σ_n' is the effective normal stress, defined as (Jaeger et al. 2007):

$$\sigma_n' = \sigma_n - P, \tag{2.9}$$

where compressive stresses are taken to be positive. For fractures with shear stress less than the frictional resistance to slip, shear deformation is assumed to be negligible.

Force balance requires that the effective normal stress of open fractures is zero. Because the fluid inside open fractures cannot support shear stress, the walls are stress free (Crouch and Starfield 1983). These stress conditions can be stated:

$$\sigma_n' = 0, \tag{2.10}$$

$$\tau - \eta v = 0. \tag{2.11}$$

Relationships are used to relate effective normal stress and cumulative shear displacement to void and hydraulic aperture. These relationships are chosen to be consistent with laboratory derived relations and so that there is not any discontinuity as elements transition between being open and closed. The aperture of a closed fracture is defined as (Willis-Richards et al. 1996; Rahman et al. 2002; Kohl and Mégel 2007):

$$E = \frac{E_0}{1 + 9\sigma'_n/\sigma_{n,Eref}} + D_{E,eff} \tan \frac{\varphi_{Edil}}{1 + 9\sigma'_n/\sigma_{n,Eref}}, \tag{2.12}$$

where E_0, $\sigma_{n,Eref}$, and φ_{Edil} are specified constants. $D_{E,eff}$ is defined as equal to D if $D < D_{E,eff,max}$, and equal to $D_{E,eff,max}$ otherwise. The constants are allowed to be different for hydraulic aperture, e, and void aperture E. Non-zero φ_{Edil} corresponds to pore volume dilation with slip, and non-zero φ_{edil} corresponds to transmissivity enhancement with slip.

The void and hydraulic aperture of an open preexisting fracture is defined as:

$$E = E_0 + D_{E,eff} \tan \varphi_{Edil} + E_{open}, \tag{2.13}$$

$$e = e_0 + D_{e,eff} \tan \varphi_{edil} + E_{open}, \tag{2.14}$$

where E_{open} is the physical separation between the fracture walls.

The hydraulic and void apertures of newly formed fractures are treated differently than preexisting fractures. A value E_{hfres} is defined as the residual aperture of a newly formed fracture. Hydraulic aperture, e, is set equal to void aperture, E. The aperture of an open, newly formed fracture is:

$$E = E_{hfres} + E_{open}, \tag{2.15}$$

and the aperture of a closed, newly formed fracture is:

$$E = E_{hfres} \exp(-\sigma_n K_{hf}), \tag{2.16}$$

where K_{hf} is a specified stiffness for closed hydraulic fracture elements.

If closed, the transmissivity of a newly formed fracture is defined as:

$$T = T_{hf,fac} E_{hfres}, \tag{2.17}$$

and if open, transmissivity is defined as:

$$T = T_{hf,fac} E_{hfres} + (E_{open})^3/12, \tag{2.18}$$

where $T_{hf,fac}$ is a specified constant. This treatment of transmissivity for newly forming fractures is used so that, if desired, they can be assigned a relatively high residual transmissivity. This might be desirable as a very simple way of approximating the effect of proppant in newly formed fractures, which would tend to cause higher residual transmissivity after closure (Fredd et al. 2001).

2.2 Initial Conditions

In the results shown here, the fluid pressure and stress state (defined by P, σ_{xx}, σ_{yy}, and σ_{xy}) were assumed to be constant everywhere in the model at the beginning of a simulation. In natural systems, the stress state or pore pressure may be spatially

heterogeneous. For example, varying mechanical properties between layers can lead to significant stress heterogeneity and is believed to be responsible for vertical confinement of fractures (Warpinski et al. 1982; Teufel and Clark 1984). If desired, spatially variable stress state could be incorporated into the model simply by changing the stress state of individual elements prior to initiating the simulation.

2.3 Methods of Solution

In this section, the numerical methods of solution for the governing equations are described. The simulator solves the unsteady-state mass balance equation, Equation 2.1, while simultaneously satisfying constitutive relations, updating stresses due to deformation, and satisfying the mechanical equations of force equilibrium. The fluid flow equations are solved using the finite volume method and the mechanical deformation problem is solved using the boundary element method of Shou and Crouch (1995). Time stepping is performed with the implicit Euler method, in which all equations and all unknowns are solved simultaneously in a coupled system of equations during every time step (Aziz and Settari 1979).

2.3.1 Iterative Coupling

During each time step, the simulator solves for the primary variables: pressure, P, open aperture, E, and shear displacement, D, based on equations governing conservation of mass and momentum (details in Sects. 2.3.4, 2.3.5, 2.3.6, and 2.3.7). Iterative coupling is used (Kim et al. 2011) to solve the system of equations. Iterative coupling is a strategy for solving a system of equations in which the system is split into parts and then solved sequentially until convergence. Figure 2.1 is a diagram of the iterative coupling approach.

At the beginning of each time step, the shear stresses equations (Eqs. 2.8 and 2.11; or equivalently, 2.31 and 2.32) are solved as a system of equations with sliding displacements as unknowns while pressure and normal displacements are held constant (Sect. 2.3.5). Next the mass balance and normal stress equations (Eqs. 2.1, 2.10, and 2.12; or equivalently, 2.24, 2.26, 2.27) are solved with pressure and normal displacements as unknowns holding shear displacements fixed (Sect. 2.3.4). The shear stress residuals change during the solution to the flow and normal stress equations, and so after solving the flow and normal stress equations, shear stress residuals are rechecked. Convergence occurs if the shear stress residuals are below a certain threshold, *itertol*. The simulator iterates until convergence. In the simulations in this book, convergence typically occurred in less than ten iterations (most commonly around three), with more iterations required for longer time steps.

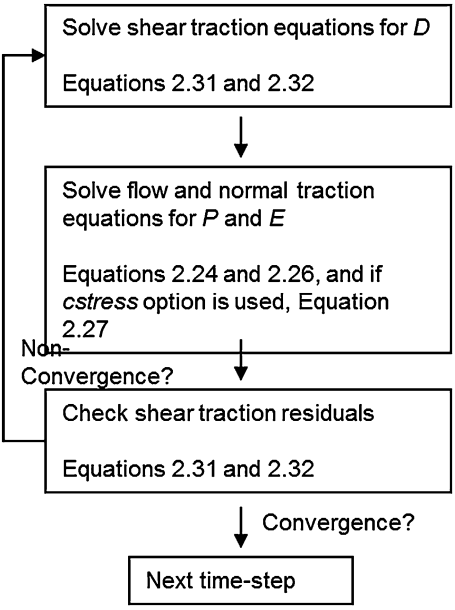

Fig. 2.1 Summary of the iterative coupling approach for a single time step

2.3.2 *Fracture Deformation: Displacement Discontinuity Method*

Stresses induced by deformation are solved with the boundary element method for fracture deformation, the Displacement Discontinuity (DD) Method. Quadratic basis functions are used according to the method of Shou and Crouch (1995). The method of Shou and Crouch (1995) calculates the stresses induced by shear and normal displacement discontinuities by satisfying the equations of quasistatic equilibrium and compatibility for small strain deformation in an infinite, two-dimensional, homogeneous, isotropic, linearly elastic medium in plane strain. The problem reduces to finding the induced stresses $\Delta\tau$ and $\Delta\sigma$ at each element i caused by the shear displacements ΔD and opening displacements ΔE of each element j. Stresses and displacements are linearly related so that:

$$\Delta\sigma_{n,i} = \sum_{j=1}^{n} (B_{E,\sigma})_{ij} \Delta E_j, \tag{2.19}$$

$$\Delta\sigma_{n,i} = \sum_{j=1}^{n} (B_{D,\sigma})_{ij} \Delta D_j, \tag{2.20}$$

$$\Delta\tau_i = \sum_{j=1}^{n} (B_{E,\tau})_{ij} \Delta E_j, \tag{2.21}$$

$$\Delta\tau_i = \sum_{j=1}^{n} (B_{D,\tau})_{ij} \Delta D_j, \tag{2.22}$$

where $B_{E,\sigma}$, $B_{D,\sigma}$, $B_{E,\tau}$, and $B_{D,\tau}$ are matrices of interaction coefficients calculated according to Shou and Crouch (1995). Interaction coefficients do not change during a simulation, and so they can be calculated once and then stored in memory.

The method of Shou and Crouch (1995) assumes strain plane deformation, which implies that the thickness of the deforming medium is infinite in the out-of-plane dimension. In three dimensions, the spatial extent of stress perturbation caused by a deforming fracture scales with the smallest dimension of the fracture (either width or length). Therefore, an implication of plane strain is that the spatial extent of stresses induced by fracture deformation scales linearly with the length of the fracture. In layered formations (more common in gas shale than in EGS), hydraulic fractures are typically confined to mechanical layers and can have length significantly greater than height. In this case, plane strain is not a good assumption. The Olson (2004) adjustment factor can be used to account for the effect of fracture height. Every interaction coefficient is multiplied by a factor:

$$G_{adj,ij} = 1 - \frac{d_{ij}^{\beta}}{\left(d_{ij}^2 + (h/\alpha)^2\right)^{\beta/2}}, \tag{2.23}$$

where h is the specified formation height, d is the distance between the centers of the two elements i and j, α is an empirical value equal to one, and β is an empirical value equal to 2.3 (Olson 2004). If h is very large, then the adjustment factor is nearly one, and the method reduces to plane strain.

2.3.3 Stresses Induced by Normal Displacements of Closed Fractures

The simulator has the option to include (the *cstress* option) or not include (the *nocstress* option) the stresses induced by normal displacements of closed fractures.

The *nocstress* method can be justified because closed cracks experience very small opening displacements in response to significant changes in normal stress. From Barton et al. (1985), closed joints are most compliant at normal stresses below 10 MPa. Reducing normal stress from 10 to 0 MPa leads to a normal displacement of 0.1–0.25 mm, depending on the stiffness of the joint. If a closed, 10 m long fracture experiences 0.1 mm of opening displacement, the approximate stress change (assuming plane strain) due to that displacement is less than 0.1 MPa (from Eq. 2.46). Induced stresses would be even less for larger fractures because fracture stiffness is inversely proportional to size (Eq. 2.46). As a result, the stresses induced by opening displacement of closed elements can be neglected with minor effect on the results unless the joints are very small or unusually compliant.

On the other hand, the *cstress* method may be useful because not all closed fractures are actually thin joints. For example, fault zones may be treated as

fractures in a DFN, but in reality they can be centimeters to meters thick and be composed of a complex zone of porous material and fracturing (Wibberley et al. 2008; Faulkner et al. 2010). In this case, the fractures represented in a DFN are not literally crack-like features. The void and hydraulic apertures are effective values that include the total effect of all sources of fluid storage and transmissivity in the fault zone. Because fault zones can have very high effective void aperture (if they are thick, porous zones), they can contain a very significant amount of fluid, and as a result induce significantly greater displacement, strain, and stress than if they were thin cracks.

While the overall approach of the *cstress* option could be useful for modeling thick fault zones, there are some problems with the current implementation. The model calculates stresses induced by fracture normal displacements using the Shou and Crouch (1995) Displacement Discontinuity Method. This method is intended to calculate the stresses induced by opening of a crack—a displacement discontinuity in which a solid is parted and physically separated along a discrete plane. The physical process of fluid filling a porous, finite thickness fault zone is obviously different from the opening of a crack. One difference is that for the same amount of injected fluid, stresses induced by the opening of a crack are greater because the fluid is emplaced into a much thinner zone, resulting in a much greater strain (because strain is the ratio of displacement and size). Therefore it is not correct to model poroelastic swelling of a fault zone as the opening of a crack. This problem could be resolved in future work by replacing the Shou and Crouch (1995) method with a BEM designed for volumetric, poroelastic strain. Changing the BEM would affect the value of the interaction coefficients between elements, but the overall numerical approach of the *cstress* method (Sect. 2.3.3) would be unchanged.

2.3.4 Solution to the Fluid Flow and Normal Stress Equations

The fractures are discretized into discrete elements (Sect. 2.4.2). The unsteady-state mass balance equation, Equation 2.1, is solved implicitly using the finite volume method, which leads to a nonlinear system of equations that must be solved at each time step. Equations representing conditions for normal stress are included in the fluid flow system of equations.

With the *cstress* method, there is a $(2n) \times (2n)$ system of equations (n is the total number of elements), with pressure and void aperture of each element as unknowns. With the *nocstress* method, there is a $(n + m) \times (n + m)$ system of equations (m is the number of open elements), with pressure of each element and void aperture of each open element as unknowns. In many simulations, m is much smaller than n, and so the *cstress* method can be significantly more computationally intensive.

From the finite volume method, a mass balance residual equation (from Eqs. 2.1 and 2.4) is written for each individual element m:

$$R_{m,mass} = \sum_{q=1}^{Q} T_{g,qm}^{n+1} \left(\frac{\rho}{\mu_l}\right)_{qm}^{n+1} \left(P_q^{n+1} - P_m^{n+1}\right) + s_m^{n+1} - 2a_m h \frac{(E\rho)_m^{n+1} - (E\rho)_m^n}{dt^{n+1}}, \tag{2.24}$$

where the superscript n denotes the previous time step, and superscript $n + 1$ denotes the current time step, Q is the number of elements connected to element m, q is a dummy subscript referring to each of the elements connecting to element m, a is the element half-length, T_g is the geometric part of the transmissibility between two elements (referred to as geometric transmissibility, see Eq. 2.25 for a definition), s_m is a source term such as a well (units of mass per time; positive s is flow into an element), and dt is the duration of the time step. The equations are solved fully implicitly by evaluating all terms at the upcoming time step. An exception is that when using the *cstress* option, the geometric transmissibility term is evaluated explicitly. The term $(\rho/\mu_l)_{qm}$ is evaluated with upstream weighting.

In this dissertation, the term "transmissivity" is used to refer to the ability of fluid to flow through a fracture (in general), and the term "transmissibility" is used to refer to the ability for fluid to flow between two numerical elements. The geometric transmissibilities are calculated using harmonic averaging. For a connection between adjacent elements not at a fracture intersection, geometric transmissibility is calculated as:

$$T_{g,qm} = \frac{2h\left(e_q^3/a_q\right)\left(e_m^3/a_m\right)}{12\left(e_q^3/a_q\right) + \left(e_m^3/a_m\right)}. \tag{2.25}$$

Geometric transmissibilities at fracture intersections are calculated according to the method of Karimi-Fard et al. (2004), which gives a way to calculate geometric transmissibilities between elements at fracture intersections without requiring a zero-dimensional element at the center of the intersection. In the Karimi-Fard et al. (2004) method, geometric transmissibility between two elements at a fracture intersection depends on all elements at the intersection.

For open elements (with either *nocstress* or *cstress*), the additional residual equation is (from Eq. 2.10):

$$R_{E,o} = \sigma_n', \tag{2.26}$$

and the additional unknown is E_{open} (or equivalently, E).

With the *cstress* method, fracture aperture is an additional unknown (and leads to an additional equation) for each closed element. For closed elements with the *cstress* method, the additional residual equation is (from Eq. 2.12):

$$R_{E,c} = E - \frac{E_0}{1 + 9\sigma'_n/\sigma_{n,Eref}} - D_{E,eff} \tan \frac{\varphi_{Edil}}{1 + 9\sigma'_n/\sigma_{n,Eref}}, \tag{2.27}$$

with E as the additional unknown.

With the *nocstress* method, stresses induced by changes in aperture of closed elements are neglected. It is not necessary to include an additional unknown and equation for closed elements (such as Eq. 2.27) because the empirical relations for fracture void and hydraulic aperture (Eq. 2.12) can be substituted directly into the fluid flow residual equation (Eq. 2.24). The hydraulic aperture is treated as a function of P, D, and σ_n, independent of E, and substituted into the fluid flow residual equation in both the *cstress* and the *nocstress* methods.

The system of equations is solved using an iterative method similar to Newton–Raphson iteration. In Newton–Raphson iteration, an iteration matrix is defined such that:

$$J_{ij} = \frac{\partial R_i}{\partial X_j}, \tag{2.28}$$

where i reflects the row number, j reflects the column number, J_{ij} is an entry in the iteration matrix, R_i is an entry in the residual vector, and X_j is an entry in the vector of unknowns (Aziz and Settari 1979). In Newton–Raphson iteration, the iteration matrix is called the Jacobian. However, we do not use the term Jacobian because, as discussed in this section, the iteration matrix used is not a full Jacobian. The algorithm makes a series of guesses for X until certain convergence criteria are met. For each iteration, X is updated according to:

$$X_{new} = X_{prev} + dX, \tag{2.29}$$

where:

$$J\,dX = -R. \tag{2.30}$$

After each update of X, the residual vector is recalculated. Stresses are induced by changes in fracture aperture (changes in aperture of all elements if *cstress* is used, changes in aperture for open elements if *nocstress* is used), and so as a part of the residual update, the stresses at each element are updated based on the changes in aperture. The stress update requires matrix multiplication according to Eqs. 2.19 and 2.21. The two most computationally intensive steps in solving the system of equations are solving Eq. 2.30 and updating the stresses.

In the DD method, normal displacements of each element affect the stress at every other element. As a result, the columns of the full Jacobian matrix corresponding to the normal displacements are dense. Solving a large, dense system of equations, as required by Eq. 2.30, is extremely intensive computationally, and if attempted directly, would severely limit the practical size of the problem that could be solved. To handle this difficulty, an incomplete Jacobian matrix is used as an iteration matrix instead of a full Jacobian matrix.

The iteration matrix is equal to the full Jacobian matrix with interaction coefficients of absolute value below a certain threshold neglected. This approach can effectively solve the system of equations because in plane strain, the magnitude of interaction coefficients decays with the inverse of the square of distance (using the correction of Olson 2004, coefficients decay even faster). As a result, the interaction coefficients of immediate neighbors are much larger than interaction coefficients of distant elements.

The threshold for inclusion in the iteration matrix is set to at 5 % of the value of the self-interaction coefficient (the effect of an element's opening displacement on its own normal stress) for open elements and 30 % for closed elements (if the *cstress* option is used). Using these thresholds, the iteration matrix typically includes five to ten interaction coefficients for each open element and even fewer for each closed element.

Neglecting values in the iteration matrix does not affect the accuracy of the solution; it only decreases the quality of the guesses based on Eq. 2.28. After each dX update is calculated, the stresses caused by the changes in opening displacement are fully calculated according to Eqs. 2.19 and 2.21 without neglecting interaction coefficients.

Using an incomplete Jacobian increases the number of iterations required to reach convergence (relative to using a full Jacobian), but hugely reduces the computational burden of solving Eq. 2.30. Because element interactions are affected mainly by near-neighbors (and time steps are short enough that variables change slightly between time steps), convergence is still possible in a reasonable number of iterations. Typically, the simulator achieves convergence in three or four iterations.

The iteration matrix is an unsymmetric sparse matrix. The publicly available code UMFPACK is used to solve the iteration matrix (Davis 2004a, b; Davis and Duff 1997, 1999). Within UMFPACK, the publicly available codes AMD (Amestoy et al. 1996, 2004; Davis 2006), BLAS (Lawson et al. 1979; Dongarra et al. 1988a, b, 1990a, b), and LAPACK (Anderson et al. 1999) are used.

Several criteria are used to judge convergence in the flow/opening stress equations. The change in the fluid pressure from iteration to another must be less than 0.001 MPa, and the change in opening displacement must be less than 1 micron. A normalized residual vector is calculated by multiplying each stress residual equation by 0.1, each mass balance residual by $dt/(2a_iE_ih\rho_i)$, and (if flow rate is specified) the wellbore residual equation by $1/s_i$. For convergence, the infinity norm of the normalized residual is required to be less than 10^{-4}. The Euclidean norm (scaled by the number of elements) is required to be less than 10^{-5}. If a constant pressure boundary condition is specified, the total flow rate into/out of the system is calculated, and the change in the calculated flow rate between iterations is required to be less than 10^{-5} kg/s.

If the nonlinear solver does not converge within a specified number of iterations, the time step is discarded and repeated with a smaller dt. In testing, it was found that nonconvergence was uncommon.

2.3.5 Solution to the Shear Stress Equations

The shear stress equations are solved with cumulative displacement, D, as the unknown (or equivalently, v or ΔD) holding fluid pressure and normal displacements constant. The sliding velocity v during a time step is equal to ΔD, the change in sliding displacement during the time step, divided by dt, the duration of the time step.

Kim et al. (2011) analyzed several iterative coupling schemes between flow and deformation for stability and convergence. They found that when solving the deformation part of the problem, it was better to hold mass content of each element constant than to hold pressure constant in each element. In this work, pressure was held constant while solving the shear deformation equations. In future work, we will experiment with holding mass content constant, and this may improve convergence rate.

The residual equations for closed and open elements are (from Eqs. 2.8 and 2.11):

$$R_{D,closed} = |\tau - \eta v| - \mu\sigma_n - S_0, \tag{2.31}$$

$$R_{D,open} = |\tau - \eta v| - S_{0,open}, \tag{2.32}$$

The shear stress residual equations are solved as a system of equations. The simulator identifies elements that have negative residual and a sliding velocity of zero and categorizes them as being "locked." The shear stiffness of locked elements is assumed to be infinite, and so locked elements have zero sliding velocity. Because the shear displacements of locked elements are not changing, they are excluded from the system of equations.

To be strictly correct, the cohesion term, $S_{0,open}$, should not be included for open elements because open fractures are not able to bear shear stress. The term is included because it offers a numerical convenience and has little effect on the results (as long as $S_{0,open}$ is small). Without the $S_{0,open}$ term, cohesion vanishes abruptly when closed elements become open elements. This causes very rapid sliding, which forces the simulator to take a large number of very short time steps. Because this process can happen frequently during a simulation (every time an element opens), it can significantly reduce efficiency. Possibly, abrupt loss of cohesion during fracture opening is a realistic process, and perhaps it is a cause of microseismicity. We are not aware of any discussion of this process in the literature. From the point of view of numerical modeling, the process is an inconvenience that increases simulation run-time drastically. The inclusion of the $S_{0,open}$ term prevents this process from occurring.

In the Shou and Crouch (1995) method, shear deformation of an element affects stress at every other element. The result is that the system of equations formed from Eqs. 2.31 and 2.32 is dense. As with the normal stress equations (Sect. 2.3.4), an iterative method similar to Newton–Raphson iteration is used. An iteration matrix is formed from the full Jacobian matrix, but entries with absolute value

below a certain threshold are set to zero. All interaction coefficients less than a certain factor, $J_{mech,thresh}$ (in the simulations in this book, 0.01 was used), of an element's self-interaction coefficient are removed from the iteration matrix, resulting in a sparse system of equations. A series of iterations are performed until the infinity norm of the shear stress residuals is less than a prescribed tolerance, *mechtol*. Special complications are discussed in Sects. 2.3.6 and 2.3.7. In the simulations in this book, convergence typically occurred in less than five or ten iterations.

The iteration matrix is an unsymmetric sparse matrix. The publicly available code UMFPACK is used to solve the matrix (Davis 2004a, b; Davis and Duff 1997, 1999).

Compared to solving the system of equations directly, the iterative method radically improves the scaling of computation time with size. Direct solution of a dense matrix scales like n^3, where n is the number of elements. Assuming that solving the iteration matrix is a negligible cost, the iterative method reduces the problem to several matrix multiplications, which scale like n^2 (with direct multiplication). As discussed in Sects. 2.5.1, 3.5, and 4.5, an efficient matrix multiplication technique is used that further reduces the problem scaling to between n and $n\log(n)$.

Because the iteration matrix is a general sparse matrix, solving the matrix may not necessarily be a trivial computational expense for larger problems or for larger values of $J_{mech,thresh}$. If the cost of solving the iteration matrix system became prohibitive (in testing for this book, it did not), a banded iteration matrix could be used. A banded iteration matrix would require more iterations because it would not be able to include interactions between nearby elements in adjacent fractures (which are particularly important at fracture intersections). However, the solution of banded matrices is very efficient, and so solution time of the iteration matrix could be guaranteed to be small.

2.3.6 Inequality Constraints on Fracture Deformations

To enforce realistic behavior, two inequality constraints are imposed for fracture deformations. The walls of open fractures may not interpenetrate ($E_{open} \geq 0$), and a fracture may not slide backwards against the direction of shear stress ($\tau\Delta D \geq 0$). During every iteration of the flow/normal stress subloop, prior to updating stresses caused by the changes in displacement, the model checks each element to see if the applied displacement will violate an inequality constraint. If an applied displacement will result in the violation of an inequality constraint, the applied displacement is adjusted so that equality will be satisfied (resulting in E_{open} and ΔD being equal to zero).

Adjustments to enforce the constraints are made frequently by the simulator. When the iteration matrix is solved, the algorithm is unaware of the constraints, and so whenever an element transitions from open to closed or from sliding to not

sliding, the update overshoots zero (E_{open} or ΔD equal to zero), and an adjustment must be applied. These adjustments tend to be small and typically have a minor effect on convergence.

However, because adjustments are nonsmooth perturbations to the residual equations, they have the potential to cause nonconvergence. Convergence problems did not occur for any of the simulations performed for this book, but with testing, it was found that convergence could sometimes be a problem when solving the shear stress residual equations on very complex, dense, and/or poorly discretized fracture networks. If nonconvergence occurs, the simulator automatically reduces time step duration. With sufficient reduction in time step, displacements can always be made small enough to assure convergence. However, frequent time step reduction due to convergence failure leads to poor efficiency and is not desirable for optimal performance.

The shear stress residual equations can fail to converge if a complex cluster of intersecting fractures is poorly discretized. In this case, a group of elements can interact in such a way that the iteration matrix consistently attempts to make updates that violate the constraint. A cycle results as the updates violate the constraint and then are reset. Unlike the normal stress element residual equation (Eq. 2.26), the shear stress residual equation of closed elements (Eq. 2.31) contains both normal and shear stress, which increases the potential for complex interactions between elements.

The tendency to for nonconvergence is affected by the number of elements included in the iteration matrix. With more elements included in the iteration matrix, nonconvergence is more likely because there is greater potential for complex interactions between neighboring elements. However, the system converges more rapidly if more elements are included. Therefore, a trade-off exists between efficiency (including more elements in the iteration matrix, which reduces the number of iterations required) and robustness (including fewer elements in the iteration matrix to prevent the possibility of reverse sliding). Including too many elements in the iteration matrix can lead to reduced efficiency if the time required to solve the iteration matrix becomes nonnegligible compared to the time required to update stresses. The most robust iteration matrix is probably one that has zeros everywhere except the main diagonal, but this matrix may require hundreds of iterations for convergence. For these reasons, the user specified parameter $J_{mech,thresh}$, which determines the number of elements included in the iteration matrix, has an important impact on efficiency and robustness. For the simulations in this book, $J_{mech,thresh}$ equal to 0.01 was used.

In the simulations in this book, convergence failure of the shear stress residual equations did not ever occur. However, convergence failure is possible in poorly discretized networks. If this is happens, the best solution is to refine the discretization. An alternative approach is to use a larger value of $J_{mech,thresh}$.

2.3.7 Changing Mechanical Boundary Conditions

As discussed in Sects. 2.3.4 and 2.3.5, different forms of the stress equilibrium equation are solved depending on whether an element is open, sliding, or locked (Eqs. 2.26, 2.27, 2.31, and 2.32). A major issue for the simulator is that element status may change during a time step, and so it is not known in advance which equations to solve for each element. The issue of changing boundary conditions is handled in two ways: frequent checking of element status between iterations and time stepping.

The use of time stepping is advantageous because deformation is always small for a sufficiently short time step. Time step duration plays a direct role in the residual equations: in the accumulation term of Eq. 2.24 and the radiation damping term of Eqs. 2.31 and 2.32. Reducing time step sufficiently always enables the simulator to converge because the residual equations do not contain discontinuities (although there are discontinuities in the derivative of the residual equations).

Typically, time step reduction to enable convergence is not necessary because the combination of iterative methods and frequent checking of element status allows convergence to be achieved in the vast majority of cases. Iterative methods are used in both the flow/normal stress and the shear stress subloops. The effect of these iterative methods is that fractures are deformed gradually as they converge to the solution. Changing element status can be problematic because it creates discontinuities in the derivative of the residual equations that worsen convergence. However, because the equations are solved with iterative methods that apply a series of small deformations, the effect of the discontinuities is dampened and typically convergence can be achieved. The need for gradual deformation is another reason why including too many elements in the iteration matrix (using a $J_{mech,thresh}$ that is too low) could be problematic for robustness.

As these gradual deformations are applied, element status is checked constantly. Elements are checked for opening and closing after every iteration in the flow/normal stress subloop and checked for opening/sliding/locking after every iteration in the shear stress subloop. An exception is that elements that are locked at the beginning of the shear stress subloop are not checked during every iteration of that subloop—they are assumed to remain locked. If dynamic friction weakening is used (Sect. 2.5.3), the status of all elements is checked after every iteration of the shear stress subloop. After the shear stress subloop has converged (before beginning the flow/normal stress subloop), element status is checked for every element, even the previously locked elements. Finally, after the flow/normal stress subloop, before evaluating the overall coupling error for the iterative coupling, the element status is checked a final time.

Convergence problems could hypothetically occur if elements entered a cycle in which status switched back and forth from one iteration to another. In testing, cycles were rarely observed even for large problems that had many elements that are opening, sliding, or locked.

Because time stepping is used, typically the deformation equations are solved when the solution (the deformation at the end of the time step) is quite close to the initial guess (the deformation at the beginning of the time step). However, in Sect. 3.2.1, it is demonstrated that the method described in this book can be used for solving contact problems where the initial guess is not close to the solution.

2.3.8 Formation of New Tensile Fractures

The model has the ability to model the propagation of new tensile fractures, but it requires that the location of potential new tensile fractures be specified in advance. The potential new fractures are discretized prior to simulation. Before potential fracture elements become "real" elements (become "active"), they are considered "inactive," which means that they have zero transmissivity and cannot slide or open. Inactive elements are not included in any of the systems of equations. However, the stress state at inactive elements is updated constantly throughout the simulation. Figures 2.2, 2.3, and 2.4 show examples of fracture networks containing both natural and potentially forming fractures.

The process by which an element is activated is discussed in Sect. 2.3.8. When an element is activated, it is given an aperture equal to E_{hfres} (see Eqs. 2.15 and 2.16), set at 10 microns. Because the E_{hfres} is increased from zero to 10 microns, activation of an element does not strictly conserve mass. However, because E_{hfres} is small, the error is small relative to the total amount of fluid injected during stimulation. E_{hfres} could be made smaller, but practically, elements with very small apertures can be problematic for the simulator. Once activated, an element is never deactivated.

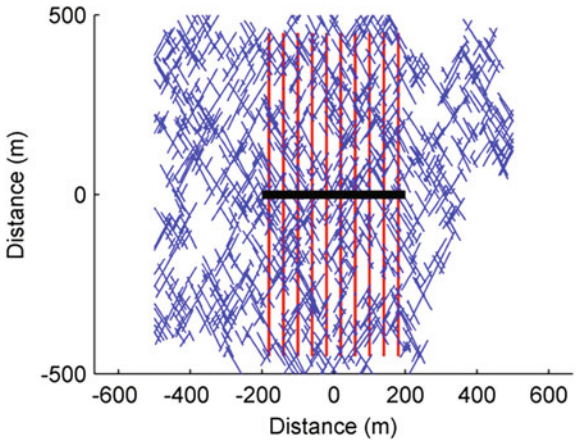

Fig. 2.2 An example of a fracture network with prespecified deterministic potential hydraulic fractures. The *black line* is the (*horizontal*) wellbore; the *blue lines* are preexisting fractures, and the *red lines* are the potentially forming hydraulic fractures

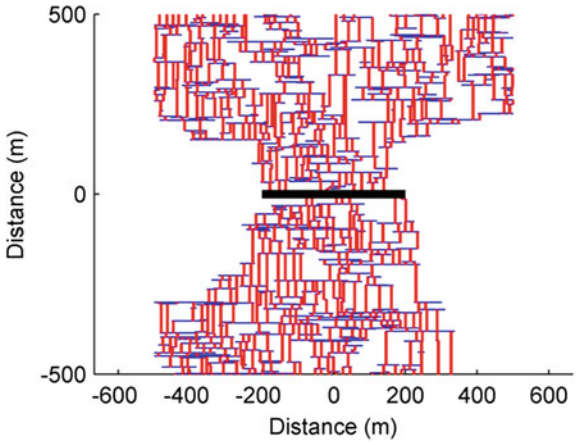

Fig. 2.3 An example of a fracture network with potential hydraulic fractures. The *black line* is the wellbore; the *blue lines* are preexisting fractures, and the *red lines* are the potentially forming hydraulic fractures

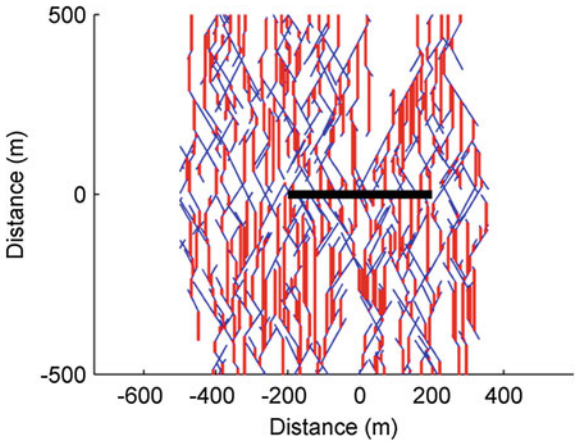

Fig. 2.4 An example of a fracture network with potential hydraulic fractures. The *black line* is the wellbore; the *blue lines* are preexisting fractures, and the *red lines* are the potentially forming hydraulic fractures

2.3.9 Adaptive Time Stepping

With adaptive time stepping, the duration of each time step is varied in order to optimize efficiency and accuracy. Time steps are selected based on the maximum change in a variable δ_i^n, defined to measure how quickly stress is changing at each element. δ_i^n is defined as the sum of the absolute value of the change in effective normal stress and the absolute value of the change in shear stress at each element i during the time step n. The duration of the next time step is selected using the method suggested by Grabowski et al. (1979):

$$dt^{n+1} = dt^n \min_i \left(\frac{(1+\omega)\eta_{t\,\arg}}{\delta_i^n + \omega\eta_{t\,\arg}} \right), \tag{2.33}$$

where η_{targ} is a user specified target for the maximum change in δ and ω is a factor that can be between zero and one (in this book, ω was set to one). If $\delta_i > 4.0\eta_{targ}$ for any element, the time step is discarded and repeated with a smaller value of dt.

Several secondary conditions are used to select time step duration. If convergence failure occurs in the shear stress residual equations, the flow/opening residual equations, or the overall iterative coupling, the time step duration is cut by 80 % and repeated. The time step duration is sometimes adjusted so that the end of the time step coincides with the prespecified duration of the simulation or a prespecified adjustment in the wellbore boundary conditions (Sect. 2.3.10).

2.3.10 Wellbore Boundary Conditions

Wellbore boundary conditions can be specified by rate or pressure. For injection, the user specifies a maximum injection pressure, P_{injmax}, and a maximum injection rate, q_{injmax}. For production, the user specifies a maximum production rate, $q_{prodmax}$, along with a minimum production pressure, $P_{prodmin}$. A schedule of wellbore control parameters is specified by the user. At each time step, the simulator identifies which constraint should be applied such that the other is not violated and implements the appropriate boundary condition.

For constant rate boundary conditions, the source term s in Eq. 2.24 is nonzero for elements connected to the wellbore. The specified rate is enforced such that it is equal to the sum of the source terms of all elements connected to the wellbore:

$$S = \sum_k s_k, \tag{2.34}$$

where S is the specified injection/production rate (positive for injection) and k cycles through every element connected to the wellbore. Equation 2.34 is added to the system of flow equations, and P_{inj} is the corresponding unknown. The source term of each element connected to the wellbore, s_k is:

$$s_k = \frac{T_{g,wk}(P_{inj} - P_k)\rho_{wk}}{\mu_{l,wk}}. \tag{2.35}$$

The geometric transmissibility between a fracture element and the wellbore, $T_{g,wk}$, is calculated from Eq. 2.25 with the assumption that e_w is infinite (although obviously a wellbore does not literally have a hydraulic aperture). With a very large e_w, Equation 2.25 reduces to:

$$T_{g,wk} = 2he_k^3/(12a_k). \tag{2.36}$$

If a rate of zero is specified, the wellbore remains connected to the formation and there can be cross-flow between fractures through the wellbore. Pressure drop

within the wellbore is assumed negligible. In this case, Equation 2.34 is still included in the flow equations, but S is set to zero.

Constant pressure boundary conditions are implemented by including an element in the flow equations with a very large (effectively infinite) volume and hydraulic aperture. Because the element has a very large volume, its pressure remains constant regardless of fluid flow in or out of the element.

2.4 Spatial Domain

2.4.1 Generation of the Discrete Fracture Network

The fracture network used by the simulator can be generated deterministically or stochastically. If specified deterministically, the user explicitly specifies the locations of the fractures (or the potentially forming fractures).

Simple techniques are used for stochastic fracture network generation. Fractures are generated sequentially with an accept/reject algorithm. The total number of fractures in the network is specified by the user. Before identifying the locations of any of the fractures, a population of lengths and orientations is chosen according to prespecified statistical distributions. The locations of the fractures are then determined in order from largest fracture to smallest. For each fracture, a candidate location is identified, and certain checks are performed before "accepting" the candidate location. If the checks are not satisfied, a new candidate location is selected, and the process is repeated until an acceptable location is found.

Candidate locations are chosen at random within a specified spatial domain. In order to avoid boundary effects, the candidate locations are located in a spatial domain larger than the problem domain. After all the fractures have been generated, the network is cropped to the size of the problem domain.

Candidate fracture locations are accepted or rejected on the basis of several checks designed to avoid numerical difficulties associated with the use of the Displacement Discontinuity Method (Shou and Crouch 1995). As discussed in Sect. 2.5.6, low-angle fracture intersections cause numerical difficulties. To avoid this problem, fractures are not allowed to intersect (or come within one meter of intersecting) at an angle below a specified threshold (in Models B and C in this book, the threshold was 20°). In addition, fracture intersections are not allowed to be closer than one meter.

As discussed in Sect. 2.3.8, new tensile fractures can form, but the potential locations of these new fractures must be specified in advance. There are several simple algorithms that are used to identify the potential locations of these new tensile fractures. In all cases, the potential new hydraulic fractures are assumed to be perpendicular to the remote least principal stress. This is a simplification because, while newly formed fractures should form perpendicular to the least

principal stress, deformations during stimulation cause stress perturbation that can rotate the principal stresses locally from its original orientation.

One way to locate the potential new hydraulic fractures is deterministically. To ensure compatibility with the accept/reject requirements for fracture intersections, deterministic potential fractures are located prior to stochastic generation of fractures. An example is shown in Fig. 2.2. The network shown in Fig. 2.2 was used in Model B (Sect. 3.3.1).

Two other algorithms are used for locating potential new tensile fractures. These algorithms are applied after the generation of the stochastic preexisting fracture network (rather than before, as with the deterministic method). These two algorithms can allow propagating fracture to terminate against natural fractures or propagate across natural fractures. In the results shown in this book, the algorithms were used to create fracture networks where propagating fractures terminate against natural fractures.

One algorithm randomly selects a few locations along the wellbore to initiate potential new fractures, then propagates them away from the wellbore until they intersect preexisting fractures. Next, a few random locations along each of the intersected preexisting fractures are chosen, more potential fractures are initiated and propagated, and the process is repeated. A sample discretization using this method is shown in Fig. 2.3. To avoid numerical difficulties, the newly formed fractures are not permitted to be closer than one meter from each other.

Another algorithm locates the potential new fractures off the tips of the preexisting fractures. These fractures are then propagated away from the wellbore until they reach a preexisting fracture, and then the cycle is repeated. An example is shown in Fig. 2.4. The network shown in Fig. 2.4 is the network used for Model C in Sect. 3.3.2.

After all of the fractures have been generated, a breadth-first graph-search algorithm is used to identify which fractures are connected to the wellbore through a continuous pathway through preexisting or potential fractures. Because the matrix permeability is assumed to be zero, fractures that do not have a continuous pathway to the wellbore are hydraulically isolated from the injection. These fractures are omitted from the simulation.

2.4.2 Spatial Discretization

Because the matrix permeability is assumed to be negligible and mechanical deformation is calculated using the boundary element method, it is only necessary to discretize the fractures, not the area around the fractures. An example of a typical discretization is shown in Fig. 2.5. Element size is not required to be constant.

The boundary element method of Shou and Crouch (1995) is inaccurate within a factor (less than one) of an element's half-length. To minimize inaccuracy, the discretization is refined when fractures are close together.

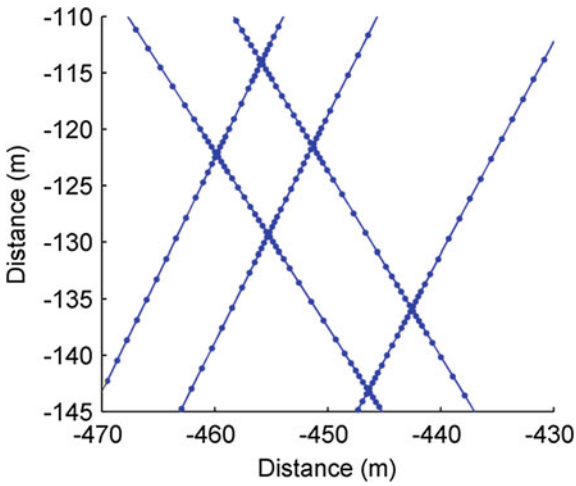

Fig. 2.5 Example of a fracture network discretization. Each dot is located at the center of an element. Note the refinement around the fracture intersections

The discretization is performed in two parts. In the initial discretization, the fractures are discretized with a roughly uniform element half-length, a_{const}. In the second phase, the discretization is refined iteratively.

In the initial discretization, the algorithm starts at one end of each fracture and moves across it, generating one element at a time with a length of $2a_{const}$. Fracture intersections cannot occur in the middle of an element, and so an element is terminated if an intersection is reached or the end of the fracture is reached. Termination of elements at intersections or fracture endpoints results in elements that are not $2a_{const}$ long. If a newly created element has a half-length below a certain threshold, $0.5a_{const}$, it is combined with the previous element. An element is not combined with the previous element if the previous element is on the other side of an intersection or if there isn't a previous element because the new element is the first element on the fracture. If the endpoint of a fracture is less than $0.1a_{const}$ from a fracture intersection, the end of the fracture is removed, leaving a "T" shaped intersection.

After generation of the initial discretization, an iterative algorithm is used for further refinement. During each iteration, two conditions are checked for each element, and if an element violates either condition, it is split in half. A minimum element half-length is set, a_{min}, below which an element is not split further. This process is repeated until no further elements are split.

The first condition enforces refinement around fracture intersections:

$$a_i < l_c + d_{ij} l_f, \tag{2.37}$$

where a_i is the half-length of element i, d_{ij} is the distance from the center of element i to intersection j, and l_c and l_f are constants. The second condition prevents elements from being too close together:

$$l_k a_i < D_{ij}, \tag{2.38}$$

where l_k is equal to a specified value, either l_s if elements i and j belong to the same fracture or l_o if i and j belong to different fractures. For elements of the same fracture, D_{ij} is the distance between the center of element j and the edge of element i. For elements of different fractures, D_{ij} is the smallest of three numbers: the distance between the center of element j and either (1) the center of element i or (2) either of the two endpoints of element i.

To avoid evaluating Eq. 2.38 n^2 times every iteration, the spatial domain is divided into a grid, and Eq. 2.38 is only evaluated between elements in neighboring gridblocks or in the same gridblock.

2.5 Special Simulation Topics

2.5.1 Efficient Matrix Multiplication

In the Shou and Crouch (1995) method, deformation at every element affects stress at every other element. As a result, updating stress requires multiplication of a dense matrix of interaction coefficients—a process that scales like n^2. As problem size grows, computation time and RAM requirements quickly become prohibitive.

Fortunately, approximate methods for efficient matrix multiplication are available. Two techniques are the fast multipole method (Morris and Blair 2000) and hierarchical matrix decomposition (Rjasanow and Steinbach 2007). The model in this book uses Hmmvp, a publicly available implementation of hierarchical matrix decomposition and adaptive cross approximation (Bradley 2012). Prior to running simulations, Hmmvp is used to perform matrix decomposition and approximation of the four matrices of interaction coefficients (Eqs. 2.19, 2.20, 2.21, and 2.22) for a given fracture network. The decomposition is stored and loaded into memory at the beginning of a simulation. Within the simulator, Hmmvp is used to perform the stress updating matrix multiplications. In Sects. 3.5 and 4.5, it is demonstrated that Hmmvp drastically reduces the memory and computation requirements for matrix multiplication.

2.5.2 Crack Tip Regions

Special treatment is needed to model the progressive opening of both preexisting and newly forming fractures. As discussed in Sect. 2.3.8, the simulator is capable of modeling the propagation of new tensile fractures, but the locations of these fractures must be specified in advance. The simulator needs to have conditions to handle three situations: (1) the initiation of a new fracture (the activation of the first element on a newly forming fracture), (2) the extension of a newly forming

fracture (the activation of subsequent elements on the newly forming fracture), and (3) the progressive opening of a preexisting fracture.

Initiation of new tensile fractures is handled in a very simple way. On each potentially forming hydraulic fracture, the (not yet activated) elements that are touching preexisting fractures or the wellbore are identified and labeled "initiation" elements. Initiation elements are "linked" to their adjoining elements on the neighboring fracture or the wellbore. At the end of each time step, a check is performed on all initiation elements to see if they should be activated. During the check, the fluid pressure of each initiation element is assumed to be equal to the fluid pressure that is highest among the elements that it is linked to. If the effective normal stress of an initiation element is less than zero (tensile), the fracture is initiated. To avoid discretization dependence, all elements on the potentially forming fracture that are within one meter of the intersection (including the initiation element itself) are activated.

Once an element on a potentially forming hydraulic fracture has been activated, a second algorithm is used to extend the fracture. The stress intensity factor at a crack tip is estimated using the Displacement Discontinuity Method according to the equation (Schultz 1988):

$$K_I = \frac{G}{4\pi(1-\upsilon)} E_{open} \sqrt{\frac{2\pi}{a}}, \tag{2.39}$$

where E_{open} is the opening of the crack tip element and a is its half-length. If the stress intensity factor reaches a critical threshold, $K_{I,crithf}$, then the fracture is allowed to propagate.

To propagate a fracture, a process zone is defined as the region of the newly forming fracture 2.0 m ahead of the crack tip. If the critical stress intensity factor is reached at the crack tip element, elements within the process zone are activated. For application of Eq. 2.39, a crack tip element is permitted to be any element in the fracture that has been activated. It is not required to be the outermost active element.

A special treatment is also used to handle the propagation of opening along a preexisting fracture. In this case, only part of a preexisting fracture has opened. The location of the transition from open to closed on the fracture is considered an effective crack tip.

Without a special treatment, propagation of opening along a preexisting fracture could be unrealistically slow. In order for opening to propagate along a preexisting fracture, fluid must be able to flow from the open element at the effective crack tip into the adjacent closed element. Because geometric transmissibility between elements is calculated using a harmonic average (Eq. 2.25), the rate of flow between a high transmissivity element and low transmissivity element will be limited by the low transmissivity element. Because of this, without a special adjustment, the propagation of opening down a preexisting fracture will be limited by the transmissivity of the closed elements ahead of the effective crack tip. This is not realistic because the crack tip should be able to propagate at a rate proportional

to the (high) transmissivity of the open fracture behind the tip as fluid flows in behind it and progressively opens the crack. This is the same reason why a crack can propagate through very low permeability matrix: propagation depends on fluid flowing in behind the crack tip, not flowing ahead of the crack tip.

Opening induces tensile stresses ahead of the crack tip, but these stresses are unable to open the elements ahead of the effective crack tip because of a poroelastic response. Because of the low compressibility of water (relative to the fractures) and conservation of mass, effective normal stress of an element must be nearly constant unless there is fluid flow into or out of the element (otherwise the void aperture and resulting mass content of an element would change, from Eqs. 2.12 and 2.24). Because of this effect, the tensile stresses induced ahead of the crack tip produce a reduction in fluid pressure that allows the effective normal stress to remain nearly constant and prevents opening.

To correct for this problem, a special adjustment is used to increase element transmissivity ahead of effective crack tips. The stress intensity factor at an effective crack tip is calculated using Eq. 2.39. If the stress intensity factor reaches a critical value, $K_{I,crit}$ (permitted to be different than $K_{I,crithf}$), then elements within 2.0 m of the effective crack tip (belonging to the same fracture) are placed in a process zone. The hydraulic aperture of the elements in the process zone is assigned a process zone hydraulic aperture, e_{proc} (equal to 106 microns), unless their hydraulic aperture is already higher than e_{proc}. The processes zone hydraulic aperture is high enough that fluid can rather quickly flow into the process zone.

As an aside, it is worth mentioning why a special adjustment is *not* needed for shear stimulation. In shear stimulation, transmissivity enhancement due to induced slip advances along a preexisting fracture. An effective shear crack tip can be defined at the boundary between where slip and transmissivity enhancement have taken place and where they have not. Shear stress concentration occurs ahead of the effective shear crack tip, just as tensile stress concentration occurs ahead of an effective opening crack tip. However, unlike for tensile stresses, there is not a poroelastic response that acts to counteract the effect of the stress concentration. In this case, the induced shear stress can cause shear deformation and transmissivity enhancement ahead of effective crack tip without requiring any fluid to flow ahead of the crack tip, a process we refer to as Crack-like Shear Stimulation (Sects. 3.4.2.2 and 4.4.2 in McClure 2012).

2.5.3 Dynamic Friction Weakening

Seismicity occurs when friction weakens rapidly on a fault, resulting in a rapid release of shear stress through slip. The leading theory to describe friction evolution on a fault is rate and state friction, and it is widely used in earthquake modeling (Dieterich 2007). In rate and state friction, the coefficient of friction changes over time as a function of sliding velocity and the past sliding history of

the fracture. According to rate and state friction, the coefficient of friction is (Segall 2010):

$$\mu_f = f_0 + a_{rs} \ln \frac{v}{v_0} + b \ln \frac{\theta v_0}{d_c}, \tag{2.40}$$

where f_0 is a constant of order 0.6–1.0, a_{rs} and b are constants of order 0.01, v_0 is a specified reference velocity, θ is a state variable that changes over time as friction evolves on the fracture, and d_c is a characteristic weakening distance on the order of microns (though in some applications, much larger values are used). In McClure and Horne (2011) of Chap. 1, a numerical approach was demonstrated that couples rate and state friction evolution, deformation, and fluid flow.

Coupling seismicity modeling with fluid flow is useful because in some cases seismicity is triggered by injection or other human activities (McGarr 2002).

Rate and state friction is significantly more intensive computationally than simulations using constant coefficient of friction. One reason rate and state friction models are so computationally intensive is that they are numerically unstable unless a very finely resolved spatial discretization is used (LaPusta 2001). Another reason is that rate and state friction requires a very large number of very short time steps in order to simulate seismic events accurately because frictional weakening is very non-linear.

In McClure and Horne (2011) of Chap. 1, explicit third-order Runge–Kutta time stepping was used. Sliding velocity was calculated by treating velocity as an unknown and calculating the velocity at each element to enforce the frictional equilibrium equation. This approach has significant difficulty when applied in settings where effective normal stress is very low. At low effective normal stress, small perturbations in shear stress can lead to large changes in sliding velocity, making accuracy challenging for explicit time stepping methods unless very small time steps are used. Possibly, implicit time stepping would perform better for settings with low normal stress, but that approach has not been tested.

As an alternative to rate and state friction, static/dynamic friction, was used to compute the results shown in this book. With static/dynamic friction, if the shear stress of an element exceeds the Coulomb failure criterion, Equation 2.8, then the element is considered to be sliding and its coefficient of friction is instantaneously lowered to a new dynamic value, μ_d. The sudden lowering of friction causes rapid sliding and can result in cascading sliding and weakening of friction on neighboring elements. The result is a process that mimics earthquake nucleation and propagation. Once the sliding velocity of a sliding element has gone below a certain threshold, friction is instantaneously restrengthened to its original static value.

When static/dynamic friction is used, checks are performed for frictional weakening and element status for all elements after every iteration in the shear stress subloop (see Sect. 2.3.7). Checks for friction restrengthening are performed at the end of each time step.

Static/dynamic friction has certain drawbacks, but it is a reasonable compromise between realism and efficiency. Static/dynamic friction is tested in Sect. 3.3.1 and

discussed in Sect. 4.2.5. In McClure and Horne (2011) of Chap. 1, results from rate and state friction modeling were compared to static/dynamic friction modeling from McClure and Horne (2010b) of Chap. 1, and the results were qualitatively similar.

In McClure and Horne (2010b) of Chap. 1, static/dynamic friction was used, but a radiation damping term was not used, and so all slip during a seismic event was effectively instantaneous and simultaneous. That was a rather problematic approach, and in the present model, the inclusion of a radiation damping term prevents slip from occurring instantaneously (though weakening and restrengthening of friction is instantaneous).

Because static/dynamic friction allows a seismic event to nucleate at a single element (and friction weakens instantaneously), there is an inherent discretization-dependence to the results. Earthquake models with this property have been referred to as "inherently discrete" (Ben-Zion and Rice 1993). In contrast, rate and state simulations that use an adequate spatial discretization require nucleation to occur on a cooperating patch of several elements, and these models are convergent to grid refinement.

A major assumption of the model in this book, quasistatic equilibrium, breaks down at very high slipping velocity, where dynamic stress transfer effects can play an important role. The radiation damping term is used to approximate the effects of dynamic stress transfer, but it is not necessarily accurate for calculations involving more than a single, planar fault (discussed by LaPusta 2001). Full dynamic simulations could be used to solve the problems more accurately, but these are extremely computationally intensive and would not be feasible for large, complex fracture networks.

2.5.4 Alternative Method for Modeling Friction

We have developed an alternative method for earthquake modeling that is intended to replicate results from rate and state friction but with much better efficiency. The method has not been extensively tested, and because it remains a work in progress, results using the method are not given in this book. However, the method is summarized in this section because it is a potential topic for future work.

An ideal method for modeling seismic events should be convergent to grid refinement, give a reasonably accurate answer with a coarse refinement, and give results similar to rate and state friction simulation. There are two particular problems that need to be overcome to make efficient earthquake simulation possible. Problem (1): crack tip discretization error (with a coarse discretization) causes difficulty in modeling rupture propagation. Problem (2): groups of elements must be able to collectively interact to nucleate a rupture.

With crack tip discretization error (Problem 1), the issue is that the stress concentration at the crack tip is underestimated with a coarse discretization. With rate and state friction, friction weakening ahead of a rupture tip occurs because a concentration of stress causes displacement to occur ahead of the tip while the

coefficient of friction is still high. If the concentration of stress ahead of the rupture tip is too weak because of inadequate discretization (the initiating rupture patch is discretized by one or a few elements), friction weakening may be artificially limited, and a seismic event may "fizzle out" shortly after nucleation.

The challenge of modeling rupture initiation (Problem 2) is that in order for results to be convergent to grid refinement, multiple elements must be able to cooperate to nucleate a single seismic event. With rate and state friction, there is a characteristic minimum patch size for nucleation of an earthquake (Segall 2010). Therefore, with sufficient discretization refinement, nucleation patches are composed of a large number of elements. Any method where individual elements are able to nucleate ruptures will not be convergent to grid refinement.

Static/dynamic friction does not handle the rupture initiation issue (Problem 2) because a single element is able to nucleate an event, regardless of element size. This happens because in static/dynamic friction the characteristic length scale of friction weakening is zero (an element does not have to displace over any distance in order to experience a drop in friction). As a consequence, the simulation results are discretization dependent.

A strategy called "RSQSim" has been proposed as an efficient method for modeling earthquakes (Dieterich 1995; Dieterich and Richards-Dinger 2010). In this strategy, a seismic event can nucleate from a single element. Semi-analytical treatments of state evolution are used to calculate the timing of earthquake nucleation. RSQSim requires that elements must be larger than the minimum nucleation patch size, and as a result it is not able to model multiple elements collectively nucleating (it does not solve Problem 2), and as with static/dynamic friction, results are discretization dependent. Dieterich (1995) suggested handling the crack tip discretization problem (Problem 1) with a special multiplying factor applied to the elements adjacent to rupture tips. However, it is not clear whether this method is robust because it is effectively a heuristic tuning parameter. Because it is not theoretically derived, it is not clear whether it can be guaranteed to work in all cases. Ideally, it should be possible to derive the crack tip adjustment theoretically without needing to tune it using trial and error. Another issue is that it is unclear how RSQSim could be coupled with fluid flow.

In the following paragraphs, our proposed alternative method for efficient earthquake simulation is described.

The default setting for an element is "stuck." Stuck elements have zero sliding velocity. Stuck elements are converted to "sliding" elements if their shear stress exceeds their frictional resistance to slip according to the Coulomb law (Eq. 2.8) with a constant, static coefficient of friction. Once converted to a sliding element, the element slides according to a velocity strengthening model, equivalent to the rate and state friction expression for friction (Eq. 2.40) with the b coefficient set to zero.

A special condition is applied to sliding elements to determine if they should begin "nucleation." When an element is nucleating, a displacement weakening law is applied such that the coefficient of friction weakens linearly as a function of displacement over a specified displacement distance until it reaches a specified minimum value. Simultaneously, the velocity strengthening term continues to be

used. This approach is equivalent to using the rate and state friction law (Eq. 2.40) with a displacement weakening f_0 term and the b coefficient equal to zero.

It can be shown that during nucleation of a rupture using rate and state friction with the aging law for state evolution (Segall 2010), friction evolution is equivalent to linear displacement weakening:

$$\frac{\partial\theta}{\partial t} = 1 - \frac{v\theta}{d_c} \sim -\frac{v\theta}{d_c} \Rightarrow \log(\theta) = \frac{-vt}{d_c} + C = \frac{-D}{d_c} + C \Rightarrow \frac{\partial\mu_f}{\partial D} = -\frac{b}{d_c}, \quad (2.41)$$

where it has been assumed that during rupture, state is decreasing rapidly, and so $|v\theta/d_c| \gg 1.0$.

The minimum value of f_0 (the value at which friction no longer weakens with displacement) can be chosen so that it is equivalent to using a constant f_0 in the rate and state expression with a non-zero b parameter and with the state variable equal to its steady state value for sliding at 1.0 m/s. The steady state value for the state variable can be calculated for a given sliding velocity by setting the time derivative of state equal to zero in the aging law:

$$\frac{\partial\theta}{\partial t} = 1 - \frac{v_{ss}\theta_{ss}}{d_c} = 0 \Rightarrow \theta_{ss} = \frac{d_c}{v_{ss}}. \quad (2.42)$$

It is important to carefully select the condition to determine when an element should begin nucleation (begin to experience displacement weakening). If an element (or a patch of sliding elements) begins to experience displacement weakening, but the patch is smaller than the characteristic minimum patch size, then sliding will relieve shear stress faster than friction weakening induces slip, and a frictional instability (rapid weakening of friction) will not occur. Frictional instability requires that the weakening of friction occurs faster than sliding relieves shear stress. This principle is the basis of our nucleation condition.

The nucleation condition we propose is:

$$\frac{\partial|\tau|}{\partial D} = -\frac{\partial\mu_f}{\partial D}\sigma_n'. \quad (2.43)$$

In Eq. 2.43, the coefficient of friction derivative term on the right-hand side is a constant value because (by definition) weakening is linear with displacement. The left-hand side of Eq. 2.43 is evaluated numerically for each element by determining the actual change in shear stress and displacement experienced by the element during the time step. The shear stress change of an element is affected by its own sliding and also by the sliding of all other elements. If a patch of elements is sliding, the term on the left-hand side of Eq. 2.43 is equivalent to the stiffness of the entire slipping patch. If a single element is sliding, it is equivalent to the stiffness of the single element. If Eq. 2.43 is satisfied, then the slipping patch has become large enough (and its equivalent stiffness low enough) that the initiation of displacement weakening will lead to frictional instability and rapid acceleration of sliding velocity. Testing with well-resolved discretizations found that this method successfully nucleated events when slipping patches reached the same patch size

as the full rate and state simulations. The method also worked well with coarse discretizations, even discretizations so coarse that a single element was larger than the minimum nucleation patch size.

A condition is needed for returning sliding elements to the "stuck" state. Recall that if an element is not "stuck," then it is sliding according to a velocity strengthening law (and may experience displacement weakening of friction if it has experience nucleation). If the sliding velocity of an element goes below a certain threshold, it is reset to "locked," and its sliding velocity returns to zero. Different thresholds are used for elements that have nucleated and elements that have not nucleated (the threshold is higher for the nucleated elements). If a nucleated element is returned to the "stuck" state, its coefficient of friction is returned to its initial value (the displacement weakening of friction is reset).

The crack tip discretization error problem is solved by applying a mechanism to enforce "nucleation" at the crack tip. Nucleation is enforced by reducing the f_0 to the minimum value it would reach due to the displacement weakening (which is a specified model parameter derived using Eq. 2.42).

An appropriate condition is needed to determine when to enforce the crack tip adjustment. If nucleation is applied too readily, then ruptures will propagate further than they should. If nucleation is applied too sparingly, ruptures may not propagate far enough. The goal is that simulated ruptures with coarse discretizations will propagate the same distance as if full rate and state simulation were performed with a well resolved discretization. The crack tip adjustment should have no effect if a large number of elements are sliding because the discretization is well resolved and no adjustment is needed. It should also be possible to derive the adjustment in advance, rather than tuning it by trial and error.

It was found that a stress intensity factor approach worked as a method for determining when to apply nucleation at the crack tip. The stress intensity factor (also discussed in Sect. 2.5.2) may be calculated for mode II deformation using the Schultz (1988) method (Eq. 2.39), with sliding displacement D replacing opening displacement E_{open}. With this method, nucleation is enforced at the crack tip if the stress intensity factor exceeds a mode II fracture toughness. It should be possible to derive the fracture toughness from rate and state parameters, but we have not yet done so. We found with trial and error that if an appropriate fracture toughness was used, then the distance of rupture propagation was insensitive to discretization refinement and consistent with a full rate and state simulation.

2.5.5 Adaptive Domain Adjustment

During some simulations, there are large regions of the spatial domain where stresses and fluid pressures change very slowly during all or some of the simulation. Depending on the specifics of the problem, fractures distant from the injector may experience virtually zero fluid flow and be in a stress state such that they are not close to either sliding or opening. Given this situation, there is an

opportunity to reduce computational effort by not updating stresses at these elements during every time step and/or by not including them in the flow simulation equations. This strategy is referred to as adaptive domain adjustment. Adaptive domain adjustment is tested in Sects. 3.3.1 and 4.2.4.

Elements that are not deforming and experience virtually zero fluid flow are identified and placed in a *nochecklist*. All other elements are placed in a *checklist*. The stresses on elements in the *nochecklis*t are not updated during every time step. Cumulative deformations between *nochecklist* updates are tracked and at intervals the *nochecklist* elements are updated from the cumulative *checklist* deformations. Because linearly elastic deformations are path-independent, updating stresses periodically results in exactly the same final stresses as if they were updated frequently.

As an additional optimization, *nochecklist* elements are removed from the system of fluid flow equations. Removing the *nochecklist* elements from the fluid flow equations causes them to have effectively zero transmissivity.

An algorithm is used to sort elements into the *nochecklist*. To aid in categorization, a separate list called an *activelist* is kept. Once an element is placed in the *activelist*, it remains there for the duration of the simulation. Elements can be added to the *activelist* for the following reasons: connection to the wellbore, sliding residual greater than a specified negative value, *slidetol* (sliding occurs when sliding residual become positive), effective normal stress less than a specified positive value, *opentol* (opening occurs when normal stress becomes negative), or a perturbation from initial fluid pressure greater than 0.1 MPa. In Simulations B5 and B9 (the two simulations in this book that use adaptive domain adjustment), *opentol* was 4.0 and *slidetol* was −2.0. The entire problem domain is gridded into a 30 × 30 grid, and blocks containing an element in the *activelist* are identified as active blocks. All elements contained in an active block or a block surrounding the active blocks are placed in the *checklist*. All other elements are placed in the *nochecklist*. Overall, the algorithm leads to a region of *checklist* elements around the injector well that spreads out gradually over time as the region of stimulation grows.

2.5.6 Strain Penalty Method

Inaccuracy in the BEM calculations can occur when elements are in close proximity relative to their size (Crouch and Starfield 1983). The best solution to this problem, described in Sect. 2.4.2, is to refine the discretization in locations where fractures are in close proximity. However, this strategy is not practical at low-angle fracture intersections, where neighboring fractures lie very close together over significant distances. Adequately discretizing these geometries requires a large number of extremely small elements, which is numerically undesirable in the fluid flow calculations. To avoid excessive discretization refinement, a minimum element size is specified in the discretization algorithm (Sect. 2.4.2). As discussed

in Sect. 2.4.1, one strategy to avoid these difficulties is to avoid creating fracture networks that contain low-angle fracture intersections.

If low-angle intersections are included in a simulation, elements can interact in unstable ways. Displacements and stresses can grow extremely large, and displacement can form in strange, unrealistic patterns (see examples in Sect. 3.4). Not only are these behaviors unrealistic, in severe cases they can cause simulations to be unusable. Some inaccuracy at low-angle fracture intersections is perhaps acceptable, but it is not acceptable for low-angle intersections to cause problems so severe that they prevent a simulation from continuing.

An algorithm, referred to as the strain penalty method, is used to minimize the effects of inaccuracies at low-angle fracture intersections. The algorithm identifies where large strains are beginning to develop and applies penalty stresses to prevent the strains from growing further. This approach could be considered a crude way to mimic rock failure, which is the process in nature that prevents extreme concentrations of stress and strain.

There is no theoretical basis for the strain penalty method. It is intended as a way to prevent catastrophic numerical error. It does not ensure that the solution is completely accurate. With discretization refinement, the region of inaccuracy can be limited to a small region very near the center of the intersection (Sect. 3.4), which minimizes error.

Perfect accuracy at intersections is probably a false goal because in reality, very complex deformations occur at fracture intersections that do not conform to the assumption of small strain, linearly elastic deformation made by the Shou and Crouch (1995) boundary element method. Therefore, numerical accuracy is overwhelmed by error from the model assumptions. This is not an issue unique to our model. Fracture intersections are very challenging for all numerical methods to describe, not just the boundary element method.

In most simulations in this book—Models A, B, and C, the strain penalty method was not used (it was not needed because low-angle intersections were not present in these networks). In Sect. 3.4, the strain penalty method was tested on Model D, which contains a very low-angle fracture intersection (Figs. 3.27, 3.28, 3.29, and 3.30).

At any location on a (zero-curvature) fracture, strain due to varying displacement discontinuity can be defined as:

$$\varepsilon_k = \frac{dD_k}{dx}, \tag{2.44}$$

where k refers either normal, n, or shear, s, displacement discontinuity and x refers to the distance along the fracture. The derivatives for both modes of deformation are calculated using finite difference approximations at the boundaries between elements. A threshold strain, $\varepsilon_{k,lim}$, is set at each element boundary, and if the absolute value of strain at any element edge exceeds the threshold strain, a penalty stress is applied. The subscript k can refer to n, strain in normal displacement or

s strain in shear displacement. After a penalty stress is applied, the limit $\varepsilon_{k,lim}$ at the interface is updated to be equal to the absolute value of the current value of ε_k.

The penalty stress is calculated according to:

$$\Delta\sigma_{k,strainadj} = (\varepsilon_k - \varepsilon_{k,\lim})\frac{G}{1-v}. \tag{2.45}$$

The penalty stress can be applied to both elements at the interface or only to one. The algorithm determines if the motion of each element in the preceding time step acted to increase or decrease the absolute value of the strain at the element interface. If only one element did, the full adjustment is applied to that element. If both did, the adjustment is divided between them proportionally based on their effect on the change in the strain.

The penalty stress is applied at the beginning of the subsequent time step. To prevent excessively large penalty strains from being applied during the subsequent time step, the adaptive time stepping equation, Equation 2.33, is applied with $\delta_{strainadj}$ defined as being equal to the largest absolute value of $\Delta\sigma_{k,strainadj}$, and $\eta_{targ,strainadj}$ defined to be a value equal to one tenth of η_{targ}. As with δ and η_{targ} (explained in Sect. 2.3.9), if the value of $\delta_{strainadj}$ exceeds $4\eta_{targ,strainadj}$, the time step is rejected and repeated with smaller value of dt.

2.5.7 Neglecting Stresses Induced by Deformation

A major purpose of the model in this book is to couple deformation with fluid flow. However for purposes of testing and comparison, it is useful to neglect the stresses induced by deformation (Sects. 3.3.1 and 4.2.6). To neglect stresses induced by deformation, all nondiagonal interaction coefficients in the boundary element matrices (Eqs. 2.19, 2.20, 2.21, and 2.22) are set equal to zero. In this case, when an element opens or slides, it affects its own stress, but not the stress at surrounding elements. This assumption has often been made in modeling of shear stimulation, apparently because it simplifies the design of the model significantly (Bruel 1995, 2007; Sausse et al. 2008; Dershowitz et al. 2010).

Neglecting the off-diagonal interaction coefficients causes element stiffness to be discretization dependent because in the Shou and Crouch (1995) method, self-interaction coefficients are a function of element size. To avoid discretization dependence when neglecting stress interaction, self-interaction coefficients are defined so that they are independent of element size. By treating a fracture as a single, constant displacement boundary element, the stiffness of a fracture, K_{frac}, can be calculated as (Crouch and Starfield 1983):

$$K_{frac} = \frac{1}{\pi(1-\upsilon)}\frac{1}{a_{frac}}, \tag{2.46}$$

where a_{frac} is the half-length of the fracture. The self-interaction coefficients of each element for effect of opening deformation on normal stress and shear deformation on shear stress are set to the fracture stiffness given by Eq. 2.46. No adjustment is needed for the self-interaction coefficients that relate opening deformation to shear stress and shear deformation to normal stress because they are always zero.

References

Amestoy, P.R., Davis, T.A., Duff, I.S.: An approximate minimum degree ordering algorithm. SIAM J. Matrix Anal. Appl. **17**(4), 886–905 (1996). doi:10.1137/S0895479894278952

Amestoy, P.R., Davis, T.A., Duff, I.S.: Algorithm 837: AMD, an approximate minimum degree ordering algorithm. ACM Trans. Math. Softw. **30**(3), 381–388 (2004). doi:10.1145/1024074.1024081

Anderson, E., Bai, Z., Bischof, C., Blackford, S., Demmel, J., Dongarra, J., Du Croz, J., Greenbaum, A., Hammarling, S., McKenney, A., Sorensen, D.: LAPACK Users' Guide, 3rd edn. Society for Industrial and Applied Mathematics, Philadelphia (1999)

Aziz, K., Settari, A.: Petroleum Reservoir Simulation. Applied Science Publishers, London (1979)

Barton, N., Bandis, S., Bakhtar, K.: Strength, deformation and conductivity coupling of rock joints. Int. J. Rock Mech. Min. Sci. Geomech. Abstr. **22**(3), 121–140 (1985). doi:10.1016/0148-9062(85)93227-9

Ben-Zion, Y., Rice, J.R.: Earthquake failure sequences along a cellular fault zone in a three-dimensional elastic solid containing asperity and nonasperity regions, J. Geophys. Res. **98**(B8), 14109–14131 (1993), doi:10.1029/93JB01096

Bradley, A.M.: H-matrix and block error tolerances, *arXiv:1110.2807v2*, source code available at http://www.stanford.edu/∼ambrad, paper available at http://arvix.org/abs/1110.2807 (2012)

Bruel, D.: Heat extraction modelling from forced fluid flow through stimulated fractured rock masses: application to the Rosemanowes Hot Dry Rock reservoir. Geothermics **24**(3), 361–374 (1995). doi:10.1016/0375-6505(95)00014-H

Bruel, D.: Using the migration of the induced seismicity as a constraint for fractured Hot Dry Rock reservoir modelling. Int. J. Rock Mech. Min. Sci. **44**(8), 1106–1117 (2007). doi:10.1016/j.ijrmms.2007.07.001

Crouch, S.L., Starfield, A.M.: Boundary Element Methods in Solid Mechanics: with Applications in Rock Mechanics and Geological Engineering. Allen & Unwin, London, Boston (1983)

Davis, T.A.: A column pre-ordering strategy for the unsymmetric-pattern multifrontal method. ACM Trans. Math. Softw. **30**(2), 165–195 (2004a). doi:10.1145/992200.992205

Davis, T.A.: Algorithm 832: UMFPACK, an unsymmetric-pattern multifrontal method. ACM Trans. Math. Softw. **30**(2), 196–199 (2004b). doi:10.1145/992200.992206

Davis, T.A.: Direct Methods For Sparse Linear Systems. SIAM, Philadelphia (2006)

Davis, T.A., Duff, I.S.: An unsymmetric-pattern multifrontal method for sparse LU factorization. SIAM J. Matrix Anal. Appl. **18**(1), 140–158 (1997). doi:10.1137/S0895479894246905

Davis, T.A., Duff, I.S.: A combined unifrontal/multifrontal method for unsymmetric sparse matrices. ACM Trans. Math. Softw. **25**(1), 1–19 (1999). doi:10.1145/305658.287640

Dershowitz, W.S., Cottrell, M.G., Lim, D.H., Doe, T.W.: A discrete fracture network approach for evaluation of hydraulic fracture stimulation of naturally fractured reservoirs, ARMA 10-475. Paper presented at the 44th U.S. Rock Mechanics Symposium and 5th U.S.-Canada Rock Mechanics Symposium, Salt Lake City, Utah (2010)

Dieterich, J.H.: Earthquake simulations with time-dependent nucleation and long-range interactions. Nonlinear Process. Geophys. **2**, 109–120 (1995)

Dieterich, J.H., Richards-Dinger, K.B.: Earthquake recurrence in simulated fault systems. In: Savage, M.K., Rhoades, D.A., Smith, E.G.C., Gerstenberger, M.C., Vere-Jones, D. (eds.) Seismogenesis and earthquake forecasting: the frank evison volume II, pp. 233–250, Springer Basel (2010)

Dieterich, J.H.: 4.04—Applications of rate- and state-dependent friction to models of fault slip and earthquake occurrence. In: Gerald, S. (ed.) Treatise on Geophysics pp. 107–129, Elsevier, Amsterdam (2007)

Dongarra, J.J., Croz, J.D., Hammarling, S., Hanson, R.J.: An extended set of FORTRAN Basic Linear Algebra Subprograms. ACM Trans. Math. Softw. **14**(1), 1–17 (1988a). doi:10.1145/42288.42291

Dongarra, J.J., Croz, J.D., Hammarling, S., Hanson, R.J.: Algorithm 656: an extended set of basic linear algebra subprograms: model implementation and test programs. ACM Trans. Math. Softw. **14**(1), 18–32 (1988b). doi:10.1145/42288.42292

Dongarra, J.J., Du Croz, J., Hammarling, S., Duff, I.S.: A set of level 3 basic linear algebra subprograms. ACM Trans. Math. Softw. **16**(1), 1–17 (1990a). doi:10.1145/77626.79170

Dongarra, J.J., Du Croz, J., Hammarling, S., Duff, I.S.: Algorithm 679: a set of level 3 basic linear algebra subprograms: model implementation and test programs. ACM Trans. Math. Softw. **16**(1), 18–28 (1990b). doi:10.1145/77626.77627

Faulkner, D.R., Jackson, C.A.L., Lunn, R.J., Schlische, R.W., Shipton, Z.K., Wibberley, C.A.J., Withjack, M.O.: A review of recent developments concerning the structure, mechanics and fluid flow properties of fault zones. J. Struct. Geol. **32**(11), 1557–1575 (2010). doi:10.1016/j.jsg.2010.06.009

Fredd, C.N., McConnell, S.B., Boney, C.L., England, K.W.: Experimental study of fracture conductivity for water-fracturing and conventional fracturing applications. SPE J. **6**(3), 288–298 (2001). doi:10.2118/74138-PA

Grabowski, J.W., Vinsome, P.K., Lin, R., Behie, G.A., Rubin, B.: A fully implicit general purpose finite-difference thermal model for in situ combustion and steam, SPE 8396. Paper presented at the SPE Annual Technical Conference and Exhibition, Las Vegas, Nevada (1979). doi:10.2118/8396-MS

Holmgren, M.: XSteam: water and steam properties according to IAPWS IF-97. <www.x-eng.com> (2007)

Jaeger, J.C., Cook, N.G.W., Zimmerman, R.W.: Fundamentals of Rock Mechanics, 4th edn. Blackwell Pub, Malden (2007)

Karimi-Fard, M., Durlofsky, L.J., Aziz, K.: An efficient discrete-fracture model applicable for general-purpose reservoir simulators. SPE J. **9**(2), 227–236 (2004). doi:10.2118/88812-PA

Kim, J., Tchelepi, H., Juanes, R.: Stability, accuracy, and efficiency of sequential methods for coupled flow and geomechanics. SPE J. **16**(2), 249–262 (2011). doi:10.2118/119084-PA

Kohl, T., Mégel, T.: Predictive modeling of reservoir response to hydraulic stimulations at the European EGS site Soultz-sous-Forêts. Int. J. Rock Mech. Min. Sci. **44**(8), 1118–1131 (2007). doi:10.1016/j.ijrmms.2007.07.022

Lapusta, N. (2001), Elastodynamic analysis of sliding with rate and state friction, PhD thesis, Harvard University

Lawson, C.L., Hanson, R.J., Kincaid, D., Krogh, F.T.: Basic Linear Algebra Subprograms for FORTRAN usage. ACM Trans. Math. Softw. **5**(3), 308–323 (1979). doi:10.1145/355841.355847

Liu, E.: Effects of fracture aperture and roughness on hydraulic and mechanical properties of rocks: implication of seismic characterization of fractured reservoirs. J. Geophys. Eng. **2**(1), 38–47 (2005). doi:10.1088/1742-2132/2/1/006

McClure, M.W.: Modeling and Characterization of Hydraulic Stimulation and Induced Seismicity in Geothermal and Shale Gas Reservoirs. Stanford University, Stanford, California (2012)

McGarr, A., Simpson, D., Seeber, L.: 40. Case histories of induced and triggered seismicity. In: Lee, W.H.K., Kanamori, H. (eds.) International Geophysics, pp. 647–661, Academic Press (2002)

Morris, J.P., Blair, S.C.: Efficient Displacement Discontinuity Method using fast multipole techniques. Paper presented at the 4th North American Rock Mechanics Symposium, Seattle, WA (2000), <http://www.osti.gov/energycitations/servlets/purl/791449-ZAgLs5/native/>

Olson, J.E.: Predicting fracture swarms—the influence of subcritical crack growth and the crack-tip process zone on joint spacing in rock. Geol. Soc., London, Spec. Publ. **231**(1), 73–88 (2004). doi:10.1144/GSL.SP.2004.231.01.05

Rahman, M.K., Hossain, M.M., Rahman, S.S.: A shear-dilation-based model for evaluation of hydraulically stimulated naturally fractured reservoirs. Int. J. Numer. Anal. Meth. Geomech. **26**(5), 469–497 (2002). doi:10.1002/nag.208

Rice, J.R.: Spatio-temporal complexity of slip on a fault. J. Geophys. Res. **98**(B6), 9885–9907 (1993). doi:10.1029/93JB00191

Rjasanow, S., Steinbach, O.: The Fast Solution of Boundary Integral Equations, 1st edn. Springer, New York (2007)

Sausse, J., Dezayes, C., Genter, A., Bisset, A.: Characterization of fracture connectivity and fluid flow pathways derived from geological interpretation and 3D modelling of the deep seated EGS reservoir of Soultz (France). Paper presented at the Thirty-Third Workshop on Geothermal Reservoir Engineering, Stanford University (2008), <https://pangea.stanford.edu/ERE/db/IGAstandard/record_detail.php?id=5270>

Schultz, R.A.: Stress intensity factors for curved cracks obtained with the Displacement Discontinuity Method. Int. J. Fract. **37**(2), R31–R34 (1988). doi:10.1007/BF00041718

Segall, P.: Earthquake and Volcano Deformation. Princeton University Press, Princeton (2010)

Shou, K.J., Crouch, S.L.: A higher order Displacement Discontinuity Method for analysis of crack problems. Int. J. Rock Mech. Min. Sci. Geomech. Abstr. **32**(1), 49–55 (1995). doi:10.1016/0148-9062(94)00016-V

Teufel, L.W., Clark, J.A.: Hydraulic fracture propagation in layered rock: experimental studies of fracture containment. SPE J. **24**(1), 19–32 (1984). doi:10.2118/9878-PA

Warpinski, N.R., Schmidt, R.A., Northrop, D.A.: In-situ stresses: the predominant influence on hydraulic fracture containment. J. Petrol. Technol. **34**(3), 653–664 (1982). doi:10.2118/8932-PA

Wibberley, C.A.J., Yielding, G., Toro, G.D.: Recent advances in the understanding of fault zone internal structure: a review. Geol. Soc., London, Spec. Publ. **299**(1), 5–33 (2008). doi:10.1144/SP299.2

Willis-Richards, J., Watanabe, K., Takahashi, H.: Progress toward a stochastic rock mechanics model of engineered geothermal systems. J. Geophys. Res. **101**(B8), 17481–17496 (1996). doi:10.1029/96JB00882

http://www.springer.com/978-3-319-00382-5

Discrete Fracture Network Modeling of Hydraulic Stimulation
Coupling Flow and Geomechanics
McClure, M.; Horne, R.N.
2013, X, 90 p. 42 illus., 41 illus. in color., Softcover
ISBN: 978-3-319-00382-5