TEST11: 2D validation of crack opening displacement: Pulling and sliding experiment to compute crack surface displacement for crack that divides a material into two equal halves.

TEST12: 2D crack initiation and propagation under application of monotonically increasing tension and shear forces: Computes v-field and displacement for material subjected to tension and shear forces to simulate crack initiation and propagation.

TEST13: 2D crack initiation and propagation under application of monotonically increasing tension and shear forces in the presence of weak (seed) points and pre-existing cracks (using VFRectangularCrackCreate): Simulates crack propagation by tension, shear and a combination of both. Tension force is applied by displacement boundary condition normal to the pulled boundary while shear force is implemented by specify displacement component tangential to the pulled surface. Specifying both displacements normal and perpendicular to a surface implements the combination of tension and shear forces.

TEST14: 3D case: Crack initiation test. Computation of pressure required to initiate propagation of fracture of any given length.

TEST15: 2D (hydraulic fracturing) static line crack case: Computes v-field and displacement for pressure loaded line crack of any given length.

TEST16: 2D (hydraulic fracturing) propagating line crack case: Computes v-field and displacement for volume driven propagation of line crack of any given length.

TEST17: 2D (hydraulic fracturing) multiple propagating line crack case: Computes v-field and displacement for volume driven propagation of arbitrary number of line crack of any length using VFRectangularCrackCreate.

TEST18: 3D (hydraulic fracturing) static penny-shaped crack case: Computes v-field and displacement for pressure loaded penny-shaped crack of any given length.

crack.

TEST19: 3D (hydraulic fracturing) propagating penny-shaped crack case: Computes v-field and displacement for volume driven propagation of penny-shaped crack of any given radius.

TEST20: 3D (hydraulic fracturing) propagating penny-shaped crack case: Computes v-field and displacement for volume driven propagation of penny-shaped crack of any given radius.

TEST21: 3D (hydraulic fracturing) propagation of multiple penny-shaped cracks: Computes v-field and displacement for volume driven propagation of multiple penny-shaped cracks of arbitrary radii, using VFPennyCrackCreate.

TEST22: 3D (hydraulic fracturing) propagation of multiple penny-shaped cracks: Computes v-field and displacement for volume driven propagation of multiple penny-shaped cracks of arbitrary radii, using VFPennyCrackCreate.

TEST23: 3D penny-shaped surface area: Computes v-field to calculate penny shape crack surface area.

TEST24: 3D KSP MixedFEMFlowSolver Validation: Solution of velocity and pressure for flow problem with exact pressure, p = sin(2Πx)sin(2Πy)sin(2Πz).Normal veolocity condition is implemented on all boundaries.

TEST25: 3D KSP MixedFEMFlowSolver Validation: Solution of velocity and pressure for flow problem with exact pressure, p = sin(2Πx)sin(2Πy)sin(2Πz).Pressure condition is implemented on all boundaries.

TEST26: 3D KSP MixedFEMFlowSolver Validation: Solution of velocity and pressure for flow problem with exact pressure, p = cos(Πx)cos(Πy)cos(Πz)/(3Π2). Normal velocity boundary condition is implemented on all boundaries.

TEST27: 3D KSP MixedFEMFlowSolver Validation: Solution of velocity and pressure for flow problem with exact pressure, p = cos(Πx)cos(Πy)cos(Πz)/(3Π2). Pressure condition is implemented on all boundaries.

TEST28: 3D SNES MixedFEMFlowSolver Validation: Solution of velocity and pressure for flow problem with exact pressure, p = sin(2Πx)sin(2Πy)sin(2Πz).Normal velocity boundary condition is implemented on all boundaries.

TEST29: 3D SNES MixedFEMFlowSolver Validation: Solution of velocity and pressure for flow problem with exact pressure, p = sin(2Πx)sin(2Πy)sin(2Πz).Pressure boundary condition is implemented on all boundaries.

TEST30: 3D SNES MixedFEMFlowSolver Validation: Solution of velocity and pressure for flow problem with exact pressure, p = cos(Πx)cos(Πy)cos(Πz)/(3Π2). Normal velocity boundary condition is implemented on all boundaries.

TEST31: 3D SNES MixedFEMFlowSolver Validation: Solution of velocity and pressure for flow problem with exact pressure, p = cos(Πx)cos(Πy)cos(Πz)/(3Π2). Pressure condition is implemented on all boundaries.

TEST32: 3D TS MixedFEMFlowSolver Validation: Solution of velocity and pressure for flow problem with exact pressure, p = sin(2Πx)sin(2Πy)sin(2Πz).Normal velocity boundary condition is implemented on all boundaries.

TEST33: 3D TS MixedFEMFlowSolver Validation: Solution of velocity and pressure for flow problem with exact pressure, p = sin(2Πx)sin(2Πy)sin(2Πz).Pressure boundary condition is implemented on all boundaries.

TEST34: 3D TS MixedFEMFlowSolver Validation: Solution of velocity and pressure for flow problem with exact pressure, p = cos(Πx)cos(Πy)cos(Πz)/(3Π2). Normal velocity boundary condition is implemented on all boundaries.

TEST35: 3D TS MixedFEMFlowSolver Validation: Solution of velocity and pressure for flow problem with exact pressure, p = cos(Πx)cos(Πy)cos(Πz)/(3Π2). Pressure condition is implemented on all boundaries.

TEST36: 1D TS. Flow problem with source term = 1 and Homogeneous pressure boundary conditions on all sides. Analytical solution is p = x(x-1)/2.

TEST37: 1D TS. Test for transient solver <http://tutorial.math.lamar.edu/Classes/DE/SolvingHeatEquation.aspx>. Zero source term and homogeneous pressure boundary condition on all sides. Initial solution is p (x,0)= 6sin(pi\*x/Lx).

TEST38: 2D TS Coupled flow and fracture test.

TEST39: 2D TS Coupled flow and fracture test using VFRectangularCrackCreate to create arbitrary fracture.

TEST40: 2D SNES. Single well problem/Poisson problem with dirac source term at (xo,yo). Analytical solution is given by p(x,y)=-1/(2\*pi)\*ln(r). r=|x-xo|. Implemented with pressure boundary conditions with values from analytical solution.

TEST41: 3D SNES. Single well problem/Poisson problem with dirac source term at (xo,yo,zo). Analytical solution is given by p(x,y,z)=-1/(4\*pi\*r). r=|x-xo|. Implemented with pressure boundary conditions with values from analytical solution.