

```

1 # Mesh Generation:
2 rmin, rmax, ncells, nlayers = 1.22, 2.22, 256, 64
3 meshld = CircleManifoldMesh(ncells, radius=rmin, degree=2)
4 mesh = ExtrudedMesh(meshld, layers=nlayers, extrusion_type="radial")
5
6 -----
7 # Constants, unit vector, initial condition
8 Ra = Constant(1e5)
9 r = sqrt(X[0]**2 + X[1]**2)
10 k = as_vector((X[0], X[1])) / r
11 Told.interpolate(rmax-r + 0.02*cos(4.*atan_2(X[1],X[0]))*sin((r-rmin)*pi))
12
13 -----
14 # UFL for Stokes equations incorporating Nitsche:
15 C_ip = Constant(100.0) # Fudge factor for interior penalty term used in weak imposition of BCs
16 p_ip = 2 # Maximum polynomial degree of the _gradient_ of velocity
17
18 # Stokes equations in UFL form:
19 stress = 2 * mu * sym(grad(u))
20 F_stokes = inner(grad(v), stress) * dx - div(v) * p * dx + dot(n, v) * p * ds_tb - (dot(v, k) * Ra * Ttheta) * dx
21 F_stokes += -w * div(u) * dx + w * dot(n, u) * ds_tb # Continuity equation
22
23 # nitsche free-slip BCs
24 F_stokes += -dot(v, n) * dot(dot(n, stress), n) * ds_tb
25 F_stokes += -dot(u, n) * dot(dot(n, 2 * mu * sym(grad(v))), n) * ds_tb
26 F_stokes += C_ip * mu * (p_ip + 1)**2 * FacetArea(mesh) / CellVolume(mesh) * dot(u, n) * dot(v, n) * ds_tb
27
28 -----
29 # Nullspaces and near-nullspaces:
30 x_rotV = Function(V).interpolate(as_vector((-X[1], X[0])))
31 V_nullspace = VectorSpaceBasis([x_rotV])
32 V_nullspace.orthonormalize()
33 p_nullspace = VectorSpaceBasis(constant=True) # Constant nullspace for pressure n
34 Z_nullspace = MixedVectorSpaceBasis(Z, [V_nullspace, p_nullspace]) # Setting mixed nullspace
35
36 # Generating near_nullspaces for GAMG:
37 nns_x = Function(V).interpolate(Constant([1., 0.]))
38 nns_y = Function(V).interpolate(Constant([0., 1.]))
39 V_near_nullspace = VectorSpaceBasis([nns_x, nns_y, x_rotV])
40 V_near_nullspace.orthonormalize()
41 Z_near_nullspace = MixedVectorSpaceBasis(Z, [V_near_nullspace, Z.sub(1)])
42
43 -----
44 # Updated solve calls:
45 stokes_problem = NonlinearVariationalProblem(F_stokes, z) # velocity BC's handled through Nitsche
46 stokes_solver = NonlinearVariationalSolver(stokes_problem, solver_parameters=stokes_solver_parameters, appctx={"
mu": mu}, nullspace=Z_nullspace, transpose_nullspace=Z_nullspace, near_nullspace=Z_near_nullspace)

```