Karl_Banana 1 August 13, 2019, 10:32am

Hi, I am learning about fenics and I started with examples with TrialFunctions and TestFunctions. But after a few examples, I found most example doesn't bother to define a trial function.

Is it the same using function instead of trialfunction?

Regards
Juntao

2 Likes

klunkean 2 August 13, 2019, 12:20pm

TrialFunctions are needed if you want to define a linear variational problem directly as a(u,v)=L(v) and call `solve(a==L,...)`. Here, u is a TrialFunction and v a TestFunction.

In the case you define the variational problem as F(u,v)=0 and call `solve(F==0, ...)` the problem is treated as nonlinear and u is a Function and v is a TestFunction.

The reason for these different definitions, however, is unclear to me.

Consider the following Poisson example where I demonstrate both ways of defining the same problem:

```
mesh = UnitIntervalMesh(8)
V = FunctionSpace(mesh, "Lagrange",1)

# example RHS
f = Expression("1.-pow(x[0]-.5,2)", element = V.ufl_element())

### Define as a(u,v) = L(v)
u = TrialFunction(V)
v = TestFunction(V)
# define bilinear and linear forms directly
a = inner(grad(u),grad(v))*dx
L = f*v*dx
# define function for storing solution
sol = Function(V)
solve(a==L, sol, bcs = DirichletBC(V, Constant(0.),lambda x,on_boundar

### Define as form F(u,v) = 0
u = Function(V)
```

```
v = TestFunction(V)
F = inner(grad(u), grad(v))*dx - f*v*dx
solve(F==0, u, bcs = DirichletBC(V, Constant(0.),lambda x,on_boundary:
```

1 Like

Weak formulation for square of gradient

Karl_Banana 3 August 13, 2019, 3:44pm

Hi, klunkean,

Thank you so much. I would never be able to figure this out myself. Though I'm still confused about the subtle difference. Does F==0 implies a non-linear solving, thus a function is used instead of trial function? I don't know. But still, thanks a lot.

---

klunkean 4 August 13, 2019, 8:24pm

Yes, if you call `solve(F==0, ...)` a NonlinearVariationalProblem is created in the background and solved by Newton iteration. If the problem F(u,v)=0 is linear, however, the solver will converge in one step.

1 Like