Define and update preconditioners

November 1, 2022

1 Define and update preconditioners

A preconditioner is defined for a bilinear-form, and aims at providing a cheap, approximative inverse of the matrix. The matrix is restricted to the non-dirichlet (free) degrees of freedom, provided by the underlying FESpace.

The canonical way is to define the preconditioner after the bilinear-form, but before calling Assemble:

[ ]:
```
a = BilinearForm(fes)
a += SymbolicBFI(grad(u)*grad(v))
c = Preconditioner(a, "local")
a.Assemble()
```

The preconditioner registers itself with the bilinear-form. Whenever the form is updated, the preconditioner is updated as well.

You can define the preconditioner after assembling, but then you have to call manually c.Update()

The ratio if this ordering is that some preconditioners (e.g. bddc, amg, ...) require access to the element-matrices, which are only available during assembling.

The preconditioners included in NGSolve are the following. Additional user-defined preconditioners can be implemented in plug-ins. An example is given in MyLittleNGSolve

Name	Preconditioners
local	Jacobi / block-Jacobi
direct	a sparse direct factorization
multigrid	h-version and high-order/low-order multigrid
bddc	p-version domain decomposition