
calcifer

Team COOP

Feb 22, 2022

Contents

1	Thermal diffusion, solver by calcifer.	3
2	calcifer_pde package	5
2.1	Submodules	5
2.2	calcifer_pde.boundary module	5
2.3	calcifer_pde.calcifer module	5
2.4	calcifer_pde.diff module	5
2.5	calcifer_pde.domain module	5
2.6	calcifer_pde.geometry module	5
3	Indices and tables	7

Under construction

Calcifer is a Poisson-like PDE finite difference solver on 2-D structured I.J Grids. Grids are curvilinear.

- create a **Geometry** object.
- define the computational **Domain**, based upon this **Geometry**.
- Solve the problem. Heat conduction is solver in **HeatSolve** from Calcifer.

CHAPTER 1

Thermal diffusion, solver by calcifer.

A thermal diffusion test with calcifer is done like this:

```
from calcifer_pde.domain import Domain
from calcifer_pde.geometry import Square

def heat_solve(dom, k_coeff=1.0):
    init_field = np.random.random_sample(dom.shp1d)
    sterm_l = np.zeros_like(dom.lapl)
    sterm_r = np.zeros(dom.shp1d)
    # Left Hand Side
    lhs_csr = dom.lapl * k_coeff + sterm_l
    # Right-Hand Side
    rhs_csr = np.zeros(dom.shp1d) + sterm_r
    lhs_csr_bc, rhs_csr_bc, grad_n_bc = apply_bc(dom, lhs_csr, rhs_csr)

    out_ld, info = scp.linalg.bicgstab(lhs_csr_bc, rhs_csr_bc, x0=init_field)
    if info == 0:
        print(".    ^^ Resolution succesfull.")
    elif info > 0:
        print(".    t(-_t) Resolution failed.")
    else:
        print(".    == Convergence not reached.")
    temp = out_ld.reshape(dom.shp2d)
    return temp

geo = Square(nx=80, ny=100, len_x=1.0, len_y=1.0)
dom = Domain(geo)
dom.switch_bc_vmax_neuman(0.0)
dom.switch_bc_vmin_neuman(0.0)
dom.switch_bc_umin_dirichlet(200.0)
dom.switch_bc_umax_dirichlet(100.0)
sol = heat_solve(dom, k_coeff=22.0)
```

One can solve an other PDE by creating a different problem than `heat_solve`

2.1 Submodules

2.2 calcifer_pde.boundary module

2.3 calcifer_pde.calcifer module

2.4 calcifer_pde.diff module

2.5 calcifer_pde.domain module

2.6 calcifer_pde.geometry module

CHAPTER 3

Indices and tables

- `genindex`
- `modindex`
- `search`