Firedrake

pyop3: A new domain-specific language for automating high-performance mesh-based simulation codes

Connor Ward

January 2023

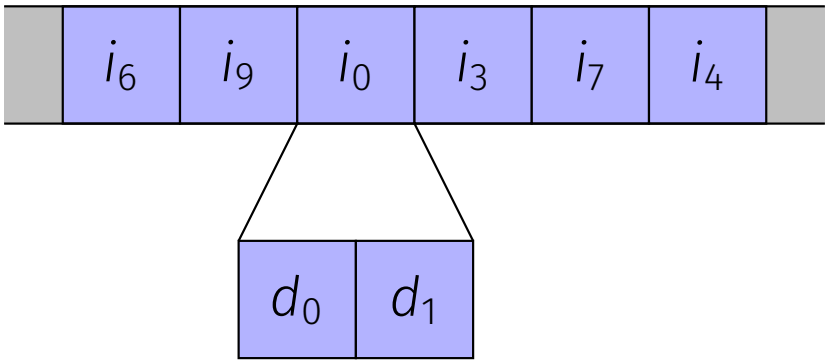
Imperial College London

PyOP2

- Used for stencil computations
- Handles all the mesh data structures
- Firedrake applications include residual assembly and interpolation

PyOP2 data model

- Vector data is stored by PyOP2 `Dats`
- These associate a fixed inner shape (d_m) with a set of possibly unordered nodes (i_n)
- `Mixed Dats` and `Dats` for extruded meshes are also possible



Introducing pyop3

- Domain-specific language embedded in Python for automating stencil computations
- Uses code generation to produce fast code

Stencil library wishlist

Features:

- ☐ Orientation (e.g. unstructured hexes)
- ☐ p-adaptivity
- ☐ Mixed meshes

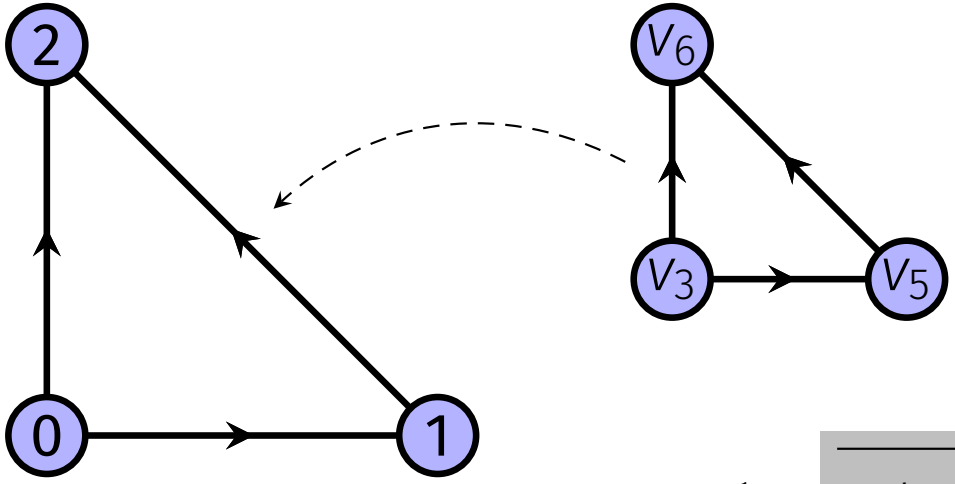
Performance:

- ☐ Mesh partial structure (e.g. extruded)[1]
- ☐ Mesh numbering (e.g. DoFs up extruded columns)[1]

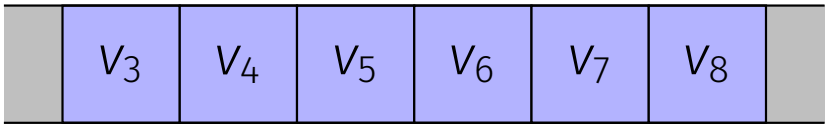
[1] Achievable in PyOP2 but very complicated and hard to extend

Claim: pyop3's new data layout abstraction enables all of these.

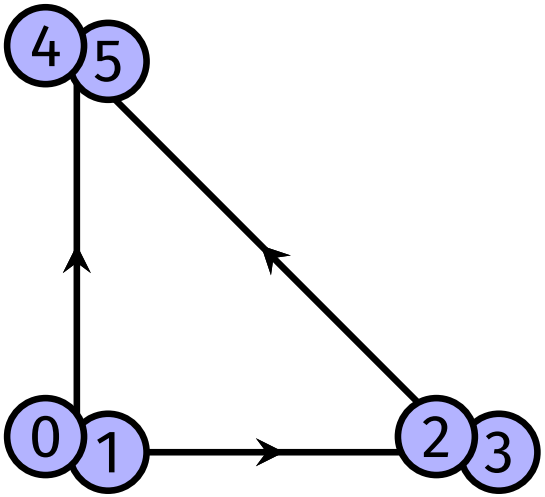
Starting simple: P1

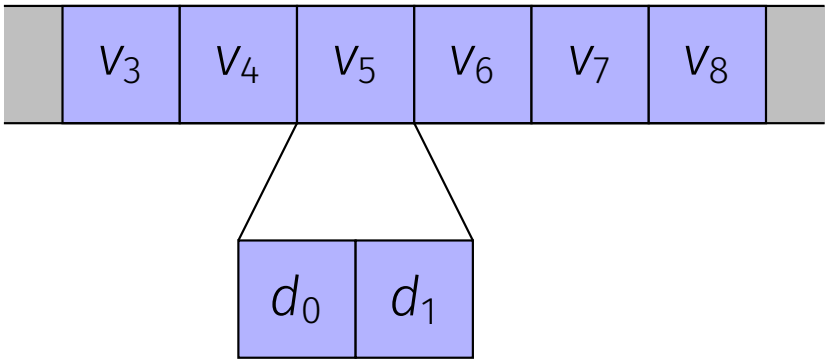



```
1  root = (
2    MultiAxis()
3    .add_part(AxisPart(nverts))
4  )
```



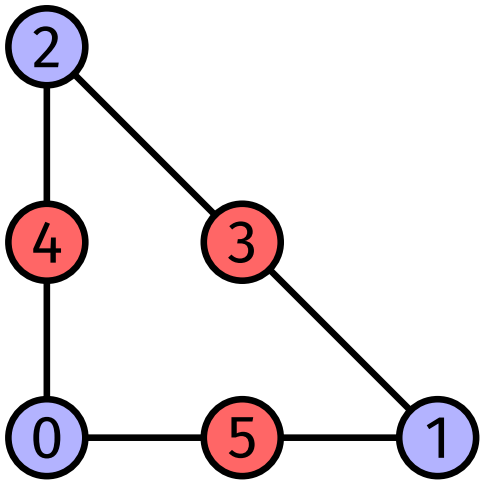
Adding shape: vector P1

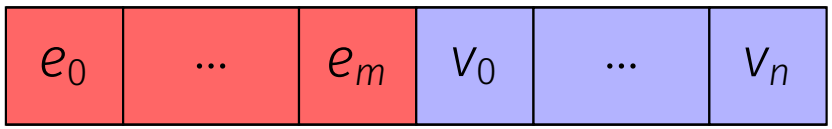





```
root = (
  MultiAxis()
  .add_part(AxisPart(nverts))
  .add_subaxis(AxisPart(2))
)
```

Multiple entities: P2

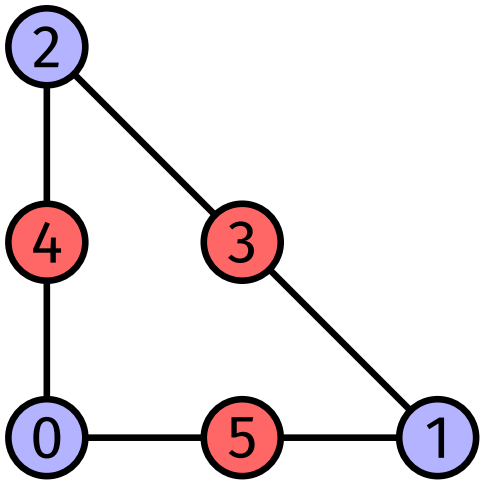


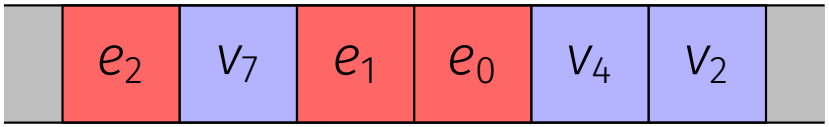


- ☑ p-adaptivity
- ☑ Mixed meshes

```
1  root = (
2    MultiAxis()
3    .add_part(AxisPart(nedges))
4    .add_part(AxisPart(nverts))
5  )
```

Now with renumbering

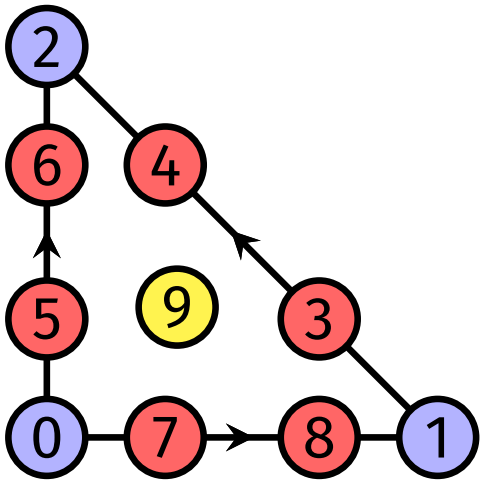


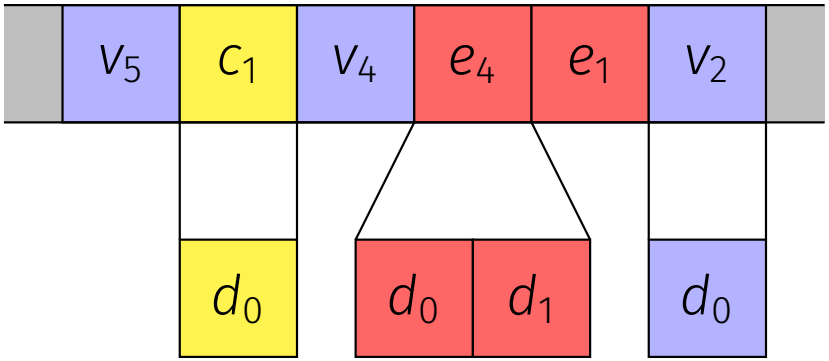


☑ Mesh numbering

```
1  root = (
2    MultiAxis()
3    .add_part(AxisPart(
4      nedges,
5      numbering=[4,2,5,...],
6    ))
7    .add_part(AxisPart(
8      nverts,
9      numbering=[3,0,1,...],
10   ))
11 )
```

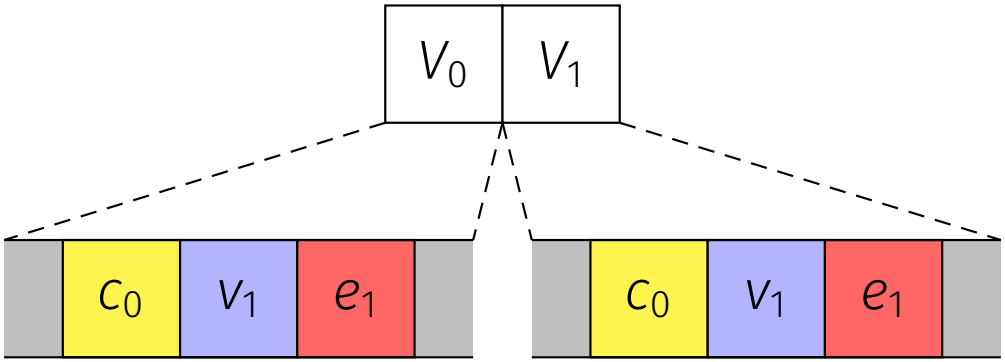
More complicated inner shape: P3






```
1  root = (
2    MultiAxis()
3    .add_part(AxisPart(ncells, "cells"))
4    .add_part(AxisPart(nedges, "edges"))
5    .add_part(AxisPart(nverts, "verts"))
6    .add_subaxis("edges", AxisPart(2))
7  )
```

Mixed




```
root = (
  MultiAxis()
   .add_part(AxisPart(1, "V0"))
   .add_part(AxisPart(1, "V1"))
   .add_subaxis("V0", ...)
   .add_subaxis("V1", ...)
)
```

Ragged

Stencil library wishlist

Features:

☐ Orientation
☑ p-adaptivity
☑ Mixed meshes

Performance:

☐ Mesh partial structure
☑ Mesh numbering

Orientation: P3

Orientation: Raviart-Thomas

Partially-structured meshes: extruded

Stencil library wishlist

Features:

☑ Orientation
☑ p-adaptivity
☑ Mixed meshes

Performance:

☑ Mesh partial structure
☑ Mesh numbering

Final bits

Things I missed

Summary

Appendix

Partially-structured meshes: refined

Axis swapping