Progetto PCS 2024

Simone Domenico Morandi, Elisabetta Pautasso

Parte 1 - Definizione delle strutture

Struttura per i dati delle fratture.

```
struct Fractures
{
 unsigned int N_frac;
 vector<unsigned int> frac_id;
 vector<unsigned int> N_vert;
 vector<vector<Vector3d>> frac_vertices;
 vector<pair<vector<unsigned int>,
    vector<unsigned int>>> trace_type;
};
```

- **trace_type** memorizza la tipologia di traccia per ogni frattura.

Parte 1 - Definizione delle strutture

Struttura per i dati delle tracce.

```
struct Traces
{
 vector<unsigned int> traces_id = {};
 vector<vector<unsigned int>> traces_gen = {};
 vector<vector<Vector3d>> traces_points = {};
 vector<double> traces_length = {};
};
```

Parte 1 - Calcolo delle Tracce

Find_Traces riempie la struttura Traces e completa trace_type.

```
void Find_Traces(Fractures, Traces);
```

Parte 1 - Calcolo delle Tracce

Find_Traces riempie la struttura Traces e completa trace_type.

```
void Find_Traces(Fractures, Traces);
```

↓ ↓ ↓

```
vector<Vector3d> Calculate_Bounding_Box(
                    fracture_points);
```

↓ ↓ ↓

Parte 1 - Calcolo delle Tracce

Find_Traces riempie la struttura Traces e completa trace_type.

```
void Find_Traces(Fractures, Traces);
```

↓ ↓ ↓

```
vector<Vector3d> Calculate_Bounding_Box(
                    fracture_points);
```

↓ ↓ ↓

```
Vector4d pianoFrattura(point_1,point_2,point_3,
                       epsilon);
```

↓ ↓ ↓

Parte 1 - Calcolo delle Tracce

Find_Traces riempie la struttura Traces e completa trace_type.

```
void Find_Traces(Fractures, Traces);
```

↓ ↓ ↓

```
vector<Vector3d> Calculate_Bounding_Box(
                    fracture_points);
```

↓ ↓ ↓

```
Vector4d pianoFrattura(point_1,point_2,point_3,
                       epsilon);
```

↓ ↓ ↓

```
pair<Vector4d, Vector4d> equazioneRetta(
                       point_1,point_2,epsilon);
```

Parte 1 - Ordinamento

- Pseudo-codice per la funzione di ordinamento delle tracce:

```
for(pair : trace_type)
{
    sort_pair(Tips, Ids);
    lastZeroIndex = binary_search(Tips);
    if(sorted(lengths))
    {
        sort_pair(lengths, Ids);
    }else{
        sort_pair(first_half, first_half_Ids);
        sort_pair(second_half, second_half_Ids);
    }
}
```

Parte 1 - Plot su Paraview

DFN con 50 fratture:

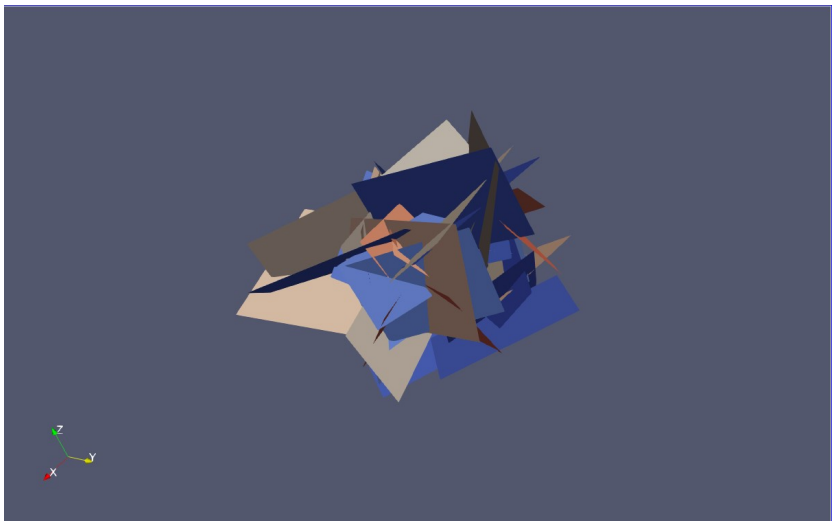

Figura: Fratture.

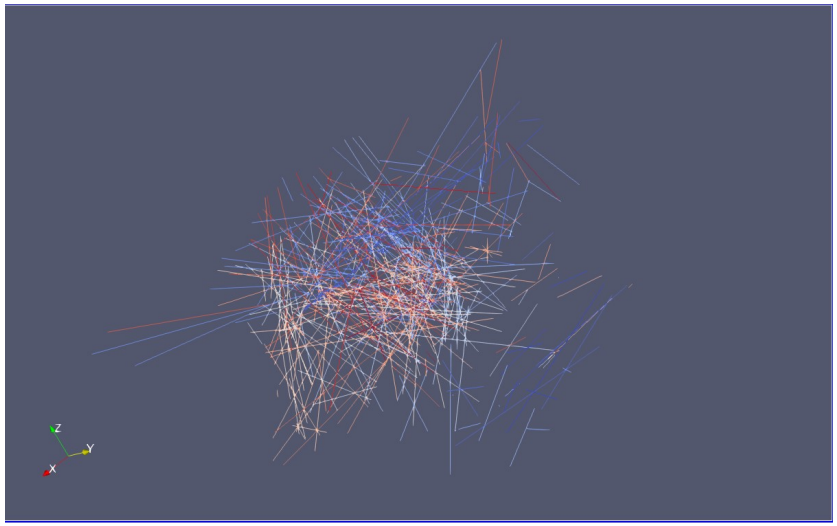

Figura: Tracce.

Parte 1 - Plot su Paraview

DFN con 200 **fratture:**

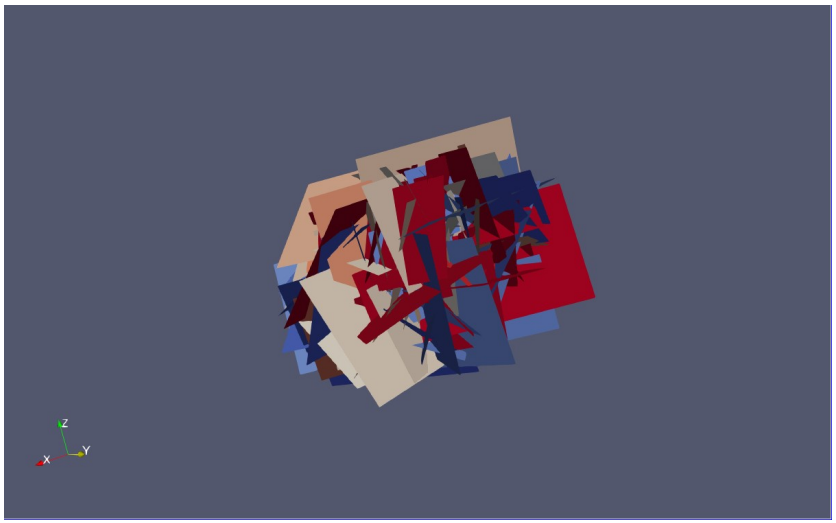

Figura: Fratture.

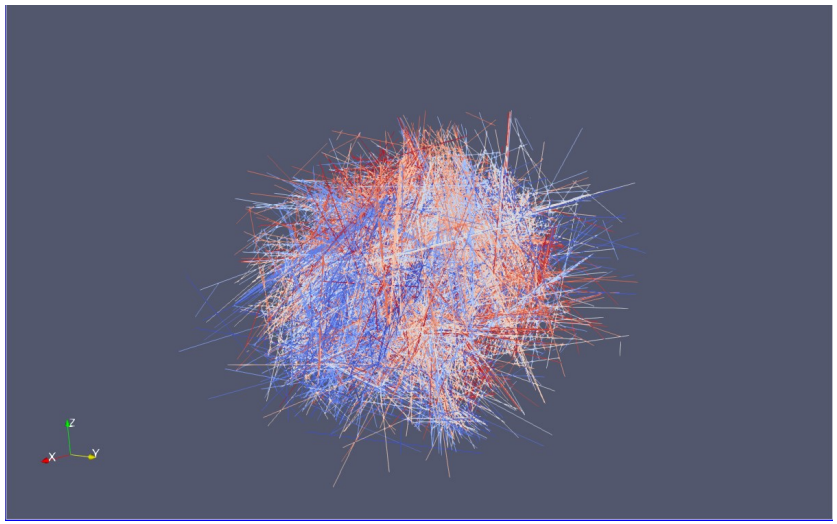

Figura: Tracce.

Parte 2 - Calcolo dei sotto-poligoni

Il taglio delle fratture viene gestito dalle seguenti funzioni:

```
bool cutPolygons(Fractures, Traces,
    found_polygons)
```

Parte 2 - Calcolo dei sotto-poligoni

Il taglio delle fratture viene gestito dalle seguenti funzioni:

```
bool cutPolygons(Fractures, Traces,
    found_polygons)
```

```
pair<vector<Vector3d>,vector<Vector3d>>
    subPolygons(frac_vertices, traces_points,
     tip)
```

Parte 2 - Calcolo dei sotto-poligoni

Il taglio delle fratture viene gestito dalle seguenti funzioni:

```
bool cutPolygons(Fractures, Traces,
    found_polygons)
```

↓ ↓ ↓

```
pair<vector<Vector3d>,vector<Vector3d>>
    subPolygons(frac_vertices, traces_points,
     tip)
```

↓ ↓ ↓

```
vector<Vector3d> extendTraceToEdges(
    frac_vertices, traces_points)
```

Parte 2 - Polygonal Mesh

```
    struct PolygonalMesh
{
    unsigned int NumberOfCell0Ds = 0;
    vector<unsigned int> IdCell0Ds = {};
    vector<Vector3d> CoordinatesCell0Ds;
    unsigned int NumberOfCell1Ds = 0;
    vector<unsigned int> IdCell1Ds = {};
    vector<vector<unsigned int>> VerticesCell1Ds
        = {};
    unsigned int NumberOfCell2Ds = 0;
    vector<unsigned int> NumberOfVertices = {};
    vector<unsigned int> NumberOfEdges = {};
    vector<vector<unsigned int>> VerticesCell2Ds
        = {};
    vector<vector<unsigned int>> EdgesCell2Ds =
       {};
};
```

Parte 2 - Plot su Paraview

DFN con 3 fratture:

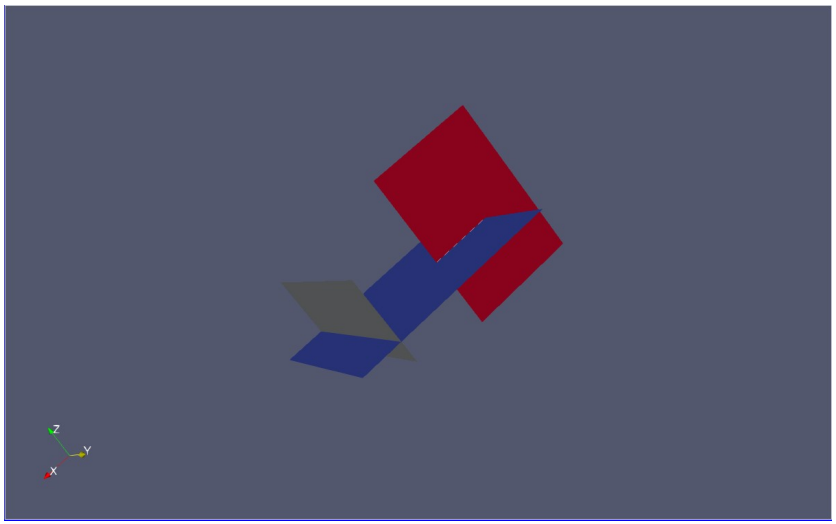

Figura: Fratture.

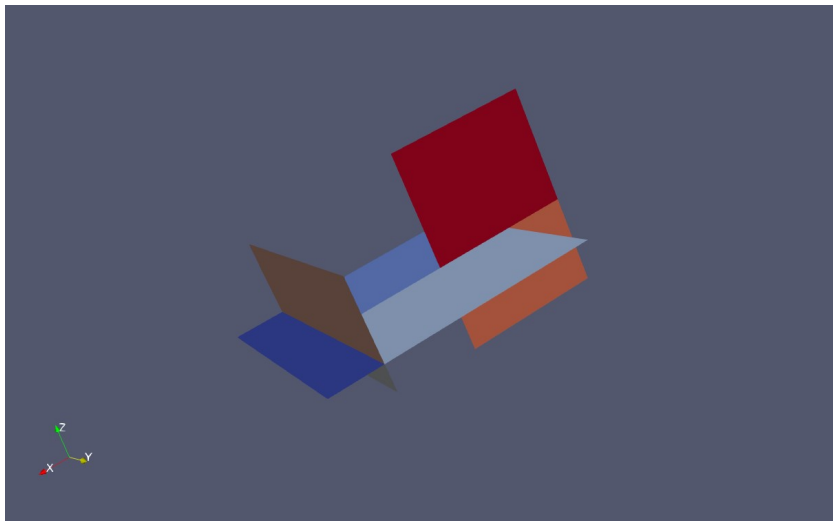

Figura: Sotto-poligioni.

Parte 2 - Plot su Paraview

DFN con 50 fratture:

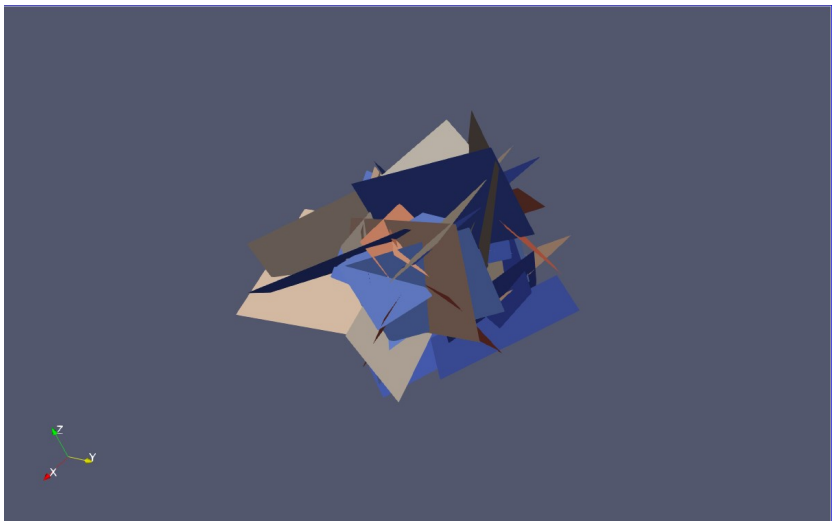

Figura: Fratture.

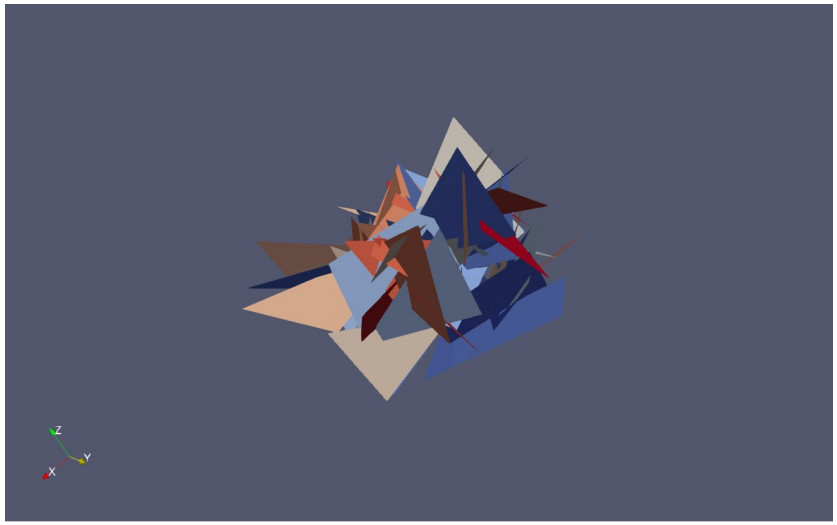

Figura: Sotto-poligioni.

Parte 2 - Plot su Paraview

DFN con 200 **fratture:**

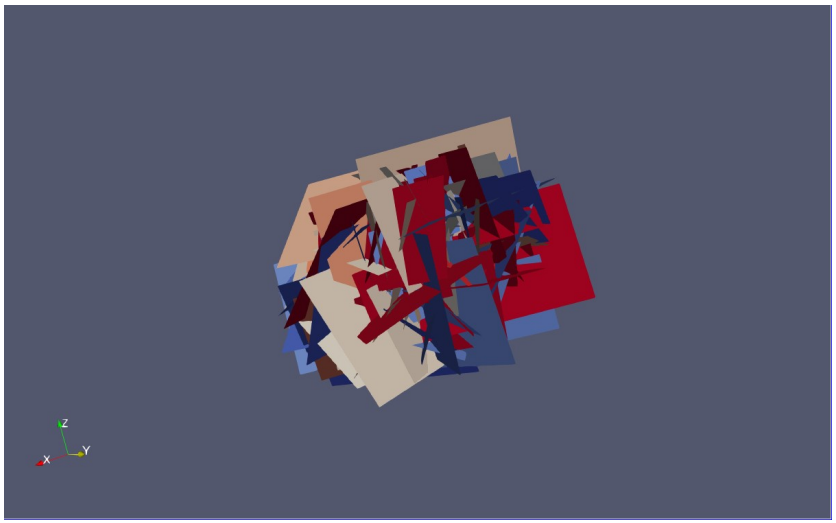

Figura: Fratture.

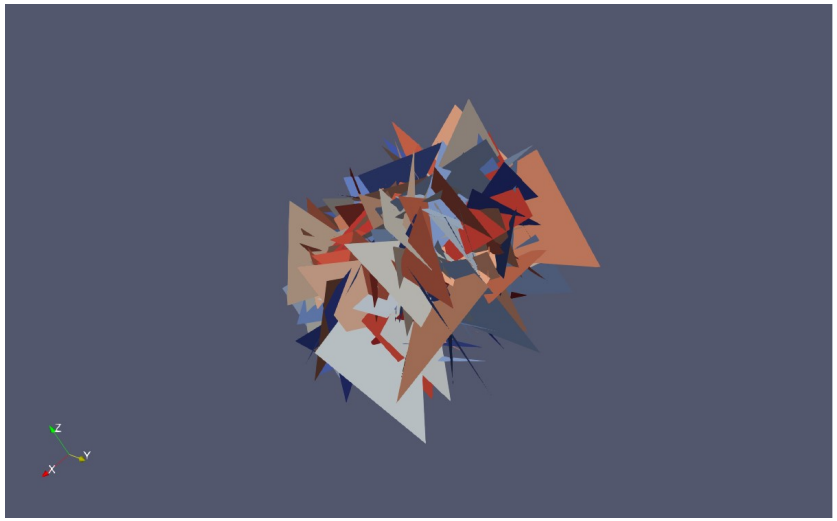

Figura: Sotto-poligioni.