

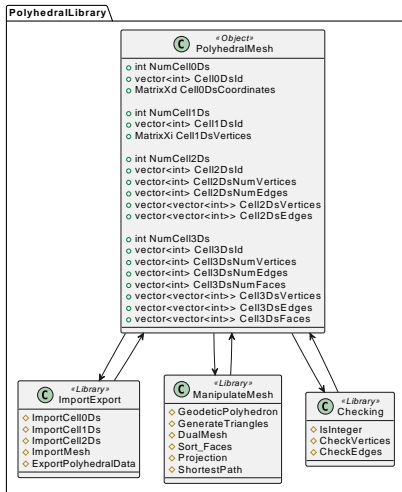
Programmazione e Calcolo Scientifico

Discussione Orale

Allemand Alessandro, Cannone Gabriele, Ceria Federico

23 giugno 2025

PolyhedralMesh

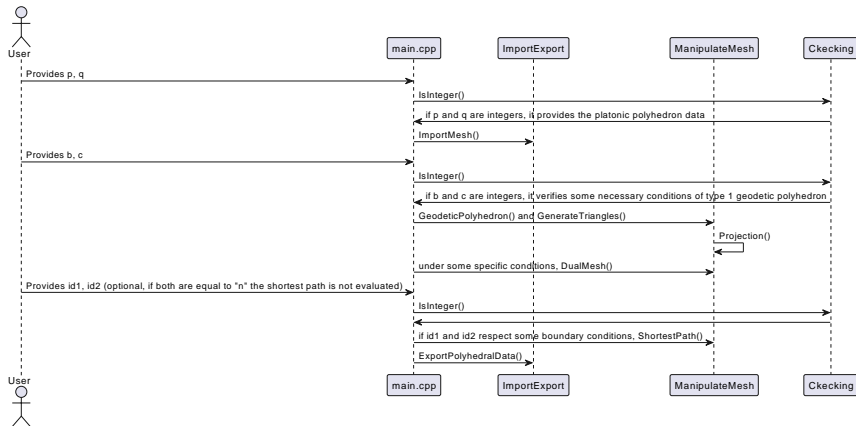


La struct PolyhedralMesh contiene al suo interno, ciascuno con coordinate e ID di riferimento:

- ▶ i vertici della mesh (Cell0Ds);
- ▶ gli spigoli della mesh (Cell1Ds);
- ▶ le facce della mesh (Cell2Ds);
- ▶ il poliedro nella sua interezza (Cell3Ds), necessario per l'export.

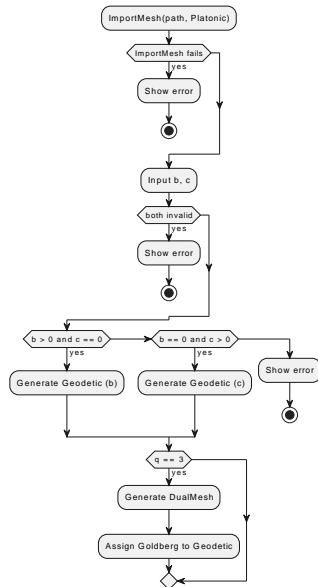
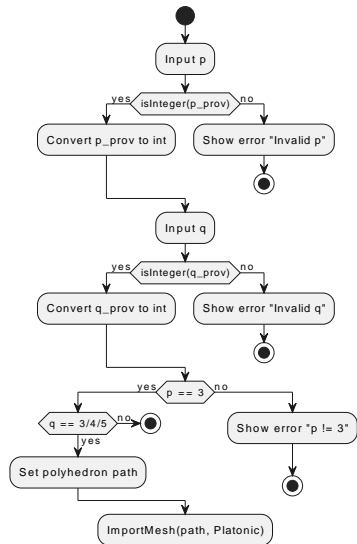
La struct comunica con le funzioni di import ed export all'inizio e alla fine del processo (vedere il Sequence Diagram), con le funzioni di manipolazione della mesh per poter generare i solidi e con quelle di controllo per evitare errori di sovrapposizione e di accesso ad elementi fuori range.

Sequence Diagram



La figura mostra le interazioni tra il main e le diverse funzioni contenute in Utils.cpp. Nelle prossime slide verrà ben analizzato il percorso logico sotto forma di flow chart.

Activity Diagram I



Activity Diagram II



Le funzioni: generazione dei poliedri

PolyhedralLibrary::ManipulateMesh



«Funzione»
GeodeticPolyhedron

+input: Platonic Polyhedron
+input: segments (= b or c for type 1 geodetic)
+output: Geodetic Polyhedron
-Generazione dei vertici del poliedro geodetico.
-Costruzione degli spigoli (e verifica duplicati).
-Costruzione delle facce (e verifica duplicati).
-Proiezione sulla sfera (normalizzazione).
-Assegnazione delle celle 0D, 1D, 2D, 3D.



«Library»
Checking

CheckVertices
CheckEdges



Projection()

PolyhedralLibrary::ManipulateMesh



«Funzione»
DualMesh

+input: GeodeticPolyhedron
+output: Dual
-Generazione dei vertici del duale (coincidenti con le facce del geodetico).
-Costruzione degli spigoli: per ogni punto, si individuano i centroidi delle facce tra i cui vertici si trova questo punto e li si collega.
-Costruzione delle facce: gli spigoli appena generati racchiudono le facce del duale.
-Proiezione sulla sfera (normalizzazione).
-Assegnazione delle celle 0D, 1D, 2D, 3D.



«Library»
Checking

CheckVertices
CheckEdges



Projection()

Si noti che a meno della fase iniziale di DualMesh, nella quale si vanno a definire nuovamente vertici e facce del duale in base alla regola di Eulero a partire dal solido geodetico, una volta individuati i nuovi vertici, spigoli e facce la restante costruzione è analoga a quella del geodetico.

Le funzioni: il cammino minimo

PolyhedralLibrary::ManipulateMesh



«Funzione»

ShortestPath

+input: Polyhedron, Id1, Id2, Weight

+output: PathEdges, Path

-Creazione della lista di adiacenza dei vertici.

-Costruzione della matrice dei pesi Weight con distanze euclidee tra vertici adiacenti.

-Applicazione algoritmo di Dijkstra.

-Ricostruzione del percorso e stampa di esso.

-Calcolo della lunghezza del cammino.

La funzione utilizza l'algoritmo di Dijkstra, un esempio di Single Source Shortest Path (SSSP): ciò è legittimo perché la matrice dei pesi Weight è composta da soli elementi positivi (le distanze euclidee tra due nodi). Come ci si aspetta per la dimensione del problema, il tempo di valutazione è quasi istantaneo: per tale algoritmo, la complessità è $O((n + m) \log(n))$, con n numero di nodi e m numero di spigoli.

Le funzioni: strumenti di controllo

Checking



«Funzione»

CheckVertices

- input: Matrix, contenente i punti
- input: Vector3d (punto candidato)
- input: duplicate_id
- output: bool

Verifica della presenza del punto contenuto in Vector3d nella Matrix. Se lo è, in duplicate_id viene salvato l'id del punto.



«Funzione»

CheckEdges

- input: Matrix, v1, v2
- input: existing_edge_id
- output: bool

Verifica che lo spigolo di vertici v1 e v2 non sia già presente. In tal caso, salvataggio dell'id corrispondente in existing_edge_id.



«Funzione»


IsInteger


- input: string
- output: bool

Verifica che il contenuto della stringa sia un numero intero, in modo da poterlo convertire in type int senza errori.

Le prime due funzioni vengono impiegate nel corso della generazione dei poliedri; la terza a monte della generazione, in quanto è necessario verificare che gli elementi in input siano effettivamente interi.

Le funzioni: adattamenti

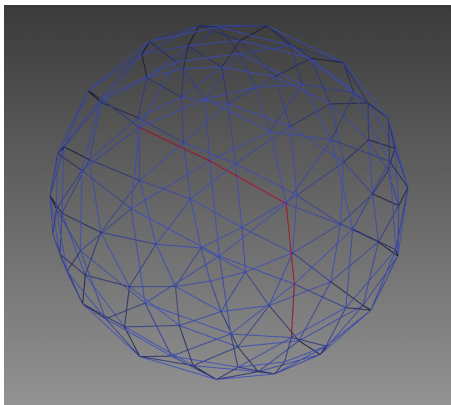
 «Funzione» Projection
○ input: PolyhedralMesh
Normalizzazione di ogni punto sulla sfera unitaria.

 «Funzione» SortFaces
○ input: unsorted_faces ○ input: Polyhedron
○ output: SortedFaces
Dato il vettore "unsorted_faces" di facce adiacenti ad un vertice comune, questa funzione le riordina affinché ogni faccia condivida uno spigolo con la faccia successiva del vettore "ordered faces".

La funzione Projection è una semplice normalizzazione delle coordinate dei vertici del poliedro.

Ben più interessante è il ruolo di SortFaces, che ovvia al problema di riordinare le facce: essa, mediante due cicli annidati, riordina le facce in base agli spigoli in comune e restituisce le stesse facce trovate in precedenza riordinate secondo tale criterio.

Risultato finale



L'output finale del lavoro, con $\{p, q\} = \{3, 4+\}_{5,0}$ (corrispondente ad un ottaedro) e il cammino minimo tra i vertici 3 e 14 evidenziato in rosso. I due file Cell0Ds.inp e Cell1Ds.inp sono stati convertiti ad un formato .vtu per mezzo di Mesh Converter, e i nuovi file sono stati inseriti su Kitware Glance.