

UNIVERSITÀ DEGLI STUDI DI PADOVA
FACOLTÀ DI INGEGNERIA



Corso di Laurea Magistrale in Ingegneria Informatica

Un algoritmo per la preservazione della topologia nella generalizzazione di isolati urbani

Laureando:

Roberto Zampieri

Relatore:

Prof. Massimo Rumor

Correlatore:

Ing. Sandro Savino

Laboratorio G.I.R.T.S.

Dipartimento di Ingegneria dell'Informazione

12 Ottobre 2015

UNIVERSITÀ DEGLI STUDI DI PADOVA

Abstract

Ingegneria Informatica

Dipartimento di Ingegneria dell'Informazione

La generalizzazione cartografica è un processo essenziale nell'ambito della cartografia. Il suo obiettivo è generare una mappa ad una scala minore, partendo da una mappa di scala maggiore. Le astrazioni della realtà, anni fa, erano svolte dai cartografi, che ridisegnavano a mano ogni dettaglio all'interno della cartografia. Per ovviare ai lunghi tempi da cui derivava questo lavoro a mano, è nata l'idea di automatizzare questo processo: è da qui che parte il progetto di ricerca CARGEN.

Il progetto CARGEN ha lo scopo di sviluppare strumenti in grado di automatizzare il processo di generalizzazione. Di nostro interesse sarà la generalizzazione dell'edificato. In particolare sono già presenti algoritmi che eseguono la generalizzazione di singoli edifici, ma non è mai stato trattato un complesso di edifici adiacenti tra loro.

Nel contesto urbano sono presenti molti blocchi composti da edifici contigui; applicandovi la soluzione adottata da CARGEN si ottiene una violazione dei vincoli topologici: edifici, che inizialmente erano adiacenti e non si sovrapponevano, dopo la fase di semplificazione risultano sovrapposti o addirittura non adiacenti tra loro; naturalmente una semplificazione di questo tipo non risulta corretta.

Questa tesi si pone come obiettivo la realizzazione di un algoritmo in grado effettuare la generalizzazione di isolati urbani garantendo il rispetto dei vincoli topologici.

Indice

Abstract	iii
Indice	v
Elenco delle figure	vii
1 La Cartografia e GIS	1
1.1 La cartografia	1
1.2 GIS	3
1.3 Il processo cartografico	5
2 La generalizzazione	7
2.1 Definizione di generalizzazione	7
2.2 Il processo di generalizzazione di una mappa	9
2.2.1 Generalizzazione del modello	9
2.2.2 Generalizzazione cartografica	10
2.3 Generalizzazione automatica	10
2.4 Vantaggi della generalizzazione	11
2.5 Gli operatori della generalizzazione	12
2.6 Il progetto CARGEN	15
3 Generalizzazione degli edifici	17
3.1 Le problematiche da affrontare	17
3.2 Semplificazione delle geometrie	19
3.3 Soluzione adottata da CARGEN	20
3.3.1 Implementazione della soluzione	22
3.3.2 Casistiche dell'algoritmo	24
3.3.3 Risultati ottenuti, limiti e ipotesi di miglioramento	26
4 Cargen-Edifici-Topologici	29
4.1 Strumenti utilizzati	29
4.2 Approccio	31
4.3 Implementazione della soluzione	34
4.3.1 Passi dell'algoritmo	35
4.3.2 Costruzione del grafo planare	36
4.3.3 Casistiche dell'algoritmo	37
4.3.3.1 Caso 0°	38

4.3.3.2	Caso 90°	43
4.3.3.3	Caso 180°	48
4.3.4	Semplificazione edge interni	52
4.4	Risultati ottenuti	56
5	Conclusioni	61

Elenco delle figure

1.1	GIS (Geographic Information System)	3
1.2	Confronto tra modello raster e modello vettoriale	4
2.1	Generalizzazione: in alto le dimensioni delle carte sono riportate coerentemente con il valore della scala, in basso le tre carte sono riportate secondo la stessa distanza grafica	8
2.2	Tipologie di processo della generalizzazione	9
2.3	Immagine da Google Maps	11
2.4	Metodo tradizionale per la produzione cartografica	12
2.5	Generalizzazione per la produzione cartografica	12
2.6	Operatori di generalizzazione nel progetto AGENT	14
3.1	Esempio dell'algoritmo di Douglas-Peucker applicato ad una linea spezzata	18
3.2	Esempi di semplificazione delle geometrie	20
3.3	Esempio di semplificazione: caso frequente in centri abitati dove la semplificazione porta ad una sovrapposizione tra due edifici	21
3.4	Trasformazione degli edifici applicando regole di Sester	22
3.5	Ampliamento del range degli angoli nell'algoritmo di Sester	22
3.6	Spike non riconosciuto dall'algoritmo di Sester	23
3.7	Caso 0° di Sester	24
3.8	Caso 90° di Sester	25
3.9	Caso 180° di Sester	25
3.10	Esempio di una corretta semplificazione di un singolo edificio	26
3.11	Esempio di errore topologico dovuto alla sovrapposizione di edifici	26
3.12	Errore topologico, dovuto alla sovrapposizione di edifici, riscontrato nella soluzione provata da Rossella Baldin. Inizialmente sono presenti tre edifici (due adiacenti e uno singolo), ma dopo la semplificazione ce ne sono solo due.	27
4.1	Fusione applicata precedentemente al processo di semplificazione	31
4.2	Esempio di grafo planare orientato con edge, nodi, segmenti e muri	33
4.3	Diagramma E-R	34
4.4	Parametri algoritmo proposto	35
4.5	Semplice esempio di grafo planare: in rosso sono presenti gli edge, mentre i cerchi azzurri rappresentano i nodi	37
4.6	Esempio di semplificazione muri nel caso 0°	38
4.7	Semplificazione muri e gestione dei nodi della parte di muro che precede/succede il muro adiacente	41
4.8	Semplificazione muri e gestione dei nodi dei muri adiacenti	42

4.9	Esempio di semplificazione dei muri e gestione dei nodi del muro \overline{BC} la cui lunghezza è minore della lunghezza critica	42
4.10	Esempio di semplificazione muri nel caso di estensione dei muri adiacenti .	44
4.11	Esempio di semplificazione muri nel caso di spike	45
4.12	Esempio di semplificazione muri e gestione dei nodi nel caso di Spike in cui si ha un aumento dell'area della geometria	47
4.13	Esempio di risoluzione di errore topologico nella casistica Spike	47
4.14	Esempio di casistica 180°	48
4.15	Esempio di semplificazione dei muri con angolo di 0° nella casistica 180° .	49
4.16	Esempio di semplificazione dei muri con angolo di 180° nella casistica 180°	49
4.17	Esempio di semplificazione dei muri e gestione dei nodi nel caso di riduzione dell'area con un'angolo di 180° nella casistica 180°	51
4.18	Esempio di semplificazione di un edge interno in cui il primo e l'ultimo segmento non sono paralleli tra loro	53
4.19	Esempio di trasformazione in una L di un edge interno durante la fase di semplificazione	53
4.20	Processo di semplificazione di un edge interno	54
4.21	Edge interno in cui gli estremi sono entrambi contenuti in due muri	55
4.22	Edge interno in cui un estremo è contenuto in due muri e l'altro estermo in un muro	55
4.23	Esempio di semplificazione di una coppia di edifici adiacenti	57
4.24	Esempio di semplificazione di un'insieme di edifici adiacenti : a) edifici originali b) semplificazione con Sester lunghezza critica pari a 4 metri c) semplificazione con Sester lunghezza critica pari a 8 metri	58
4.25	Esempio di semplificazione di un'insieme di edifici adiacenti : a) edifici originali b) semplificazione con Sester lunghezza critica pari a 2 metri c) semplificazione con Sester lunghezza critica pari a 4 metri	59
4.26	Esempio di semplificazione di un'insieme di edifici adiacenti : a) edifici originali b) semplificazione con Sester lunghezza critica pari a 2 metri c) semplificazione con Sester lunghezza critica pari a 4 metri	59

Capitolo 1

La Cartografia e GIS

In questo capitolo verranno, per prima cosa, fornite alcune definizioni relative ai concetti di cartografia, carta geografica e scala. Successivamente saranno illustrate, in modo generale, le varie classificazioni e tipologie delle carte geografiche e i sistemi informativi geografici (GIS), per concludere infine con la descrizione del processo cartografico, processo tramite il quale si realizza una mappa.

1.1 La cartografia

*La **cartografia** è la scienza che si occupa di riprodurre l'aspetto della Terra o di una sua parte con la minima distorsione possibile, in modo che le informazioni contenute sulla carta siano il più possibile accurate.*

– Enc. Italiana Grolier, 1987

*La **carta geografica** è una rappresentazione grafica, riprodotta in scala e secondo simboli convenzionali, di una parte o di tutta la superficie terrestre, o di alcuni suoi aspetti particolari.*

– Diz. Devoto, 1979

Un aspetto molto importante nella definizione di carta geografica è il concetto di scala. Essa è il rapporto di riduzione delle lunghezze sulla carta rispetto a quelle nella realtà. La misura della scala è indicata da un rapporto $1 : n$, dove n è il numero di volte che sulla carta è stata ridotta la distanza topografica fra due punti. Perciò più grande è il rapporto (e quindi più piccolo è il numero n), maggiori sono i dettagli che si vedono. Se si

vuole rappresentare un piccolo territorio con molti dettagli si utilizzerà una grande scala, viceversa, per rappresentare un grande territorio con i soli elementi principali basta una piccola scala.

Si può definire una classificazione delle mappe basata sulla scala che metta in evidenza la relazione tra valore di scala adottato e informazioni trasmesse dalla mappa, e ricavare in base a questo criterio quattro gruppi principali di mappe:

- **Piante o mappe.** Hanno una scala maggiore di 1:10.000 (cioè possono avere valori compresi tra 1:1 e 1:10.000). Le piante rappresentano la planimetria di città, mentre le mappe riguardano essenzialmente la raffigurazione di zone rurali. Si tratta di carte molto dettagliate perché riguardano porzioni relativamente piccole di territorio. Fanno parte di questa categoria le carte tecniche regionali (CTR);
- **Carte topografiche.** La loro scala è compresa tra 1:10.000 e 1:150.000. Esse includono aree maggiori e sono meno dettagliate delle piante e delle mappe, ma riescono comunque ad evidenziare particolari molto piccoli come una casa isolata o un ponte;
- **Carte corografiche.** La loro scala va da 1:150.000 a 1:1.000.000. Raffigurano zone assai estese ed il numero dei particolari è assai ridotto ma comunque significativo;
- **Carte geografiche.** Hanno scala inferiore ad 1:1.000.000 e servono per rappresentare aree molto estese della superficie terrestre, quali ad esempio uno stato o un continente.

La storia della cartografia, in Italia, inizia nel 1872, con la fondazione dell'Istituto Topografico Militare (diventato nel 1882 Istituto Geografico Militare), che nello stesso anno iniziò la rilevazione del territorio italiano per la realizzazione della nuova Carta Topografica d'Italia (1921) alla scala 1:100.000. Negli ultimi anni dello scorso secolo la cartografia ha conosciuto un momento di grande sviluppo, che può essere considerato come una specie di rivoluzione, grazie all'introduzione dell'informatica e la nascita dei primi Sistemi Informativi Geografici (GIS), potenti strumenti di analisi dei dati geografici che permettono non solo la visualizzazione ma anche l'elaborazione e la manipolazione degli elementi di una carta.

1.2 GIS

Un GIS è una potente serie di strumenti per acquisire, memorizzare, estrarre, trasformare e visualizzare dati spaziali dal mondo reale.

– Burrough, 1986

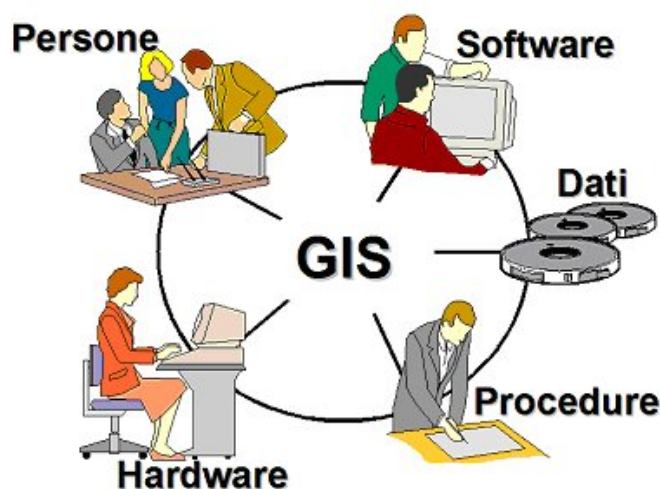


FIGURA 1.1: GIS (Geographic Information System)

Con l'avvento dei GIS la cartografia subisce un cambiamento epocale. I GIS, infatti, permettono di manipolare oggetti spaziali e di estrarre, in vari modi, una grande quantità di informazioni dalle mappe. In questo modo la mappa da elemento statico si trasforma in una sorta di oggetto dinamico. Il singolo dato geografico diventa un'entità dotata di attributi e inserita in una gerarchia relazionale; a questa entità corrisponde, oltre che una rappresentazione grafica, un contenuto logico e semantico. In altre parole, un GIS permette di integrare operazioni che tipicamente offre un database con i vantaggi offerti dalla visualizzazione della mappa. Le entità geografiche, elementi base del GIS, vengono memorizzate all'interno di un database geografico (GeoDB), detto anche database spaziale, la cui funzione è quella di memorizzare, interrogare e manipolare informazioni geografiche e dati spaziali.

Più precisamente, un sistema GIS è costituito da risorse e procedure. Le risorse coinvolte sono risorse umane, infrastrutturali e di dati; le procedure, invece, si suddividono in acquisizione, archiviazione, elaborazione, presentazione e trasmissione.

I dati sono in assoluto l'elemento più prezioso del sistema perché hanno un costo di produzione elevato. Ogni dato, che viene chiamato feature, è costituito da due componenti:

- componente spaziale: contiene informazioni relative alla posizione (geografica), alla geometria (forma e dimensione) e alla topologia (relazioni spaziali con altri dati) del dato;
- componente non spaziale: formata da dati descrittivi, o attributi, e metadati associati alla componente spaziale che velocizzano le operazioni di query spaziale.

Nella fase di acquisizione, che è seguita da un processo di astrazione e generalizzazione, è necessario memorizzare i dati. Il modo in cui le informazioni spaziali vengono memorizzate dipende da quale modello dello spazio viene adottato dal GIS. Ci sono due modelli utilizzabili:

- il modello raster in cui lo spazio viene visto come un mosaico, dove ogni tassello, o pixel, contiene uno o più valori (ad esempio, il colore o l'altitudine). Il pixel rappresenta il luogo dove una grandezza assume un certo valore. Un file raster risulta allora costituito da una matrice quadrata dove ogni cella è un pixel; le dimensioni di tale file tendono ad essere molto elevate;
- il modello vettoriale che è formato da una serie di primitive geometriche:
 - il punto, definito da una coppia (x,y) o da una terna di coordinate (x, y, z) ;
 - la linea, costituita da una sequenza ordinata di punti;
 - il poligono, anch'esso costituito da una sequenza ordinata di punti, dove però l'ultimo punto della sequenza corrisponde al primo; di consuetudine è utilizzato per rappresentare laghi, edifici o confini amministrativi .

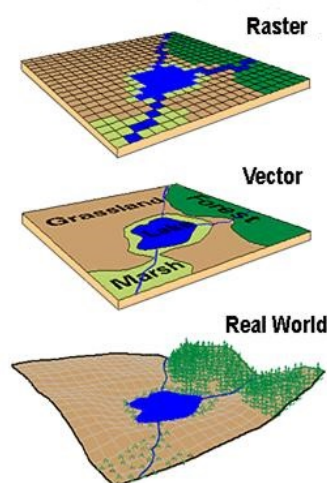


FIGURA 1.2: Confronto tra modello raster e modello vettoriale

Comparando il modello raster e quello vettoriale emerge che quest'ultimo si presta molto di più alla manipolazione, all'elaborazione e all'interrogazione dei dati, i quali si possono integrare facilmente in un database relazionale dove vengono trattati come valori di una colonna di una tabella (solitamente chiamata "Geometry") e di seguito processati.

Una volta che i dati sono stati acquisiti e opportunamente memorizzati secondo uno dei due modelli appena illustrati, è possibile effettuare su di essi operazioni di elaborazione e trasformazione degli elementi geografici tramite degli strumenti di analisi forniti dal GIS. Alcune di queste operazioni sono:

- l'overlay topologico in cui si effettua una sovrapposizione tra gli elementi di due temi per creare un nuovo tema, ad esempio si sovrappone il tema dei confini di un parco con i confini dei comuni in modo da determinare le superfici di competenza;
- le query spaziali, ovvero interrogazioni dei dati a partire da criteri spaziali quali vicinanza, inclusione, sovrapposizione etc. . . ;
- il buffering che permette di creare un poligono che circonda la geometria originaria;
- la segmentazione che applicata su un elemento lineare determina il punto alla distanza specificata dall'inizio dell'elemento;
- la network analysis che applicata su una rete di elementi lineari, ad esempio la rete stradale, determina i percorsi minimi tra due punti;
- la spatial analysis che effettua un'analisi spaziale di varia tipologia utilizzando un modello raster, come ad esempio l'analisi di visibilità;

1.3 Il processo cartografico

Il processo cartografico, ovvero il processo di realizzazione di una carta, è il frutto di una sequenza ordinata di operazioni.

1. Definizione delle caratteristiche finali della mappa. Le caratteristiche tecniche principali sono la scala, la superficie di riferimento, la proiezione e il tipo di rappresentazione (conforme, equivalente, equidistante). In questa prima fase si definisce, dunque, come debba avvenire l'astrazione della realtà. In questo modo si produce per le tradizionali mappe cartacee la legenda della carta e per le mappe digitali la definizione di un GeoDB.
2. Analisi delle caratteristiche finali della mappa.

3. Raccolta dei dati. I dati possono essere raccolti a seconda che la mappa sia il frutto di una rilevazione o di una derivazione. Nel primo caso, gli elementi del territorio vengono rilevati durante un sopralluogo con l'aiuto di sistemi di posizionamento satellitare (GPS) e fotografie aeree. Nel secondo caso, se la mappa viene derivata, vengono utilizzati dati provenienti da una cartografia preesistente, la quale dovrà avere un livello di dettaglio maggiore (e quindi un'alta scala) rispetto alla carta che si deve realizzare.
4. Produzione della mappa. Una volta che il cartografo entra in possesso di tutte le informazioni (dati e specifiche) necessarie, avvia il processo di generalizzazione che rappresenta la parte fondamentale per la costruzione della mappa. Durante questa fase, il cartografo deve prendere una serie di decisioni atte a soddisfare le specifiche definite ma anche a garantire i requisiti classici di ogni mappa, quali la leggibilità e l'usabilità. In pratica egli dovrà creare un'astrazione della realtà geografica che faciliti la comprensione e la comunicazione dell'informazione.
5. Collaudo. Si verifica la correttezza e la consistenza della carta. Tra le attività più importanti vi è il controllo finale sul terreno mediante operazioni di misura e verifica della rappresentazione cartografica: vengono confrontati sul campo i dati riportati nella mappa con rilevazioni effettuate tramite strumenti ad alta precisione, come ad esempio il GPS differenziale.

Capitolo 2

La generalizzazione

In questo capitolo verrà trattato il tema della generalizzazione. Sarà definito il concetto di generalizzazione, verrà descritto il processo che permette la generalizzazione di una mappa e saranno esposte le ragioni che inducono a realizzarlo; verranno esaminati alcuni modelli concettuali della generalizzazione proposti in letteratura ed infine si tratteranno gli operatori di generalizzazione che saranno descritti e classificati in base alla loro funzione.

2.1 Definizione di generalizzazione

La selezione e la rappresentazione semplificata dei dettagli che meglio si adatta alla scala e o allo scopo di una mappa.

– ICA, 1973

Osservando una carta geografica, si può notare come gli elementi della realtà vengono classificati, ridefiniti e posizionati, sempre secondo un ordine logico e grafico. Se non fossero compiute queste azioni di classificazione e ridefinizione, l'informazione cartacea sarebbe sensibilmente inferiore rispetto l'informazione reale.

Ne consegue che non ha senso tentare di rappresentare nella mappa tutta la realtà; è necessario semplificare le informazioni geografiche per soddisfare i vincoli di rappresentazione e per raggiungere la finalità della carta, mettendo in evidenza gli elementi importanti e al tempo stesso rimuovendo quelli irrilevanti. Il risultato sarà dunque una astratta e più o meno limitata parte della realtà.

Lo scopo della generalizzazione è proprio quello di astrarre la realtà per rappresentarla in modo che essa risulti chiara, comprensibile, leggibile e in accordo con la scala di rappresentazione scelta.

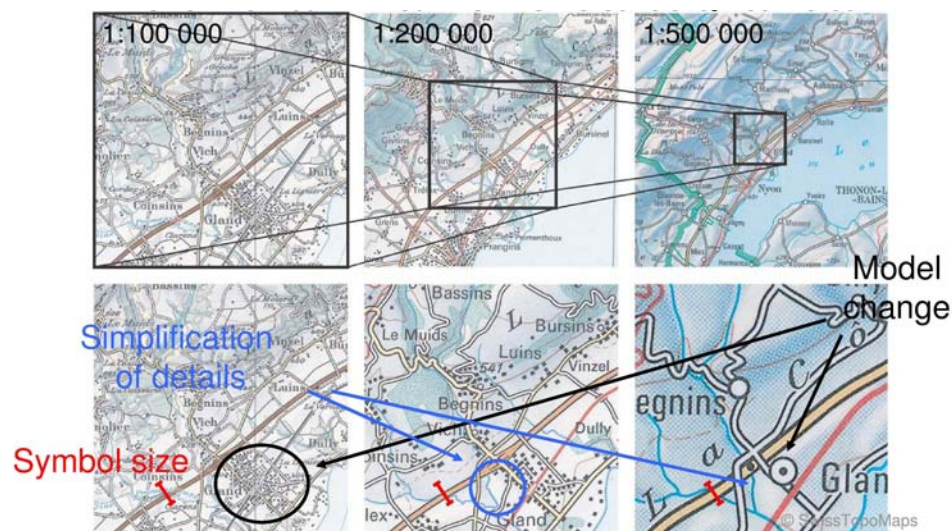


FIGURA 2.1: Generalizzazione: in alto le dimensioni delle carte sono riportate coerentemente con il valore della scala, in basso le tre carte sono riportate secondo la stessa distanza grafica

Secondo McMaster e Shea (1992) i fattori che devono essere tenuti in considerazione dal processo di generalizzazione sono :

1. Riduzione della complessità: il passaggio ad una scala inferiore porta naturalmente diversi oggetti ad entrare in conflitto per lo stesso spazio. Ridurre il numero di questi oggetti consente di far risaltare gli elementi più importanti e di mantenere la mappa leggibile.
2. Mantenimento dell'accuratezza spaziale: più la scala è alta più l'accuratezza spaziale deve essere rispettata, limitando al minimo l'errore dovuto alla diversa posizione degli oggetti nella mappa rispetto alla realtà.
3. Mantenimento dell'accuratezza degli attributi: in particolare nelle mappe tematiche, l'obiettivo è quello di minimizzare le alterazioni non intenzionali degli attributi delle feature.
4. Mantenimento della qualità estetica: dai colori utilizzati alla simbologia, dal bilanciamento allo stile tipografico; bisogna valutare tutti i vari fattori che influenzano l'estetica complessiva di una mappa.
5. Mantenimento di una logica gerarchica: elementi di una stessa categoria, ma di importanza o dimensioni differenti, devono essere differenziati, in accordo con lo scopo della mappa.
6. Coerente applicazione delle regole di generalizzazione: allo scopo di ottenere una generalizzazione imparziale e coerente, il cartografo deve determinare esattamente

quali algoritmi applicare, in quale ordine ed i parametri di input necessari per ottenere il risultato voluto quando si opera ad una data scala.

2.2 Il processo di generalizzazione di una mappa

La generalizzazione viene comunemente distinta in due tipologie di processo:

- **map compilation** se i dati disponibili sono il frutto di una rilevazione, cioè di un'acquisizione diretta;
- **map generalization** se i dati sono quelli provenienti da una cartografia preesistente.

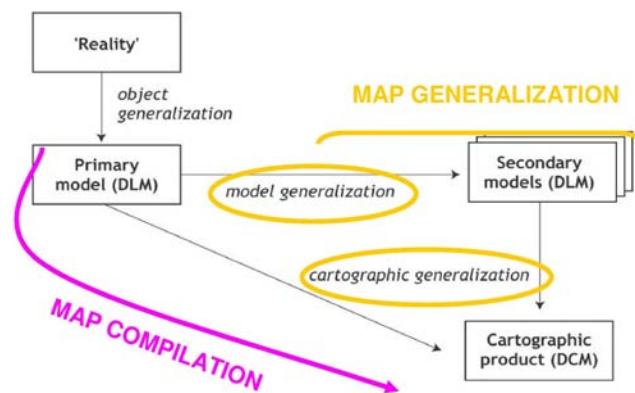


FIGURA 2.2: Tipologie di processo della generalizzazione

Entrambe le tipologie devono affrontare le medesime problematiche riguardanti il contenuto della mappa e la sua rappresentazione tanto che, nella pratica, non esiste una reale distinzione. Infatti la map generalization può essere vista come una parte della map compilation. La vera differenza sta all'origine: nella produzione di una mappa rilevata bisogna infatti costruire un modello astratto partendo dalla realtà, mentre nel creare una mappa derivata è necessario partire da un modello della realtà.

2.2.1 Generalizzazione del modello

Per mezzo della generalizzazione del modello, il risultato che si ottiene è la creazione di un modello astratto, rappresentativo della realtà. La scelta relativa a quali aspetti della realtà riportare e quali invece eliminare avviene in funzione degli scopi per cui viene realizzata la mappa. Ad esempio, in una carta politica verrà dato risalto ai confini amministrativi mentre elementi fisici come i rilievi o le depressioni non verranno riportati.

Questo processo porta alla definizione di una tassonomia degli elementi della realtà cioè ad una loro suddivisione in classi. Anche i parametri che portano a distribuire gli oggetti in più classi variano a seconda delle funzioni della mappa che si vuole realizzare: una carta civile può classificare le strade in urbane ed extraurbane mentre una carta militare può enfatizzare maggiormente una distinzione delle vie di comunicazione basata sulla loro larghezza e tipo di fondo.

Nella cartografia digitale, la generalizzazione del modello corrisponde alla definizione di un GeoDatabase. Durante la fase di definizione del database è fondamentale ricordare che una sua modellazione e strutturazione efficiente rappresenta un presupposto necessario per la buona riuscita della successiva fase di generalizzazione cartografica.

2.2.2 Generalizzazione cartografica

La seconda componente della generalizzazione è quella cartografica, che agisce sulle informazioni geografiche delle feature.

Una volta definito il modello astratto della realtà, si passa alla ridefinizione delle geometrie, allo scopo di fornire una coerente rappresentazione grafica al modello dei dati. Questa fase è di fondamentale importanza per garantire una mappa accurata e comprensibile. Il cartografo, per ridefinire le geometrie, si deve affidare alla sua interpretazione personale. Non esiste infatti nessuno schema rigido che si possa applicare sistematicamente ad ogni singolo caso: per esempio un agglomerato di case potrebbe venire rappresentato come un unico blocco da un primo operatore e da due o più blocchi da un secondo operatore.

Tra i fattori che possono influenzare le scelte compiute dal cartografo troviamo la scala di rappresentazione, e, successivamente il contesto. In relazione al valore di scala, un centro urbano potrebbe venire rappresentato con un punto affiancato dal nome della città oppure da un suo sottoinsieme di strade ed edifici. Il contesto invece determina il livello di importanza attribuito ad un oggetto, che può determinare la sua eliminazione o la precisione con cui lo si rappresenta. Ad esempio, un'etichetta posizionata in una zona densamente edificata potrebbe essere omessa in modo da garantire la massima leggibilità della mappa.

2.3 Generalizzazione automatica

L'avanzamento delle tecnologie informatiche ha permesso di automatizzare l'intera procedura di generalizzazione, riducendo notevolmente fatica e tempo che l'operatore umano deve dedicarvi e migliorando in termini di qualità e di complessità il prodotto finale.

Ecco perchè il campo della generalizzazione automatica desta sempre più interesse, stimolando la produzione scientifica in questo settore.

La realizzazione di un processo di generalizzazione automatico si deve comunque scontrare con alcuni ostacoli (Mackaness, 2007):

- Complessità del processo di progettazione: la mappa è un complesso mix di pattern metrici e topologici che solitamente sono molto interdipendenti. Interpretare correttamente queste forme e individuare le caratteristiche della mappa richiede una conoscenza sia cartografica che geografica.
- Trasformazione delle informazioni dovuta ad un cambiamento di scala: le mappe, al variare della scala, mettono in luce caratteristiche geografiche diverse. Il processo di generalizzazione deve, perciò, essere in grado di estrapolare queste differenti caratteristiche in funzione della scala, a partire dalla stessa base di dati.

2.4 Vantaggi della generalizzazione

Consideriamo Google Maps, il più famoso servizio, accessibile dal relativo sito web, che consente la ricerca e la visualizzazione di carte geografiche di buona parte della Terra. Ogni livello di zoom è una nuova mappa su diversa scala.



FIGURA 2.3: Immagine da Google Maps

Nella produzione cartografica tradizionale, dalla realtà di interesse, inizialmente, si effettua un'acquisizione degli elementi del territorio con l'aiuto di sistemi di posizionamento satellitare (GPS) e fotografie aeree; successivamente si passa alla fase di "disegno" della mappa che potrà essere aggiornata. Tale procedimento viene ripetuto per ogni scala di interesse. Risulta abbastanza semplice capire che tale metodo richiede una quantità di tempo e di budget molto elevata ed inoltre gli aggiornamenti sono asincroni.



FIGURA 2.4: Metodo tradizionale per la produzione cartografica

Utilizzando la generalizzazione si effettuerebbe un'unica acquisizione a grande scala e le mappe a scala più piccola verranno generate, da essa, in automatico. Ciò comporta vantaggi :

- Economici: rilievo e restituzione non sono necessari, produzione di mappe a diverse scale viene realizzata in modo veloce.
- Organizzativi: le cartografie prodotte, essendo che derivano dalla stessa mappa, sono tra loro consistenti (e di conseguenza anche i dati) ed inoltre è possibile la centralizzazione degli aggiornamenti.

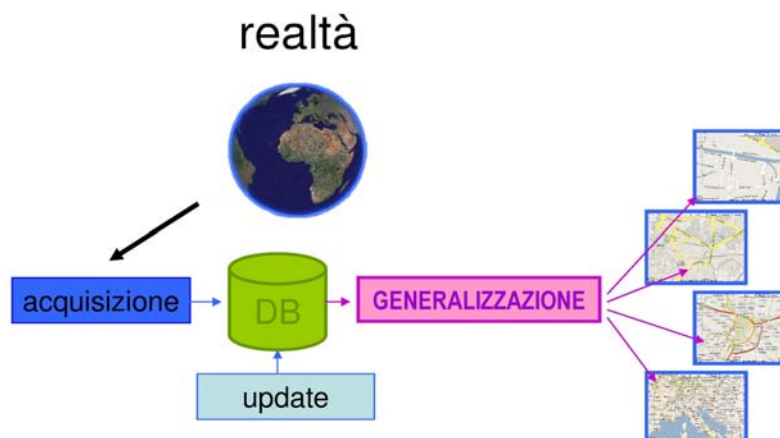


FIGURA 2.5: Generalizzazione per la produzione cartografica

2.5 Gli operatori della generalizzazione

Quando si deve generalizzare una carta ci sono numerosi vincoli da soddisfare in quanto il progetto ha delle specifiche che devono essere rispettate. Un vincolo è una specifica di progetto che deve essere rispettata nella soluzione ad un problema di generalizzazione. Tra i più importanti vincoli si possono citare i seguenti:

- vincoli topologici: legati a connessione, adiacenza e inclusione tra elementi;
- vincoli spaziali: legati alla conservazione delle forme e degli allineamenti;
- vincoli semantici: legati alle relazioni logiche che intercorrono tra gli oggetti e i loro insiemi;
- vincoli di disegno: legati all'aspetto grafico della carta.

Allo scopo di soddisfare questi vincoli, una grande varietà di soluzioni sono state studiate nel corso degli anni e prendono il nome di operatori di generalizzazione. Gli operatori sono stati sviluppati sia emulando le pratiche manuali dei cartografi, sia tramite studi puramente matematici. Ogni operatore, comunque, prende vita mediante un algoritmo di generalizzazione che lo implementa.

Diverse classificazioni sono state proposte per dare un ordine logico a questi operatori, in modo da facilitare i cartografi nella scelta dell'operatore che più si presta a soddisfare un particolare vincolo. Un'importante classificazione è quella proposta da McMaster e Shea (1992) che distingue operatori che trasformano le geometrie e operatori che trasformano gli attributi. Più tardi nel 1997, venne lanciato il progetto AGENT (Automated GEneralization New Technology) con lo scopo di utilizzare l'intelligenza artificiale, (un approccio multi-agente) assegnando ad ogni oggetto da generalizzare una entità autonoma dotata di

- vincoli da rispettare
- un set di operatori da usare
- dei vicini con cui "lottare".

Di nostro interesse, per questa tesi, risulta essere l'operatore di semplificazione su singoli oggetti che sono rappresentati come edificati.

Attribute transformation	Classification	Thematic Selection	Select a subset of feature classes that are relevant to an application.			
		Thematic Aggregation	Changing thematic resolution (moves along a classification hierarchy)			
Spatial transformation	Individual objects	Simplification	Weeding	A representation of the original line using a subset of its initial coordinates, retaining those points which are considered to be most representative of the line.		
			Unrestricted Simplification	A simplified representation of the original line is computed. Instead of using a subset of initial coordinates, the new line may choose any point of the space and may even consist of more points.		
		Collapse	The decomposition of features of n dimensions in features of n-1 or even n-2 dimensions.			
	Enhancement	Enhancement with regard to geometric constraints	Enlargement	Constant enlargement in all directions (scaling).		
			Exaggeration	Exaggerate important parts = Enlargement with change of shape.		
		Enhancement with regard to semantic constraints	Smoothing	Change the geometry of an object to improve the aesthetic quality.		
			Fractalization			
	Individual objects or Set of objects	Selection / Elimination	Selection	Select the most important objects from a cluster/network to represent the original feature.		
			Elimination	Eliminate unimportant objects from the map.		
	Set of objects	Aggregation	Join features to 1 object	Fusion	Aggregation of two connected objects of the same nature.	
				Merge	Join disjoint objects	
		Join feature to several objects	Combine	Combine a set of objects to one object of higher dimensionality.		
Typification			An initial set of objects is transformed into a new (generalized) group. It is not clear after the transformation which original object(s) created a new one; the new objects are merely placeholders. The initial group might be built of disjoint objects (such as buildings) or be created through segmentation of one single object (such as road segments). The former type is called structuration, the latter one schematisation.			

FIGURA 2.6: Operatori di generalizzazione nel progetto AGENT

2.6 Il progetto CARGEN

Il progetto CARGEN il cui nome significa CARTographic GENeralization inizia nel 2006 come una collaborazione tra il Dipartimento di Ingegneria dell'Informazione dell'Università di Padova, la Regione Veneto e l'agenzia cartografica nazionale italiana IGM (Istituto Geografico Militare). E' uno dei primi (se non l'unico) in Italia che si occupa dello studio dei problemi di generalizzazione cartografica ed è attivo non solo in un ambito nazionale, ma entra a far parte attivamente gli sviluppi della comunità di ricerca internazionale.

CARGEN si propone di studiare e sviluppare un processo automatico per generalizzare il geodatabase IGM in scala 1: 25000 dal geodatabase Regionale in scala 1: 5000.

Gli algoritmi sviluppati durante il progetto variano su tutto il campo della generalizzazione e comprendono, tra gli altri, la selezione e la potatura dell'idrografia, la semplificazione delle reti stradali, generalizzazione di edifici e displacement.

Generalizzare in maniera automatica una mappa significa caratterizzare le classi di oggetti che la compongono e, successivamente, determinare una serie di procedure specifiche di generalizzazione per ciascuna tipologia di elementi. Tali procedure dovranno essere in grado di agire sulla geometria dell'oggetto riuscendo ad operare la giusta generalizzazione in relazione al contesto in cui l'oggetto è situato, considerando anche le relazioni e le geometrie con gli elementi circostanti, qualsiasi tipologia essi siano.

Se pensiamo ad esempio alla generalizzazione della viabilità stradale, ferroviaria o dell'idrografia, c'è un approccio ed una serie di strategie da adottare profondamente diverse da quelle necessarie per il trattamento dei fabbricati: la presenza di una strada - e di qualsiasi altra tipologia di oggetti - in prossimità di un edificio condiziona in maniera determinante il risultato della generalizzazione cartografica. Quindi il tema dell'edificato ha un peso specifico notevole e la sua trattazione risulta essere complessa.

Di tutti i temi trattati nel progetto CARGEN ci è utile, per questa tesi, la generalizzazione di edifici.

Capitolo 3

Generalizzazione degli edifici

In questo capitolo si parlerà del progetto CARGEN, delle problematiche da affrontare nella generalizzazione di edifici con una particolare attenzione alla semplificazione delle geometrie intese come edificati . Una volta spiegato tutto ciò verrà fornita ed illustrata la soluzione adottata da CARGEN per effettuare la generalizzazione degli edifici. Verranno mostrati i risultati ottenuti e al fine di migliorare la soluzione già proposta verranno avanzate alcune ipotesi di miglioramento.

3.1 Le problematiche da affrontare

La generalizzazione dell'edificato comprende tutta una serie di problematiche da risolvere:

1. La **selezione dell'edificato** atto a identificare gli oggetti interessati alla procedura di generalizzazione. Utilizzato per dividere in zone l'intero insieme degli edifici.
2. Lo **soltimento dell'edificato** atto a migliorare la chiarezza e la leggibilità rappresentativa senza comunque sconvolgere le caratteristiche della zona in cui si sta operando. Questa operazione avviene mediante l'eliminazione degli edifici di ridotte dimensioni in quelle aree di alta concentrazione di oggetti (ad esempio all'interno di nuclei abitati).
3. La **fusione di elementi omogenei** distinti originariamente per differenze volumetriche, ma posti in relazione geometrica per la condivisione di uno o più lati.

4. La **semplificazione delle geometrie** intesa come riduzione del numero di coordinate necessarie a determinare la geometria di un edificio. Questa problematica prevede due fasi:

- riduzione del numero dei vertici del poligono;
- eliminazione di alcuni segmenti costituenti il perimetro della geometria.

La prima fase può essere risolta utilizzando l'algoritmo di Douglas-Peucker [Douglas & al., 1973]. Lo scopo dell'algoritmo è, data una linea spezzata composta da segmenti, trovare una linea spezzata simile con meno punti. La linea semplificata consiste in un sottoinsieme dei punti che definiscono la linea originale. In particolare si hanno le seguenti fasi :

- Costruzione di una linea di trend fra i punti estremi della linea originale;
- Individuare il punto più lontano dalla linea di trend e confrontare tale distanza con una soglia (parametrica) fissata.
- Se la distanza è sopra la soglia accetta il punto e identifica due nuove linee di trend. Ritornare al passo 2 su ogni linea.

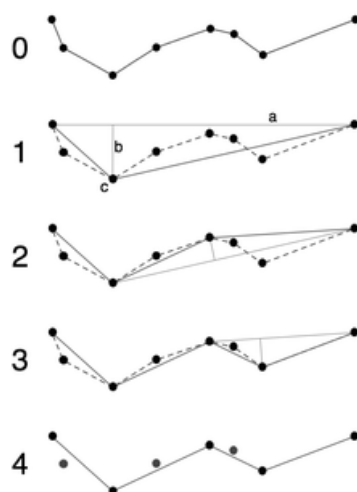


FIGURA 3.1: Esempio dell'algoritmo di Douglas-Peucker applicato ad una linea spezzata

Tale algoritmo può essere applicato anche su poligoni e in particolare al perimetro del poligono (nel caso di poligoni composti o poligoni con fori è necessario operare su ogni anello singolarmente); è veloce nella sua implementazione di base soprattutto su strutture con pochi vertici (un edificio tipico ha spesso meno di 20 vertici) ed è in grado di mantenere la forma più caratteristica della linea di ingresso.

Un problema di implementazione di base è che non è topologicamente sicuro: la semplificazione di una linea chiusa può infatti creare auto-intersezioni. Una versione topologica sicura esiste [Saalfeld 1999], ma ha un elevato costo computazionale.

Inoltre, l'utilizzo dell'algoritmo Douglas- Peucker su edifici o oggetti di forma rettangolare non è completamente raccomandabile in quanto tende ad eliminare gli angoli, rendendo gli angoli rimanenti molto acuti con la conseguente perdita della forma originale dell'edificio.

Per risolvere tale problema si passa direttamente alla fase successiva: eliminando alcuni segmenti del perimetro della geometria si eliminano anche i vertici associati. Vedremo in seguito come si effettuerà tutto ciò.

5. Lo **squaring degli angoli** delle geometrie. Questa è un'operazione molto pericolosa e che consente di approssimare gli angoli tra ogni due segmenti del perimetro. Questo procedimento consente di diminuire il numero di angoli presenti negli edifici della mappa e di avvicinare i restanti il più possibile ad un multiplo dell'angolo retto.
6. Il **displacement degli edifici** utile a eliminare eventuali sovrapposizioni o a evidenziare la distanza (poco visibile) tra gli edifici generalizzati e altre features presenti nella mappa (altri edifici, grafo stradale, grafo ferroviario, ecc ...).

3.2 Semplificazione delle geometrie

Di interesse per questa tesi sono i processi che portano alla rappresentazione semplificata degli edifici. Tali procedure dovrebbero essere in grado di agire sulla geometria operando una semplificazione adatta al contesto in cui l'oggetto stesso è inserito. Per esempio, se si sta lavorando con un edificio in un centro storico, una variazione troppo accentuata dell'area interna sicuramente comporterà sovrapposizioni di geometrie adiacenti, cosa che in aperta campagna molto probabilmente non si verifica .

I motivi che portano a dare importanza al processo di semplificazione sono molteplici sia a livello implementativo sia a livello della forma del prodotto finale. Per quanto riguarda la prima categoria vi sono due principali vantaggi che derivano direttamente dalla semplificazione delle geometrie:

- minor numero di dati da memorizzare con conseguente minore dimensione, in termini di memoria, del database;
- geometrie meno complesse con vantaggi a livello computazionale nelle successive operazioni di gestione e manipolazione.

A livello della forma finale del prodotto, geometrie troppo complesse e di difficile lettura, portano a valutare il risultato finale in maniera negativa.

Per aumentare la percezione di qualità del nostro prodotto si cerca di risolvere il problema della complessità delle geometrie degli edifici eliminando i segmenti (quindi anche i vertici) che, a causa della riduzione della scala, rendono la geometria troppo complessa o che non risultano caratterizzanti per la percezione del suo andamento sulla carta. Con l'eliminazione di alcuni vertici, si riduce anche l'occupazione in memoria.

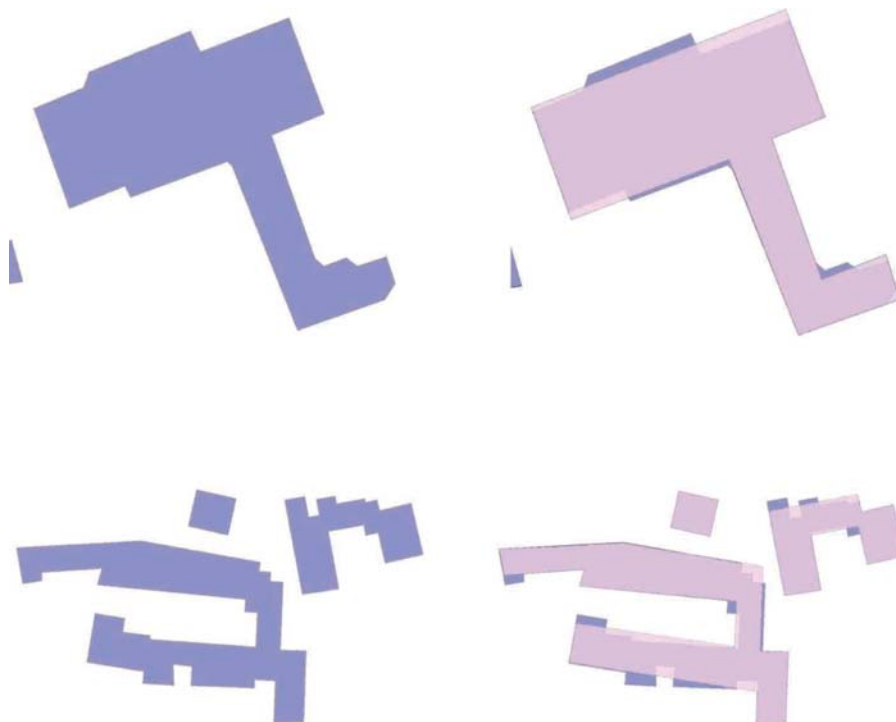


FIGURA 3.2: Esempi di semplificazione delle geometrie

La semplificazione dovrà riuscire ad avvicinarsi quanto più possibile al compromesso ottimo, ovvero eliminare il maggior numero di dati (vertici e segmenti) modificandone il meno possibile la dimensione e la forma. È inoltre necessario che le geometrie modificate non vadano a creare delle incongruenze all'interno della carta, andando a sovrapporsi, ad esempio, ad un altro edificio, ad una strada o a qualsiasi altre feature (vedi Figura 3.3).

3.3 Soluzione adottata da CARGEN

In una riduzione di scala, alcuni tratti del perimetro risultano superflui, in quanto nella rappresentazione finale risulterebbero difficilmente o per niente visibili.

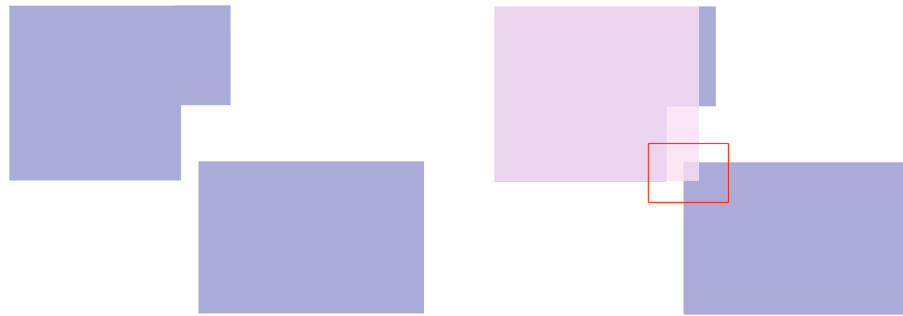


FIGURA 3.3: Esempio di semplificazione: caso frequente in centri abitati dove la semplificazione porta ad una sovrapposizione tra due edifici

Per effettuare questo passo di semplificazione si è deciso di partire dall'idea di soluzione presentata da Monika Sester nel 2000. Il suo algoritmo si basa sulla teoria Least Squares Adjustment (LSA) e sulla lunghezza delle facciate dell'edificio. Si definisce una lunghezza critica al di sotto della quale la facciata non risulta più caratterizzante e quindi soggetta a rimpiazzo o eliminazione. Le azioni di adeguamento vengono svolte consecutivamente e applicate alla geometria di volta in volta.

L'algoritmo lavora localmente in accordo con alcune semplici regole, dipendenti dalla geometria dei segmenti adiacenti a quello in esame. La decisione di sostituire o eliminare il segmento in esame dipende dalla geometria che esso forma con il suo precedente e successivo:

- **Inclusione/Esclusione:** l'angolo tra il segmento precedente e quello successivo è di circa 180° . L'azione prevista in questo caso è che il segmento più breve tra il precedente e il successivo venga sottratto a quello più lungo, di fatto eliminandolo assieme a quello in esame.
- **Offset:** l'angolo tra il segmento precedente e quello successivo è di circa 0° . L'azione intrapresa è simile al caso di 180° , il segmento più lungo tra il precedente e il successivo viene esteso e di conseguenza il più corto rimosso.
- **Corner:** l'angolo tra il segmento precedente e quello successivo è di circa 90° . Vengono prolungati il segmento precedente e quello successivo fino al punto di intersezione. Anche in questo caso il segmento in esame viene eliminato.

Questi tre casi vengono iterati fino a quando tutti i segmenti del perimetro della geometria non siano di dimensione maggiore alla lunghezza critica. Una volta conclusi i passi della ricorsione, la geometria viene considerata semplificata e quindi salvata.

I principali vincoli da mantenere durante la semplificazione degli edifici sono:

- conservazione degli angoli retti;
- conservazione della collinearità;
- conservazione del parallelismo.

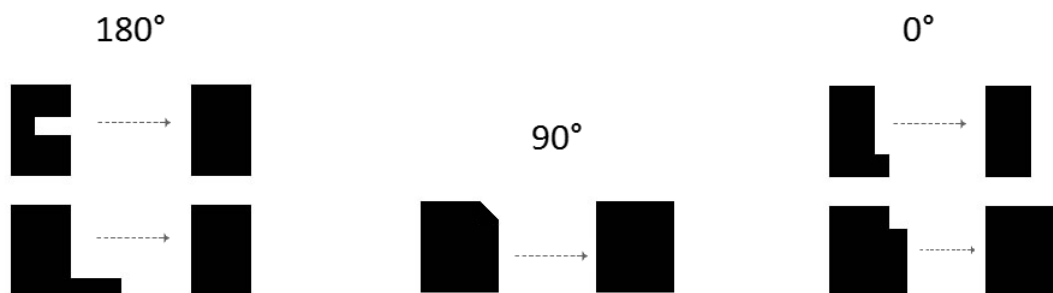


FIGURA 3.4: Trasformazione degli edifici applicando regole di Sester

3.3.1 Implementazione della soluzione

La prima cosa che è stata modificata è l'ampliamento della gamma di angoli trattati: nell'attuazione originale, l'algoritmo semplifica solo gli edifici con soli angoli di 0° , 90° e 180° mentre nei nostri dati di input alcuni lati dell'edificio sono collegati tra loro con angoli che non sono trattati nella realizzazione originale. Perciò si sono introdotti dei range di $\pm 15^\circ$ attorno a queste tre angolature principali: gli angoli interni a questi range vengono trattati secondo le regole del caso cui fanno capo.

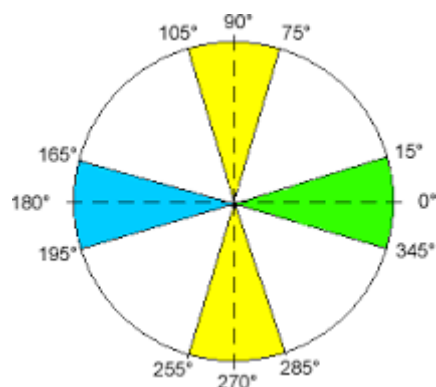


FIGURA 3.5: Ampliamento del range degli angoli nell'algoritmo di Sester

Con riferimento all'ampliamento del range presente nella Figura 3.5 vediamo ora quali sono i passi della soluzione adottata da CARGEN.

1. Per un edificio vengono individuati tutti i vertici ed i segmenti che lo compongono.
2. I segmenti vengono ordinati in base alla lunghezza, dal più corto al più lungo.
3. Viene esaminato il segmento più corto:
 - se è più lungo della lunghezza critica, la geometria non può essere semplificata e l'algoritmo termina;
 - se è più corto allora si comincia ad analizzare la geometria.
4. Viene calcolato l'angolo formato dal segmento precedente e quello successivo rispetto al segmento in esame. Nonostante l'approccio ampliato dell'algoritmo, si è scoperto che esistono alcuni casi che l'algoritmo originale non può gestire:
 - presenza di spike ovvero sporgenze o rientranze che non rientrano nei range delle angolature principali;

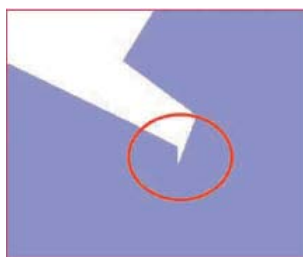


FIGURA 3.6: Spike non riconosciuto dall'algoritmo di Sester

- nei casi di angoli compresi nei range di 0° e 180° la variazione di area potrebbe essere troppo consistente e pertanto non accettabile in quanto l'edificio semplificato potrebbe avere un'area molto maggiore o inferiore rispetto all'area dell'edificio originale;

Per la gestione di questi problemi sono stati aggiunti quindi dei casi alle regole originali dell'algoritmo:

- nel caso di spike se i prolungamenti del segmento precedente e del segmento successivo si intersecano, il caso viene ricondotto a 90° ;
 - se i segmenti adiacenti formano un angolo nel range di 0° , la geometria viene modificata in maniera da ridurre al minimo la differenza di area;
 - se i segmenti adiacenti formano un angolo nel range di 180° , viene calcolata la differenza di area e, in caso sia troppo elevata, non vengono effettuate modifiche
5. In base all'angolo calcolato, tenendo sempre in considerazione la Figura 3.5, ci si riporta al caso adeguato, scegliendo tra 0° , 90° e 180° .

3.3.2 Casistiche dell'algoritmo

Ora che abbiamo capito la logica su cui si basa la soluzione adottata da CARGEN, vediamo nel dettaglio come vengono implementate le tre casistiche.

Consideriamo un generico edificio in cui P e C sono i vertici del segmento precedente, C e S i vertici di quello in esame e S e D i vertici del successivo. I tre segmenti saranno quindi \overline{PC} (precedente), \overline{CS} (in esame), \overline{SD} (successivo). Aggiungiamo inoltre il precedente del precedente $\overline{PC-1}$ e il successivo del successivo $\overline{SD+1}$

Caso 0°

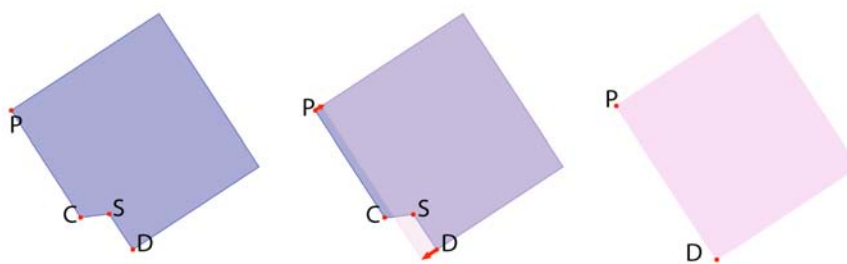


FIGURA 3.7: Caso 0° di Sester

Per migliorare il risultato finale si tenta di preservare il più possibile l'area iniziale. Per fare ciò :

- viene verificato quale tra precedente e successivo sia il più lungo (da ora in avanti \overline{L});
- viene calcolata un'approssimazione dell'area che verrebbe aggiunta/esclusa:

$$Area_{\text{approssimata}} = \overline{L} * \overline{PC} \quad (3.1)$$

- si approssima un rettangolo costruito sulla somma dei lati \overline{PC} e \overline{SD} . Di questo rettangolo se ne calcola l'altezza h :

$$h = \frac{Area_{\text{approssimata}}}{\overline{PC} + \overline{SD}} \quad (3.2)$$

- h viene aggiunta o sottratta a $\overline{PC-1}$ e $\overline{SD+1}$. Tra tutti i possibili casi si sceglie quello che meglio preservi l'area originale;
- i vertici C e S andranno eliminati, mentre P e D saranno spostati al nuovo valore e faranno parte della nuova geometria.

Caso 90°

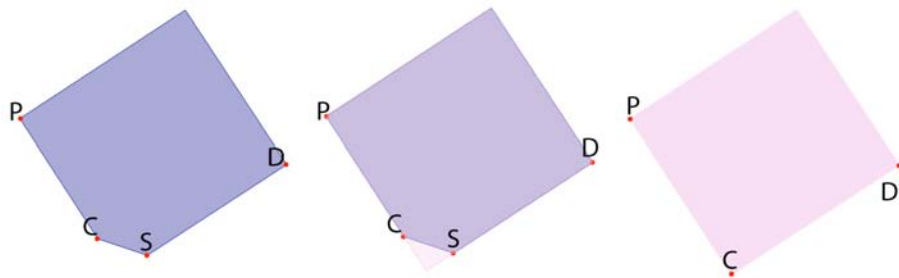


FIGURA 3.8: Caso 90° di Sester

In questo caso l'algoritmo segue le specifiche originali.

Caso 180°

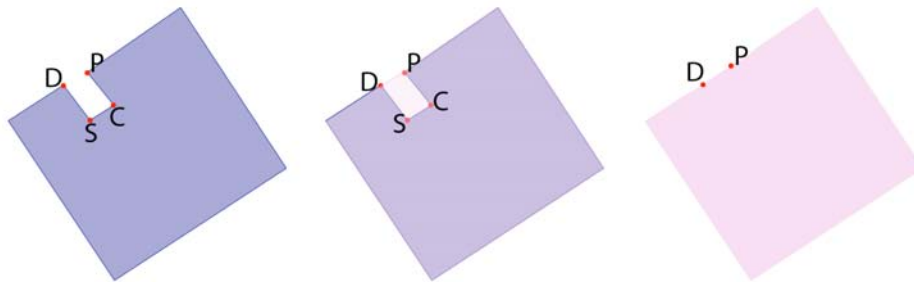


FIGURA 3.9: Caso 180° di Sester

- Viene calcolata l'area che andrebbe aggiunta o esclusa. Se quest'area supera un valore accettabile, l'algoritmo non opera nessun cambiamento.
- Viene verificato quale tra \overline{PC} e \overline{SD} è il più breve:
 - se \overline{PC} risulta più breve, $\overline{PC} - \epsilon$ viene prolungato fino ad intersecare \overline{SD} ;
 - se \overline{SD} risulta più breve, $\overline{SD} + \epsilon$ viene prolungato fino ad intersecare \overline{PC} ;
- I vertici C e S vengono cancellati

3.3.3 Risultati ottenuti, limiti e ipotesi di miglioramento

Applicando l'algoritmo appena descritto, si scopre che esso va bene solo per singoli edifici (Figura 3.10).

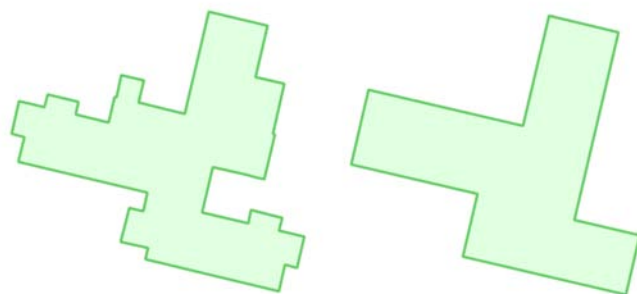


FIGURA 3.10: Esempio di una corretta semplificazione di un singolo edificio

In particolare il problema topologico della sovrapposizione non viene risolto, in quanto si analizza una sola geometria per ogni passo; nel caso di edifici adiacenti non viene considerato l'intero blocco di edifici, comprendente tutte le geometrie da processare, ma ogni edificio viene processato singolarmente ed in maniera indipendente dagli altri. Il risultato di tale processo è la sovrapposizione degli edifici come riportato in Figura 3.11.

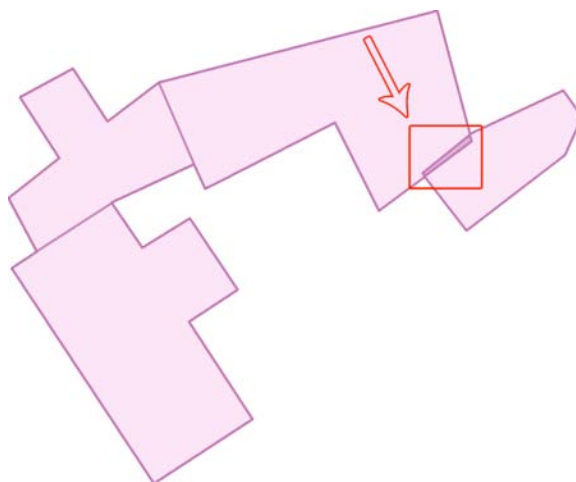


FIGURA 3.11: Esempio di errore topologico dovuto alla sovrapposizione di edifici

Un tentativo che è stato provato da Rossella Baldin per superare tale problema è stato effettuare una sorta di fusione degli edifici adiacenti: dati tutti gli edifici adiacenti è possibile vederli come un unico edificio. Tale edificio, successivamente, può essere processato utilizzando l'algoritmo di Sester. Il problema topologico tra edifici adiacenti

non si presentà più, ma si presenta invece il problema della sovrapposizione tra edifici indipendenti tra loro, cioè che non condividono alcun lato. Dalla Figura 3.12, possiamo notare che, oltre alla sovrapposizione, il prodotto finale lascia un pò a desiderare: la semplificazione è avvenuta, ma la forma dell'edificio originale e la sua dimensione si sono modificate notevolmente e ciò può essere visto come un fatto negativo.

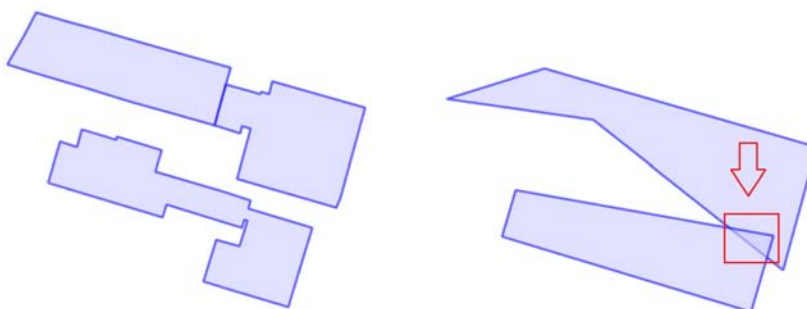


FIGURA 3.12: Errore topologico, dovuto alla sovrapposizione di edifici, riscontrato nella soluzione provata da Rossella Baldin. Inizialmente sono presenti tre edifici (due adiacenti e uno singolo), ma dopo la semplificazione ce ne sono solo due.

Tale approccio non è ideologicamente corretto in quanto particolari caratteristiche degli edifici (altezza, tipologia, ecc. . .) li avevano resi non fondibili prima della semplificazione e pertanto alla fine della semplificazione dovranno tornare ad essere entità distinte.

L'ipotesi di miglioramento proposta consiste nella creazione di una struttura che riesca a mantenere il patrimonio informativo delle geometrie originali, riesca a fornire il perimetro esterno e consenta di far propagare le modifiche a tutte le geometrie che ne hanno bisogno. La struttura, indipendentemente dalla presenza o meno delle adiacenze, non cambierà. Questa soluzione, che prenderà il nome di "Cargen-Edifici-Topologici", ha lo scopo fondamentale di eseguire la semplificazione degli edificati non violando i vincoli topologici.

Capitolo 4

Cargen-Edifici-Topologici

In questo capitolo verrà presentata la soluzione che permette di risolvere le problematiche descritte nell'ultimo paragrafo del capitolo precedente. Questa soluzione, che prende il nome di "Cargen-Edifici-Topologici", eliminerà il concetto di errore topologico dovuto alla sovrapposizione o alla disgiunzione di edifici che, prima del processo di semplificazione, erano adiacenti tra loro; verrà descritto l'approccio utilizzato e la struttura, su cui si basa l'algoritmo, capace di mantenere il patrimonio informativo delle geometrie originali. Successivamente verrà esposto nel dettaglio la soluzione implementata, verranno fornite alcune immagini relative ai risultati ottenuti ed infine, nell'ultimo capitolo, si effettuerà un'analisi riguardo svantaggi, vantaggi e possibili sviluppi futuri della soluzione realizzata.

4.1 Strumenti utilizzati

JTS Topology Suite

La JTS Topology Suite è una libreria open source, scritta interamente in linguaggio Java, che fornisce una modellazione ad oggetti per le geometrie lineari in uno spazio euclideo. In questa libreria sono definite tre geometrie fondamentali, Point, LineString e Polygon, che rappresentano rispettivamente la geometria puntuale, lineare e areale. La JTS mette a disposizione numerose funzioni geometriche, tra le quali è importante citare:

1. operatori topologici che realizzano le funzioni di intersezione, differenza, unione;
2. creazione di buffer (sia positivo, che negativo) intorno alle geometrie;
3. costruzione dell'involuppo convesso di geometrie;

4. semplificazione delle geometrie, ad esempio mediante l'algoritmo di Douglas-Peucker;
5. costruzione del Minimum Bounding Box;
6. triangolazione.

Oltre a queste funzioni, la JTS fornisce l'implementazione di indici spaziali, come il quadTree ed il KdTree, che offrono un modo veloce per la risoluzione di query spaziali. La versione utilizzata nell'ambito di questa tesi è la JTS 1.13.

OpenJump

OpenJump è un Desktop GIS open source che permette di visualizzare, modificare e interrogare dati spaziali. È scritto in linguaggio Java, si basa sulla JTS ed è in grado di gestire file raster, vettoriali e database (PostGis, Oracle, ArcSDE). Una caratteristica peculiare è la sua architettura modulare, che permette di estendere di molto le funzionalità di base, potendo integrare, per esempio, il proprio codice mediante la realizzazione di uno o più plugin.

In OpenJump, la creazione di un plugin, relativo al proprio codice, offre al programmatore il grosso vantaggio di poter visionare tramite l'interfaccia grafica gli effetti della propria applicazione. Il plugin diventa così, uno strumento essenziale nello sviluppo di nuovi algoritmi che manipolano geometrie e permette di avere una risposta grafica e istantanea, garantendo un grande aiuto nella fase di testing e debug.

La modalità con cui OpenJump gestisce le feature (oggetti astratti dal mondo reale) si basa sull'utilizzo dei layer (livelli) che svolgono il ruolo di contenitori. Ogni layer è in grado di contenere le feature relative ad uno specifico schema dati, chiamato FeatureSchema. Quest'ultimo, specifica il nome e la tipologia degli attributi che costituiscono la feature, similmente a quanto accade nelle tabelle dei database. Un layer rappresenta, quindi, una vera e propria tabella, il cui schema dati è specificato dal FeatureSchema. Il layer è interrogabile per mezzo di query, che possono essere sia spaziali che non spaziali. Per migliorare le query spaziali, è possibile associare al layer uno degli indici spaziali forniti dalla JTS.

La versione di OpenJump utilizzata nello svolgimento di questa tesi è la 1.6.0, rilasciata nell'aprile 2013.

Eclipse

Allo scopo di creare un plugin, il programmatore necessita di un editor per la scrittura di codice sorgente, un compilatore (interprete) che traduce (compila) una serie di istruzioni

presenti nel codice sorgente, scritte in un linguaggio di programmazione, in istruzioni di un altro linguaggio (assembly o linguaggio macchina), un linker che prende in input il file oggetto prodotto dal compilatore e crea il file binario comprensibile al computer che può essere eseguito, un loader che si occupa di far partire il programma e un debugger che consente di individuare errori durante la scrittura del codice sorgente.

Tutte queste funzionalità possono essere raggruppate in un unico ambiente di sviluppo detto IDE (Integrated Development Environment). In particolare usando la JTS si è optato per l'IDE Eclipse in quanto è l'ambiente di sviluppo più noto e usato in ambito Java (ma anche per molti altri linguaggi). Per sviluppare il plugin di OpenJump relativo alla tesi, utilizzando Eclipse, è stato necessario configurare l'IDE ¹.

4.2 Approccio

L'attuale algoritmo di semplificazione analizza una geometria alla volta. Come fare ad inserire la gestione del contesto ed avere un incremento della complessità computazionale accettabile? Continuare a gestire geometrie singole e per ognuna verificare che non ci siano problemi nel contesto? Assolutamente no in quanto decisamente costoso.

Analizzare geometrie adiacenti come entità distinte non è assolutamente sufficiente: molto meglio considerarle come un'unica entità. Si potrebbe affermare che in questo modo si lavori sulla geometria e sul suo contesto in contemporanea.

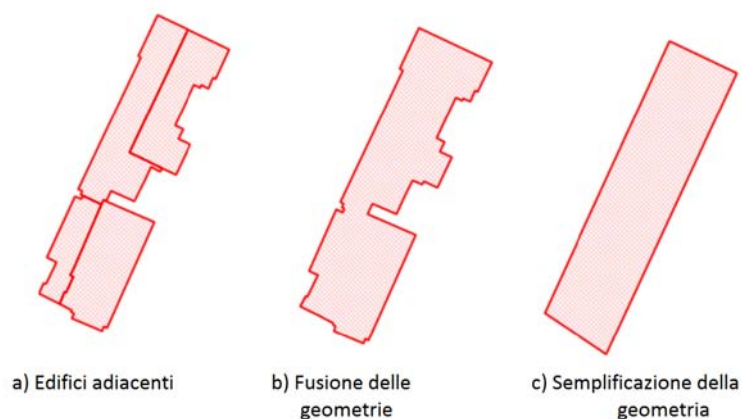


FIGURA 4.1: Fusione applicata precedentemente al processo di semplificazione

Mentre la fusione applicata precedentemente alla semplificazione delle geometrie è un processo irreversibile, la “fusione” degli edifici adiacenti deve essere assolutamente reversibile. Questo perché le loro particolari caratteristiche li avevano resi non fondibili

¹<https://milkator.wordpress.com/2012/11/08/develop-own-openjump-plugin-with-eclipse>

prima della semplificazione e pertanto alla fine della semplificazione dovranno tornare ad essere entità distinte.

Per rendere reversibile la nostra “fusione” sono stati pensati due approcci diversi:

- approccio “*a posteriori*”: si mette in atto la fusione delle geometrie adiacenti. Il ring esterno viene dato in input all’algoritmo di semplificazione che lo elabora seguendo i canoni che utilizzerebbe per un semplice edificio. Dal perimetro semplificato si tenta di ricostruire la divisione originale degli edifici (Figura 4.1)
- approccio “*topologico*”: non si opera una vera e propria fusione delle geometrie. L’algoritmo si appoggia ad una struttura che dà la possibilità di ricavare il perimetro esterno (sul quale avverrà la semplificazione) ma nel contempo consente di seguire tutte le modifiche e rifletterle su tutta la parte che ha bisogno di essere modificata.

L’approccio “*a posteriori*” non è applicabile in quanto una volta ottenuta l’entità semplificata non è possibile tornare indietro e riavere le singole geometrie.

L’approccio “*topologico*” è più complicato da realizzare, ma è l’unico che permetta di risolvere le problematiche dovute alla sovrapposizioni di edifici e alla suddivisione originale delle geometrie. Come accennato precedentemente, è necessaria una struttura di appoggio che riesca a mantenere il patrimonio informativo delle geometrie originali, riesca a fornire il perimetro esterno e consenta di far propagare le modifiche a tutte le geometrie che ne hanno bisogno. La struttura adottata è un grafo planare orientato che è composta sostanzialmente da due componenti:

- **Nodo**: è un vertice della geometria da cui escono un minimo di tre edge. Questo ha importanti proprietà e quindi deve avere un trattamento diverso rispetto ad un semplice vertice.
- **Edge**: composto da un nodo di partenza e uno di arrivo (che gli conferiscono quindi un orientamento) e un insieme di vertici che costituiscono la geometria vera e propria dell’edge. Se un edge appartiene al perimetro allora viene chiamato “edge perimetrale”, altrimenti “edge interno”. A sua volta un edge può essere composto da:
 - **Segmento**: parte di edge delimitata da due vertici; nel caso in cui l’edge abbia solo il nodo di partenza e il nodo di arrivo l’edge coincide con il segmento.
 - **Muro**: è il classico muro di un edificio che va da ginocchio a ginocchio e può contenere vari segmenti e vari nodi; con questo termine si fa sempre riferimento al perimetro dell’edificio.

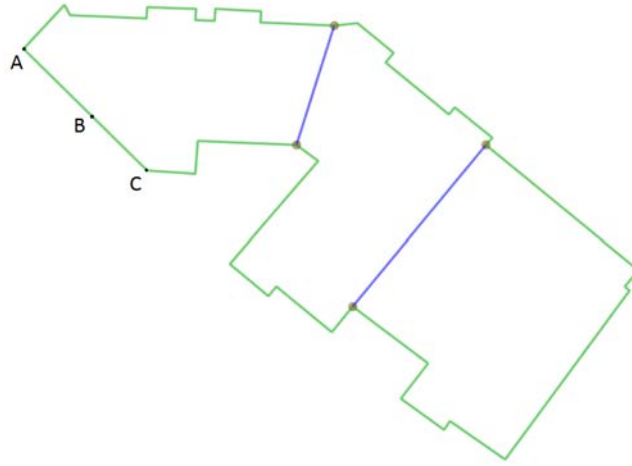


FIGURA 4.2: Esempio di grafo planare orientato con edge, nodi, segmenti e muri

Dalla Figura 4.2 è possibile vedere i nodi (raffigurati con il cerchio marrone) da cui partono almeno 3 edge: 2 edge perimetrali di colore verde e 1 edge interno di colore blu. La parte di perimetro delimitata dalle lettere A e C rappresenta il muro, mentre \overline{AB} e \overline{BC} costituiscono alcuni dei segmenti dell'edge.

Per capire come gli elementi appena descritti si relazionano tra loro può essere utile il diagramma in Figura 4.3.

La geometria che viene processata dall'algoritmo di semplificazione è l'unione degli edge perimetrali del grafo. Restando distinti i nodi dai semplici vertici, l'approccio della semplificazione varia a seconda che si sia in un caso o nell'altro. I dettagli del trattamento verranno discussi nei prossimi paragrafi, ma è bene fare una precisazione. Se ad essere modificato è un vertice, l'algoritmo potrà agire seguendo le casistiche definite dall'approccio originale. Diversamente se nell'analisi è presente un nodo, questo non potrà sicuramente essere cancellato e la variazione di posizione dovrà riflettersi in tutti gli edge (perimetrali ed interni) che poggiano su di lui.

Una volta che si è semplificata la geometria esterna e le relative modifiche sono state propagate, si è passati alla semplificazione degli edge interni in modo da completare l'intero processo di semplificazione del gruppo di edifici adiacenti.

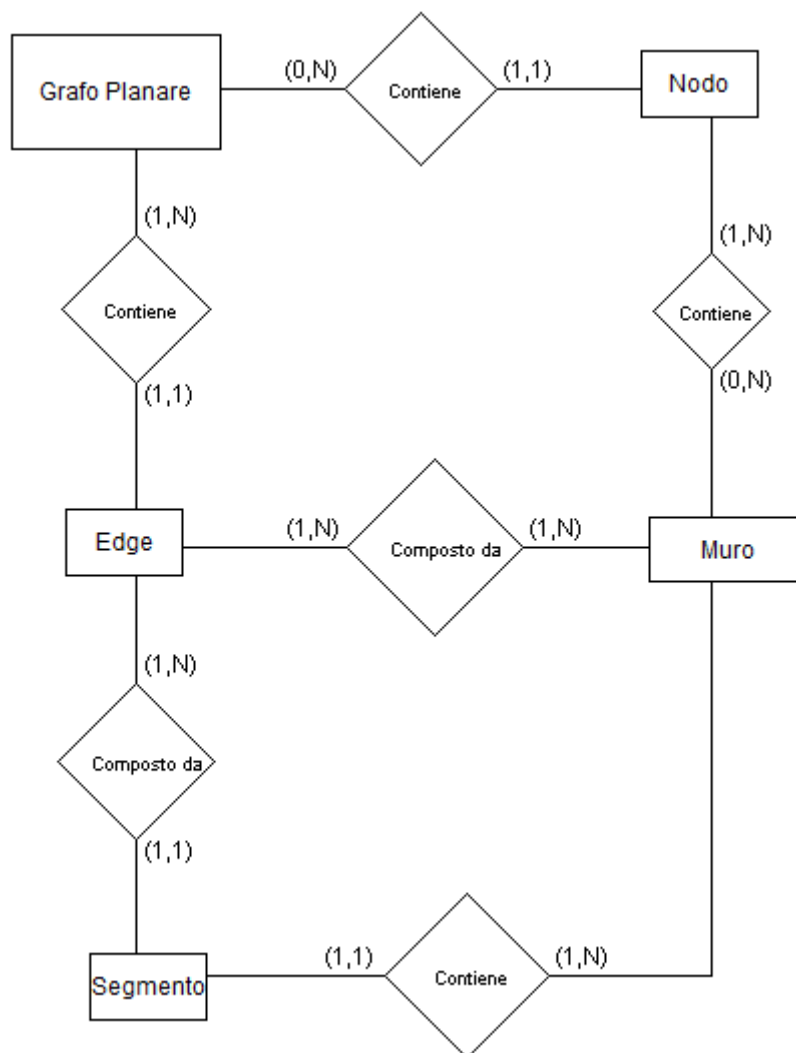


FIGURA 4.3: Diagramma E-R

4.3 Implementazione della soluzione

La soluzione ricorsiva proposta non varierà a seconda degli oggetti scelti: indipendentemente dal fatto che si voglia semplificare un gruppo di edifici oppure il singolo edificio, l'algoritmo proposto applicherà Sester agli edge perimetrali presenti. Una novità rispetto agli algoritmi sviluppati fino a questo momento è la possibilità di selezionare quante e quali geometrie semplificare: l'utente, tramite il puntatore del mouse, deciderà quali edifici dovranno essere processati. Inoltre, per dare massima libertà di utilizzo, si avrà a disposizione una shell in cui si potranno inserire:

- Sester lunghezza critica: lunghezza al di sotto della quale il muro può essere eliminato/rimpiazzato. Possono essere inseriti numeri interi o numeri razionali positivi.

- Sester range angoli: non essendo garantita la presenza esclusivamente di angoli 0° , 90° e 180° , vengono considerati range di angoli. Possono essere inseriti solo numeri naturali.
- Sester min area: parametro utilizzato nella casistica 180° per la valutazione dell'area che viene aggiunta o rimossa. Possono essere inseriti numeri interi o numeri razionali positivi.
- Sester min dist: parametro utilizzato nella casistica 90° per la valutazione della distanza tra il muro che deve essere eliminato/rimpiazzato e la coordinata data dall'intersezione tra i muri adiacenti al muro sotto la lunghezza critica. Possono essere inseriti numeri interi o numeri razionali positivi.
- Simplify edge inside: spunta che indica se l'utente vuole procedere con la semplificazione degli edge interni. Se è presente avviene la semplificazione, altrimenti no.

Tutti questi parametri vengono inseriti dall'utente quando inizia l'esecuzione dell'algoritmo.

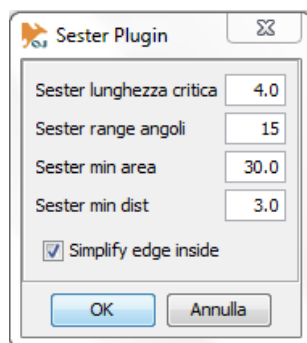


FIGURA 4.4: Parametri algoritmo proposto

4.3.1 Passi dell'algoritmo

Verrano ora descritti i passi dell'algoritmo proposto:

1. Costruzione del grafo planare in base alle geometrie selezionate dall'utente.
2. Trovo tutti i ring dell'edificio (interni e perimetro) e li ordino (in modo che gli edge che li compongono siano consecutivi e stessa orientazione).
3. Di tutti i ring trovati si cerca il perimetro da cui ricavo i muri.
4. I muri vengono ordinati in base alla lunghezza, dal più corto al più lungo.

5. Considero il numero di muri :
 - Se ho tre muri (un triangolo) non lo semplifico e si passa al punto 10;
 - Se ho più di tre muri trovo il muro più corto.
6. Viene esaminato il muro più corto:
 - Se è più lungo della lunghezza critica, la geometria non può essere semplificata e si passa al punto 10;
 - Se è più corto allora si comincia ad analizzare i muri del perimetro.
7. Trovo i muri adiacenti al muro più corto. Viene calcolato l'angolo formato tra essi. Ovviamente la gestione dei range attorno alle misure principali resta inalterata, come del resto la gestione di tutti gli angoli non compresi in quella casistica.
8. In base all'angolo calcolato e al range inserito dall'utente ci si riporta al caso adeguato, scegliendo tra 0° , 90° e 180° . Se l'angolo non è coperto da nessuna casistica si ripartirà dal punto 6 dove si passerà ad esaminare il più corto tra tutti i restanti muri. Ricordiamo che in caso di spike i muri verranno gestiti secondo la casistica 90° .
9. Si ritorna al punto 4.
10. Se sono presenti edge interni e l'utente inserisce la spunta descritta precedentemente si passa alla semplificazione degli edge interni. Successivamente l'algoritmo termina.

4.3.2 Costruzione del grafo planare

Il processo di costruzione del grafo planare si divide in due fasi:

- **Creazione degli edge:** vengono isolate le singole linee che compongono le geometrie in input. Successivamente si creano gli edge effettuando l'unione delle linee in base al numero di vertici incontrati: se i vertici sono di grado 2 faranno parte dello stesso edge, altrimenti termino la creazione dell'edge. Di ogni edge vengono impostati *leftID* e *rightID* che indicano l'identificatore (ID) dei due edifici su cui poggia: se -1 allora non tocca nessun edificio, altrimenti l'ID della geometria toccata. Ciò implica la conoscenza degli edge perimetrali ($leftId = -1$ o $rightId = -1$) e degli edge interni ($leftId \neq -1$ e $rightId \neq -1$).
- **Creazione dei nodi:** una volta creati gli edge, i nodi coincideranno con i loro estremi. In particolare se ho un edificio singolo non ci saranno edge interni ed il

nodo verrà rimosso, altrimenti per ogni nodo vengono memorizzati tutti gli edge che lo toccano. In questo modo ogni modifica della sua posizione verrà propagata agli edge interessati.

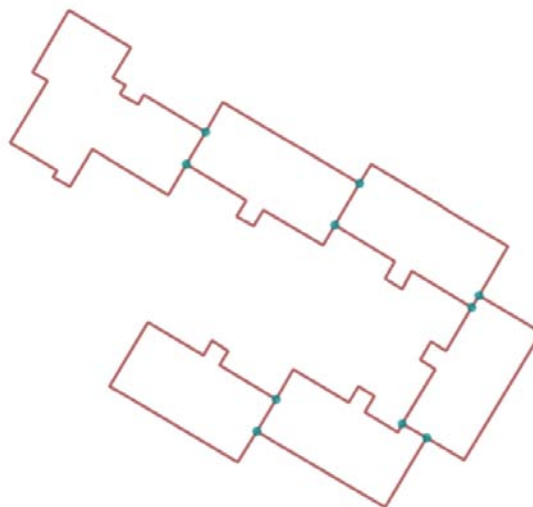


FIGURA 4.5: Semplice esempio di grafo planare: in rosso sono presenti gli edge, mentre i cerchi azzurri rappresentano i nodi

Il grafo, quindi, possiede tutto il patrimonio informativo necessario per poter ripristinare la suddivisione originale degli edifici selezionati. Inoltre, lavorando comunque con un'astrazione della realtà, ci sono anche altri vantaggi:

- ogni edge è presente singolarmente, mentre in due geometrie adiacenti il lato di contatto è presente due volte;
- il processo di semplificazione risulta più semplice.

La realizzazione del grafo planare è il primo passo fondamentale verso la realizzazione del nostro algoritmo. Grazie al fatto di poter distinguere edge perimetrali ed edge interni, la creazione dei muri risulterà più semplice. Essi, insieme ai nodi, saranno gli elementi principali su cui si fondano le casistiche che andremo a descrivere di seguito.

4.3.3 Casistiche dell'algoritmo

Nel funzionamento dell'algoritmo le casistiche 0° , 90° e 180° rappresentano la parte più importante e più significativa dell'intero progetto. Per ogni casistica verrà descritto come si ottiene la semplificazione dei muri di interesse e come i nodi appartenenti ai muri verranno gestiti. Per quest'ultima fase, si nomineranno alcune funzioni il cui scopo è la conservazione della geometria originale degli edge interni con alcuni piccoli aggiustamenti dovuti al caso.

4.3.3.1 Caso 0°

Consideriamo il caso di Figura 4.6 in cui il muro \overline{BC} è più corto della lunghezza critica mentre \overline{AB} e \overline{CD} sono i muri adiacenti. Saranno spiegate ora le procedure che portano alla semplificazione dei muri e la gestione dei nodi.

Semplificazione dei muri

1. Dati i muri \overline{AB} e \overline{CD} indichiamo con \overline{L} la lunghezza del muro più lungo, mentre con \overline{l} la lunghezza del muro più corto.

2. Viene calcolato il punto H che giace sul muro \overline{BC}

$$H = \frac{\overline{L}}{\overline{L} + \overline{l}} \quad (4.1)$$

3. Viene creata una semiretta di angolazione BA che parte da H: l'intersezione tra essa ed il muro che precede/succede \overline{AB} sarà la coordinata E.

4. Viene creata una semiretta di angolazione CD che parte da H: l'intersezione tra essa ed il muro che precede/succede \overline{CD} sarà la coordinata F.

5. Aggiorno le coordinate dei muri che precedono/succedono \overline{AB} e \overline{CD} in base alle nuove coordinate E e F. Potrei avere un'estensione o una riduzione di tali muri.

6. I muri \overline{AB} , \overline{BC} , \overline{CD} vengono rimpiazzati dal nuovo muro \overline{EF} .

7. Controllo che \overline{EF} non intersechi altri muri al di fuori del suo precedente e successivo: se è così procedo con la gestione dei nodi altrimenti esco e termino la casistica 0° .

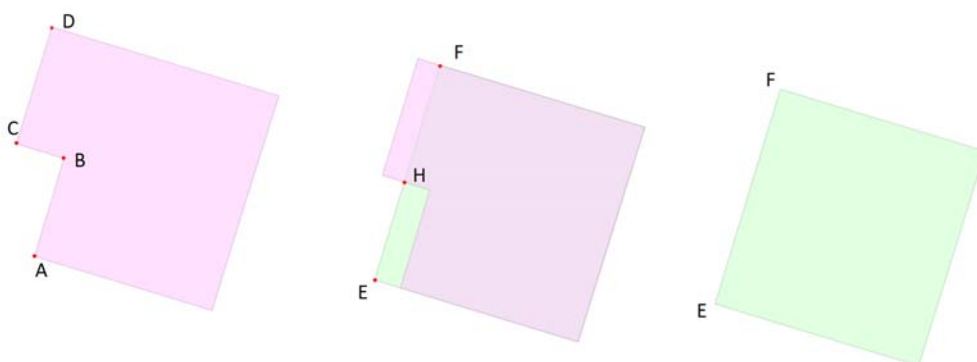


FIGURA 4.6: Esempio di semplificazione muri nel caso 0°

Per migliorare il risultato finale, si tenta di preservare il più possibile l'area iniziale. Dall'esempio di Figura 4.6 si nota che :

- l'area (definita dai vertici H,C,D,F) viene sottratta alla geometria.
- l'area (definita dai vertici H,B,A,E) viene aggiunta alla geometria.
- la preservazione dell'area dell'edificio dipende dalla coordinata H.

Gestione dei nodi

La gestione dei nodi non avviene per tutti i muri dell'edificio, ma solo per quelli che sono direttamente coinvolti nel processo di semplificazione. Guardando la Figura 4.6 bisogna prestare attenzione ai muri \overline{AB} , \overline{BC} e \overline{CD} , alla parte di muro compresa tra le coordinate D e F e alla parte di muro compresa tra le coordinate A ed E. Per queste ultime due sia che vengono aggiunte, sia che vengono rimosse, ogni nodo presente dovrà essere gestito altrimenti la semplificazione potrebbe non risultare corretta.

Le funzioni che saranno utilizzate sono:

- *NodiMuroEliminatoPP_SS()* che esegue i seguenti passi:
 1. calcolo intersezione tra il muro \overline{EF} e il muro in esame. Definisco questa coordinata G;
 2. per ogni nodo presente nel muro in esame guardo quanti edge interni lo toccano:
 - un edge interno :
 - * se edge tocca \overline{EF} aggiorno il nodo a G ed elimino tutti i vertici dell'edge che sono esterni al perimetro della geometria;
 - * se edge non tocca \overline{EF} aggiorno il nodo alla nuova posizione G e il secondo/penultimo vertice nel caso in cui il secondo/penultimo segmento dell'edge interno sia parallelo \overline{EF} così da mantenere il parallelismo.
 - almeno 2 edge interni :
 - * se esiste almeno un edge che interseca \overline{EF} prendo il primo edge calcolo l'intersezione e aggiorno gli edge interessati a quella intersezione;
 - * se nessun edge che interseca \overline{EF} creo un nuovo nodo in G e creo un nuovo edge che va da G alla posizione del nodo.
- *gestisciEdgeMuroTouchRes()* che realizza i seguenti passi:

1. Creo nodo in intersect (coordinata passata come parametro della funzione) se non è già presente nella lista dei nodi della geometria.
2. Ordino i nodi in base alla distanza tra nodo e la coordinata data dall'intersezione tra il muro in esame ed un muro che viene passato come parametro della funzione.
3. Se il primo nodo ha 1 edge interno estendo edge fino al secondo nodo, aggiungo tale edge come edge interno del secondo nodo ed elimino primo nodo.
4. Creo edge interno per ogni coppia di nodi consecutivi.

Questa funzione viene utilizzata solo nel caso in cui ci siano almeno due nodi appartenenti al muro in esame.

- *multipleEdgeP_S()* che compie i seguenti passi:

1. se non esiste l'intersezione tra \overline{EF} e il primo/ultimo segmento dell'edge creo un nuovo nodo come proiezione su \overline{EF} del nodo in esame e creo l'edge che li congiunge.
2. se esiste l'intersezione tra \overline{EF} e il primo/ultimo segmento dell'edge aggiorno tutti gli edge alla nuova coordinata calcolata come la proiezione su \overline{EF} del nodo in esame.

Il primo passo che viene svolto è l'eliminazione di tutti gli edge interni che sono completamente contenuti nelle aree definite dai vertici H,C,D,F e dai vertici H,B,A,E.

Vediamo ora in base ai muri specificati come si effettua la gestione dei nodi che vi appartengono. Indicheremo con il termine muro in esame il muro la cui lunghezza è inferiore alla lunghezza critica e muro adiacente il suo precedente/successivo.

Parte di muro che precede/succede il muro adiacente

1. Calcolo il numero di nodi presenti nella parte di muro che precede/succede il muro adiacente:
 - se ho un solo nodo richiamo la funzione `NodiMuroEliminatoPP_SS()`;
 - se ho almeno due nodi:
 - se esiste un edge interno del nodo che interseca \overline{EF} richiamo la funzione `NodiMuroEliminatoPP_SS()`;
 - se non esiste un edge interno del nodo che interseca \overline{EF} richiamo la funzione `gestisciEdgeMuroTouchRes()`;

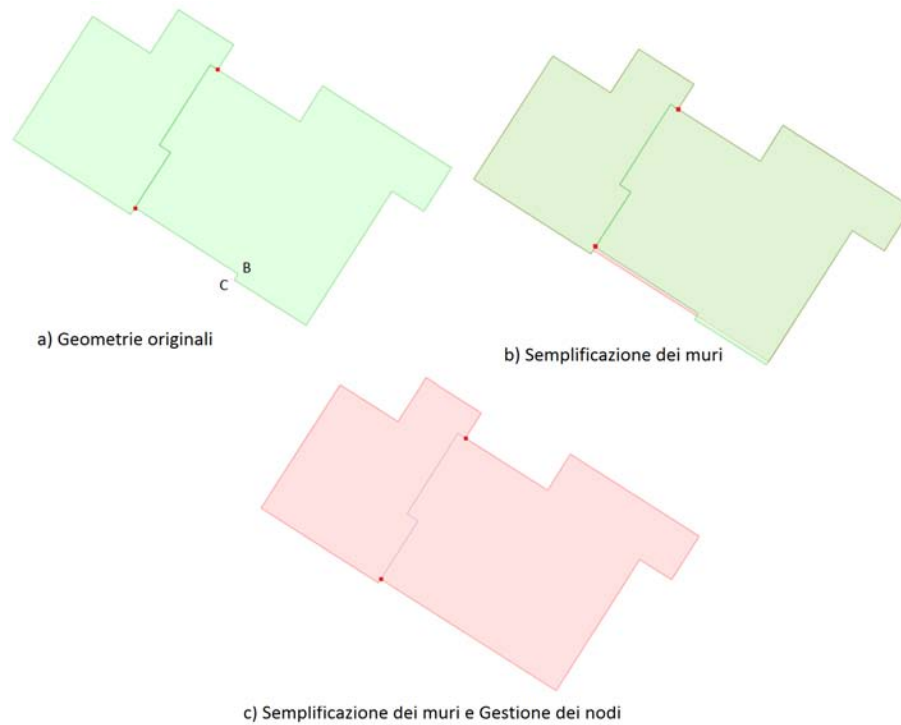


FIGURA 4.7: Semplificazione muri e gestione dei nodi della parte di muro che precede/succede il muro adiacente

Nella Figura 4.7 inizialmente troviamo un gruppo di edifici da semplificare (a); una volta che si è individuato il muro in esame \overline{BC} e si è calcolato l'angolo formato tra i muri adiacenti (0° in questo caso) si passa alla semplificazione dei muri (b). Notiamo che il nodo appartenente alla parte di muro che precede/succede il muro adiacente non viene toccato. Rimanendo in questa posizione si avrebbe un errore della semplificazione in quanto il nodo deve appartenere ad un muro. Applicando però la gestione dei nodi descritta precedentemente il nodo viene spostato in una nuova posizione consona alle regole descritte e senza alcun errore di semplificazione (c).

Muri adiacenti

Per ogni nodo presente nel muro adiacente al muro \overline{BC} guardo il numero di edge interni che lo toccano:

- un edge interno:
 - se l'edge non interseca il muro \overline{EF} lo estendo fino alla coordinata data dalla loro intersezione e aggiorno il nodo alla nuova posizione;
 - se l'edge interseca il muro \overline{EF} aggiorno il nodo all'intersezione ed elimino eventuali coordinate esterne alla geometria;

- almeno due edge interni: elimino le coordinate esterne al poligono, prendo il primo edge e calcolo la sua intersezione con \overline{EF} ; richiamo la funzione `multipleEdgeP_S()`.

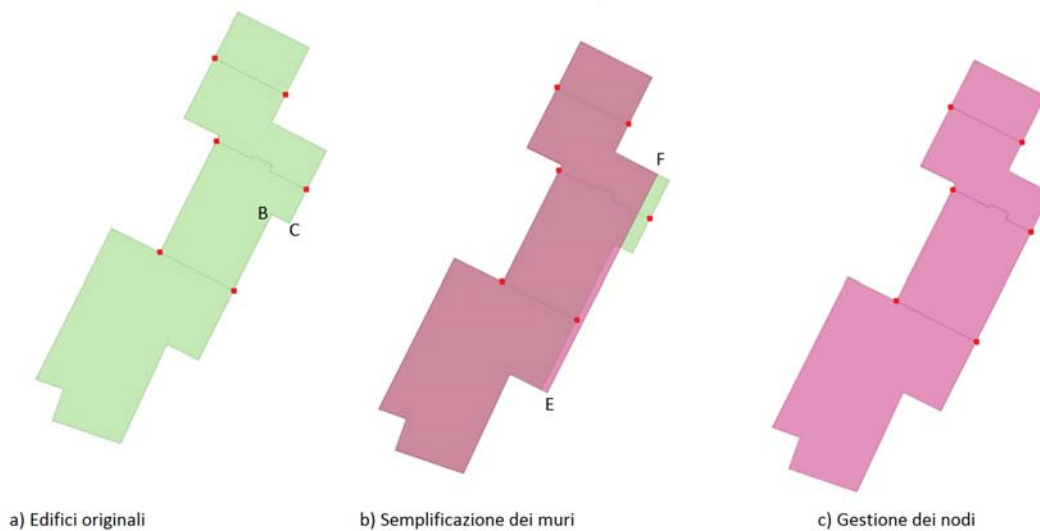
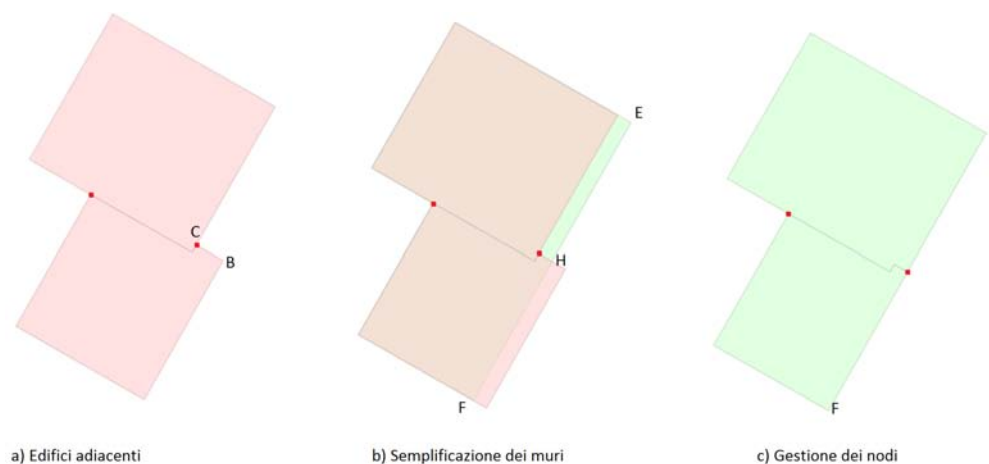


FIGURA 4.8: Semplificazione muri e gestione dei nodi dei muri adiacenti

Nella Figura 4.8 troviamo, inizialmente, un gruppo di edifici da semplificare (a); una volta che si è individuato il muro in esame \overline{BC} e si è calcolato l'angolo formato tra i muri adiacenti (0° in questo caso) si passa alla semplificazione dei muri (b). I nodi che appartengono ai muri adiacenti non vengono toccati. Applicando però la gestione dei nodi descritta sopra otteniamo la semplificazione dell'intero gruppo di edifici (c).

Muro in esame


 FIGURA 4.9: Esempio di semplificazione dei muri e gestione dei nodi del muro \overline{BC} la cui lunghezza è minore della lunghezza critica

Controllo che il muro \overline{BC} abbia almeno un nodo. Se è così immagino \overline{BC} come composizione di due muri: \overline{BH} e \overline{HC} . Se ogni muro contiene i nodi richiamo la funzione `gestisciEdgeMuroTouchRes()` in cui la coordinata `intersect` è `H`, altrimenti nessun nodo è gestito in quanto non ce ne sono.

4.3.3.2 Caso 90°

In questo caso il muro \overline{BC} è più corto della lunghezza critica e \overline{AB} e \overline{CD} sono i muri adiacenti. In base all'angolazione tra muro precedente, successivo guardo se entro nella casistica in cui devo estendere i muri adiacenti, altrimenti rientro nella casistica dello spike. Spieghiamo ora le procedure che portano alla semplificazione dei muri e la gestione dei nodi.

Estensione dei muri adiacenti

Semplificazione dei muri

In riferimento alla Figura 4.10:

1. Calcolo la coordinata (E) data dall'intersezione delle estensioni dei muri adiacenti.
2. Calcolo la distanza tra E ed il punto intermedio del muro \overline{BC} :
 - Se tale distanza è maggiore della soglia specificata dall'utente termino;
 - Se è minore della soglia specificata dall'utente rimuovo muri adiacenti e al loro posto inserisco i nuovi muri \overline{AE} e \overline{DE} ; anche il muro \overline{BC} viene eliminato.
3. Vengono individuati quali nodi appartengono al muro \overline{BC} ; se non ce ne sono si termina la semplificazione, altrimenti viene calcolata l'area della geometria dopo la semplificazione. Confronto tale valore con l'area prima della semplificazione in modo da capire come effettuare la gestione dei nodi del muro \overline{BC} .

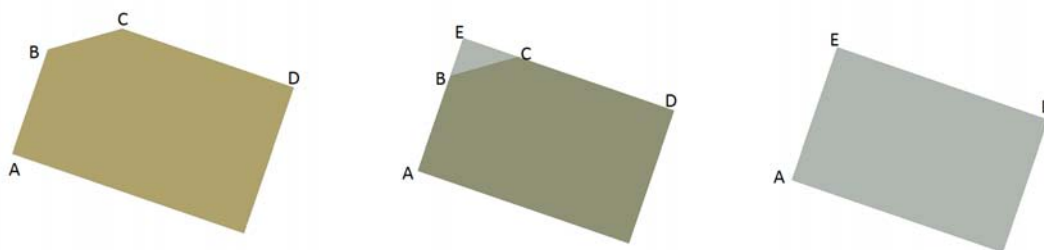


FIGURA 4.10: Esempio di semplificazione muri nel caso di estensione dei muri adiacenti

Gestione dei nodi

Di tutti i muri interessati dal processo di semplificazione l'unico su cui riponiamo particolare attenzione è il muro \overline{BC} . Guardando i muri adiacenti si intuisce che qualsiasi nodo appartenente ad essi, dopo la semplificazione, non subirà alcuna variazione di posizione visto che è avvenuta solo un'estensione. Questo fatto ci basta per farci capire che i nodi che dovranno essere gestiti saranno solo quelli presenti nel muro \overline{BC} .

Come descritto precedentemente, dalla variazione del valore dell'area dipende la gestione dei nodi presenti in \overline{BC} .

- Aumento area
 1. calcolo i nodi in \overline{BC} che non sono in posizione B e C;
 2. se ho 1 nodo conto quanti edge interni lo toccano:
 - se ho un edge interno valuto se la sua estensione tocca \overline{AE} o \overline{ED} ;
 - se ho più edge interni estendo l'edge più lungo aggiungendo nodo in \overline{AE} o \overline{ED} ;
 3. se ho almeno 2 nodi richiamo la funzione `gestisciEdgeMuroTouchRes()` descritta nell'acasistica 0°;
 4. se non ci sono nodi termino la gestione.
- Riduzione area
 1. Per ogni nodo guardo se tutti edge interno che lo tocca intersecano i muri \overline{AE} o \overline{ED} o entrambi:
 - se gli edge del nodo intersecano solo \overline{AE} prendo il primo edge e aggiorno il nodo alla nuova coordinata data dall'intersezione;
 - se gli edge del nodo intersecano solo \overline{ED} prendo il primo edge e aggiorno il nodo alla nuova coordinata data dall'intersezione;

- se gli edge intersecano sia \overline{AE} sia \overline{ED} aggiorno tutti gli edge, e di conseguenza anche il nodo, alla coordinata E.

Spike

Semplificazione dei muri

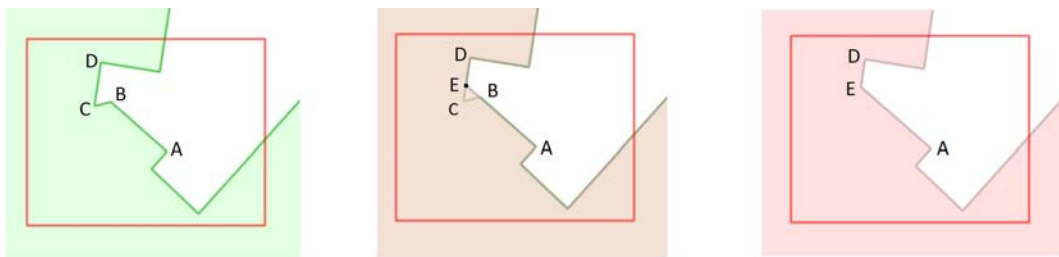


FIGURA 4.11: Esempio di semplificazione muri nel caso di spike

1. Calcolo la coordinata (E) data dall'intersezione delle estensioni dei muri adiacenti.
2. Rimuovo muri \overline{AB} e \overline{CD} ed al loro posto inserisco i muri \overline{AE} e \overline{ED} ; anche il muro \overline{BC} viene rimosso.
3. Viene calcolata l'area della geometria dopo la semplificazione. Confronto tale valore con l'area prima della semplificazione in modo da capire come effettuare la gestione dei nodi del muro \overline{BC} e della parte di muro delimitata dalle coordinate C ed E. Tale parte verrà indicata con \overline{CE} .

Gestione dei nodi

Per la gestione dei nodi verrà richiamata la funzione *gestisciCorto()* che prevede i seguenti passi:

1. Per ogni nodo:
 - (a) Se c'è un solo edge interno aggiungo come prima/ultima coordinata il punto medio (che sta alla metà) della parte di muro delimitata dalle coordinate B ed E ed elimino il nodo.
 - (b) Se ci sono più edge interni creo edge tra punto medio e nodo e aggiungo tale edge ad entrambi i nodi.

Come descritto precedentemente per la gestione dei nodi bisogna capire se l'area della geometria subisce una riduzione o un aumento dopo il processo di semplificazione dei muri.

In caso di

- Aumento area
 1. Trovo i nodi in \overline{BC} che non sono in posizione B e C.
 2. Se ho almeno 1 nodo creo un nuovo nodo dato dalla coordinata del punto medio (che sta alla metà) della parte di muro delimitata dalle coordinate B ed E e richiamo la funzione *gestisciCorto()*.
- Riduzione area
 1. Trovo i nodi presenti nel muro \overline{BC} che non coincidono i suoi estremi. Se ce ne sono:
 - (a) per ogni nodo presente nel muro prendo il primo edge e calcolo la sua intersezione con la parte di muro delimitata dalle coordinate B ed E.
 - (b) se c'è intersezione aggiorno tutti gli edge a quella intersezione;
 - (c) se non c'è l'intersezione ho un errore topologico in cui l'edge non attraversa \overline{BE} ma attraversa \overline{AB} . Per risolvere il problema si calcola l'intersezione tra l'edge e \overline{AB} e si aggiorna il nodo a quella posizione.
 2. Rifai il punto 1 per la parte di muro delimitata dalle coordinate C ed E.

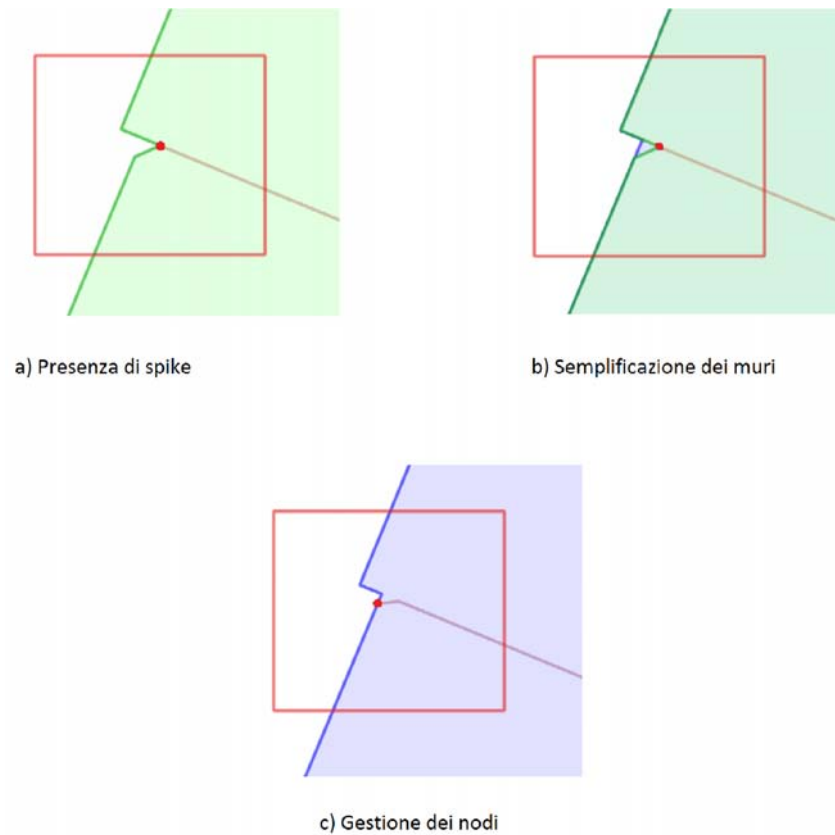


FIGURA 4.12: Esempio di semplificazione muri e gestione dei nodi nel caso di Spike in cui si ha un aumento dell'area della geometria

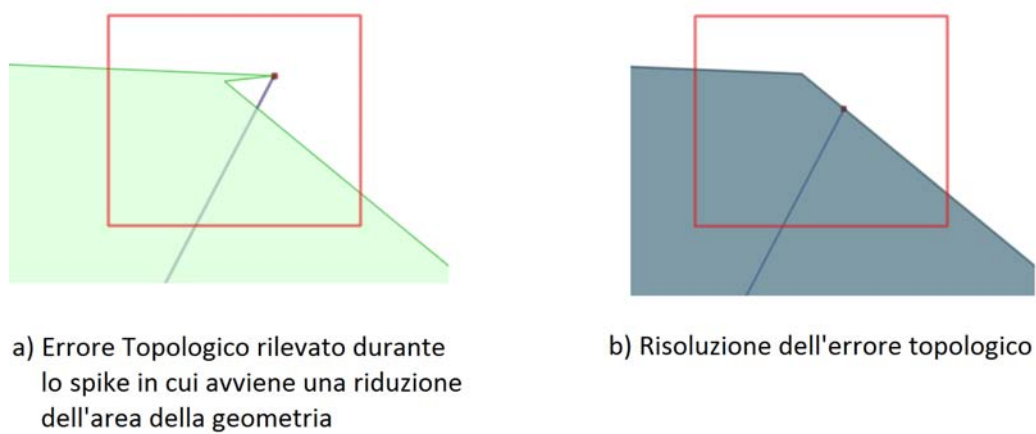


FIGURA 4.13: Esempio di risoluzione di errore topologico nella casistica Spike

4.3.3.3 Caso 180°

Per questa casistica le indicazioni fornite precedentemente per le casistiche 0° e 90° vengono modificate: indichiamo con \overline{CD} il muro più corto della lunghezza critica, \overline{BC} e \overline{DE} i muri adiacenti mentre con \overline{AB} e \overline{EF} i muri che precedono/succedono i muri adiacenti.

Per capire le indicazioni è stato riportato l'esempio di Figura 4.14.

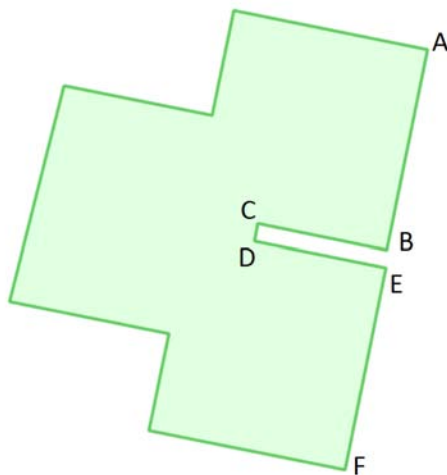


FIGURA 4.14: Esempio di casistica 180°

Semplificazione dei muri

Viene calcolato l'angolo tra i muri \overline{AB} e \overline{CD} . In base a tale valore possiamo cadere in uno dei seguenti valori:

- 0°: gestisco i muri \overline{AB} , \overline{BC} e \overline{CD} come la casistica 0° descritta precedentemente. \overline{BC} sarà il muro in esame, mentre \overline{AB} e \overline{CD} saranno i muri adiacenti.
- 90°: gestisco i muri \overline{AB} , \overline{BC} e \overline{CD} come la casistica 90° descritta precedentemente. \overline{BC} sarà il muro in esame, mentre \overline{AB} e \overline{CD} saranno i muri adiacenti. Fanno parte di questa casistica anche gli angoli di 45°.
- 180°:
 1. Calcolo la coordinata G come intersezione tra l'estensione dei muri \overline{DE} e \overline{AB} .
 2. Valuto l'area formata dalle coordinate B,C,D,G e se è maggiore di una certa soglia fissata dall'utente termino la semplificazione, altrimenti sostituisco i muri \overline{AB} e \overline{DE} con i nuovi muri \overline{AG} e \overline{GE} e rimuovo i muri \overline{BC} e \overline{CD} .

3. Viene calcolata l'area della geometria dopo la semplificazione. Confronto tale valore con l'area prima della semplificazione in modo da capire come effettuare la gestione dei nodi.

- -1: angolo non compreso nelle casistiche di Sester e termino semplificazione.

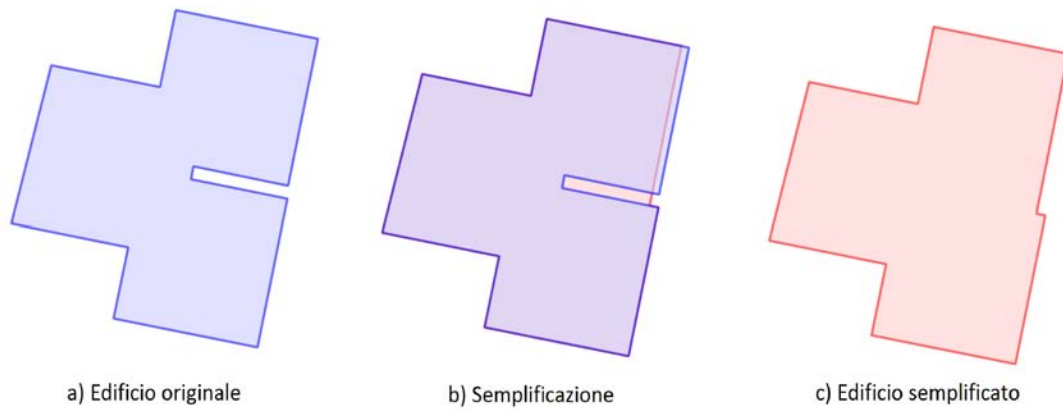


FIGURA 4.15: Esempio di semplificazione dei muri con angolo di 0° nella casistica 180°

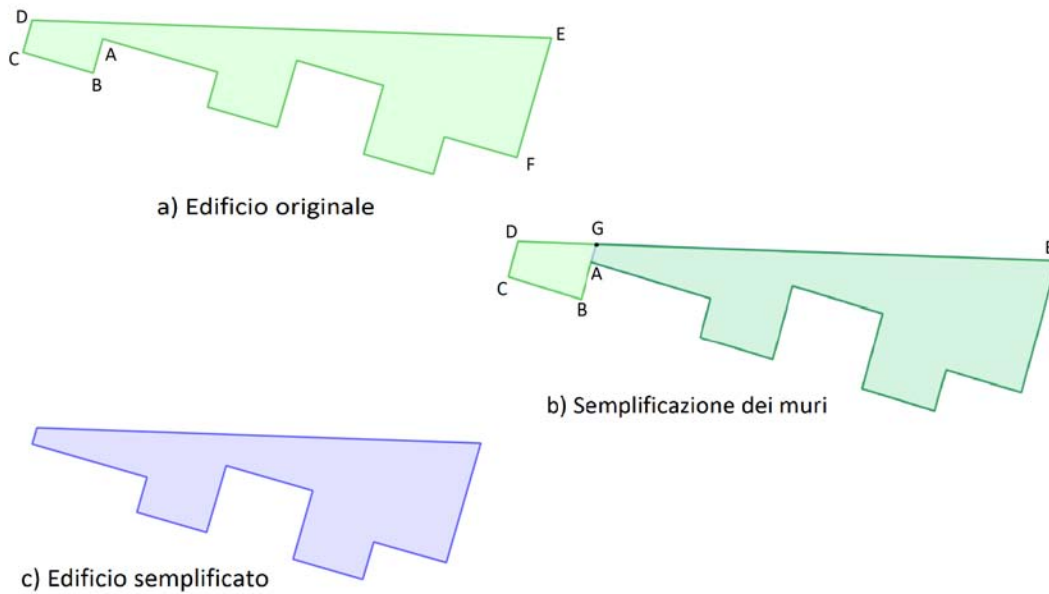


FIGURA 4.16: Esempio di semplificazione dei muri con angolo di 180° nella casistica 180°

Gestione dei nodi

Per quanto riguarda la gestione dei nodi nel caso in cui l'angolo trovato sia di 0° o 90° è stata già trattata nei paragrafi precedenti, quindi ci concentriamo esclusivamente sulla casistica in cui l'angolo trovato è di 180° .

La gestione dei nodi non avviene per tutti i muri presenti, ma solo per quelli che sono direttamente coinvolti nel processo di semplificazione. Guardando la Figura 4.16 bisogna prestare attenzione ai muri \overline{AB} , \overline{BC} , \overline{CD} e alla parte di muro compresa tra le coordinate D e G. A tale proposito verrà utilizzata la seguente funzione:

- *GestioneNodi180MuriTouchArea()* in cui per ogni nodo:
 1. se ha un solo edge interno guardo l'altra estremità dell'edge:
 - se è interna alla geometria semplificata calcolo intersezione tra edge e AG e aggiorno la posizione del nodo eliminando tutte le coordinate che non sono nel poligono;
 - se è esterna alla geometria semplificata allora elimino l'edge relativo. Ciò implica la perdita di una proprietà cioè di un edificio (vedi Figura 4.17);
 2. se più edge toccano il nodo elimino tutti gli edge che hanno l'altra estremità all'esterno della geometria semplificata e verifico che il nodo abbia almeno un edge interno. Se non è così elimino il nodo, altrimenti tra tutti gli edge che rimangono prendo il primo (ricordando che ad ogni nodo è associata una lista di tutti gli edge interni che lo toccano), calcolo intersezione con \overline{AG} e aggiorno tutti gli edge interessati a quell'intersezione.

Come descritto precedentemente, dalla variazione del valore dell'area dipende la gestione dei nodi presenti nei muri sopra citati. In caso di

- Aumento area
 1. Tutti i nodi presenti nella parte di muri di muro delimitata dalle coordinate D e G vengono gestiti richiamando la funzione *gestisciEdgeMuroTouchRes()* descritta precedentemente;
 2. Tutti i nodi presenti nel muro \overline{AB} vengono gestiti richiamando la funzione *gestisciEdgeMuroTouchRes()* descritta precedentemente;

3. Se esiste almeno un nodo non gestito dai casi 1 e 2 creo un nodo nel punto medio di \overline{AG} e richiamo la funzione `gestisciCorto()` per i muri \overline{CD} e \overline{BC} .

- Riduzione area

1. Richiamo la funzione `GestioneNodi180MuriTouchArea()` per i muri \overline{AB} , \overline{BC} , \overline{CD} e per la parte di muro delimitata dalle coordinate D e G.

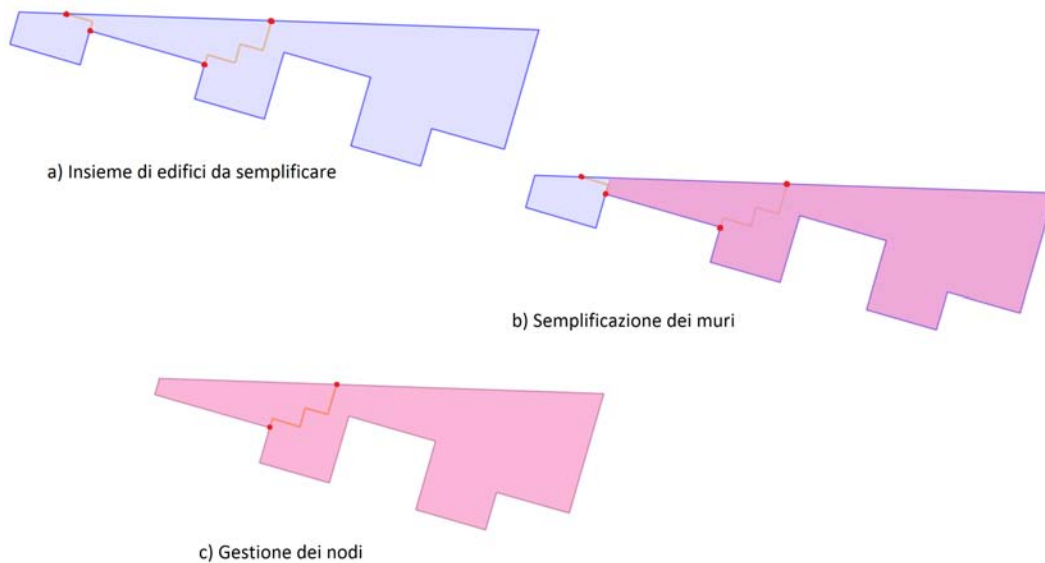


FIGURA 4.17: Esempio di semplificazione dei muri e gestione dei nodi nel caso di riduzione dell'area con un'angolo di 180° nella casistica 180°

Nell'esempio di Figura 4.17 si può notare come la gestione dei nodi porta alla perdita di un'edificio. Ciò può essere giustificato dal fatto che l'area che è stata rimossa sia minore di una certa soglia definita dall'utente. Sta all'utente, quindi, decidere il livello di dettaglio della semplificazione: un'area grande può portare alla perdita dell'edificio ma consente una maggiore semplificazione, mentre un'area piccola non consente la semplificazione ma la presenza degli edifici non varia. Pertanto tutti gli edge che hanno un qualsiasi nodo la cui posizione è esterna rispetto alla geometria semplificata devono essere eliminati con la conseguente perdita dell'edificio in esame.

4.3.4 Semplificazione edge interni

Una volta terminata la semplificazione dei muri si passa, se l'utente lo desidera, alla semplificazione degli edge interni. Ogni edge interno può essere composto da vari segmenti i quali sono tra loro ortogonali.

La prima soluzione che è stata pensata è stata quella utilizzare l'algoritmo di Douglas-Peucker già descritto in precedenza. Sicuramente il numero dei vertici che caratterizza l'edge verrà ridotto con il conseguente raggiungimento dell'obiettivo della semplificazione, ma la qualità del prodotto finale ne risente in quanto i segmenti che comporranno l'edge semplificato non saranno ortogonali come in precedenza, perdendo di fatto la caratteristica originale, ed inoltre si può incappare in errori topologici dovuti alla sovrapposizione di edge semplificati. Per queste ragioni si è passati ad un'altra soluzione che raggiunga l'obiettivo fissato ma nello stesso tempo preservi il più possibile l'originalità degli edge iniziali.

La soluzione adottata consiste in due fasi:

1. Semplificazione dell'edge in base al suo andamento originale.
2. Modifica dell'edge in base all'angolazione dei muri che compongono la geometria semplificata.

Per scelta di implementazione, negli edge che saranno semplificati la coordinata from non verrà modificata, mentre la coordinata to potrà subire variazioni.

I passi che vengono svolti nella prima fase sono:

1. Per ogni edge interno:
 - (a) Memorizzo quali e quanti muri contengono la coordinata from e to (che coincidono con i nodi).
 - (b) Calcolo l'angolo tra primo e ultimo segmento dell'edge:
 - i. Se non sono paralleli estendo primo e ultimo segmento, elimino tutte le coordinate e aggiungo la loro intersezione come seconda coordinata;
 - ii. Se sono paralleli controllo che ci siano almeno 4 coordinate. Se è così:
 - A. edge viene trasformato in una L.

B. controllo se la seconda coordinata interseca un'altro edge interno o un muro. Se è così aggiorno in nodo all'intersezione ed elimino ultima coordinata; in questo modo l'edge passa da tre coordinate a due.

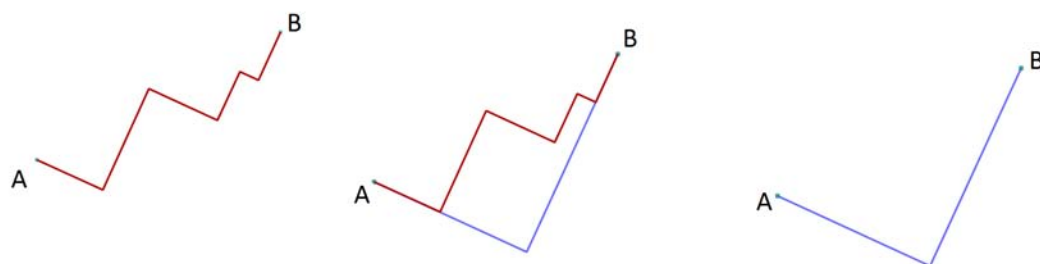


FIGURA 4.18: Esempio di semplificazione di un edge interno in cui il primo e l'ultimo segmento non sono paralleli tra loro

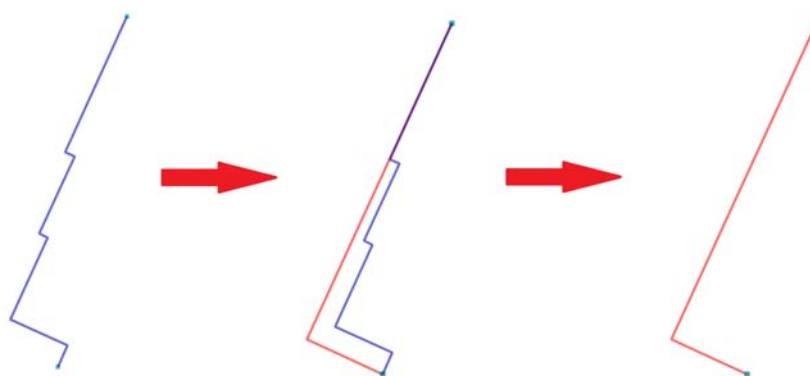


FIGURA 4.19: Esempio di trasformazione in una L di un edge interno durante la fase di semplificazione

In Figura 4.18 possiamo vedere la semplificazione dell'edge in cui il primo segmento e l'ultimo non sono paralleli tra loro descritta precedentemente al passo i), mentre in Figura 4.19 vengono forniti tutti i passaggi per la creazione della L di un edge con almeno quattro coordinate in cui in primo e l'ultimo segmento sono paralleli tra loro.

Di seguito, in Figura 4.20 possiamo vedere tutti i passaggi che portano alla semplificazione di un edge interno in cui il primo e l'ultimo segmento sono paralleli tra loro e la seconda coordinata interseca un muro:

a) Edge originale

- b) Edge originale viene semplificato: si ottiene una L composta dai nodi A e B e dal vertice C.
- c) Il vertice C interseca il muro: elimino il nodo A e lo sposto in C.
- d) Edge semplificato

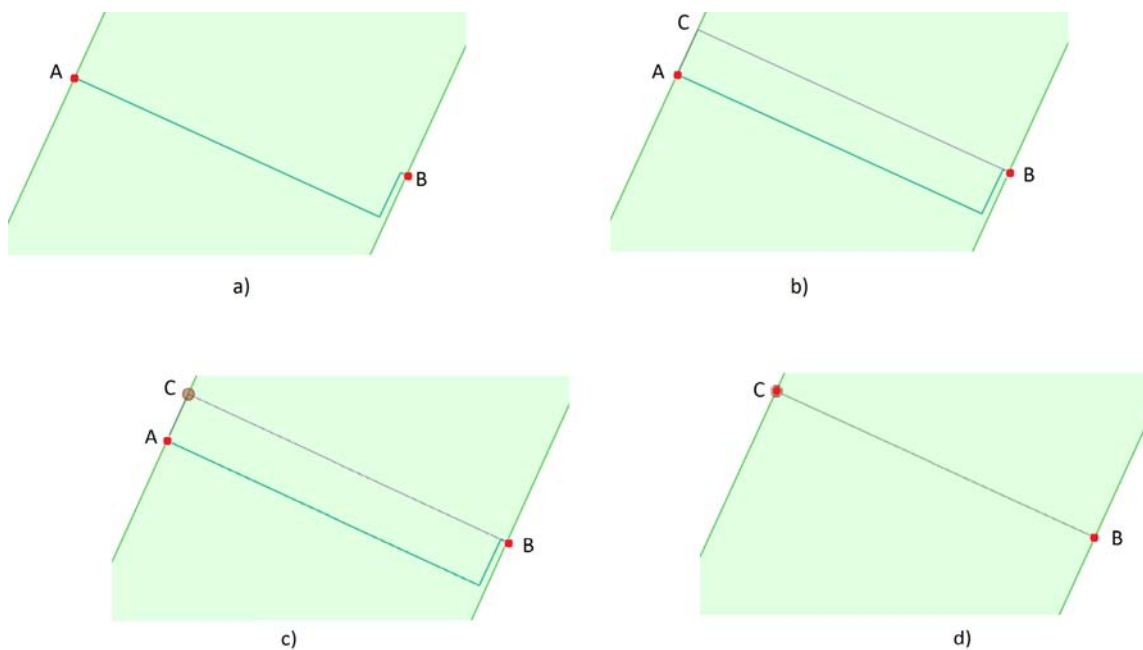


FIGURA 4.20: Processo di semplificazione di un edge interno

Terminata la prima fase, tutti gli edge interni avranno al più due o tre coordinate.

Nella seconda fase vengono ricalcolati quanti e quali muri contengono i nodi, che coincidono con gli estremi dell'edge, perchè potrebbero aver subito variazioni. A seconda di quanti muri vengono conteggiati e al numero di coordinate che determinano l'edge ci sarà un'ulteriore semplificazione:

- Per tutti gli edge interni in cui gli estremi sono entrambi contenuti in due muri :
 - Tra i quattro muri trovo il muro più corto che lo chiameremo m .
 - Tra i restanti tre muri trovo il muro perpendicolare che non è adiacente a m e lo chiameremo M .

- Estendo m e M e la loro intersezione sarà la nuova seconda coordinata dell'edge.

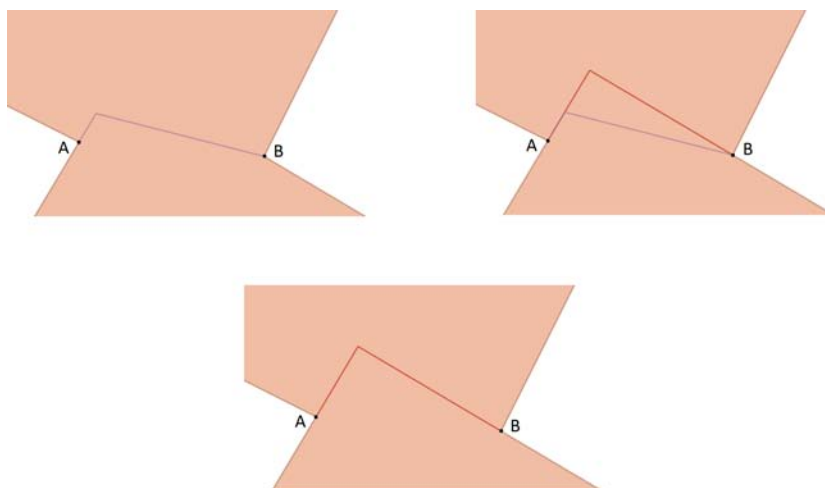


FIGURA 4.21: Edge interno in cui gli estremi sono entrambi contenuti in due muri

- Se un estremo è contenuto in due muri e l'altro estremo in uno o viceversa:
 - tra la coppia di muri adiacenti viene individuato il muro “più parallelo” all'edge in esame;
 - viene calcolata l'intersezione tra il muro che contiene il singolo nodo e l'estensione del muro “parallelo”; a tale intersezione corrisponderà in nuovo valore del nodo.

Al termine di tale operazione l'edge avrà solo due coordinate.

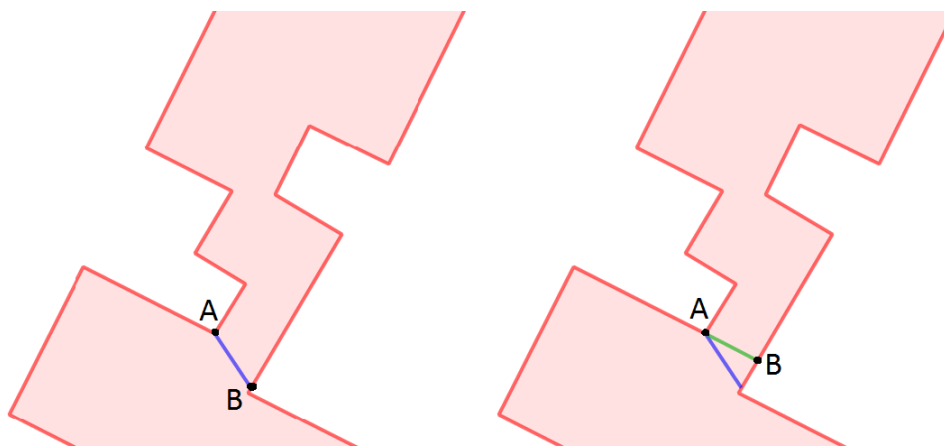
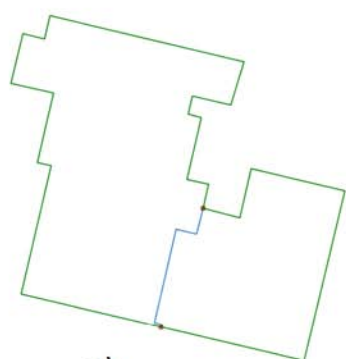


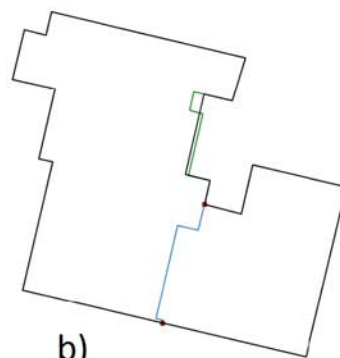
FIGURA 4.22: Edge interno in cui un estremo è contenuto in due muri e l'altro estremo in un muro

4.4 Risultati ottenuti

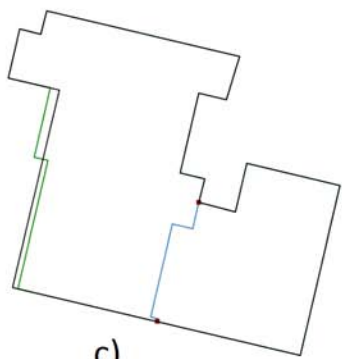
Ora che sono stati descritti i passi dell'algoritmo, la semplificazione dei muri, la gestione dei nodi e la semplificazione degli edge interni possiamo mostrare i risultati che abbiamo ottenuto. In seguito saranno mostrate alcune figure raffiguranti tutti i passi del processo relativo ad una coppia di edifici adiacenti, lasciando, come ultimo passo, la semplificazione degli edge interni.



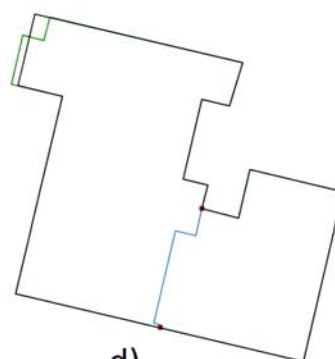
a)



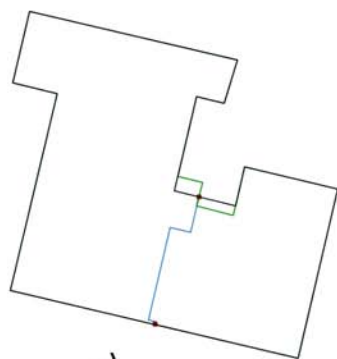
b)



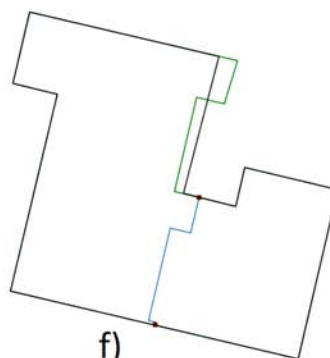
c)



d)



e)



f)

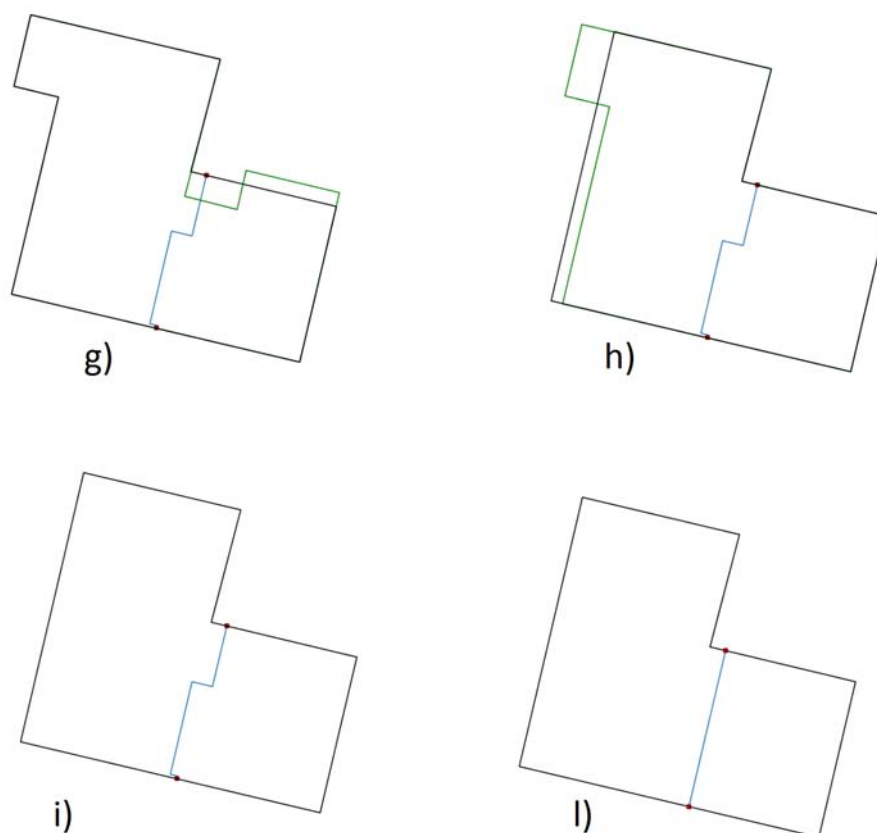


FIGURA 4.23: Esempio di semplificazione di una coppia di edifici adiacenti

Nella Figura 4.23 possiamo vedere tutte le fasi relative al processo di semplificazione: in verde si vede la struttura dell'insieme di edifici prima della semplificazione, in nero il risultato della semplificazione mentre in blu l'edge interno. Si parte dall'insieme di edifici da semplificare (a) e si esegue passo passo tutte la semplificazione dei muri con la relativa gestione dei nodi (da b ad h) fino ad ottenere in risultato finale (i) ; infine si passa alla semplificazione dell'unico edge presente (l). Per questo tipo di semplificazione si è utilizzato il parametro “Sester lunghezza critica” pari a 4.0 metri.

Forniamo ora alcune immagini per capire i diversi tipi di semplificazione, sullo stesso gruppo di edifici, dovuti al parametro “Sester lunghezza critica”. Per ogni valore utilizzato si potrà ottenere un nuovo insieme di edifici e quindi sarà dell'utente capire come far variare tale valore a seconda di cosa vuole ottenere e a quale livello di dettaglio.

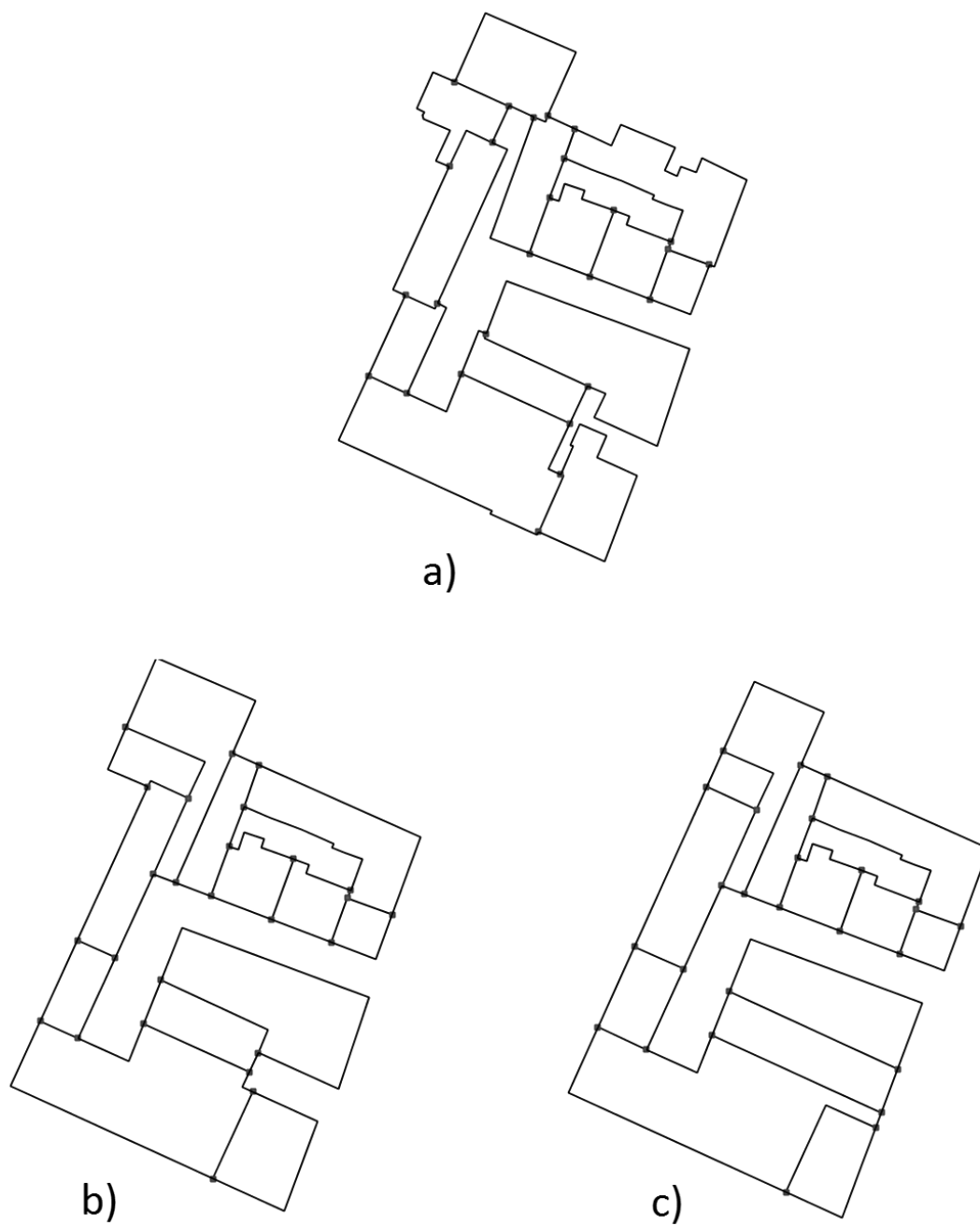


FIGURA 4.24: Esempio di semplificazione di un'insieme di edifici adiacenti :
 a) edifici originali b) semplificazione con Sester lunghezza critica pari a 4 metri
 c) semplificazione con Sester lunghezza critica pari a 8 metri

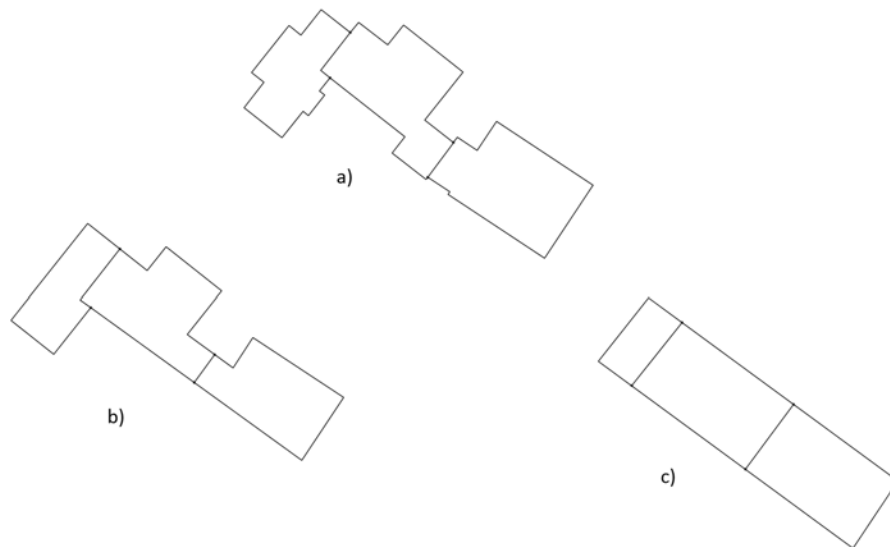


FIGURA 4.25: Esempio di semplificazione di un'insieme di edifici adiacenti :
 a) edifici originali b) semplificazione con Sester lunghezza critica pari a 2 metri
 c) semplificazione con Sester lunghezza critica pari a 4 metri

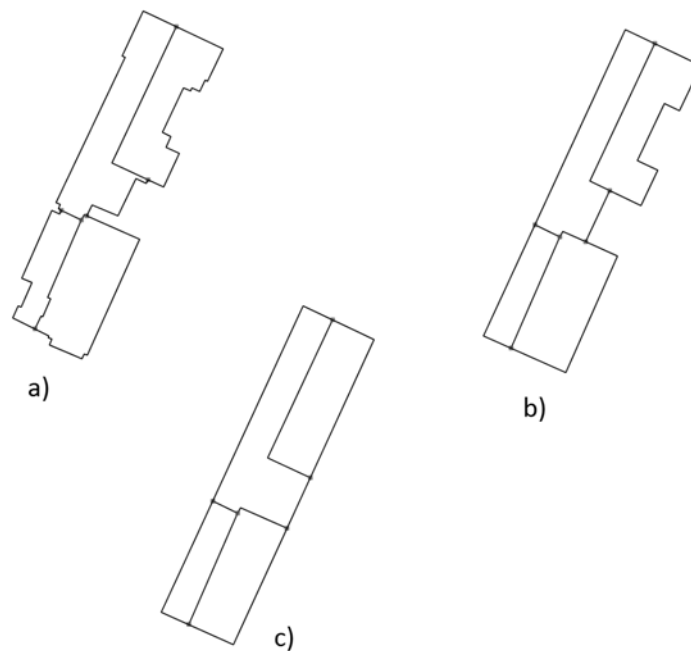


FIGURA 4.26: Esempio di semplificazione di un'insieme di edifici adiacenti :
 a) edifici originali b) semplificazione con Sester lunghezza critica pari a 2 metri
 c) semplificazione con Sester lunghezza critica pari a 4 metri

Capitolo 5

Conclusioni

Questo lavoro di tesi si poneva l'obiettivo di preservare la topologia di un insieme di edifici adiacenti durante il processo di generalizzazione. Il lavoro svolto è stato implementato come un plug-ins per OpenJump. In questo modo è possibile effettuare dei test e visualizzare i risultati ottenuti.

L'algoritmo sviluppato è stato testato su vari dati ed in particolare sugli edifici di Mestre e, come si può vedere dalle immagini fornite nel capitolo precedente, si hanno dei buoni risultati. Naturalmente, vista la varietà degli edifici e la dimensione originale dell'insieme di edifici da semplificare, trovare un'unica soluzione che vada bene a tutti è molto difficile. Infatti se un edificio è collocato in aperta campagna potrà subire notevoli variazioni, mentre un edificio sito in un centro abitato dovrà subire limitate modifiche visto il contesto in cui è inserito. Per questo motivo si è scelto di creare una shell nella quale sarà l'utente a scegliere quali parametri utilizzare per la semplificazione.

L'introduzione della gestione contemporanea di più edifici garantisce la correttezza topologica dell'azione di semplificazione in quanto l'insieme di edifici viene visto come un unico edificio composto da muri e da edge interni. Rispetto agli algoritmi precedenti, la semplificazione che si ottiene è migliore dato che lavorare sull'intero ring esterno del cluster di edifici presenta la possibilità di risolvere alcuni casi che altrimenti non esisterebbero. Un ulteriore punto di forza è la riduzione del numero di lati da processare visto che ogni lato condiviso dagli edifici verrà processato una sola volta.

L'inserimento di questo meccanismo crea un incremento della complessità temporale del processo di semplificazione dell'edificato. Vengono aggiunte la creazione del grafo e l'aggiornamento della sua geometria. Attualmente questi processi hanno un costo non trascurabile. L'incremento del tempo di esecuzione, portato dal

grafo planare, è bilanciato in parte dalla riduzione del numero dei lati da processare. Tuttavia questa riduzione, non bilancia il tempo delle operazioni sul grafo. Volendo dare uno sguardo ai possibili sviluppi futuri, nonostante si siano risolti i problemi legati alla correttezza topologica della struttura in esame, restano da affrontare alcuni importanti problemi quali la semplificazione dei vari ring interni che possono essere presenti nelle geometrie, l'utilizzo frequente di funzioni già presenti nel package open-source Java Topology Suite (JTS) che presenta una grande versatilità e può essere utilizzato anche in altri passi nel processo di generalizzazione in modo da ridurre l'incremento della complessità temporale. Caratteristica di questo algoritmo è la selezione manuale dell'insieme di edifici da semplificare; si potrebbe in futuro identificare automaticamente il gruppo di edifici da semplificare e attuare l'algoritmo proposto.

In una visione generale, invece di concentrarci esclusivamente sugli edifici, l'algoritmo, con qualche leggera modifica, potrebbe essere testato su qualsiasi tipo di struttura topologica riconducibile ad un grafo come ad esempio un'area divisa in lotti (una provincia con i vari comuni, un terreno frazionato per distinguere le varie proprietà, ecc. . .).

Bibliografia

- [1] Baldin R. (2010) Context awareness and typification in building generalisation, Tesi di Laurea Magistrale, Università degli Studi di Padova, 2010.
- [2] Bergamin Filippo (2009) Miglioramenti agli algoritmi CARGEN per la generalizzazione degli edifici, Tesi di Laurea Triennale, Università degli Studi di Padova, 2009.
- [3] Dizionario Devoto-Oli (1979) Vocabolario illustrato della lingua Italiana Selezione dal Reader's Digest, Milano.
- [4] Douglas D. e Peucker T. (1978) Algorithms for the Reduction of the Number of Points Required to Represent a Digitized Line or its Caricature, *The Canadian Cartographer*.
- [5] Enciclopedia Italiana Grolier (1987) Vol. 4, Cartografia, pp 293 – 296.
- [6] Mackaness W. A. (2007) Understanding Geographic Space, in *Generalisation of Geographic Information: Cartographic Modelling and Applications*, Mackaness, W. A., Ruas, A. and Sarjakoski, L. T. (eds.), pp. 1–10, Elsevier, Oxford.
- [7] McMaster R. B. e Shea K. S. (1992) *Generalization in Digital Cartography*, Association of American Geographers, Washington D.C.
- [8] Sandro Savino, Generalizzazione Cartografica, Lezione del corso di Sistemi Informativi Territoriali A.A 2013-2014.