

## **Valutazione dell'accessibilità con software open source**

Prof. L. La Franca (\*), Ing. A. Lo Burgio (\*\*), Ing. G. Ferrante (\*\*\*)  
Università di Palermo, Viale delle Scienze, edificio 8  
Tel 091 23840411 Fax 091 423105

(\* Professor of Air Transport - luigi.lafranca@unipa.it

(\*\*) Ph.D. Student in Tecnica ed Economia dei Trasporti - antonino.loburgio@unipa.it

(\*\*\*) Dottore in Ingegneria Civile – Laurea Triennale

### **Sommario**

*Questo studio vuole affrontare la tematica della valutazione dell'accessibilità territoriale attraverso l'uso di software GIS (Geographic Information System) open source. In particolare si è affrontato il tema dell'accessibilità ai terminal aeroportuali e per testare il metodo è stata condotta un'applicazione sulle infrastrutture aeroportuali siciliane.*

### **Abstract**

*In this work we want to focus on accessibility to the airport terminals. If the access time to the terminals is not fast enough, the airplane will not be competitive compared to the other modal transport for the short distance. The aim of this work is to estimate the accessibility through the use of a GIS (Geographic Information System) open source software. To validate the method we have conducted an application on Sicilian airport infrastructures. The study focused on topological measures of accessibility, which express the site's reachability in terms of connections offered by the transport system. It was considered the passive accessibility, in terms of time, from all Sicilian towns towards the airport terminals. This measure can be represented by isochrones, which allow to determine the number of potential destinations that can be reached in a predetermined interval of travel time. This measure does not consider the subjective perceptions and personal preferences, which implies that all opportunities are equally desirable with reference to travel time or the type of opportunity.*

L'accessibilità ha da sempre ricoperto un ruolo di fondamentale importanza per l'uomo, per lo sviluppo delle società e per rendere sempre più agevoli gli spostamenti, a prescindere dalle cause che li originano.

In questo lavoro si vuole focalizzare l'attenzione sull'accessibilità ai terminal aeroportuali, in quanto la velocità e la facilità di accesso sono aspetti strategici per il trasporto aereo. Se il tempo di accesso ai terminal non fosse sufficientemente rapido, negli spostamenti a breve distanza l'aereo perderebbe il vantaggio che ha sulle altre scelte modali.

Lo scopo che ci si è posti con questo lavoro è quello di valutare l'accessibilità territoriale attraverso l'uso di software GIS (Geographic Information System) open source. Per validare il metodo si è condotta un'applicazione sulle infrastrutture aeroportuali siciliane. A tal fine è stato preliminarmente necessario analizzare la rete stradale siciliana, per creare un quadro generale che ha permesso di mettere in evidenza il grado di accessibilità che caratterizza i tre principali aeroporti dell'isola:

- Falcone Borsellino di Palermo-Punta Raisi
- Vincenzo Bellini di Catania-Fontanarossa
- Vincenzo Florio di Trapani-Birgi

Lo studio si è concentrato sulle misure topologiche dell'accessibilità, che esprimono la raggiungibilità di un sito in funzione delle connessioni offerte dal sistema di trasporti. E' stata, quindi, considerata l'accessibilità passiva a partire da tutti i comuni siciliani verso i terminal aeroportuali in termini di tempo.

Tale misura può essere rappresentata da curve isocrone, che permettono di determinare il numero di destinazioni potenziali che possono essere raggiunte in un prefissato intervallo di tempo di viaggio. Questa misura non tiene conto della percezione degli individui e delle loro preferenze; ciò implica che tutte le opportunità sono egualmente desiderabili con riferimento al tempo di viaggio o al tipo di opportunità.

Le isocrone sono state tracciate per testare il potenziale della rete stradale a vuoto e sotto l'ipotesi semplificativa che l'offerta dei tre aeroporti, in termini di voli disponibili, sia uguale.

## **SOFTWARE OPEN SOURCE UTILIZZATI**

L'elemento di novità che si vuole mettere in evidenza in questo lavoro è quello dell'implementazione di un sistema che consenta di tracciare in maniera agevole delle isocrone di accessibilità attraverso l'utilizzo di software GIS open source, che oltre ad essere gratuiti, sono molto adattabili alle diverse esigenze tecniche. In particolare sono stati utilizzati (vedi loghi in fig.1):

- Quantum GIS
- PostgreSQL
- PostGIS

- PgRouting

### **QuantumGIS**

QuantumGIS (QGIS) è un Sistema Informativo Geografico a codice aperto (open source). Il progetto è nato nel maggio 2002, attualmente gira su molte piattaforme Unix, su Windows, e OSX. QGIS è sviluppato in Qt e C++. Ciò fa sì che QGIS appaia comodo nell'uso e piacevole e semplice da usare nell'interfaccia grafica GUI (graphical user interface). L'obiettivo iniziale era di fornire un visualizzatore di dati GIS, ma attualmente QGIS ha oltrepassato questo punto nel suo sviluppo, ed è usato da molti per il loro lavoro quotidiano nel campo GIS. Permette, inoltre, di eseguire un elevatissimo numero di query attraverso i suoi tools e visualizzarne i risultati.

### **PostgreSQL**

PostgreSQL, conosciuto anche come Postgres, è il più completo database server open source e fra questi è l'unico a supportare il modello object-relational. L'utilizzo di tale applicazione è stata fondamentale per la stesura di questo lavoro. Infatti, è proprio attraverso postgresQL che è stato possibile creare tutti i database su cui andare a caricare (tramite Java) le mappe e su cui eseguire le query.

### **PostGIS**

PostGIS è l'estensione spaziale del server PostgreSQL che introduce i tipi di dato geometrico e le funzioni per lavorare con essi. Fornisce i tipi di dati specificati negli standard dell'Open Geospatial Consortium. In particolare, è un geodatabase e fornisce il sistema di gestione dati sui quali è basato un GIS. Consente di poter archiviare su database anche dati geografici e di eseguire operazioni su di essi. Le geometrie che si possono manipolare ed usare grazie a PostGis sono: point, line, polygon, multipoint, multiline, multipolygon e geometry collections.

### **PgRouting**

PgRouting è una estensione open source per il database spaziale PostGIS che aggiunge al famoso DBMS (*Database Management System*) relazionale funzionalità per il calcolo del cammino minimo su una topologia stradale. PgRouting fornisce una varietà di tools per poter effettuare ricerche Shortest Path, e in particolare fornisce delle funzioni *core* sulla base delle quali è possibile sviluppare delle funzioni *wrapper* per la navigazione di reti stradali reali, che richiedono algoritmi complessi che supportino restrizioni di vario genere (come sensi unici, divieti, aree pedonali, tipologia di strada, etc.) e che coinvolgano anche parametri *time-dependent*.

## MISURA DELL'ACCESSIBILITA'

Come già accennato, la misura dell'accessibilità presa in considerazione in questo lavoro è quella ottenuta attraverso il tracciamento delle isocrone, che altro non sono che delle linee luogo dei punti caratterizzate dallo stesso tempo di viaggio verso un punto fissato e con una determinata scelta modale. La scelta modale presa in considerazione è l'auto privata. A tal proposito, è stato necessario costruire il grafo della rete stradale siciliana caratterizzando ogni arco con una serie di attributi: tipo di strada, nome della strada, lunghezza, numero di carreggiate, senso di marcia, velocità di percorrenza a vuoto, etc. (fig.2).

Un attributo di particolare interesse è il *traveltime* (tempo di viaggio), espresso in minuti, dato dal rapporto fra la lunghezza e la velocità di percorrenza a vuoto dell'arco stradale  $V_0$ , cioè la massima velocità costante che un veicolo isolato può tenere, con sicurezza, lungo un tronco stradale; anche il tempo di viaggio è quindi da intendersi a vuoto. Per costruire le isocrone si considera il percorso minimo che bisogna compiere, a partire da ogni punto della rete stradale, verso un punto stabilito, che corrisponde in questo caso con l'infrastruttura aeroportuale. Per individuare il percorso minimo è stato implementato l'algoritmo di Dijkstra che consente di trovare il percorso minimo fra due nodi di una rete chiusa.

### L'algoritmo di Dijkstra

L'algoritmo deve il suo nome all'informatico Edsger Dijkstra e permette di trovare i cammini minimi (Shortest Paths) in un grafo ciclico orientato con pesi non negativi sugli archi: in particolare l'algoritmo può essere utilizzato o per trovare il cammino minimo che unisce due nodi del grafo, oppure per trovare quelli che uniscono un nodo d'origine a tutti gli altri nodi e quindi, estendendolo, per trovare tutti i cammini minimi da ogni nodo ad ogni altro nodo della rete. L'algoritmo visita i nodi nel grafo in maniera simile a una ricerca in ampiezza o in profondità. In ogni istante, l'insieme  $N$  dei nodi del grafo è diviso in tre parti: l'insieme dei nodi visitati  $V$ , l'insieme dei nodi di frontiera  $F$ , che sono successori dei nodi visitati, e i nodi sconosciuti, che sono ancora da esaminare. Per ogni nodo  $z$ , l'algoritmo tiene traccia di un valore  $d_z$ , inizialmente posto uguale a infinito, e di un nodo  $u_z$ , inizialmente indefinito.

L'algoritmo consiste semplicemente nel ripetere il seguente passo: si prende dall'insieme  $F$  un qualunque nodo  $z$  con  $d_z$  minimo, si sposta  $z$  da  $F$  in  $V$ , si spostano tutti i successori di  $z$  sconosciuti in  $F$ , e per ogni successore  $w$  di  $z$  si aggiornano i valori  $d_w$  e  $u_w$ .

L'aggiornamento viene effettuato con la regola:

$$d_w \leftarrow \min[d_w, d_z + p_a]$$

dove  $a$  è l'arco che collega  $z$  a  $w$ . Se il valore di  $d_w$  è stato effettivamente modificato, allora  $u_w$  viene posto uguale a  $z$ .

L'aggiornamento di  $u_w$  ci permette di ricordare che, al momento, il cammino di peso minimo che conosciamo per arrivare da  $x$  in  $w$  ha come penultimo nodo  $z$ .

L'algoritmo parte con  $V = \emptyset$ ,  $F = \{x\}$ ,  $d_x = 0$  e prosegue finché  $y$  non viene visitato, o finché  $F = \emptyset$ : in questo caso,  $y$  non è raggiungibile da  $x$  lungo un arco orientato. Se usciamo solo nel secondo caso, alla fine dell'algoritmo,  $d_z$  contiene, per ogni nodo  $z$ , il peso di un cammino minimo da  $x$  a  $z$ ; inoltre, il vettore  $u$  permette di ricostruire l'albero dei cammini minimi con origine in  $x$ .

Alla base di questi problemi c'è lo scopo di trovare il percorso minimo (più corto, più veloce, più economico, ect.) tra due punti, uno di partenza e uno di arrivo. Con il metodo che vedremo è possibile ottenere non solo il percorso minimo tra un punto di partenza e uno di arrivo ma il percorso minimo tra un punto di partenza e tutti gli altri punti della rete.

## **METODOLOGIA E APPLICAZIONE**

Implementare l'algoritmo in una rete particolarmente grande può richiedere del tempo. Solitamente tale operazione è delegata a dei calcolatori opportunamente sviluppati per fare ciò. In questo caso si farà ricorso a PgRouting, un tool che aggiunge in QGIS tutta una serie di funzioni, fra le quali proprio quella di implementare l'algoritmo di Dijkstra. Per poter fare questo è necessario prima di tutto partire dalla rete stradale e da questa ricavarci i nodi. Ad ogni ramo della rete bisogna associare due nodi, un nodo di inizio (startpoint) e un nodo di fine (endpoint). Si procede per gradi. Il primo passo è quello di creare una tabella in cui oltre ai records presenti nella tabella rete\_stradale siano presenti due ulteriori campi, costituiti da entità geometriche puntiformi, che rappresentano appunto lo startpoint e l'endpoint. Attraverso un'apposita query, i campi delle due colonne aggiunte (startpoint e l'endpoint) vengono riempite con i codici dei nodi, che altro non sono che i punti di intersezione fra i vari rami stradali. Ogni ramo è ora caratterizzato da un nodo di inizio e da un nodo di fine (fig.3).

A questo punto si è pronti per lanciare l'espansione di PgRouting che, una volta avviato l'algoritmo, consentirà di ottenere un database con i percorsi minimi tra un punto preso come destinazione e i vari nodi della rete. Il risultato così ottenuto, oltre che visualizzabile graficamente, sarà consultabile anche attraverso un database generato come risultato dell'algoritmo che, oltre alle colonne già presenti, sarà composto da tre ulteriori colonne: la colonna *id* in cui sono riportati i nomi dei nodi che caratterizzano il percorso più breve, la colonna *start\_id* e la colonna *end\_id*, che rappresentano rispettivamente il punto di partenza e di arrivo. In questa elaborazione è stato chiesto all'algoritmo di utilizzare il tempo di viaggio come funzione di costo e quindi come peso di ogni arco del grafo.

## Tracciamento delle isocrone

A questo punto si può passare al tracciamento delle isocrone di accessibilità. L'idea è quella di creare una nuova tabella contenente i nodi della tabella *network*. Ad ognuno dei nodi verrà associata una funzione di costo, pari proprio al tempo necessario per andare da quel nodo all'aeroporto considerato, attraverso il percorso più breve trovato con l'algoritmo di Dijkstra. Per fare questo si deve proporre al database la query riportata in figura 4.

Analizziamo la sintassi della query: si è richiesto al database di creare una tabella chiamata *catchment\_airport* formata da tre colonne, una colonna con il nome del nodo *id*, una colonna in cui viene specificata l'entità geometrica degli elementi che costituiscono la tabella ed infine una colonna in cui viene riportata la funzione di costo. Il risultato della query può essere importato in QuantumGIS con il comando *Aggiungi vettore PostGIS*.

A questo punto, tramite il comando *interpola* di QGIS si può effettuare una interpolazione triangolare (TIN) fra i punti ottenuti, in base alla funzione di costo. Si otterrà così un *grid* che evidenzia, attraverso colori differenti, le fasce del territorio caratterizzate da un valore della funzione di costo che ricade all'interno di un intervallo. Attraverso il comando *curve di livello*, raggiungibile dal percorso *Raster - Extraction*, si possono infine ricavare le isocrone. Questo lavoro è stato fatto singolarmente per le tre più importanti infrastrutture aeroportuali siciliane. A titolo d'esempio nella figura 5 si riportano le isocrone riferite all'aeroporto di Palermo Falcone-Borsellino.

Per un'analisi ancora più completa e per mettere in mostra le potenzialità e la flessibilità degli strumenti open source utilizzati, si sono voluti sovrapporre i risultati fin qui ottenuti per i tre aeroporti, in modo da evidenziare le aree di influenza degli stessi dal punto di vista dell'accessibilità relativa alle infrastrutture stradali. Per far questo, si è proceduto ad implementare una query che consente di chiedere al database di prendere in considerazione, per ogni nodo, il tempo di viaggio minore rispetto ai tre aeroporti.

Si può a questo punto importare la tabella così ottenuta in QGIS attraverso il comando *aggiungi vettore PostGIS*, effettuare l'interpolazione del vettore così ricavato e attraverso il comando *estrai curve di livello* si ottiene quanto è riportato in figura 6.

Da quest'ultima applicazione si può notare come nell'area meridionale e sud-orientale della Sicilia ci sono zone il cui tempo di accesso a una infrastruttura aeroportuale supera le 2 ore, arrecando gravi disagi alla popolazione residente in queste fasce.

Altra applicazione sviluppata molto interessante è quella dell'intersezione, con strumenti di geoprocessing, fra le isocrone prima trovate e diversi tematismi o cosiddetti *shapefile*, che ha permesso di ricavare dei dati di rilevante importanza ai fini dell'analisi e della stima dell'accessibilità. In particolare, da queste intersezioni, si è ottenuto per ogni fascia isocrona il numero di persone residenti, la superficie territoriale e lo sviluppo della rete stradale (tab.1).

## CONCLUSIONI

I risultati ottenuti in questo lavoro si basano su un'ipotesi molto forte che è quella di considerare un'offerta di trasporto omogenea in tutti gli aeroporti siciliani, sia in termini di numero di voli che di rotte disponibili.

Un dato importante che è emerso è la conferma che utilizzare software open source non solo consente di lavorare in maniera efficiente, al pari di software con licenza tradizionale, ma consente una maggiore flessibilità d'uso e un più facile adattamento alle più svariate applicazioni.

Sicuramente uno sviluppo futuro di questa ricerca dovrà prendere in considerazione, nella valutazione dell'accessibilità alle infrastrutture aeroportuali, non solo il mezzo di trasporto privato ma anche le altre modalità di trasporto pubblico presenti, sia esse su gomma che su ferro. Inoltre, un ulteriore passo avanti dovrà essere fatto considerando l'effettiva offerta da parte dei singoli aeroporti e quindi sganciarsi dall'ipotesi di offerta omogenea.

La stessa applicazione resta interessante per studi relativi ad altre offerte di servizi (commercio, istruzione, sanità, etc.), specialmente nella valutazione programmatica delle allocazioni e nelle refluenze determinate simulando perturbazioni della rete di trasporto, mediante l'ipotesi di nuove infrastrutture o la variazione delle caratteristiche delle stesse.

## Bibliografia

- [1] De Neufville, Amedeo R. Odoni (2003), *Airport system-planning design and management*, Mc Grow Hill.
- [2] Cantarella G. E. (2001), *Introduzione alla tecnica dei trasporti e del traffico con elementi di economia dei trasporti*, UTET.
- [3] Pietro Russo, Letizia Leonardi (2010), *Database spaziali e loro interazione con QuantumGIS*, tesi di laurea: Università degli studi di Modena e Reggio Emilia, Facoltà di Ingegneria, Corso di Laurea in Ingegneria Informatica.
- [4] Renda Renato, La Franca Luigi (2004), *Accessibilità aeroportuale, il caso di Palermo*, tesi di laurea: Università degli studi di Palermo, Facoltà di Ingegneria, Corso di Laurea in Ingegneria Aerospaziale.