



POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Pianificazione delle evacuazioni di massa: approcci alla gestione dell'infrastruttura secondo modelli comportamentali

Tesi di Laurea Magistrale in
Mobility Engineering

Autore: **Leonardo Vincenzo Rossi**

Matricola: 977232

Relatore: Prof. Ing. Roberto Maja

Co-relatore: Prof. Maurizio Bruglieri

Anno Accademico: 2022-2023

*AUDENTES
FORTUNA
IUVAT.*

Alla mia famiglia!

Sommario

Le evacuazioni di massa costituiscono contemporaneamente una ricorrenza drammatica in quasi ogni paese del mondo e una sfida intellettuale notevole per un Ingegnere dei Trasporti o della Sicurezza. Le circostanze che generano l'evacuazione sono intrinseche dei luoghi in cui avvengono le emergenze, ma presentano delle caratteristiche comuni che permettono una trattazione analitica dei flussi veicolari, se ambientiamo il problema in un contesto di Pianificazione dei Trasporti su strada. In questo elaborato, dopo una revisione generale delle scelte modellistiche che possiamo attuare, incorporate in una struttura ad albero che tiene conto sia delle specifiche pratiche, sia delle risorse matematiche per trattarle, vengono presentati dei modelli di assegnazione su grafo innovativi che utilizzano la Teoria dei Giochi e la Ricerca Operativa. I vari possibili approcci al problema, divisi in funzione del diverso grado d'intervento statale, sono presentati nei minimi dettagli modellistici e poi applicati su grafi di prova e sul grafo reale della cittadina di Sioux Falls, South Dakota. **Parole chiave:** Evacuazioni di massa, Pianificazione dei Trasporti, Teoria dei Giochi, Ricerca Operativa, grafo stradale, trasporti in emergenza.

Abstract

Mass evacuations represent, at the same time, both a dramatic recurrence, in almost every nation in the world, and an interesting challenge for Transportation and Safety Engineers. Evacuation-generating circumstances are inherently related to where they take place in; however, they show some common features which allow us to describe the vehicles flow with an analytic methodology, after setting the Transportation Planning problem on a road graph. In this work, firstly we present a detailed review of the modelling choices that can be undertaken, both for the practical aspects and the mathematical tools to deal with them. All these aspects are set into a decision-tree structure. Afterwards, we describe some innovative models and algorithms for the evacuees' flow assignment with the help of Game Theory and Operations Research. The various intervention approaches, split depending on the degree of state involvement and power, are illustrated with a high level of detail and then applied to some trial graphs; the case study for application is the city of Sioux Falls, South Dakota.

Keywords: Mass evacuation, Transportation Planning, Game Theory, Operations Research, road graph, emergency transportation.

Indice

| | |
|---|------------|
| | iii |
| Sommario | i |
| Abstract | iii |
| Indice | v |
| | |
| Introduzione | 1 |
| | |
| 1 Albero degli scenari: classificazione delle evacuazioni | 3 |
| 1.1 Albero degli scenari | 3 |
| 1.2 Prevedibilità del fenomeno | 3 |
| 1.3 Livello di coercizione | 6 |
| 1.4 Trattazione dei flussi e dimensione temporale | 8 |
| 1.5 Ricorso alla Teoria dei Giochi | 9 |
| 1.5.1 Giochi contro un demone | 10 |
| 1.5.2 Giochi tra viaggiatori | 10 |
| 1.5.3 Giochi tra autorità | 11 |
| 1.5.4 Giochi tra viaggiatori e autorità | 11 |
| 1.6 Ricorso alla Programmazione Matematica | 11 |
| 1.7 Assunzioni della tesi | 12 |
| | |
| 2 Panoramica dello stato dell'arte | 15 |
| 2.1 Teoria dei Giochi e applicazioni | 15 |
| 2.1.1 Applicazioni trasportistiche | 16 |
| 2.2 Ricerca Operativa e applicazioni | 20 |
| 2.3 Pianificazione dei Trasporti | 21 |
| 2.3.1 Pianificazione classica e metodi di assegnazione | 21 |
| 2.3.2 Trasporti emergenziali, trasporti pericolosi e vulnerabilità della rete | 21 |

| | | |
|----------|---|-----------|
| 2.3.3 | Evacuazioni di massa | 22 |
| 3 | Approccio I: Imposizione Totale | 25 |
| 3.1 | Ipotesi del modello | 25 |
| 3.1.1 | Divergenze | 26 |
| 3.1.2 | Inversioni del senso di marcia | 26 |
| 3.1.3 | Formulazione ai nodi | 28 |
| 3.2 | Modello di ottimizzazione | 30 |
| 3.2.1 | Vincoli sugli archi | 30 |
| 3.2.2 | Vincoli di fattibilità | 30 |
| 3.2.3 | Vincoli sui flussi | 31 |
| 3.2.4 | Modello completo | 32 |
| 4 | Approcci II e III: Semplice Informazione e Prescrizione Parziale | 35 |
| 4.1 | Generazione domanda ai nodi | 35 |
| 4.2 | Scelta del percorso | 37 |
| 4.3 | Flussi di prossimità | 41 |
| 4.4 | Flussi e costi d'arco | 43 |
| 4.4.1 | Efficacia dell'evacuazione | 46 |
| 4.5 | Prescrizione Parziale | 47 |
| 5 | Applicazione dei modelli | 51 |
| 5.1 | Applicazione su piccoli grafi di prova | 51 |
| 5.1.1 | Grafo con 6 nodi | 51 |
| 5.1.2 | Grafo con 14 nodi | 56 |
| 5.2 | Applicazione ad un grafo reale: Sioux Falls | 59 |
| 5.2.1 | Sioux Falls, South Dakota | 59 |
| 5.2.2 | Simulazioni e risultati | 62 |
| 6 | Conclusioni e sviluppi futuri | 65 |
| | Bibliografia | 69 |
| A | Appendice A | 73 |
| A.1 | Semplice Informazione - 6 nodi | 73 |
| A.2 | Imposizione Totale - 6 nodi | 77 |

| | | |
|----------|---|-----------|
| A.3 | Funzioni ausiliarie | 82 |
| A.3.1 | Logit | 82 |
| A.3.2 | BPR | 82 |
| A.3.3 | Generazione domanda distribuzione normale | 82 |
| A.3.4 | Parametri distribuzione uscita | 82 |
| A.3.5 | Calcolo fattori di pericolo | 82 |
| A.3.6 | Dijkstra | 83 |
| B | Appendice B | 85 |
| B.1 | Parametri comuni | 85 |
| B.2 | Parametri Sioux Falls | 85 |
| B.3 | Parametri d'intervento - Imposizione Totale | 87 |
| | Elenco delle figure | 89 |
| | Elenco delle tabelle | 91 |
| | Ringraziamenti | 93 |

Introduzione

Questo elaborato è stato sviluppato con lo scopo di applicare la modellistica matematica alla pianificazione dei trasporti emergenziali propria delle evacuazioni di massa. L'obiettivo è sviluppare una serie di modelli per valutare quali siano gli approcci migliori per la gestione pratica dell'evacuazione; i risultati finali dei modelli sono funzionali allo sviluppo dei piani di emergenza e all'ottimizzazione dell'allocazione delle risorse sul territorio (Forze Armate, Forze dell'Ordine, Protezione Civile).

Le domande a cui cerchiamo di rispondere nelle pagine seguenti sono frutto di riflessioni maturate nei 3 campi (eterogenei) della modellistica matematica, della Pianificazione dei Trasporti e della Protezione Civile:

- Come inquadrare matematicamente un problema così delicato?
- È possibile applicare modelli e metodi matematici più avanzati rispetto a quelli tipici della disciplina? Se sì, quali miglioramenti otteniamo nell'attendibilità dei risultati? A che costo computazionale?
- Come possiamo gestire, da Ingegneri dei Trasporti, fenomeni di massa come la congestione o la mobilitazione di interi centri abitati in un contesto di emergenza? E per quanto riguarda i fenomeni individuali, come il panico, la fretta o l'intenzione di cambiare percorso in corso d'opera?
- Per lo Stato, è più efficace intervenire in maniera coercitiva o lasciare autonomia ai cittadini?
- Esistono opzioni intermedie, né puramente coercitive, né eccessivamente caotiche?

L'inquadramento modellistico del problema viene affrontato in maniera dettagliata nel Capitolo 1, in cui tentiamo di armonizzare le varie possibilità modellistiche all'interno di uno scenario di riferimento comune e generale, qui definito *Albero degli scenari*.

Nel Capitolo 2 illustriamo una panoramica dello Stato dell'Arte sui diversi settori da cui l'elaborato attinge.

Il Capitolo 3 presenta l'approccio puramente coercitivo, definito *Imposizione Totale*, sviluppato come applicazione di algoritmi di Programmazione Matematica. Il Capitolo 4,

suo contraltare, presenta il modello di assegnazione del traffico nel caso di completa libertà, o *Informazione Semplice*. Questo modello costituisce il cuore dell'elaborato, in quanto frutto esclusivo del lavoro dell'autore: partendo da osservazioni di natura comportamentale proprie della Teoria dei Giochi e della Psicologia, approdiamo a risultati numerici di immediata utilità. Alla fine del capitolo viene presentata anche una modalità d'intervento intermedia tra le prime due, la *Prescrizione Parziale*, anch'essa proposta innovativa non presente in letteratura.

I tre approcci vengono applicati numericamente nel Capitolo 5, prima su grafi di prova, poi sul grafo reale di Sioux Falls, South Dakota, in modo da poterne confrontare l'efficacia e la fattibilità su un caso studio reale. Le conclusioni e le proposte di miglioramento e sviluppo futuro sono riportate nel Capitolo 6.

1 | Albero degli scenari: classificazione delle evacuazioni

L'obiettivo di questa prima parte dell'elaborato è di delineare una metodologia generale di classificazione e modellizzazione matematica delle possibili situazioni emergenziali, in modo da poter ricondurre le tematiche chiave dell'evacuazione a problemi matematici afferenti all'Ingegneria dei Trasporti.

1.1. Albero degli scenari

La metodologia esposta di seguito si articola in una struttura ad albero. Ogni livello di scelta si riferisce a caratteristiche del fenomeno emergenziale o del modello matematico che verrà adottato per trattare il problema, includendo assunzioni comportamentali, strumenti matematici e contesto modellistico. Si segnalano due conseguenze auspicabili dell'utilizzo di quest'albero:

1. In linea teorica, la capacità di affrontare quasi ogni problema si presenti, a patto di farlo ricadere nella corretta "foglia" dell'albero;
2. Qualora risultasse poco intuitivo classificare il problema, è possibile estendere la ramificazione della struttura, includendo altri criteri di diversificazione delle emergenze o del modo in cui vengono affrontate, a seconda della specificità dell'evacuazione in esame.

Di seguito, si presenta l'elenco delle variabili di riferimento.

1.2. Prevedibilità del fenomeno

La prima caratteristica da valutare, nonché la più determinante per discriminare tra due modelli d'evacuazione, è la prevedibilità degli eventi scatenanti. Nel caso di un disastro naturale ragionevolmente prevedibile, a seconda delle tempistiche a disposizione e dell'attendibilità delle nostre previsioni, è possibile intervenire sul territorio e sulla

popolazione in ottica tanto preventiva, quanto protettiva. Ad esempio, supponiamo di inferire dai dati meteorologici che un dato evento idrologico, come la piena del Po, si verificherà nelle prossime 72 ore. L'allerta diramata dalla Protezione Civile, con l'ausilio delle Forze dell'Ordine e dei Vigili del Fuoco, può essere sufficiente a limitare i danni a cose e persone, qualora l'evacuazione venga gestita correttamente, grazie all'abbondante finestra temporale in cui è possibile sfollare i cittadini. Dunque, d'ora in poi parleremo in maniera del tutto equivalente di un'emergenza/un disastro prevedibile e dell'evacuazione pianificabile/differibile che ne deriva. A titolo d'esempio, ricadono in questa categoria:

- le evacuazioni derivanti da vari fenomeni idrologici, come piene, alluvioni ed inondazioni;
- il disinnescamento controllato di ordigni bellici inesplosi, ritrovati durante le attività di scavo, come avviene per i cantieri stradali;
- procedure estremamente complesse o pericolose che prevedano operazioni di detonazione o demolizione controllate.

Per contro, fenomeni del tutto imprevedibili o derivanti da incidenti artificiali generano complicazioni enormi nel processo d'evacuazione, che possiamo schematizzare come segue:

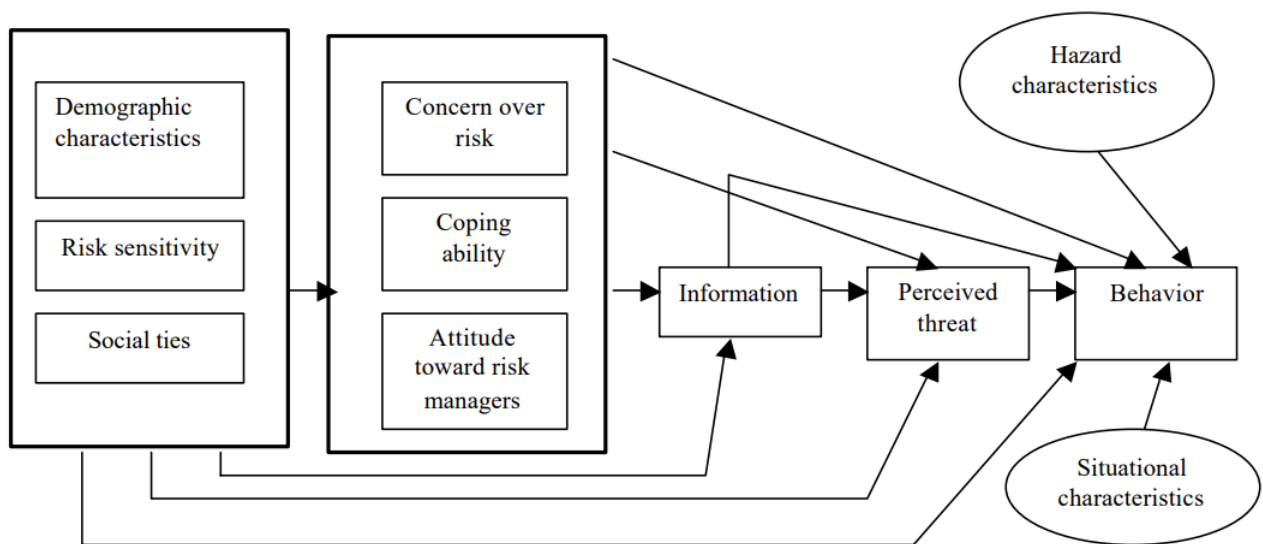


Figura 1.1: Modello comportamentale in caso di evacuazione di massa. Tratto da [2].

Questo schema di comportamento è totalmente analogo a quanto avviene in altri contesti, come gli sport e gli incidenti stradali. Ad una breve ed inevitabile fase di reazione/percezione, seguono la fase di percezione, necessaria per prendere consapevolezza dell'accaduto e decidere il da farsi, e la fase d'azione vera e propria. Nel nostro contesto, agire equivale

a mettersi in moto lungo il grafo stradale, se l'utente della rete ha deciso di andarsene dal luogo interessato dall'evento. Dal punto di vista modellistico-transportistico, possiamo affermare che, dopo la reazione, gli utenti si mettono in moto con tempistiche e modalità varie estremamente dipendenti dal contesto territoriale e dalla sensibilità individuale; perciò, in questo testo si consiglia di affrontare le suddette modalità e tempistiche da un punto di vista statistico, mediante una media in probabilità:

$$N_t = \alpha * N_{TOT} * \mathbb{P}(t) \quad (1.1)$$

dove N_t indica il numero di persone che entreranno nel grafo stradale al tempo (discreto) t , α è la percentuale delle persone totali N_{TOT} che decidono effettivamente di lasciare il luogo interessato e $\mathbb{P}(t)$ è la probabilità di immettersi nel grafo in quell'istante, derivante da una distribuzione discreta di probabilità. Questa modalità di calcolo riprende esattamente la metodica di ripartizione media dei flussi adottata da [19] (e ripresa da [45]) per la trattazione del Modello a 4 fasi. Ricadono nella seconda categoria di evacuazioni quelle causate da:

- Terremoti, eruzioni vulcaniche, tsunami (in generale, fenomeni naturali di origine tettonica e quindi di elevatissima imprevedibilità);
- Rilasci incidentali di sostanze chimiche, incendi, esplosioni di inaudita gravità, provenienti da stabilimenti industriali ad altissimo rischio (a.e. attività sottoposte a SEVESO);
- Eventi analoghi a quelli del punto precedente, ma originati da incidenti stradali o ferroviari legati al trasporto di merci pericolose (cfr. RID e ADR).

Veniamo ora ad alcune differenze sostanziali nella trattazione delle due categorie.

1. Nel caso di un'evacuazione pianificata, possiamo distinguere tre problemi trasportistici successivi: la predisposizione delle risorse sul territorio, l'evacuazione vera e propria dei cittadini e la seconda mobilitazione delle risorse, in un momento successivo, per soccorrere eventuali feriti e cercare di ripristinare la viabilità e la sicurezza. Se l'evacuazione è improvvisa, ragionevolmente le tre fasi si sovrapporranno e confonderanno, complicando sia la riuscita complessiva delle azioni di soccorso, sia la modellizzazione del problema.
2. Un evento imprevisto potrebbe comportare delle complicazioni pratiche su entrambi i fronti della domanda e dell'offerta di trasporto.

- La domanda di trasporto è elevatissima per i soccorsi, mobilitati in maniera ingente ed improvvisa. Al contrario, per i cittadini la domanda scende notevolmente: nel caso di un terremoto, il parametro α dell'eq. (1.1) serve a tener conto delle vittime e di tutti coloro che risultano impossibilitati a spostarsi dopo il terremoto (a causa di familiari da soccorrere, del pericolo scosse di assestamento, ecc.).
- L'offerta di trasporto è un fattore critico, in quanto potrebbe risultare gravemente alterata o del tutto compromessa in seguito a eventi dalle conseguenze particolarmente ingenti. Ad esempio, a livello topologico, un terremoto potrebbe interrompere fisicamente vari archi del grafo, separando intere sezioni, fino alla completa perdita di connessione di certi nodi; a livello macroscopico, le variabili del deflusso (densità k , flusso f e velocità v) verrebbero soggette a vincoli estremamente restrittivi. Nel caso di evacuazioni programmate, le restrizioni topologiche e fisiche che possono derivare sono, a livello puramente teorico, di più semplice trattazione. Riprendendo l'esempio della perdita di connessione del grafo, [44] affronta una problematica analoga, ma la complessità risulta minore, in quanto gli archi critici sono già noti preventivamente, prima dell'emergenza. Nei capitoli 3 e 4 verranno descritti dei modelli matematici per trattare proprio uno di questi casi; la relativa applicazione verrà descritta nel capitolo 5.

1.3. Livello di coercizione

La seconda variabile d'interesse è il livello di coercizione, definito qualitativamente sulla base del grado di intervento statale nella gestione dell'evacuazione. Idealmente, potremmo immaginare il livello di coercizione come un punto sul segmento $[0,1]$.

Se ad un estremo "1" descrive il caso in cui lo Stato interviene in maniera totalmente coercitiva e con poteri dittatoriali, "0" indica l'assoluta libertà decisionale ed operativa degli utenti della rete stradale. Il caso coercitivo viene qui definito *Imposizione totale*: idealmente, le Forze Armate e le Forze dell'Ordine hanno il potere, fucili alla mano, di imporre il percorso ad ogni singolo cittadino, veicolo o plotone omogeneo di veicoli, a seconda del grado di dettaglio del modello e della simulazione. All'opposto, il caso "0" rappresenta uno scenario più vicino a quello di uno stato democratico occidentale, ma pur sempre ideale, vista la situazione emergenziale. Questo caso viene definito *Semplice Informazione* e prevede come assunzione la completa informazione degli utenti; la completa informazione di cui parliamo in questo paragrafo è intesa in senso trasportistico, per

distinguere dai concetti di completa e perfetta informazione nel campo della teoria dei giochi (cfr. Capitolo 2): si ipotizza che i cittadini siano perfettamente a conoscenza dei percorsi disponibili, che siano coscienti della gravità del disastro in corso/in arrivo, che possano stimare i costi d'arco o di percorso su tutta la rete. Si assume, inoltre:

1. che la funzione utilità d'arco o di percorso riferita ad ogni utente sia non-crescente rispetto al tempo, per tenere conto del pericolo;
2. che gli utenti siano in condizione di operare liberamente una *scelta mista preventivo-adattiva*, o *re-routing*. Questa possibilità, oltre ad aggiungere maggiore realismo al modello, e a complicare l'aspetto computazionale della ripartizione dei flussi, genera anche interessanti effetti di congestione sulla rete.

Rimane da descrivere il caso intermedio, corrispondente ad un qualunque punto sull'intervallo aperto $(0, 1)$ e definito *Prescrizione Parziale*. A livello concettuale, possiamo vedere la Prescrizione Parziale come un caso particolare di Informazione Semplice, calato in un contesto molto più restrittivo sulle variabili del deflusso. Queste restrizioni possono essere quantificate come vincoli matematici e, in perfetta analogia alla sezione precedente, possiamo considerare due tipi fondamentali di restrizione:

- Vincoli fisici, corrispondenti a limitazioni dirette sulle variabili del deflusso. Ad esempio, in un contesto di macrosimulazione, potremmo imporre una velocità massima o un flusso minimo o massimo su una certa sezione della rete, o su tutti gli archi, ecc. In equazioni, riferendoci al generico arco dal nodo i al nodo j , indicando con v_{ij} la velocità sull'arco, con f_{ij} il flusso e con k_{ij} la densità veicolare:

$$\begin{cases} v_{ij} \leq v_{MAX} \\ f_{ij} \in [f_{MIN}, f_{MAX}] \\ k_{ij} \leq k_{LIM} \end{cases}$$

- Vincoli topologici. In questo caso, le restrizioni al moto degli utenti provengono da problemi di determinati archi, nodi o percorsi, come riduzioni della capacità o interruzioni di percorrenza. È il caso della perdita di connessione citata prima per il terremoto o nel caso di paesi inondatai, come affrontato da [44]. Ad esempio, se indichiamo con U_{ij} la capacità dell'arco $i \rightarrow j$ e con a_{ij} il generico elemento della matrice di incidenza di un grafo, abbiamo:

$$\begin{cases} U_{ij}^t = \frac{1}{2} U_{ij}^0 \\ a_{ij} = 0 \quad \forall i \end{cases}$$

1.4. Trattazione dei flussi e dimensione temporale

Veniamo ora alle possibili caratteristiche delle variabili del problema.

- Una trattazione macroscopica dei flussi è più intuitiva se ci riferiamo alla Pianificazione dei Trasporti classica, che segue questo schema di assegnazione:

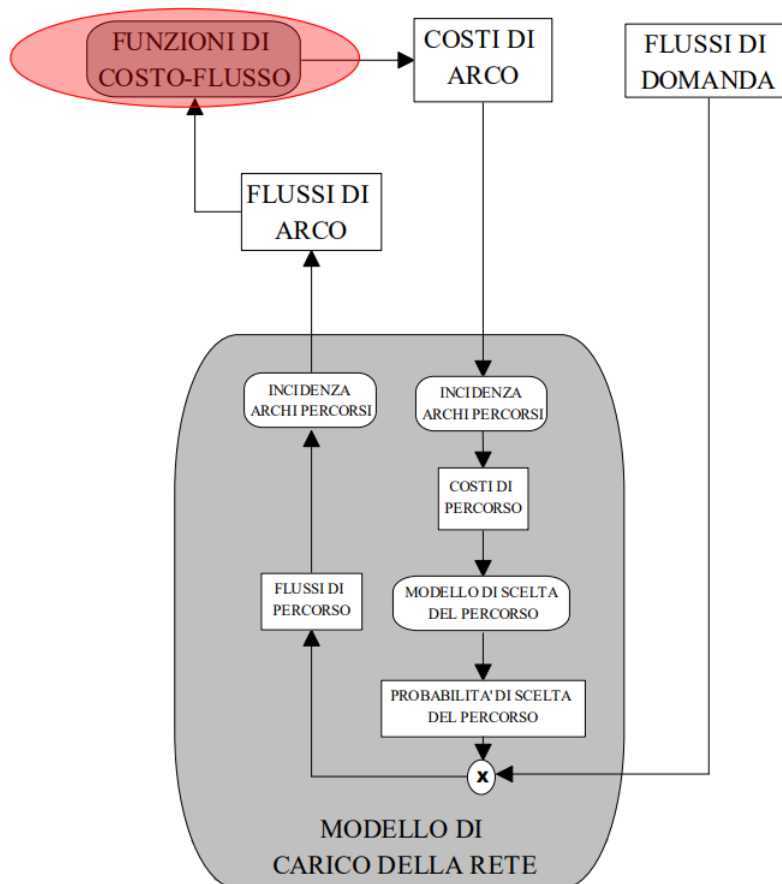


Figura 1.2: Ciclo di calcolo delle variabili d'arco e di percorso in un contesto macroscopico.

La trattazione microscopica è potenzialmente più performante, ma richiede una calibrazione minuziosa e complicata dei parametri, nonché il ricorso a simulazioni stocastiche per generare i modelli comportamentali. I modelli mesoscopici, infine, presentano potenzialità e difficoltà intermedie tra le altre due, aggregando blocchi di utenti della rete in *plottoni* discreti. In questo elaborato, nel Capitolo 4, è stato adottato un modello macroscopico innovativo, che tiene conto del comportamento individuale e calcola i flussi mediati sulle decisioni degli utenti.

- La dimensione temporale del problema e dei flussi definisce in maniera cruciale la complessità e l'affidabilità dei risultati. I modelli a tempo continuo, intrinsecamente

più realistici di quelli a tempo discreto, richiedono solitamente uno sforzo modellistico superiore e si prestano meglio a modelli microscopici. Un altro elemento di distinzione è la natura temporale dei flussi, istantanei o mediati. I flussi istantanei sono di più difficile trattazione e si adattano proficuamente a modelli microscopici o mesoscopici a tempo continuo. L'alternativa è di calcolare le medie dei flussi su un intervallo di tempo di riferimento, che nel caso dei modelli a tempo discreto deve coincidere con il passo di discretizzazione. In generale, la seconda alternativa è da preferire in quanto estremamente più conveniente nelle applicazioni ai flussi veicolari. I flussi istantanei trovano adeguato spazio nella descrizione dei flussi pedonali, ma non entriamo nel merito della materia. In analogia a quanto detto prima per i modelli macroscopici, per questo lavoro è stato raggiunto un compromesso tra le possibili scale temporali di riferimento. Infatti, il modello a tempo discreto che verrà descritto nei prossimi capitoli presenta un passo di discretizzazione talmente fine e paragonabile a quello dei movimenti degli utenti sul grafo da omogeneizzarne perfettamente i risultati con quelli di un modello a tempo continuo.

1.5. Ricorso alla Teoria dei Giochi

[35] è il testo di riferimento su cui è stata impostata questa sezione: ogni documento citato e descritto nella sezione 2.1.1 dello Stato dell'arte può essere inquadrato nell'ottica seguente. Esistono 3 possibilità di applicazione della Teoria dei giochi ai problemi trasportistici e, nello specifico, ai problemi di evacuazione:

- Un'applicazione completa della Teoria dei Giochi, impostando tutta la trattazione in funzione della risoluzione di un gioco specifico;
- Un utilizzo limitato e superficiale, che si traduce nella Pianificazione Classica con modelli di equilibrio. L'equilibrio di costi e flussi di una rete, descritto in [19], [45] e [47] tra gli altri, non è altro che un particolare *equilibrio di Nash* (vedere a.e. [31]) tra gli utenti della rete;
- Una fruizione ridotta dei modelli di Teoria di Giochi, utilizzata in maniera strumentale per supportare determinate ipotesi o per tradurre alcuni comportamenti e tendenze in vincoli ed equazioni.

In questo lavoro, è stato fatto ricorso alla terza modalità, come si vedrà nel Capitolo 4 ed in particolare nella sezione 4.1 e 4.2. Descriviamo le 4 modalità descritte in [35].

1. Giochi contro un demone;

2. Giochi tra viaggiatori;
3. Giochi tra autorità;
4. Giochi tra viaggiatori e autorità.

1.5.1. Giochi contro un demone

La prima categoria di giochi descritta è senz'altro la più frequente in letteratura. Esempi notevoli sono ad opera di G.M.H. Bell, come [7], [9], ecc. Fondamentalmente, si imposta un gioco tra due sfidanti. Il primo è comunemente definito *demone*, o *entità demoniaca*, e rappresenta figurativamente la congestione (come avviene in [42] e molti altri); talvolta il demone coincide con un vero e proprio oppositore fisico, come può essere l'organizzazione terroristica di [16]. All'altro capo, si trova lo Stato, o in generale l'*entità benevola*. Le due entità si confrontano in un *gioco a somma zero* e puntano ad esiti diametralmente opposti. Nel caso di [7], ripreso da [35], il giocatore buono cerca di minimizzare il tempo complessivo dell'evacuazione (o le vittime, o i costi, ecc.), condizionato dal fatto che il demone massimizzerà la stessa variabile per quanto gli compete (cioè intervenendo sui propri parametri di competenza). La formulazione demoniaca è estremamente flessibile e può essere modificata per tenere conto di altre variabili, come la vulnerabilità della rete, o estesa a più coppie O-D; tuttavia, ipotizzare la *natura matrigna* di Leopardi, che manda le cose nel verso peggiore possibile, può comportare una decisa e pericolosa sovrastima dei tempi e delle risorse necessarie. Si cita [11], in cui si ricercano metodi robusti di ottimizzazione per problemi soggetti ad incertezza: l'introduzione di vincoli sul numero di parametri che possono raggiungere contemporaneamente il valore corrispondente al caso peggiore permette di evitare risultati troppo conservativi. Pur non essendo strettamente orientato ad applicazioni sui trasporti, l'opera introduce delle alternative al *worst case scenario*, scenario di caso peggiore, che viene delineato dall'intervento del demone.

1.5.2. Giochi tra viaggiatori

La seguente categoria di giochi è una generalizzazione del suo capostipite: l'articolo "*Some Theoretical Aspects of Road Traffic Research*" di John Wardrop, in cui espone i principi che portano il suo nome. I giochi di questa categoria sono giochi tra N giocatori, che possono essere sia cooperativi che non-cooperativi. Nel caso dei giochi cooperativi, l'entropia generale della rete è tendenzialmente più bassa, ma sappiamo che, nella realtà, nessuno scenario di traffico è realmente cooperativo. Tuttavia, questa modalità modellistica potrebbe essere sfruttata come alternativa alla Programmazione Matematica nei casi di *Imposizione Totale*. I giochi non-cooperativi tra N giocatori sono una categoria

tanto vasta, quanto di scarsa utilità, a causa dell'estrema complessità dei modelli, che in generale si riducono, per praticità, a delle classiche attuazioni dei suddetti principi di Wardrop o dell'equilibrio di Nash. Per completezza, si citano le due possibili classi in cui questi problemi ricadono:

- Un gioco in cui ogni giocatore lotta contro gli altri $N - 1$;
- Equivalentemente, N giochi tra 2 giocatori, in cui un singolo gioca contro la folla corrispondente agli altri $N - 1$.

1.5.3. Giochi tra autorità

Questa categoria prevede una competizione diretta tra due o più autorità di trasporto, quali possono essere gestori della rete, ministeri, protezione civile, le forze armate o enti di altro genere. Oltre ad essere una branca sottosviluppata rispetto alle altre, in quanto esistono ad oggi pochissimi esempi ad esse riconducibili, questa categoria è anche poco funzionale a descrivere problemi di evacuazione, poiché non c'è concorrenza tra istituzioni!

1.5.4. Giochi tra viaggiatori e autorità

L'ultima classe di giochi trasportistici è una generalizzazione delle prime due. Nei giochi demoniaci, le funzioni utilità dei giocatori sono opposte, mentre nei giochi tra N giocatori l'utilità è uguale per tutti. Nella quarta classe di giochi, solitamente, gli N utenti presentano utilità uguali tra loro e divergenti da quella dell'entità esterna superiore, ma non necessariamente in conflitto con essa. Il caso tipico è quello di un gestore autostradale che, ponendo i propri vincoli di tariffazione, influenza il flusso degli utenti, che a loro volta si orienteranno verso il percorso ritenuto più conveniente. Questi problemi, generalmente risolti ricorrendo a soluzioni di equilibrio di Nash, presentano degli esponenti esemplari, come [41]. In un certo senso, la soluzione presentata nel seguente elaborato appartiene a questa categoria: l'entità benevola (che per noi è lo Stato) deve minimizzare il tempo; qualora il tempo superasse una soglia massima di accettabilità, l'obiettivo diventa minimizzare le vittime. Al contempo, ogni utente, in conflitto con gli altri $N - 1$, deve cercare di minimizzare il proprio tempo di percorrenza da origine (specifica) a destinazione (collettiva).

1.6. Ricorso alla Programmazione Matematica

Si inizia questa sezione precisando immediatamente che non è strettamente necessario utilizzare modelli e algoritmi di Ricerca Operativa per risolvere problemi di evacuazio-

ne; tuttavia, la trattazione di questi problemi, che pongono le basi sulla teoria dei grafi, non può che richiamare spontaneamente concetti di ottimizzazione. In generale, per reti congestionate non è possibile limitarsi alla semplice Programmazione Lineare: l'utilizzo di funzioni come la *BPR* (*Bureau of Public Roads cost function*) che legano i costi d'arco ai flussi d'arco con relazioni polinomiali, preclude la possibilità di risolvere i problemi di assegnazione quando il flusso è naturale e spontaneo; tuttavia, se rilassiamo quest'ultima ipotesi, possiamo utilizzare modelli di Programmazione Lineare, eventualmente Intera o Mista Intera, come fatto da [1] e come ripreso nel Capitolo 3. Dunque, possiamo pensare di risolvere problemi di evacuazione di *Imposizione Totale* se riusciamo ad eliminare le dipendenze non-lineari tra le variabili. Altrimenti, è possibile ricorrere alla Programmazione Non Lineare, ma la complessità di risoluzione del problema e l'attendibilità della soluzione richiedono conoscenze molto avanzate di Ricerca Operativa, superiori a quelle utilizzate per redigere questo elaborato. Un'ulteriore possibilità, attinente sia a problemi lineari che non-lineari, è il ricorso alla *Programmazione Multi-Livello*, *PML*. La maggior parte delle applicazioni prevede due livelli di ottimizzazione, ma esistono anche diversi esempi di programmi tri-livello, come accade in [16]. Solitamente, nel campo dei trasporti la Programmazione Bi-Livello, *PBL*, viene adoperata per risolvere problemi di Teoria dei Giochi. Ricollegandoci alla sezione precedente, ci riferiamo perlopiù a giochi demoniaci; è comunque possibile incontrare anche applicazioni di *PML* a giochi tra viaggiatori, se ci ricollegiamo al caso "1 vs folla", o a giochi tra viaggiatori e autorità (dato che, fondamentalmente, sono giochi demoniaci ripetuti numerose volte).

1.7. Assunzioni della tesi

Tiriamo le somme di questa sezione, specificando in un'unica sezione a quali dei casi elencati sopra ci si è ricondotti per questo elaborato.

Innanzitutto, si è optato per l'analisi di un'evacuazione programmata, come descritto nella sottosezione 1.2. Si reputa che questo tipo di problemi, pur non essendo necessariamente più semplice da formalizzare matematicamente, sia senz'altro più utile da descrivere con modelli complessi. In un contesto applicativo reale, la gestione di fenomeni di evacuazione, già ovviamente delicatissima, non concede grande spazio di manovra per l'attuazione di procedure complicate o per la loro ricezione da parte della cittadinanza; senza sacrificare il senso pratico, la scelta di operare su fenomeni prevedibili consente, in linea teorica, di avere a disposizione risultati numerici effettivamente impiegabili, con le dovute cautele, per la stesura dei piani e delle procedure di emergenza, soccorso e messa in sicurezza, al contrario dei fenomeni imprevedibili, che richiedono una trattazione intrinsecamente più condizionata dalla probabilità. L'incertezza stocastica potrebbe, da un lato, condurre alla

sovrastima di tempi/risorse necessarie/vittime, in analogia a quanto accade con i giochi demoniaci (sottosezione 1.5.1), oppure, dall'altro, alla generazione di fin troppi scenari differenti (topologicamente, per gravità, per difficoltà d'intervento, ecc.). Per fissare le idee, lo scenario considerato idealmente, pur senza fare riferimenti troppo espliciti nella stesura delle equazioni, è quello di una cittadina soggetta ad un'inondazione, in analogia a quanto sviluppato da [44] per il paese lombardo di San Rocco al Porto. Il caso studio affrontato è la città statunitense di *Sioux Falls*, centro industriale di circa 200.000 abitanti, nonché città più popolosa del Sud Dakota.

Ciò detto, la trattazione procede ad affrontare tutti e 3 gli scenari descritti nella sezione 1.3. L'obiettivo è di simulare prima cosa accadrebbe in caso di evacuazione nello scenario completamente libero corrispondente alla Semplice Informazione, per poi valutare in parallelo l'efficacia dell'Imposizione Totale di un contesto coercitivo. I modelli matematici reattivi a questi due approcci verranno presentati in ordine inverso a come sono stati simulati per rendere più chiaro il processo logico e modellistico che è stato seguito. In particolare, la modellizzazione è partita dal problema PLIM presentato nel Capitolo 3, ma gli algoritmi, a seguito di varie prove e numerosi cambiamenti nella struttura del lavoro, sono stati scritti prima per i problemi di assegnazione con Teoria dei Giochi e successivamente per risolvere quelli di Ricerca Operativa. Il terzo approccio, la Prescrizione Parziale, prevede proprio di utilizzare i risultati dell'Imposizione in una versione riorganizzata *ad hoc* dell'algoritmo di assegnazione della Semplice Informazione, nell'ottica di un'ottimizzazione ancora più marcata dei flussi da evacuare. Per quanto riguarda le variabili, è stata adottata una trattazione a tempo discreto con flussi mediati, ma in maniera tale da far confondere i risultati con quelli di una trattazione a tempo continuo, grazie all'adozione di un passo di discretizzazione temporale talmente ridotto (1 minuto) da essere paragonabile ai tempi di percorrenza sugli archi del grafo e ai tempi di reazione e decisione degli utenti. Il ricorso alla Teoria dei Giochi è parziale e puramente strumentale, come già detto, e si colloca nell'ambito dei giochi tra viaggiatori ed autorità; la Ricerca Operativa, invece, viene utilizzata ampiamente per il modello di Imposizione, formulato in Programmazione Lineare Mista Intera.

Prima di procedere con la spiegazione dei modelli e degli algoritmi sviluppati per questa tesi, si procede ad offrire una panoramica dello Stato dell'Arte nei variegati settori da cui questo lavoro ha attinto.

2 | Panoramica dello stato dell'arte

Passiamo brevemente in rassegna lo stato della letteratura di settore sui 3 campi fondamentali su cui poggia questo lavoro.

2.1. Teoria dei Giochi e applicazioni

Dato che questo elaborato non intende trattare le tematiche di teoria dei giochi in maniera perfettamente formale, citeremo soltanto i libri fondamentali consultati per approcciare questa materia complessa. Nello specifico, è stato fatto ricorso a 3 tomi classici: [31] di Robert Gibbons, [43] di Roberto Lucchetti e [52] di Karl Sigmund. Il lavoro di Gibbons è il riferimento primario per il retroterra teorico dell'elaborato, descrivendo dettagliatamente i concetti di *informazione* discussi nel capitolo precedente. Inoltre, la trattazione è sufficiente formale ma estremamente pratica: i numerosi esempi a tema micro-economico sono stati di particolare aiuto per inquadrare i problemi trasportistici nell'ottica dei giochi (il parallelismo tra micro-economia e pianificazione dei trasporti è prassi consolidata, come testimoniato da [45] e [19]). Gli altri due titoli sono molto più formali e teorici e rappresentano un punto fermo nello studio della disciplina.

Uno dei problemi classici di Teoria dei Giochi che ha avuto un'influenza su questo elaborato è il cosiddetto *duopolio di Cournot*. In un contesto microeconomico di concorrenza tra due imprese che producono beni omogenei e soddisfano la stessa domanda di mercato, queste cercheranno di massimizzare il proprio profitto lavorando sulle quantità prodotte. Si rimanda al lavoro di Gibbons o a testi di microeconomia per i dettagli; per questo lavoro, è stato utilizzato un risultato esemplare sul duopolio di Cournot nel Capitolo 4. Imponiamo determinate ipotesi di produzione: domanda di mercato a , quantità prodotta dall'azienda i -esima q_i con costo di produzione $C_i(q_i) = cq_i$, funzione di prezzo $P(q) = a - q = a - q_1 - q_2$, è noto che la strategia ottima per entrambe le imprese sarà produrre $q_1 = q_2 = \frac{a-c}{3}$, che genererà un profitto pari a $\frac{(a-c)^2}{9}$.

Un problema parallelo è il duopolio di Bertrand, in cui le imprese decidono il prezzo di vendita; il risultato è analogo al famoso *dilemma del prigioniero*. Entrambi i problemi sono generalizzabili a n oligopolisti concorrenti, per funzioni di produzione qualsiasi. La procedura generale consiste nel cercare l'intersezione tra le curve di *risposta ottima*, o migliore. L'ipotesi di decisori razionali e perfettamente informati implica che ciascun giocatore agisca per massimizzare il proprio profitto alla luce del fatto che sanno che gli altri agiranno analogamente.

2.1.1. Applicazioni trasportistiche

A partire dai lavori di Wardrop, negli ultimi 50 anni sono stati numerosi i tentativi di applicazione della Teoria dei Giochi al mondo dei trasporti. Per elaborare questa tesi, che inizialmente era stata intesa come una tesi metodologica di amplissimo spettro sulle applicazioni trasportistiche della disciplina, chi scrive ha letto più di 100 articoli sull'argomento; ovviamente, non tutti hanno lasciato il segno nella stesura del lavoro, ma citeremo senz'altro quelli che hanno condizionato assunzioni e modelli. Il prospetto generale delle possibili applicazioni della tematica è stato descritto nella sezione 1.5, in riferimento a [35].

L'autore più prolifico su questo argomento ibrido è senz'altro Michael G. H. Bell. Qui, citiamo i seguenti suoi lavori.

- [7] è l'articolo fondante della sua bibliografia. L'autore imposta un gioco demoniaco per determinare le sezioni della linea di massima vulnerabilità, in un contesto stocastico.
- [9] estende il problema al caso di n utenti che affrontano m demoni, ciascuno interessato ad una coppia O-D.
- [20] e [6], invece, modificano l'obiettivo del problema. Nel primo si massimizza l'affidabilità in senso statistico; nel secondo si minimizzano i costi complessivi dopo aver imposto, non linearmente, una dipendenza di tipo Logit.
- [8] è un'applicazione specifica al trasporto di merci pericolose.

Procedendo oltre, [28] è stato un buon riferimento per l'intuizione di applicare questa branca della matematica a problemi di trasporti in emergenza; parallelamente, anche [42] e [40] hanno spinto in questa direzione, illustrandone l'utilità nel caso di evacuazioni di edifici. Al di là degli strumenti utilizzati dagli autori, tra cui simulazioni Montecarlo e QRA (*Quantitative Risk Assessment*), gli spunti fondamentali ripresi da questi due articoli sono i modelli comportamentali e le procedure di calcolo.

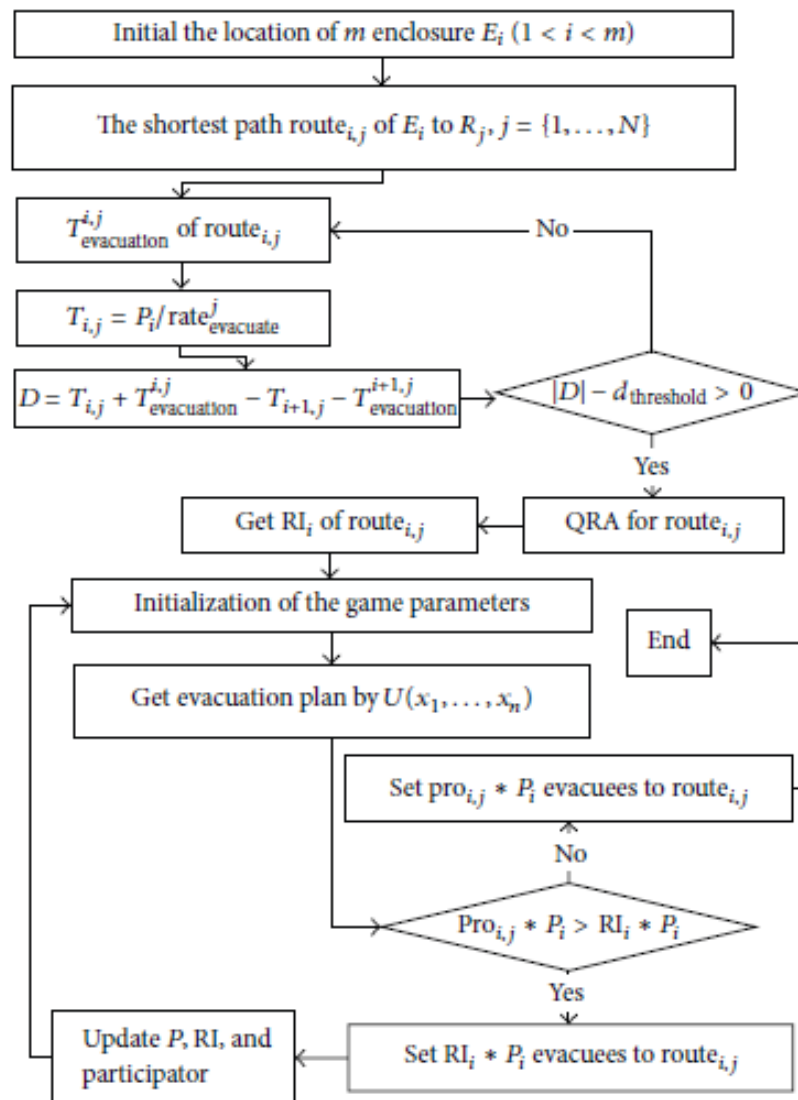


Figura 2.1: Procedura di calcolo per l'evacuazione dell'edificio in [40].

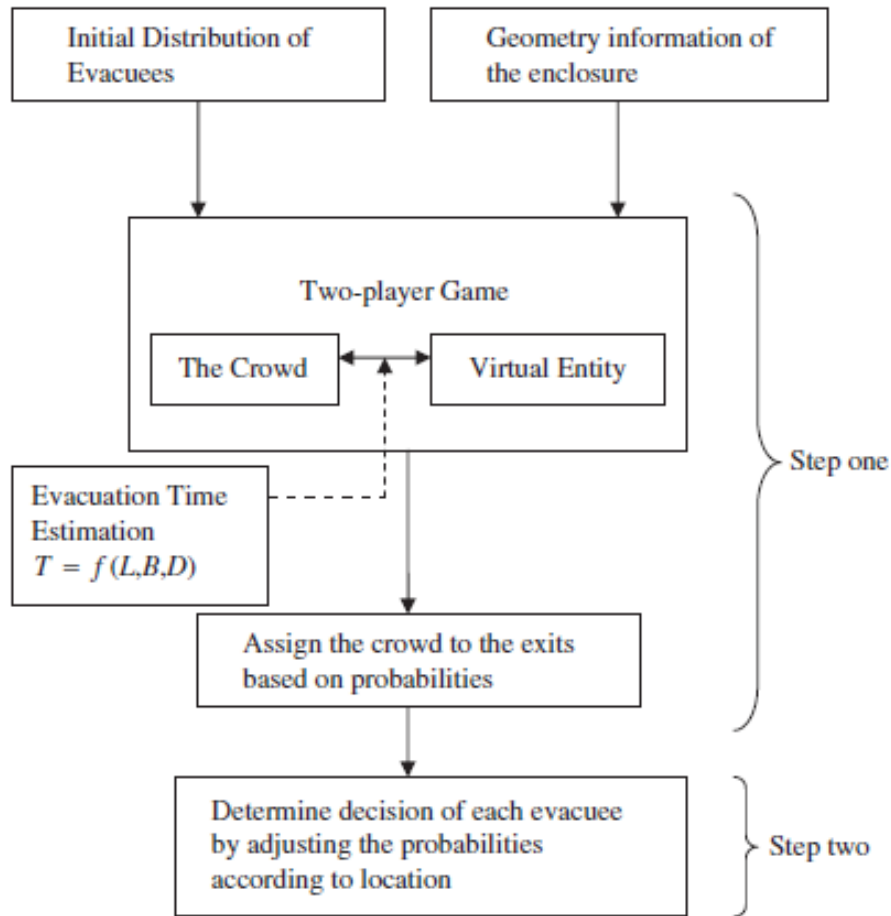


Figura 2.2: Approccio modellistico all'evacuazione dell'edificio in [42].

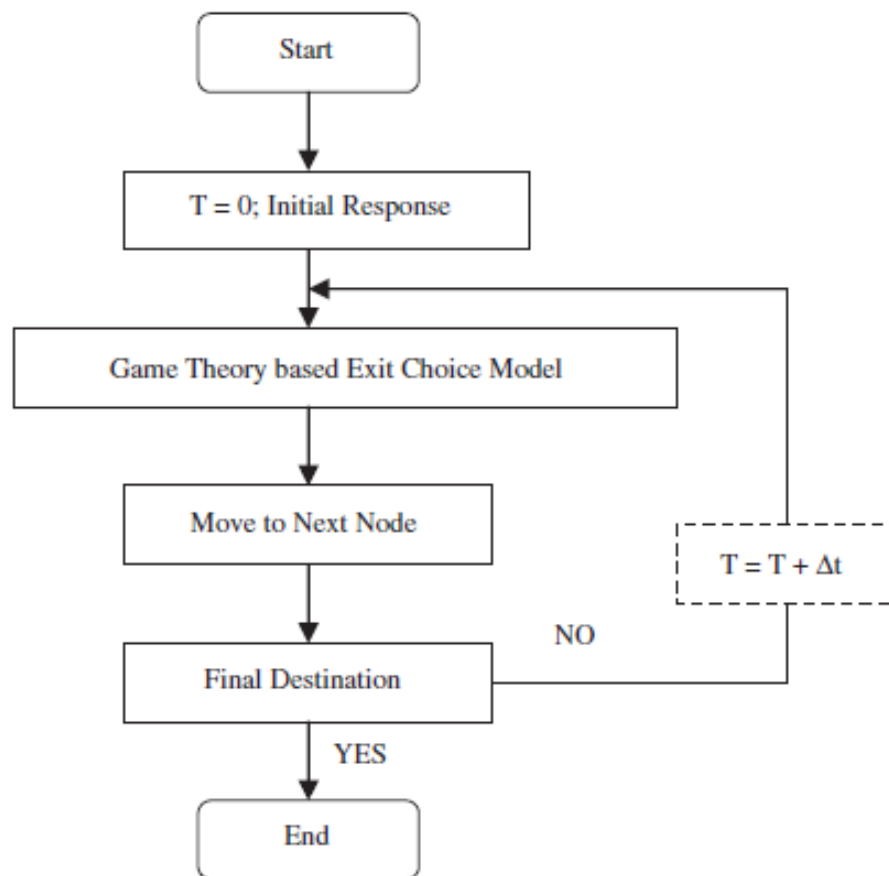


Figura 2.3: Procedura di calcolo per l'evacuazione dell'edificio in [42].

Si cita anche [24], in cui gli autori adottano una variante del modello Logit per descrivere modelli di evacuazione mediante giochi parzialmente cooperativi, traendo ispirazione dai movimenti delle formiche che seguono i feromoni della loro regina.

Un'applicazione estremamente funzionale della Teoria dei Giochi è lo studio a fini di difesa militare o spionistica, uno degli sviluppi più interessanti della cosiddetta *Interdizione Stocastica*, *Stochastic Network Interdiction*. Questi lavori, tra cui [34] e, soprattutto, [16], sono essenzialmente applicazioni molto intelligenti dei giochi demoniaci. [16] è un punto di riferimento per la letteratura di settore, poiché arriva a sviluppare persino modelli di Programmazione Trilivello. Per concludere la sezione sulla Teoria dei Giochi, occorre nominare [1] e [38], intitolati "Dynamics of Disasters". Questi volumi dal titolo altamente esplicativo non trattano soltanto di tematiche trasportistiche, ma abbracciano tutto lo spettro delle pianificazioni emergenziali. L'approccio con Teoria dei Giochi è utilizzato in forme diverse, ma concettualmente sempre simili, in molti capitoli. Il volume 1, [1], è citato numerose volte in questo elaborato.

Infine, spaziando un po' ma rimanendo nella tematica dei giochi su rete, [57] e [54], tesi di

laurea, affrontano tematiche di ingegneria delle telecomunicazioni: il primo verte sui temi della selezione di rete e dell'allocazione delle risorse; il secondo studia la competizione nella distribuzione di contenuti in reti veicolari (fisiche o digitali).

2.2. Ricerca Operativa e applicazioni

Le applicazioni dell'Ottimizzazione su grafo e della programmazione matematica sono pressoché illimitate, specialmente nell'ambito dei trasporti. In questa sezione, discutiamo dei testi consultati per trovare ispirazione per questo lavoro. Testi specifici di Ricerca Operativa, come [5], [27], [15] e [23], sono stati adoperati per sviluppare una conoscenza di base sulle metodiche di programmazione intera e non-lineare.

La maggior parte dei testi consultati per questa sezione è legata alla stretta connessione tra i giochi demoniaci o i giochi *leader-follower* e i problemi multilivello formulati per risolverli. Un gioco leader-follower è un gioco in cui uno dei due giocatori muove condizionatamente alla mossa dell'altro. La dipendenza non è necessariamente temporale: l'importante è che il giocatore *follower* muova sempre sapendo la mossa dell'altro. Di conseguenza:

- La strategia del follower sarà quella che gli permette di massimizzare il proprio profitto sapendo la scelta del leader;
- La strategia del leader consisterà nella scelta, sulle proprie variabili d'influenza, che gli permetta di massimizzare il proprio profitto prevedendo che il follower farà lo stesso.

Il problema cardine in questo frangente è il cosiddetto *duopolio di Stackelberg*, variazione del duopolio di Cournot. I risultati, naturalmente, sono diversi per i due giocatori nei due casi.

Essendoci una linea sottile, labile, tra i problemi leader-follower e i problemi di Ottimizzazione Multilivello, documenti come [16] sono degni di essere citati anche in questa sezione. Altri esempi notevoli sono [4], in cui si ottimizza il trasporto di merci pericolose sulla base delle previsioni delle scelte attuate dai trasportatori, [36], testo estremamente complesso in cui si cercano strategie per l'interdizione stocastica dei percorsi, e [49], che offre una descrizione dettagliatissima del retroterra teorico sui problemi di interdizione.

Rimangono da nominare due testi completamente diversi.

- [30] introduce dei possibili metodi di approssimazione numerica per risolvere problemi di Programmazione Bilivello.
- [11], invece, è un testo molto famoso sull'ottimizzazione robusta. Gli autori, Bertsi-

mas e Sim, sviluppano dei metodi per problemi soggetti ad incertezza sui parametri per lasciarsi alle spalle la letteratura "del caso peggiore" (vedere il paragrafo 1.5.1).

Questi due articoli hanno il merito di aver dissuaso chi scrive dall'attuazione di una formulazione multilivello, spostando così il focus su applicazioni meno rigide della Teoria dei Giochi.

Altre applicazioni notevoli della Ricerca Operativa al mondo dei trasporti sono il trasporto di merci pericolose (già citato prima e sviluppato per bene in [39] e [46]), la logistica di ultimo miglio ([12],[26]) e la valutazione della vulnerabilità della rete ([44], in cui si valuta la perdita di connessione topologica a seguito dell'inondazione del paese lombardo di San Rocco al Porto (LO)).

2.3. Pianificazione dei Trasporti

Rimane da discutere lo stato dell'arte in merito agli aspetti più applicativi della tesi, ovvero la pianificazione dei trasporti, emergenziali e non, e il tema delle evacuazioni.

2.3.1. Pianificazione classica e metodi di assegnazione

La maggior parte delle tematiche di Pianificazione dei Trasporti utilizzate per questo elaborato sono date per assodate, in quanto stabilmente affermate in letteratura, come il Modello a 4 fasi e la struttura ricorsiva delle variabili flusso di percorso-flusso d'arco-costi d'arco-costi di percorso. I testi fondamentali, citati numerose volte nei Capitoli 4 e 5, sono [45], [19], [47], [18]. Un altro testo importante è [51], libro in cui si imposta la pianificazione dei trasporti come successione di problemi di grafo con un'impostazione molto formale matematicamente, ricorrendo in maniera diffusa alla Teoria dei grafi e dei giochi anche per dimostrare problemi classici (a.e. il Paradosso di Braess).

Per completezza, si nominano anche due testi consultati per trarre ispirazione su come modellizzare la psicologia del traffico, [37] e [2].

2.3.2. Trasporti emergenziali, trasporti pericolosi e vulnerabilità della rete

[25] sviluppa un modello mesoscopico di assegnazione per modellizzare l'esodo in caso di evacuazione dal paesino lombardo di San Rocco al Porto, di cui [44] affronta la vulnerabilità dal punto di vista topologico.

Rimanendo nell'ambito di lavori svolti presso il Politecnico di Milano, [14], [21] e [17] affrontano il tema della quantificazione e gerarchizzazione della vulnerabilità di archi e

percorsi in contesti emergenziali, prevalentemente nel caso di fenomeni idrologici anomali. La stessa prerogativa è al centro di numerosi lavori del Politecnico sul trasporto di merci pericolose su strada; tra gli innumerevoli, si nominano [53], [32], [56], [10] e [50]. Si cita anche [55] a corollario della trattazione sui trasporti di merci pericolose: l'autore illustra un'applicazione di metodologie *RAMS* per il trasporto su rotaia.

2.3.3. Evacuazioni di massa

Per quanto riguarda l'aspetto specifico delle evacuazioni di massa, è difficile ritagliare uno spazio nella bibliografia che contenga titoli omogenei tra loro per modalità della trattazione e scopi. L'argomento è talmente vasto e pieno di potenziali complicazioni che, inevitabilmente, sconfinava in un numero praticamente illimitato di altri settori, come le scienze strategiche/militari, la geopolitica, l'ingegneria civile e ambientale, la psicologia e la gestione e pianificazione della sicurezza.

Un testo estremamente vasto, quasi enciclopedico, sull'argomento è [13]: l'autore, ingegnere ambientale ed esperto di pianificazione territoriale, affronta a tutto tondo il tema della gestione delle emergenze sul territorio (conseguenze, normative, metodi di prevenzione, ruoli e responsabilità, ecc.), con focus particolare sulla Protezione Civile e sul suo operato.

Testo radicalmente diverso è [33], in cui vengono illustrati i risultati attuali in letteratura sul problema della modellizzazione matematica delle evacuazioni. È essenzialmente un compendio di applicazioni utili ma molto complesse della Ricerca Operativa alla tematica emergenziale.

Avvicinandoci a quanto prodotto in questo lavoro, la tematica degli alberi di evacuazione, discussa in [1] e nel Capitolo 3, viene approfondita da [3] con modelli molto più complessi. La gestione pratica dei flussi di cittadini, invece, è descritta bene in [29], da cui si riprende il seguente schema concettuale.

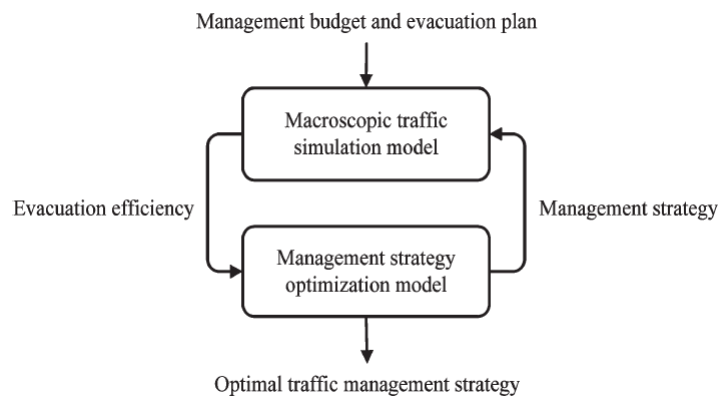


Figura 2.4: Procedura ricorsiva per la gestione ottima di un'evacuazione di massa. Tratto da [29].

Infine, degli spunti interessanti per proseguire da quanto emerge dai risultati di questa tesi sono da ricercare in [48], tesi di laurea sulla gestione dell'assegnazione degli evacuati a più rifugi differenti, sparsi sul territorio, attraverso l'attuazione di un modello PLIM, in naturale rapporto di consecutività con questo lavoro.

3 | Approccio I: Imposizione Totale

Iniziamo ora a presentare il primo dei 3 possibili approcci al problema di evacuazione. D'ora in poi, daremo per scontato di operare in un contesto di emergenza prevedibile come descritto nella sezione 1.2. L'Imposizione Totale che dà il titolo al presente capitolo dev'essere immaginata, come già detto nel Capitolo 1, come una situazione in cui lo Stato, in un contesto coercitivo e dittatoriale, ha il potere assoluto di intervento sulla gestione dell'evacuazione. Questo potere assoluto genera 2 conseguenze:

1. Ad ogni nodo, possiamo ipotizzare di riuscire a distribuire i flussi tra gli archi da esso uscenti in maniera estremamente precisa e puntuale.
2. Preliminarmente all'evacuazione, le Forze dell'Ordine, le Forze Armate, la Protezione Civile, ecc. possono essere disposte in maniera ottimale sul territorio per assicurare la riuscita del punto precedente.

A livello operativo, queste due considerazioni si possono tradurre nella possibilità di modellizzare il problema mediante algoritmi e modelli di Ricerca Operativa (come già detto nel Capitolo 2). In particolare, per il punto 2 di cui sopra si rimanda ad articoli come [16], in cui vengono modellizzati problemi di allocazione delle risorse sul territorio per questioni di intelligenza. Per quanto riguarda il punto 1, invece, seguiremo il modello riportato da Omkar Achrekar e Chrysafis Vogiatzis in [1], opportunamente modificato per adattarlo al presente lavoro.

3.1. Ipotesi del modello

Il presente modello rientra nella categoria di problemi *PLIM*, *Programmazione Lineare Mista Intera*, o *MILP* in inglese. Questa formulazione prevede una funzione obiettivo e dei vincoli lineari nelle variabili di controllo, che possono essere tanto variabili continue quanto intere. In particolare, utilizzeremo variabili continue per descrivere i flussi di

persone, variabili binarie per descrivere l'utilizzo degli archi e una variabile intera positiva *ad hoc* per la gestione delle risorse sul territorio.

Il modello di Achrekar e Vogiatzis poggia le proprie basi sul concetto di *albero d'evacuazione*, già noto in letteratura. Con questo termine ci si riferisce ad un grafo, non necessariamente orientato, in cui:

- Plotoni compatti che partono dalla stessa origine arriveranno sicuramente nella stessa destinazione, seguendo lo stesso identico percorso, ma non necessariamente in contemporanea o impiegando lo stesso tempo totale;
- Da ogni nodo esce sempre un solo arco, ad eccezione, eventualmente, dei nodi finali.

L'albero di evacuazione è una struttura ideata per soddisfare le 3 necessità fondamentali di un'evacuazione, che dev'essere *rapida, intuitiva ed inconsutile* [1]. Tuttavia, Achrekar e Vogiatzis rilassano le ipotesi del modello, aggiungendo la possibilità di introdurre delle *divergenze* e delle *inversioni di senso di marcia* ("contraflow" in originale).

3.1.1. Divergenze

Ad ogni nodo, rilassiamo l'ipotesi che esca un solo arco, pur richiedendo che il numero di archi uscenti sia sempre minore, o al limite uguale, al numero degli archi entranti. La variabile di riferimento sarà:

$$m_i = \text{numero di divergenze concesse per il nodo } i \quad (3.1)$$

3.1.2. Inversioni del senso di marcia

Definiamo:

$$x_{ij} = \begin{cases} 1 & \text{se l'arco } i \rightarrow j \text{ viene utilizzato nel piano d'evacuazione} \\ 0 & \text{altrimenti} \end{cases}$$

$$y_{ij} = \begin{cases} 1 & \text{se l'arco } i \rightarrow j \text{ viene invertito nel piano d'evacuazione} \\ 0 & \text{altrimenti} \end{cases}$$

L'idea dell'algoritmo è duplice: il risultato finale è costituito, da una parte, dallo storico dei flussi, e dall'altra dalla lista degli archi utilizzati nel piano d'evacuazione, specificando quelli eventualmente invertiti. Bloccare determinate strade permette di far confluire i flussi in maniera opportuna, nell'ipotesi di avere a disposizione un solo punto di raccolta.

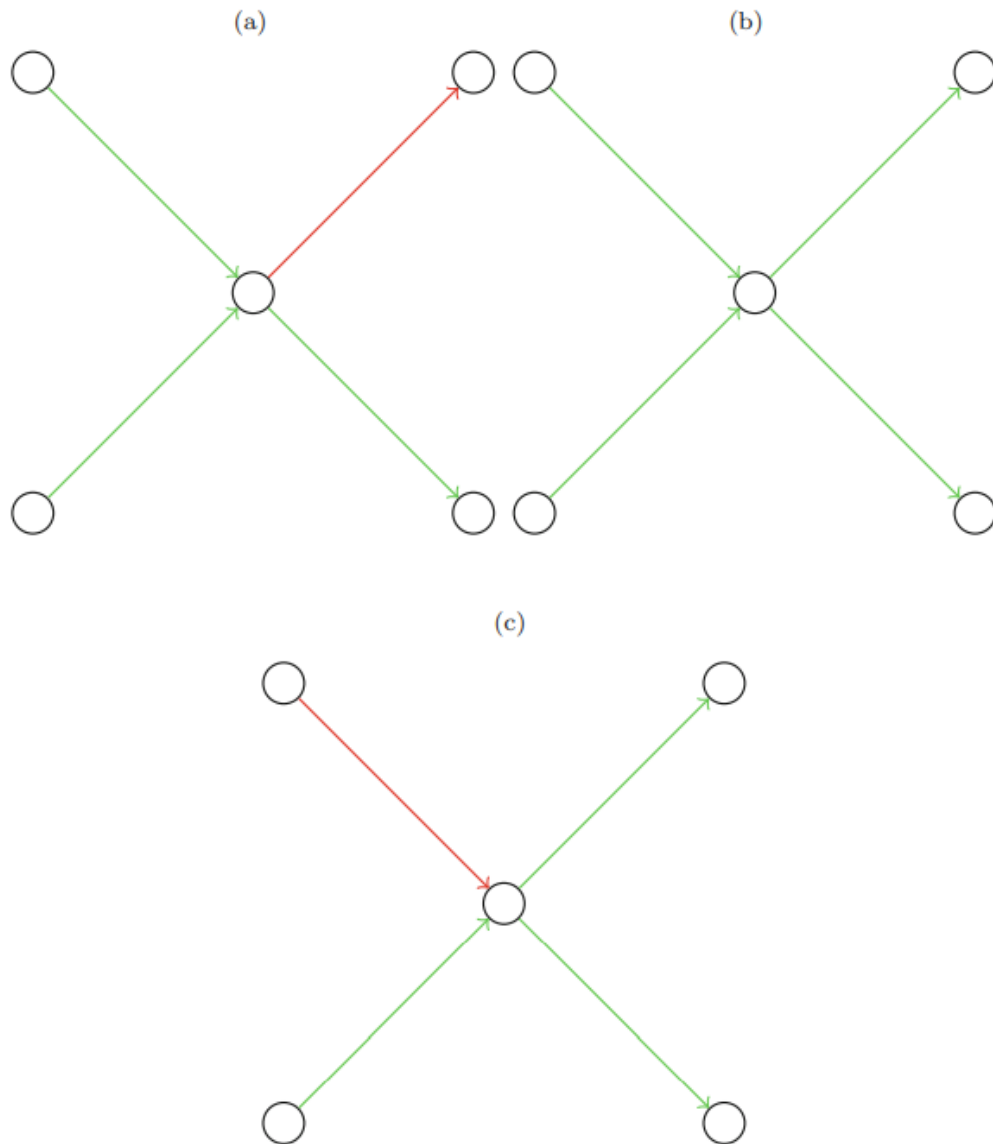


Figura 3.1: **(a)** Concesso. È una sezione ammissibile di un albero di evacuazione. **(b)** Concesso per $m_i = 1$. Introducendo la possibilità di divergere, anche questa sezione è ammissibile, poiché soddisfa le equazioni 3.9 e 3.12. **(c)** Non concesso. Viene violata la condizione che gli archi uscenti non superino quelli entranti.

Invertire il senso di marcia, invece, prevede che una determinata strada venga utilizzata nel senso opposto, per permettere di recuperare la capacità e sfruttarla nel verso più conveniente per agevolare il deflusso.

3.1.3. Formulazione ai nodi

Dopo aver introdotto m_i , x_{ij} ed y_{ij} , definiamo le variabili di controllo rimanenti del problema:

$$f_{ij}^t = \text{flusso veicolare in procinto di lasciare il nodo } i \text{ verso il nodo } j \text{ al tempo } t \quad (3.2)$$

$$\Delta q_i^t = \text{quantità di utenti che si immettono nella rete, uscendo dal nodo } i, \text{ al tempo } t \quad (3.3)$$

Il flusso f_{ij}^t è definito come flusso *in procinto di lasciare il nodo i verso il nodo j* , in maniera funzionale ad una trattazione ai nodi del problema. Questa, ripresa ed estesa nel Capitolo 4, permetterà numerose semplificazioni del problema (eliminando, ad esempio, la necessità di enumerare esplicitamente i percorsi), a patto di affrontare delle complicazioni nella quantificazione temporale dei flussi d'arco. La *domanda d'evacuazione al nodo*, invece, è stata leggermente modificata rispetto alla formulazione di Achrekar e Vogiatzis; qui, prendiamo in considerazione una distribuzione ben precisa degli evacuati sul territorio. La scelta di introdurre $\Delta q_i(t)$ segna una delle differenze dal modello originale. Gli autori hanno utilizzato delle variabili, d_i^t , per indicare la *domanda rimanente in ciascun nodo*; tali variabili, però, vengono utilizzate nei vincoli soltanto come termini incrementali, del tipo $d_i^{t-1} - d_i^t$, ad eccezione della funzione obiettivo:

$$\min \sum_{i \in V} \sum_{t=1}^T r_i^t d_i^t \quad (3.4)$$

che è stata opportunamente modificata in:

$$\min \sum_{i=1}^n \sum_{j \in FS(i)} \sum_{t=1}^T r_j^t f_{ij}^t$$

I parametri r_i^t sono definiti *fattori di pericolo*. Nell'articolo originale, i fattori di pericolo utilizzati sono stati fatti variare per valutare l'effetto sulla soluzione finale. In ogni caso, l'idea è di imporre la minimizzazione del tempo passato in ogni nodo in base alla pericolosità intrinseca della zona in cui si trovano i cittadini. Preliminarmente all'assegnazione dei coefficienti, è necessaria una fase di zonizzazione del territorio che definisca il livello di rischio di ogni nodo; tale zonizzazione è, ovviamente, fortemente dipendente dalle caratteristiche morfologiche, viabilistiche e demografiche del territorio, nonché dal tipo di emergenza che viene valutata.

Le tre zone, S_1 , S_2 e S_3 , sono ordinate per pericolosità *decrescente*: S_1 è la zona di massimo pericolo, da abbandonare il prima possibile; S_2 può essere interpretata come una zona di transizione, di elevata pericolosità, ma abbandonabile con minore urgenza; infine, S_3 è la zona di sicurezza, per cui possiamo assumere tranquillamente che i cittadini giunti nei nodi ad essa appartenenti siano definitivamente al sicuro. A titolo di esempio, nel caso di un'inondazione, possiamo zonizzare il territorio in base alla quota dei nodi, assegnando i settori più in basso alle zone di maggior pericolo. All'interno della zona S_3 troviamo, in particolare, i punti di raccolta, dei nodi sicuri che modellizzano i rifugi o le strutture di soccorso della Protezione Civile. Un'ipotesi fondamentale di questo lavoro è l'unicità del nodo finale, in quanto ipotizziamo di predisporre un solo punto di raccolta. Questa assunzione, utilizzata per motivi di semplice comodità di scrittura degli algoritmi, non pregiudica in alcun modo la generalità del modello, che può essere facilmente esteso a casi più complessi, con più punti di raccolta o più zone.

In questo elaborato, come si può evincere dall'equazione 3.7, i fattori di pericolo vengono riferiti al nodo verso cui ci si sposta, invece che a quello che si lascia. Di conseguenza, sono stati utilizzati i flussi in arrivo ai nodi al posto delle domande. I fattori di pericolo impiegati in questo capitolo sono i seguenti:

$$r_j^t = \begin{cases} t & \forall j \in S_1 \\ r_2 (= 1) & \forall j \in S_2 \\ r_3 (= 0) & \forall j \in S_3, j \neq n \end{cases}$$

$$r_n^t = t - T$$

In maniera simile a quanto descritto in [1], questa scelta dei parametri penalizza maggiormente i nodi più pericolosi, mentre favorisce di molto gli arrivi a destinazione più precoci.

3.2. Modello di ottimizzazione

Iniziamo ora una breve descrizione dei vincoli del modello di ottimizzazione PLIM.

3.2.1. Vincoli sugli archi

Innanzitutto, prevediamo che un arco possa essere invertito soltanto se abbiamo già impiegato l'intera capacità del suo opposto. A livello pratico, non avrebbe alcuna utilità, né alcun senso, invertire l'arco $i \rightarrow j$ senza aver prima utilizzato $j \rightarrow i$; al contrario, una simile scelta genererebbe soltanto confusione e disordine nell'esodo.

$$x_{ij} \geq y_{ji}, \quad \forall i = 1, 2, \dots, n, \quad j \in FS(i)$$

Imponiamo ora l'equilibrio dell'albero di evacuazione in forma rilassata:

$$\sum_{v \in FS(i)} x_{iv} \leq \sum_{k \in BS(i)} x_{ki} \quad \forall i = 1, 2, \dots, n$$

Riportiamo, inoltre, un vincolo "spezza-ciclo" impiegato dagli autori per evitare di far girare i veicoli in tondo indefinitamente.

$$\sum_{i \in S, j \in S \cap FS(i)} x_{ij} \leq |S| - 1 \quad \forall S \subset V \quad (3.5)$$

Si segnala, tuttavia, che questo vincolo non è stato utilizzato nel nostro modello, in analogia con quanto avvenuto in [3]. Nelle prove eseguite per questo elaborato, comunque, il problema dei cicli non si è presentato.

3.2.2. Vincoli di fattibilità

I due vincoli seguenti servono a tener conto delle problematiche reali nell'attuazione dell'evacuazione. Non possiamo ipotizzare di essere in grado di invertire o bloccare tutte le strade che vogliamo per motivi di fattibilità, poiché richiederebbe una predisposizione di risorse eccessivamente ingente, e per motivi di rapidità, poiché renderebbe fin troppo complicata la gestione dei flussi e l'indirizzamento dei cittadini verso i nodi successivi. I vincoli imposti sono sul massimo budget per invertire gli archi e per permettere divergenze dall'albero di evacuazione ideale:

$$\sum_{i=1}^n \sum_{j \in FS(i)} \bar{C}_{ij} y_{ij} \leq \bar{B}$$

$$\sum_{i=1}^n \hat{C}_i m_i \leq \hat{B}$$

Inoltre, imponiamo che le divergenze siano ben definite nel loro utilizzo:

$$\sum_{j \in FS(i)} x_{ij} = 1 + m_i, \quad \forall i = 1, 2, \dots, n-1$$

3.2.3. Vincoli sui flussi

Si espongono, ora, i vincoli più complicati, necessari per imporre l'equilibrio dei flussi nel tempo e nello spazio:

$$\sum_{v \in FS(i)} f_{iv}^t - \sum_{k \in BS(i)} f_{ki}^{t-w_{ij}} - \Delta q_i^t = 0 \quad \forall i = 1, 2, \dots, n, \quad \forall t = 1, 2, \dots, T-1$$

$$\min \left\{ t, \quad T-w_{ij} \right\} \\ \sum_{\tau=\max \left\{ 0, \quad t-w_{ij}+1 \right\}} f_{ij}^\tau \leq U_{ij} x_{ij} + U_{ji} y_{ji} \quad \forall i = 1, 2, \dots, n, \quad \forall t = 1, 2, \dots, T-1 \quad (3.6)$$

Il primo vincolo traduce matematicamente l'equilibrio dei flussi ad ogni nodo, mentre il secondo serve ad imporre il rispetto della massima capacità possibile per quell'arco, tenendo conto di quando sono partiti i flussi in transito sull'arco. Osserviamo che si fa riferimento ai costi d'arco w_{ij} senza specificare la loro dipendenza dal tempo né altre caratteristiche. In [1] viene assunto che i tempi siano fissi per motivi computazionali, in modo tale da limitare il problema alla PLIM e non sconfinare nel dominio della Programmazione Non Lineare (cfr. Capitolo 1). In questo testo si segue la stessa ipotesi, ma i costi d'arco, ritenuti costanti, sono assunti multipli dei costi base:

$$w_{ij} = m \tilde{c}_{ij}, \quad m \in \mathbb{N}$$

\tilde{c}_{ij} è il costo d'arco minimo, quando la strada è completamente vuota.

Nel nostro caso, è stato posto cautelativamente $m = 3$. In questo modo, possiamo tenere conto degli inevitabili rallentamenti che si generano per tenere sotto controllo la folla, il cui reindirizzamento richiede estremo dettaglio ed attenzione; inoltre, possiamo confidentemente affermare che tempi così lunghi ci assicurano che l'evacuazione reale difficilmente richiederà più tempo di quello calcolato dalla simulazione. Anche imprevisti vari lungo il percorso dovrebbero essere ammortizzati, considerando che, in questo scenario di imposi-

zione, non possono verificarsi i caotici scenari di transitorio verso le soluzioni di equilibrio a cui assistiamo, ad esempio, per il fenomeno delle "partenze intelligenti".

3.2.4. Modello completo

Riportiamo di seguito il modello di ottimizzazione completo. Si segnala un'aggiunta a quanto detto prima: sono state introdotte le variabili g_{ij}^t , definite a tratti come segue. Questa modifica è stata inserita in corso d'opera per poter tradurre il problema in sintassi MATLAB, impiegando il pacchetto "Optimization Tools" (in particolare, la funzione *intlinprog*). Riportiamo il codice completo in Appendice.

$$\min \sum_{i=1}^n \sum_{j \in FS(i)} \sum_{t=1}^T r_j^t f_{ij}^t \quad (3.7)$$

s.t.

$$x_{ij} \geq y_{ji}, \quad \forall i = 1, 2, \dots, n, \quad j \in FS(i) \quad (3.8)$$

$$\sum_{v \in FS(i)} x_{iv} \leq \sum_{k \in BS(i)} x_{ki} \quad \forall i = 1, 2, \dots, n \quad (3.9)$$

$$\sum_{i=1}^n \hat{C}_i m_i \leq \hat{B} \quad (3.10)$$

$$\sum_{i=1}^n \sum_{j \in FS(i)} \bar{C}_{ij} y_{ij} \leq \bar{B} \quad (3.11)$$

$$\sum_{j \in FS(i)} x_{ij} = 1 + m_i, \quad \forall i = 1, 2, \dots, n-1 \quad (3.12)$$

$$\sum_{v \in FS(i)} f_{iv}^t - \sum_{k \in BS(i)} g_{ki}^t - \Delta q_i^t = 0 \quad \forall i = 1, 2, \dots, n, \quad \forall t = 1, 2, \dots, T-1 \quad (3.13)$$

$$U_{ij}x_{ij} + U_{ji}y_{ji} \geq \begin{cases} \sum_{\tau=1}^t f_{ij}^{\tau} & \text{se } 1 \leq t \leq w_{ij} \\ \sum_{\tau=t+1}^{t+w_{ij}} g_{ij}^{\tau} & \text{se } w_{ij} + 1 \leq t \leq T - w_{ij} \\ \sum_{\tau=t+1}^T g_{ij}^{\tau} & \text{se } T - w_{ij} + 1 \leq t \leq T \end{cases}$$

$$f_{ij}^t \geq 0 \quad \forall i = 1, 2, \dots, n-1, \quad \forall j \in FS(i), \quad \forall t = 1, 2, \dots, T-1 \quad (3.14)$$

$$g_{ij}^t = \begin{cases} 0 & \text{se } t \leq w_{ij} \\ f_{ij}^{t-w_{ij}} & \text{se } t > w_{ij} \end{cases}$$

$$\Delta q_i^t \geq 0 \quad \forall i = 1, 2, \dots, n-1, \quad \forall t = 1, 2, \dots, T-1 \quad (3.15)$$

$$x_{ij} \in \{0, 1\} \quad \forall i = 1, 2, \dots, n-1, \quad \forall j \in FS(i) \quad (3.16)$$

$$y_{ij} \in \{0, 1\} \quad \forall i = 1, 2, \dots, n-1, \quad \forall j \in FS(i) \quad (3.17)$$

$$m_i \in \mathbb{N} \quad \forall i = 1, 2, \dots, n-1 \quad (3.18)$$

Nel capitolo 5 verranno esposti i risultati dell'algoritmo prima su due grafi di prova e poi su un grafo reale.

4 | Approcci II e III: Semplice Informazione e Prescrizione Parziale

Procediamo ora a descrivere il modello di assegnazione dinamica del traffico che è stato sviluppato per gestire il caso della semplice informazione dei cittadini. Faremo riferimento alle aree citate nello stato dell'arte, ovvero Teoria dei Giochi, Pianificazione dei Trasporti con modelli dinamici e Ottimizzazione, nonché a cenni di Statistica.

4.1. Generazione domanda ai nodi

La prima parte del modello di Informazione Semplice, nonché il primo passo dell'algoritmo, consiste nella stima della domanda di evacuazione in ciascun nodo ad ogni istante di tempo. Prioritariamente, è stata ipotizzata la distribuzione degli abitanti sul territorio, per definire le quantità Q_i . La scelta è ricaduta, per semplicità, su una distribuzione derivante dalla Normale:

1. Definita la variabile ***domanda_totale***, si genera un campione derivante da una normale standard:

$$x_i \sim \mathcal{N}(\mu, \sigma^2) \quad \forall i = 1, \dots, n - 1$$

2. Imponiamo che tutti gli elementi del campione siano non-negativi:

$$x_i \mapsto \text{abs}(x_i)$$

3. Gli elementi del campione vengono riscaldati proporzionalmente per garantire che sommino a ***domanda_totale***:

$$Q_i = \text{domanda_totale} \times \frac{x_i}{\sum_{i=1}^{n-1} x_i}$$

Il codice della funzione *generazione_domanda_distribuzione_normale* è riportato in Appendice A.

Successivamente, è stata ipotizzata la distribuzione temporale degli ingressi in rete. Le ipotesi formulate sono le seguenti:

- Prendendo in considerazione un orizzonte temporale dell'evacuazione T , è ragionevole supporre che nessuno, indipendentemente dal proprio livello di spregiudicatezza, si attardi a mettersi in fuga oltre una certa soglia αT , con $\alpha \in (0, 1)$.
- La distribuzione temporale presenterà, verosimilmente, una forma "a campana", come una Normale o una Binomiale, come avviene per la maggior parte dei fenomeni naturali o dei comportamenti spontanei.
- Gli utenti, come già riportato nel capitolo 1, sono completamente informati sulla situazione.

Dunque, in analogia con il celebre *duopolio di Cournot*, modello classico di Teoria dei Giochi citato anche da [31], è stata formulata la seguente espressione:

$$\begin{cases} \Delta q_i(t) = c_i + b_i t - a_i t^2, t = 1, 2, \dots, \alpha T \\ \Delta q_i(t) = 0, t \geq \alpha T + 1 \end{cases}$$

L'idea è che i flussi (medi) di persone siano distribuiti nel tempo in modo che non tutti decidano di uscire di casa in maniera uniforme. Per tenere conto della differente avversione/propensione al rischio degli abitanti, possiamo supporre che una minima parte della popolazione si affretti eccessivamente ad uscire o, al contrario, si attardi fino all'ultimo, in accordo con l'ipotesi di distribuzione a campana. Inoltre, l'analogia con Cournot è motivata dal fatto che utenti completamente informati si trovino al centro di "giochi mentali" per prevedere il momento migliore per scappare. Si tratta dello stesso, identico meccanismo alla base del fenomeno delle *partenze intelligenti* in occasione di ferie estive o giorni di festa.

Per determinare i parametri della distribuzione parabolica, sono necessarie 3 equazioni. La prima consiste nell'imporre la simmetria della parabola, che deve assumere il valore massimo in corrispondenza del vertice, situato in $\frac{\alpha T}{2}$.

$$b_i = \alpha T a_i \tag{4.1}$$

La seconda è la conservazione della domanda totale:

$$\sum_{t=1}^{\alpha T} \Delta q_i(t) = Q_i \quad (4.2)$$

Da essa deriva la seguente relazione:

$$\frac{Q_i}{\alpha T} = c_i + \frac{1}{6}(\alpha T + 1)(\alpha T - 1)a_i \quad (4.3)$$

Infine, bisogna fissare uno dei due termini del secondo membro dell'equazione appena scritta. In maniera totalmente arbitraria, per semplicità, è stato scelto di eguagliare i due termini, ottenendo i seguenti valori.

$$c_i = \frac{Q_i}{2\alpha T} \quad (4.4)$$

$$a_i = \frac{6c_i}{\alpha^2 T^2 - 1} \quad (4.5)$$

4.2. Scelta del percorso

La sezione successiva del modello definisce le modalità di scelta del percorso: dai costi d'arco calcoliamo i costi d'arco ipotizzati dagli utenti, poi i costi di percorso e infine le probabilità di scelta del percorso, che serviranno poi a ripartire i flussi.

Indicando con \tilde{c}_{ij} il tempo di percorrenza di un arco in condizioni di flusso nullo, riportiamo l'equazione di *stima dei costi d'arco*.

$$w_{ij}^t = r_{ij}^t \tilde{c}_{ij} \quad (4.6)$$

Il parametro r_{ij}^t , dipendente dal tempo, è l'equivalente comportamentale del fattore di pericolo del Capitolo 3. In sostanza, esso assume il ruolo di un moltiplicatore del costo d'arco base dal punto di vista dell'utente, che "vede" w_{ij}^t ogniqualvolta si trova a decidere verso quale nodo $j \in FS(i)$ dirigersi, trovandosi in prossimità del nodo i al tempo t . Si riportano di seguito le espressioni utilizzate in questo elaborato per questi nuovi fattori di pericolo.

$$r_{ij}^t = \begin{cases} x & \text{se } j \in S_3, \forall i \\ x + yt & \text{se } j \in S_2, \forall i \\ x + yt + zt^2 & \text{se } j \in S_1, \forall i \end{cases}$$

Si osservi che r_{ij}^t dipende, in generale, dal tempo e dal nodo di arrivo, ma non da quello di partenza. Idealmente, il comportamento in una situazione d'emergenza può essere schematizzato come se gli utenti agissero "un nodo alla volta": nelle situazioni caotiche, è verosimile pensare che si verifichi un vero e proprio *re-routing* progressivo. In questo modo, teniamo in conto un potenziale fattore di aumento del disordine sulla rete, quasi un'*entropia* artificiale, e non vincoliamo nel tempo la scelta del percorso, permettendo ad un cittadino spaventato di adattare dinamicamente la propria decisione. Ad esempio, nel caso di un'inondazione, come quella di [44], certi nodi non sono inizialmente pericolosi da raggiungere, ma lo diventano rapidamente nel tempo all'aumentare del livello delle acque, fino a diventare totalmente impraticabili.

La zona S_3 è caratterizzata da un parametro r_{ij} in arrivo costante nel tempo, ma non unitario. Questa scelta riflette il compromesso tra due fattori. Da una parte, dato che la terza zona è sicura, è ragionevole, dal punto di vista dell'utente, immaginare che la disutilità complessiva dell'arco scelto per giungervi non aumenti nel tempo. Al contempo, però, si presenta la necessità di mantenere sempre $r_{ij} > 1$, per evidenziare che, nello scenario emergenziale in cui ci poniamo, l'utente medio sarà diffidente verso il tempo previsto per percorrere tale arco. La trattazione dei modelli di utilità aleatoria, incluso il Logit utilizzato di seguito, è data per consolidata; si rimanda a [19], [18] o [45] per i dettagli.

In maniera analoga, è stato arbitrariamente imposto che i parametri r_{ij} seguano le espressioni lineare e quadratica riportate prima per $j \in S_2$ o S_3 . I coefficienti utilizzati sono riportati nella tabella B.4 in Appendice B e sono stati ponderati, euristicamente, affinché i risultati del Logit risultassero verosimili su reti paragonabili a quella utilizzata come caso studio; in generale, vale lo stesso discorso dei parametri del Logit o di altri modelli aleatori, che richiedono una calibrazione sulla base di un'analisi statistica reale. La situazione ipotetica descritta poco sopra, in caso di inondazione, viene ben riprodotta nel nostro caso, poiché, effettivamente, dopo un determinato intervallo di tempo la probabilità di dirigersi verso nodi in S_1 diventa sostanzialmente nulla.

Osserviamo ora una delle peculiarità di questo modello: la formulazione ai nodi. Questa caratteristica è stata ispirata dal modello di [1] e presenta i propri vantaggi e svantaggi. Uno dei vantaggi è la possibilità di evitare la cosiddetta *enumerazione esplicita dei percorsi*. Come spesso accade nei problemi di Pianificazione dei Trasporti, mentre su grafi semplici è possibile enumerare tutti i possibili percorsi da ogni origine ad ogni de-

stinazione, studiando la topologia del grafo a livello combinatorio, su grafi minimamente complessi o molto connessi tale operazione è del tutto improponibile. Una delle possibilità per ovviare al problema è la selezione di un sottoinsieme di percorsi, a.e. gli n più significativi o rapidi o intuitivi (in genere, $n = 2, 3, 4$). Si rimanda a [18] o a testi specifici sulla Teoria dei Grafi per alcuni esempi e si mostra di seguito il classico "diamond graph" utilizzato da [45] e [47] a fini didattici.

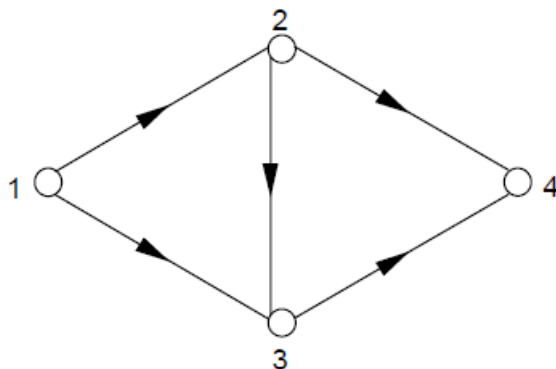


Figura 4.1: Classico grafo a diamante utilizzato a fini didattici.

| O-D Pairs | | 1-4 | | | 2-4 | | 3-4 |
|-----------|-------|-----|---|---|-----|---|-----|
| | | 1 | 2 | 3 | 4 | 5 | 6 |
| Links | Paths | | | | | | |
| | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| | 2 | 0 | 0 | 1 | 0 | 0 | 0 |
| | 3 | 1 | 0 | 0 | 1 | 0 | 0 |
| | 4 | 0 | 1 | 0 | 0 | 1 | 0 |
| | 5 | 1 | 0 | 1 | 1 | 0 | 1 |

Figura 4.2: Matrice di incidenza archi-percorsi per il grafo a diamante. Tratto da [19].

Questa soluzione, tuttavia, presenta due inconvenienti:

1. Diventa indispensabile utilizzare la *matrice di incidenza archi-percorsi*, o *ARGIM* in letteratura;

2. Si va ad aggiungere al modello (e quindi ai calcoli e ai risultati) un ulteriore elemento di incertezza dovuto all'arbitrarietà della scelta dei percorsi "più realistici" o del loro numero.

In questo elaborato, dunque, a patto di aumentare la complessità della trattazione, si è optato per la summenzionata formulazione ai nodi, integrando la seguente modalità di scelta del percorso:

$$h_{ij}^t = w_{ij}^t + \sum_{(u,v) \in DJ(j,t)} w_{uv}^t \quad (4.7)$$

In questo lavoro si fa riferimento all'algoritmo di Dijkstra, uno dei più conosciuti algoritmi per la ricerca del percorso di costo minimo da un nodo origine a un nodo destinazione.

$DJ(j, t)$ è qui definito come la successione ordinata degli archi appartenenti al percorso ottimale fino al nodo n secondo l'algoritmo di Dijkstra, a partire dal nodo j al tempo t . I costi d'arco forniti in input a Dijkstra sono i costi w_{ij}^t stimati prima. L'assunzione comportamentale sottostante è che ad ogni t l'utente riconsideri il percorso, come già detto, permettendo fenomeni di *re-routing*. Ma come avviene, di preciso, questa scelta adattiva?

Eludendo la problematica dell'enumerazione dei percorsi e riprendendo l'ipotesi presentata ad inizio sezione per parlare dei fattori di pericolo: ogni utente decide il percorso "un nodo alla volta". Tuttavia, poiché sappiamo che la ripartizione dei flussi deve avvenire in base all'utilità dei percorsi O-D e non dei singoli archi, è stato deciso di calcolare i costi di percorso per parti, partendo dall'arco più prossimo e considerando come origine O il nodo attuale (o verso cui siamo attualmente diretti). Schematicamente:

- h_{ij}^t indica, nell'equazione 4.7, il costo di percorso complessivo fino al nodo n , dove è situato il punto di raccolta.
- La prima componente sarà data da w_{ij}^t , il costo *stimato in base al pericolo crescente* dell'arco che va dal nodo i , attuale, ad uno dei suoi successori j .
- La seconda componente, $\sum_{(u,v) \in DJ(j,t)} w_{uv}^t$, è il costo minimo possibile, secondo Dijkstra, per arrivare a destinazione (n) partendo da $j \in FS(i)$.

Questa metodologia di calcolo si basa sulla Teoria dei Giochi e, in particolare, sulle nozioni di *informazione* descritte nei Capitoli 2 e 1. Si suppone che gli utenti siano *razionali* nel senso trasportistico del termine, ovvero che perseguano egoisticamente la massima utilità possibile. Al contempo, però, essi sono anche *perfettamente informati*, magari inconsciamente, ovvero sono a conoscenza che gli altri sono razionali, che gli altri sanno che sono razionali, e così via. Traducendo in un contesto reale, gli utenti sono *razionali e sufficientemente diffidenti gli uni verso gli altri*.

A questo punto, non resta che utilizzare il modello Logit per calcolare le probabilità di scelta del percorso:

$$p_{ij}^t = \frac{\exp(-\frac{1}{\theta}h_{ij}^t)}{\sum_{v \in FS(i)} \exp(-\frac{1}{\theta}h_{iv}^t)} \quad (4.8)$$

Ribadiamo che, in questo modo, possiamo suddividere i flussi tra i percorsi che escono dal nodo in cui ci si trova, dividendo gli utenti tra gli archi della stella uscente.

4.3. Flussi di prossimità

In questa sottosezione e nella successiva verranno esposti i punti più delicati del modello, quelli che hanno richiesto il maggior tempo per essere elaborati e che hanno generato più revisioni del codice.

Iniziamo la trattazione del tema dei flussi definendo una prima e fondamentale differenza rispetto al lavoro di [1]. Tale articolo ha rappresentato l'ispirazione principale per la descrizione della ripartizione degli utenti tra i nodi, ma il lavoro seguente procede oltre ponendo la distinzione, originariamente non presente, tra *flussi di prossimità* e *flussi d'arco* in senso stretto. I flussi di prossimità rappresentano l'evoluzione diretta del flusso f_{ij}^t , di cui mantengono la notazione, nel senso che sono i flussi riferiti alla quantità di utenti che, arrivata al nodo i al tempo t , decide di *orientarsi verso* il nodo j . Potremmo definirli, in maniera tanto impropria quanto singolare, *flussi di nodo*! I flussi d'arco, F_{ij}^t , invece, si attengono alla classica definizione che adottiamo nell'Ingegneria dei Trasporti (il flusso sull'arco che va da i a j al tempo t).

Entrambi i flussi sono regolati da vincoli fisici. In particolare, nel caso dei flussi di Achrekar e Vogiatzis, era stato imposto il rigido rispetto della *capacità* dell'arco: il flusso f_{ij}^t non poteva mai essere superiore alla capacità descritta nell'equazione 3.6. Nel nostro caso, il flusso corrispondente per questo scopo è il flusso d'arco F_{ij}^t , che rispetta la cosiddetta *equazione BPR-affine*, descritta nella prossima sottosezione. I flussi di prossimità, invece, dovrebbero rispettare dei vincoli sulla cosiddetta *capacità del nodo*; tuttavia, in accordo con la semplificazione operata dagli autori, anche in questo lavoro è stato stabilito di trascurare il vincolo e considerare tutte le capacità di nodo infinite. A rigore, se si volesse complicare ulteriormente il modello per tener conto di questa limitazione, si dovrebbe adottare una struttura simile a quella del *Link Transmission Model* o del *Cell Transmission Model* (vedere [47] o [18] in proposito). Innanzitutto, adeguiamo allo scopo l'equazione 3.13, che diventa la seguente:

$$\sum_{v \in FS(i)} f_{iv}^t = \Delta q_i(t) + \sum_{k \in BS(i)} f_{ki}^{t-z_{ki}^t} \quad (4.9)$$

Qui, la quantità z_{ki}^t esprime un intervallo di tempo, ricorsivamente così definito:

$$z_{ki}^t = \lfloor c_{ki}^{t-z_{ki}^t} \rfloor \quad (4.10)$$

Chiariamo meglio cosa rappresenta la quantità e come è stata calcolata. Ipotizziamo di avere 3 successori del nostro ipotetico nodo i , che chiameremo j_1, j_2, j_3 . I tempi di percorrenza al tempo t sono i costi $c_{ij_v}^t$, per $v = 1, 2, 3$. Ma se a noi interessa calcolare i flussi in uscita dal nodo, $\sum_{j_v \in FS(i), v=1,2,3} f_{ij_v}^t$, dobbiamo prima conoscere $\sum_{k_m \in BS(i), m=1,2,3} f_{k_m i}^{t-z_{k_m i}^t}$, ovvero i flussi che sono entrati in quel nodo *nel recente passato*. Supponiamo di invertire la prospettiva e consideriamo i seguenti valori esemplari riferiti ai tempi per arrivare a i dai suoi predecessori k_m , $m = 1, 2, 3$:

| m | Predecessore | Tempo da i | $z_{k_m i}^t$ |
|----------|---------------------|--------------------------------|---------------------------------|
| 1 | k_1 | 2 | 2 |
| 2 | k_2 | 3.4 | 3 |
| 3 | k_3 | 5.99 | 5 |

Tabella 4.1: Esempi di calcolo di $z_{k_m i}^t$ per un nodo con 3 successori.

Nel primo caso, i flussi in uscita da i al tempo t saranno partiti "2" periodi fa; perciò, li considereremo al tempo $t - 2$ e il relativo costo d'arco per arrivare a i sarà stato:

$$c_{ki}^{t-2} = z_{ki}^{t-2} = 2$$

Ergo, sarà stata soddisfatta l'equazione 4.10. Ciò significa che i costi d'arco, relativi a flussi partiti nel passato, sono calcolati proprio alla loro partenza, poiché gli utenti nel passato avevano "visto" quel costo lì, e non quello attuale (qualunque esso sia). Nel secondo e nel terzo caso, estraiamo la parte intera inferiore del costo c per determinare il tempo z . Ma perché? Se il tempo è 3.4, o in generale un qualunque costo d'arco non intero, si palesano tre possibilità.

1. Consideriamo il tempo in se stesso, senza modifiche;
2. Prendiamo la parte intera inferiore, *floor*;
3. Prendiamo la parte intera superiore, *ceiling*.

La prima strada, più attraente, richiederebbe il ricorso a modelli *a tempo continuo* (cfr. la sottosezione 1.4), ma non è questo il caso. Dato che abbiamo ambientato il modello in uno

scenario a tempo discreto (t in minuti, come descritto nel capitolo 5), rimane da scegliere tra le due alternative intere. Se estraessimo la parte intera superiore, ovvero $z = 4$, andremmo a considerare i flussi partiti al tempo $t - 4$; ma allora, detti flussi sarebbero già arrivati al nodo, oltrepassandolo, da 0.6 minuti! Si è optato per la parte intera inferiore, $z = 3$, per una motivazione puramente logica. I flussi partiti a $t - 3$ stanno per arrivare al nodo e impiegheranno ancora 0.4 minuti per oltrepassarlo; in compenso, al prossimo passo temporale, $t + 1$, saranno già arrivati. Di per sé, questa affermazione non rappresenta nessun motivo di scelta privilegiata rispetto all'altra opzione, che sembrerebbe essere stata scartata solo per convenzione. In realtà, la motivazione si palesa in retrospettiva, ipotizzando che il costo d'arco cui ci riferiamo sia stato visto dagli utenti al tempo $t = 0$.

- $z = 3$. Al tempo 0 gli utenti vedono $c = 3.4$. Al tempo 1, dovranno ancora muoversi per 2.4 minuti e analogamente per 1,4 minuti al tempo 2. A $t = 3$, avranno ancora 0.4 minuti di viaggio, ma tale tempo è al di sotto della finestra di discretizzazione temporale e saranno già in procinto di riversarsi tra i successori del nodo che stanno approcciando.
- $z = 4$. Ripetendo lo stesso ragionamento, al tempo 4 avranno superato il nodo, ma la ripartizione, per motivi fisici, sarà già avvenuta.

In ogni caso, una volta sbrigata la questione dell'equilibrio al nodo, non resta che ripartire i flussi tra i nodi successivi mediante il modello Logit.

$$f_{ij}^t = p_{ij}^t \left(\sum_{v \in FS(i)} f_{iv}^t \right) \quad (4.11)$$

4.4. Flussi e costi d'arco

Vediamo ora come determinare i flussi d'arco. Il flusso d'arco è stato ricondotto ad una combinazione lineare di flussi di prossimità partiti nel passato. Infatti, gli utenti presenti al tempo t sull'arco $i \rightarrow j$ non potranno che essere partiti da i , dirigendosi verso j , ad un certo tempo $t - k$ senza essere arrivati a destinazione. Da questa prospettiva, possiamo calcolare i flussi d'arco come segue:

$$F_{ij}^t = \sum_{k=1}^{n(t)} f_{ij}^{t-k} \quad (4.12)$$

L'estremo superiore della sommatoria è così definito:

$$n(t) = \max \left\{ k \mid c_{ij}^{t-k} \geq k \right\} \quad (4.13)$$

Illustriamo rapidamente alcune proprietà dell'indice $n(t)$.

Innanzitutto, analizziamo la definizione. Rispetto all'istante attuale t , $t - n(t)$ indica il momento più lontano nel tempo per cui ci aspettiamo che gli utenti partiti da i non siano ancora arrivati a j . Gli indici dei nodi sono omessi nella trattazione solo per comodità di lettura; l'indice è, ovviamente, calcolato separatamente per ogni arco in ogni istante. Fissato k , c_{ij}^{t-k} è il tempo di percorrenza dell'arco per chi parte k minuti nel passato. Per aggiornare l'indice nel tempo, analizziamo cosa succede al minuto successivo; con delle semplici manipolazioni algebriche,

$$n(t+1) = \max \left\{ k \mid c_{ij}^{t+1-k} \geq k \right\} = 1 + \max \left\{ k \mid c_{ij}^{t-k} - 1 \geq k \right\} \quad (4.14)$$

Ragionando per assurdo, possiamo affermare anche quanto segue:

$$n(t+1) \leq n(t) + 1 \quad (4.15)$$

Quest'equazione traduce un fatto fisico ovvio. Prendiamo, ad esempio, $n(t) = 4$. Ciò significa che, prendendo come riferimento il tempo t , tornando indietro nel tempo, chi è partito a $t-1, t-2, t-3$, fino a $t-4$, sarà ancora in moto lungo l'arco, mentre chi era partito ancora prima è sicuramente già arrivato, quantomeno da un tempo pari a $n(t) + 1 - c_{ij}^{t-n(t)-1} = 5 - c_{ij}^{t-5}$. Nel nostro caso, la 4.15 si tradurrebbe in $n(t+1) < 5$. Spostando il punto di riferimento in avanti a $t+1$, si presentano soltanto 2 possibilità:

1. Alcuni dei flussi partiti sono già arrivati, a.e. quelli a $t-3$ e $t-4$. Ciò significa che quelli partiti da $t-2$ in poi non lo sono, implicando

$$n(t+1) = (t+1) - (t-2) = 3 < 5 = n(t) + 1$$

2. Nessun flusso è arrivato a destinazione, ma nel frattempo sono partiti altri utenti. Perciò

$$n(t+1) = (t+1) - (t-4) = 5 = n(t) + 1$$

Se per assurdo avessimo $n(t+1) > n(t) + 1$, significherebbe che, paradossalmente, qualche flusso arrivato a destinazione al tempo t sarebbe nuovamente sull'arco al minuto successi-

vo! Non viene presentata una dimostrazione analiticamente rigorosa per non appesantire la trattazione, peraltro per un argomento di secondo piano. Applicando in serie la 4.15, otteniamo la seguente:

$$n(t) \leq t \quad (4.16)$$

La 4.15 e la 4.16 non sono particolarmente interessanti dal punto di vista del modello o dell'applicazione, ma sono state riportate in quanto punti salienti ed indispensabili dei codici implementati su MATLAB per risolvere il problema di assegnazione in esame.

Detto questo, perché abbiamo distinto tra F_{ij}^t e f_{ij}^t , vincoli di capacità a parte? Fondamentalmente per 2 motivi:

1. Per determinare, nella pratica, quali archi necessitino di maggior attenzione e quante risorse sia opportuno allocare, calcolando $\max_t F_{ij}^t$ possiamo determinare il flusso maggiore, e quindi foriero di maggiori pericoli, che abbia transitato su ogni arco;
2. Per far avanzare l'algoritmo nel tempo, abbiamo bisogno di un'equazione supplementare di chiusura che segua, a grandi linee, la formulazione *SEAM*, "Six Equations Assignment Model" ([47], [18], [22]). In sostanza, è necessario che venga garantito il seguente ciclo di flussi e costi:

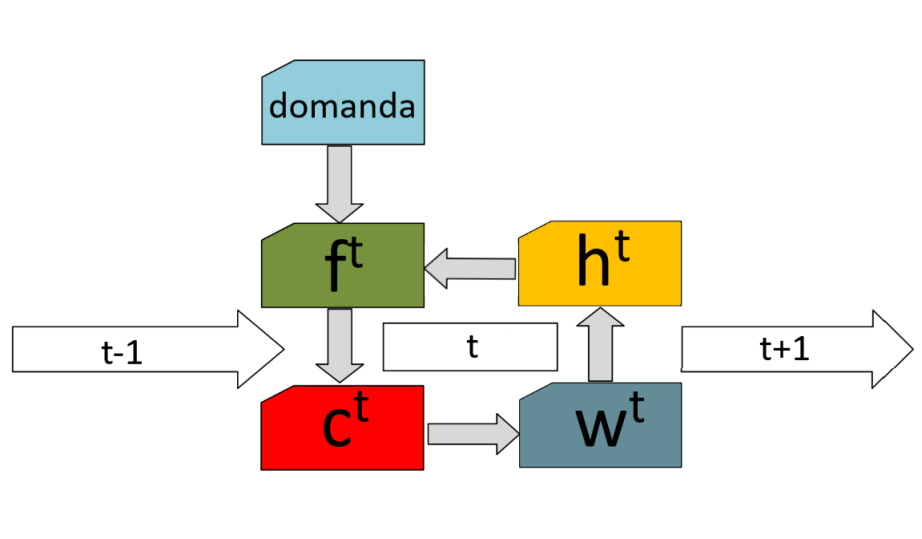


Figura 4.3: Ciclo di calcolo delle variabili d'arco e di percorso.

L'equazione di chiusura di nostro interesse è l'equazione che traduce la congestione, ovvero la dipendenza dei costi d'arco dai flussi presenti. Si è deciso di adottare la funzione

BPR-affine, tipicamente usata in letteratura:

$$c_{ij}^t = \tilde{c}_{ij} \left[1 + \beta_1 \left(\frac{F_{ij}^t}{u_{ij}} \right)^{\beta_2} \right] \quad (4.17)$$

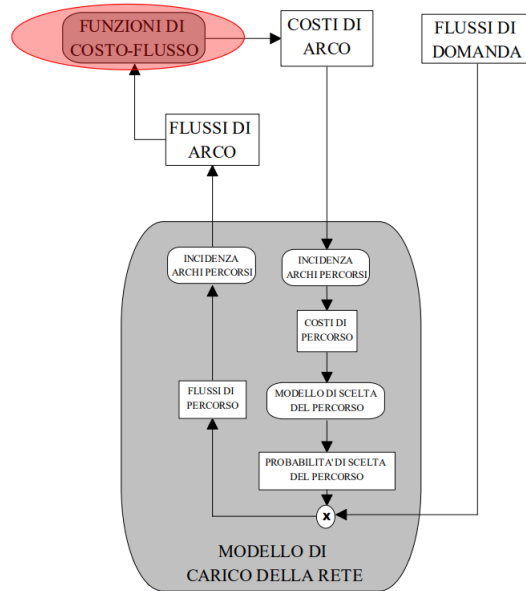


Figura 4.4: Rappresentazione generica dei modelli di assegnazione all'equilibrio. Tratto da [22].

4.4.1. Efficacia dell'evacuazione

Come nota finale del modello, si riporta la formulazione adottata per il calcolo delle potenziali vittime. Ad ogni t , il numero dei sopravvissuti certi può essere stimato come il totale di coloro arrivati al nodo destinazione n . Per sottrazione, le vittime sono state calcolate come segue:

$$\mathbf{vittime_potenziali}(t) = \sum_{i=1}^{n-1} \sum_{j \in FS(i)} F_{ij}^t \quad (4.18)$$

La quantità in esame è stata consapevolmente sovrastimata: ad esempio, vengono conteggiati anche tutti gli utenti in moto verso nodi in S_3 diversi dal nodo finale, che sappiamo essere comunque piuttosto sicuri. In ogni caso, la formula non ha pretesa di certezza, bensì solo di verosimiglianza, ed è stata adoperata perlopiù per confrontare i modelli. L'altra utilità della formula qui descritta è che, sovrastimando appunto le vittime, non appena il valore si azzera non abbiamo più alcun dubbio che tutti gli utenti, pari a *doman-*

da_totale, siano giunti a destinazione. Le iterazioni del ciclo di assegnazione, infatti, presentano la seguente doppia condizione di arresto:

```
1 while (t<=tau) && (vittime_potenziali>0)
```

4.5. Prescrizione Parziale

In riferimento alle sezioni 1.3 e 1.7, rimane da definire il terzo approccio possibile alla gestione dell'evacuazione, definito *Prescrizione Parziale*. L'idea su cui è stato basato questo terzo approccio è la seguente: appurato che il modello PLIM, oltre a determinare esattamente la distribuzione dei flussi minuto per minuto e la loro gestione, restituisce anche l'insieme di archi da invertire o bloccare del tutto, possiamo prendere per "ottimale" il nuovo grafo che viene, così, generato; ma varrà lo stesso anche per altre formulazioni, ovvero per differenti modalità di assegnazione? Qualora fosse così, l'applicazione sul nuovo grafo del secondo approccio, totalmente libero, apporterebbe dei miglioramenti al risultato complessivo.

In breve, l'applicazione prevista è la seguente:

1. Applico l'algoritmo di assegnazione di Semplice Informazione e calcolo:
 - Il tempo complessivo di evacuazione, se minore di τ ; in tal caso, inoltre, registro $\max_t F_{ij}^t$ per ogni arco.
 - Il numero stimato di vittime, se il tempo finale t soddisfa $t = \tau$ (poiché, in tal caso, avremmo un tempo necessario superiore).
 - Ritorno al punto 1 aumentando l'orizzonte della simulazione, eventualmente iterando fino a coprire tutto il tempo necessario per completare l'evacuazione.
2. Applico il modello PLIM per l'Imposizione Totale, che viene risolto dal solutore MATLAB, imponendo come orizzonte temporale della simulazione lo stesso τ del punto precedente.
3. Genero il nuovo grafo ottimale seguendo questa procedura:
 - (a) Se $a_{ij} = 0$, sicuramente $a'_{ij} = 0$ e $U'_{ij} = 0$;
 - (b) Se $a_{ij} = 1$, valuto x_{ij} e y_{ji} (tenuto conto che $y_{ji} \leq x_{ij}$ sempre).
 - Se $x_{ij} = 0$, $a'_{ij} = 0$ e $U'_{ij} = 0$;
 - Se $x_{ij} = 1$ e $y_{ji} = 0$, $a'_{ij} = a_{ij} = 1$ e $U'_{ij} = U_{ij}$;

- Se $x_{ij} = 1$ e $y_{ji} = 1$, $a'_{ij} = a_{ij} = 1$ e $U'_{ij} = U_{ij} + U_{ji}$.
4. Applico nuovamente l'algoritmo di assegnazione di Semplice Informazione sul nuovo grafo, ottenendo così dei nuovi valori finali per tempo totale, flussi massimi o vittime.

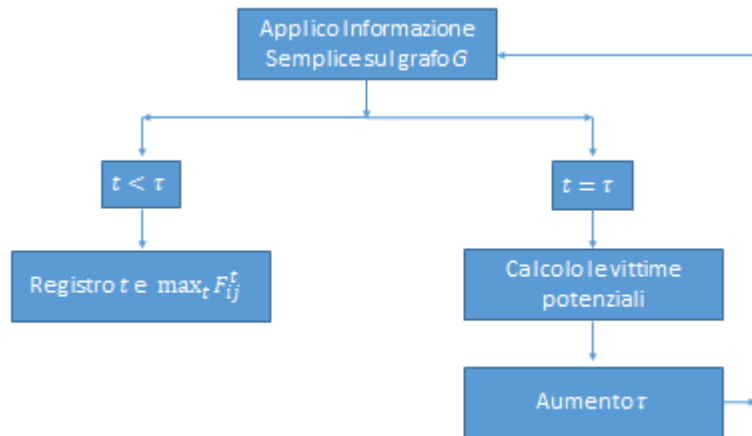


Figura 4.5: Parte 1 della procedura: confronto di tempi e vittime.

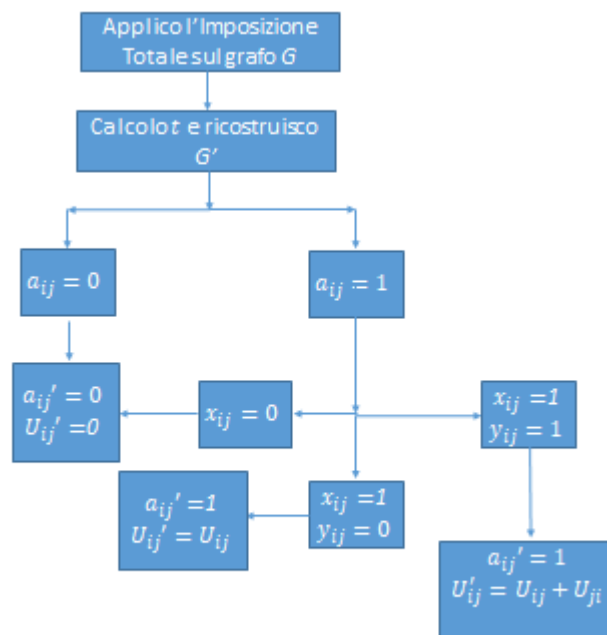


Figura 4.6: Parte 2 della procedura: calcolo delle nuove matrici di incidenza e delle capacità.

Conclusa la sequenza di calcolo, confrontiamo i risultati per valutare efficacia, convenienza e fattibilità, tenuto conto che ciascuna delle tre modalità ha i propri pro e contro e

non è detto che sia possibile applicarne una qualsiasi (a.e., banalmente, l'Imposizione Totale non è esattamente la strada più semplice da applicare in una nazione occidentale...). In generale, va precisato che non c'è nessuna garanzia di miglioramento delle prestazioni passando da un approccio all'altro. Tuttavia, euristicamente, si è osservato un comportamento generale, come si vedrà nel prossimo capitolo:

1. Si osserva un generale, ma limitatissimo, miglioramento delle prestazioni della rete passando dall'approccio 2, Semplice Informazione, al 3, Prescrizione Parziale, ovvero modificando soltanto il grafo della rete. Tuttavia, tale modifica deve soddisfare dei vincoli di fattibilità economica e pratica, per predisporre le risorse sul territorio; inoltre, dovendo predisporre il piano d'evacuazione, è importante che le variazioni apportate al grafo stradale non creino confusione nei cittadini, che potrebbero, ad esempio, trovarsi spiazzati di fronte al blocco di una strada che percorrono quotidianamente, in assenza di un'adeguata formazione ed istruzione per approcciarsi all'emergenza. Il miglioramento nella prestazione della rete cui si fa riferimento non è nella scala temporale dell'evacuazione, quanto piuttosto nelle dimensioni massime dei flussi sugli archi, che rimangono più contenute. Si specifica che solo per alcuni archi $\max_t F_{ij}^t$ decresce effettivamente, ovvero quelli che vengono mantenuti uguali in entrambi gli scenari. Per gli archi che vengono "allargati" incorporando l'inversione di marcia dell'arco in senso opposto, ovviamente i flussi $\max_t F_{ij}^t$ aumentano, ma rispetto alle capacità complessive i rapporti diminuiscono.
2. L'approccio di Imposizione Totale è generalmente migliore nella gestione del tempo. Dal punto di vista operativo, le simulazioni sono state gestite impostando come orizzonte temporale un intervallo di tempo leggermente inferiore, o al massimo uguale, a quello previsto per la Semplice Informazione, ottenendo dei miglioramenti. Inoltre, occorre osservare che, in generale, cambiare la finestra temporale modifica la soluzione in termini di flussi, ma, apparentemente, non sembrerebbe influenzare la soluzione dal punto di vista topologico. Ad esempio, nel piccolo grafo riportato nella sottosezione 5.1.1, si è posto $T_{Max} = 50$ e $\tau = 200$, ottenendo il completamento dell'evacuazione in 48 minuti per i due approcci liberi e soltanto in 25 minuti per quello coercitivo. τ è il tempo di riferimento per l'allocazione delle variabili: il modello PLIM necessita di un orizzonte di riferimento, maggiore o uguale del tempo T_{Max} previsto per l'evacuazione fisica. Curiosamente, reimpostando una finestra temporale di $\tau = 50$ minuti per la simulazione PLIM, la soluzione non si mantiene uguale a prima, nonostante già dal ventesimo minuto i flussi fossero identicamente nulli. La durata complessiva dell'evacuazione è la stessa, ma i flussi non nulli sono piuttosto diversi. La soluzione, in gergo statistico, non presenta la proprietà dell'*assenza di*

memoria (a.e. [58]). Si suppone che la motivazione sia da ricercare nelle funzioni utilizzate nella sottosezione 3.1.3 per definire i fattori di pericolo, ma non si indaga ulteriormente in merito.

Si segnala che l'osservazione di questa leggera tendenza migliorativa passando di volta in volta da un approccio più libero ad uno più impositivo è totalmente euristica e non è stata ricercata nessuna modalità di calcolo per dimostrare rigorosamente questo comportamento; addirittura, chi scrive suppone che non esista affatto un modo analitico per dimostrare tale miglioramento. Un'altra osservazione importante per giustificare questo legittimo scetticismo è che i fattori di pericolo sono definiti diversamente tra il modello PLIM e il modello di assegnazione, per tenere conto delle variazioni nella funzione utilità di chi sceglie i movimenti sul grafo (rispettivamente, lo Stato e i singoli giocatori/utenti).

5 | Applicazione dei modelli

Concludiamo la trattazione modellistica presentando i risultati relativi all'applicazione del modello PLIM e del modello di assegnazione con teoria dei giochi. Nell'ordine, si illustrano prima i risultati su due grafi di prova, di dimensioni ridotte, e poi sul grafo corrispondente ad un caso studio reale.

5.1. Applicazione su piccoli grafi di prova

I primi due grafi sono stati utilizzati per valutare in corso d'opera l'efficacia e la correttezza degli algoritmi e, successivamente, per effettuare una rapida fase di *debugging*, per poter rendere più veloce il codice ed evitare errori imprevisti.

5.1.1. Grafo con 6 nodi

Il primo dei due grafi utilizzati è schematizzato come segue (immagine Matlab):

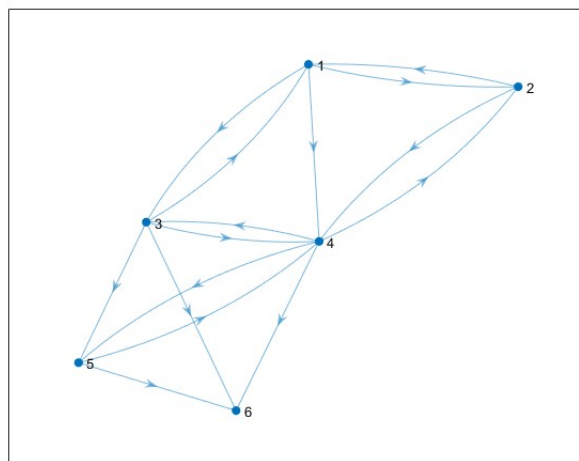


Figura 5.1: Grafo di prova con 6 nodi.

Riportiamo di seguito i parametri (totalmente arbitrari) utilizzati, insieme ai risultati

di nostro interesse e al grafo modificato che deriva dall'applicazione dell'algoritmo del Capitolo 3.

| Parametro | Valore |
|----------------------------|--------|
| zona S_1 | 1-2 |
| zona S_2 | 3-4 |
| zona S_3 | 5-6 |
| U_{ij} | 50 |
| τ | 200 |
| T_{MAX} | 50 |
| $domanda_totale$ | 1000 |
| \tilde{c}_{ij} | 1 |
| $c_{ij} = m\tilde{c}_{ij}$ | 3 |
| β_1 | 0.1 |
| β_2 | 2 |
| Q_1 | 268 |
| Q_2 | 4 |
| Q_3 | 105 |
| Q_4 | 409 |
| Q_5 | 214 |

Tabella 5.1: Parametri grafo di prova con 6 nodi.

Per semplicità, è stato imposto un unico valore per ogni arco per capacità e costo-base, pari rispettivamente a $U_{ij} = 50$ veicoli/minuto e $\tilde{c}_{ij} = 2$ (minuti). La finestra temporale a nostra disposizione per mettersi in moto è pari a 50 minuti, implicando che la generazione di domanda cessa dopo $\alpha T_{MAX} = 40$ minuti. Per la prima simulazione è stato posto $\tau = 200$. τ svolge una funzione duplice: nell'Informazione Semplice e nella Prescrizione Parziale, rappresenta il tempo in cui è *fisicamente* possibile evacuare, passato il quale possiamo tragicamente considerare gli utenti come vittime; nell'Imposizione Totale, dato che la finestra computazionale e la finestra fisica coincidono, dovendo allocare τ variabili di ottimizzazione per ogni flusso d'arco, questo tempo assume entrambi i significati. Rispettando la procedura di calcolo, punto 1, al paragrafo 4.5, il tempo τ è stato scelto abbondantemente superiore al tempo T_{MAX} , con un rapporto T_{MAX}/τ , per poter valutare l'efficacia delle scelte progettuali e la correttezza dell'algoritmo. La domanda riportata in

| | A | B | C | D | E | F | G | H | I | J |
|---|-------|---|----|----|----|----|---|----|---|---------|
| 1 | | | | | | | | | | |
| 2 | MAX F | | | | | | | t | | vittime |
| 3 | 0 | 3 | 8 | 8 | 0 | 0 | | 48 | | 0 |
| 4 | 2 | 0 | 0 | 5 | 0 | 0 | | | | |
| 5 | 2 | 0 | 0 | 6 | 8 | 14 | | | | |
| 6 | 0 | 4 | 12 | 0 | 14 | 26 | | | | |
| 7 | 0 | 0 | 0 | 10 | 0 | 25 | | | | |
| 8 | 0 | 0 | 0 | 0 | 0 | 0 | | | | |

Figura 5.2: Risultati Semplice Informazione per il grafo con 6 nodi.

tabella, Q_1, Q_2, \dots, Q_5 , è stata generata con la funzione riportata in Appendice A. La prima simulazione eseguita ha registrato un esito positivo: l'evacuazione è stata completata correttamente, senza vittime, in 48 minuti. Il margine tra T e T_{MAX} è pari solo a 2 minuti, il 4% del totale, poiché le domande totali dei nodi possono essere smaltite velocemente su un grafo così piccolo. Riportiamo un prospetto dei risultati, trasferiti da MATLAB a Excel per praticità. Osserviamo anche che i massimi flussi vengono comprensibilmente registrati sugli archi 4-6 e 5-6. La distribuzione della domanda Q_i è estremamente sbilanciata a favore degli ultimi nodi, perciò era lecito attendere un simile risultato.

Quindi, appurato che è possibile evacuare completamente in meno di 50 minuti, possiamo utilizzare questo stesso valore come limite massimo per la simulazione PLIM: come è intuibile, e come è stato anticipato nel capitolo precedente, ci aspettiamo una diminuzione del tempo necessario. In questo grafo, non solo tale impressione viene confermata, ma si rivela anche estremamente più netta di quanto pensassimo: il tempo richiesto è pari a soli 25 minuti! Dopo 19 minuti non c'è più generazione di domanda; gli ultimi flussi, molto contenuti e pari a 42 sull'arco 4-6 e 1 sull'arco 5-6, vengono smaltiti con calma nei 6 minuti successivi (pari a $m\tilde{c}_{ij}$, come nella sottosezione 3.2.3). La distribuzione della domanda utilizzata, ovviamente, è la stessa della prima simulazione.

| | A | B | C | D | E |
|----|---|---|---|---|---|
| 1 | M | | X | | Y |
| 2 | 0 | | 0 | | 0 |
| 3 | 1 | | 1 | | 0 |
| 4 | 1 | | 0 | | 0 |
| 5 | 1 | | 1 | | 0 |
| 6 | 0 | | 1 | | 0 |
| 7 | | | 0 | | 1 |
| 8 | | | 1 | | 0 |
| 9 | | | 0 | | 0 |
| 10 | | | 1 | | 0 |
| 11 | | | 0 | | 0 |
| 12 | | | 0 | | 1 |
| 13 | | | 1 | | 0 |
| 14 | | | 1 | | 0 |
| 15 | | | 0 | | 1 |
| 16 | | | 1 | | 0 |

Figura 5.3: Variabili per l'alterazione del grafo con 6 nodi a seguito dell'algoritmo PLIM.

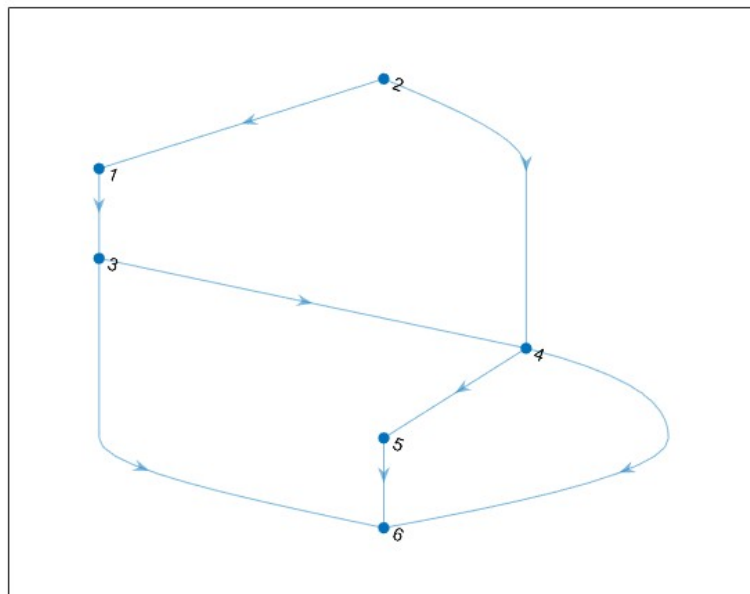


Figura 5.4: Alterazione del grafo con 6 nodi a seguito dell'algoritmo PLIM.

Le variabili m_i , x_{ij} , y_{ij} sono riportate in forma vettoriale. Mentre per il numero di divergenze concesse al nodo questo è ovvio, per le altre variabili questa enumerazione deriva dalla struttura richiesta per agevolare i calcoli nell'esecuzione della funzione `intlinprog` di MATLAB.

In prima istanza, si è tentato di utilizzare una formulazione delle variabili che rispecchiasse perfettamente la struttura algebrica delle variabili.

```

1  %DICHIARAZIONE VARIABILI DI CONTROLLO
x = optimvar('x',n-1,n,'Type','integer','LowerBound',0,'UpperBound',1);
3  y = optimvar('y',n-1,n,'Type','integer','LowerBound',0,'UpperBound',1);
f = optimvar('f',n-1,n,tau,'LowerBound',0);
5  g = optimvar('g',n-1,n,tau,'LowerBound',0);
numero_divergenze=optimvar('numero_divergenze',n-1,'Type','integer','LowerBound',0, 'UpperBound',n-1);
7  dq= optimvar('dq',n-1,tau,'Type','integer','LowerBound',0);
totale_sicurezza=optimvar('totale_sicurezza',n,tau,'Type','integer','LowerBound',0);
9  evacuation_prob = optimproblem;

```

Questa formulazione, tuttavia, presentava due problematiche:

1. Vengono generate molte più variabili del necessario, poiché introduciamo numerose variabili fittizie per flussi su archi che non esistono davvero. Computazionalmente parlando, una scelta simile diventa un suicidio per grafi minimamente più grandi di questo.
2. A causa delle numerose approssimazioni numeriche introdotte dal *Branch and Bound* di intlinprog, ed in particolare dei vincoli $x \in [0, 1]$, $x \leq 0$, che si sovrappongono in maniera totalmente inaspettata. Il risultato è che la regione ammissibile risulta vuota.

Successivamente, dopo notevoli peripezie nell'utilizzo di intlinprog, si è optato per la formulazione presente nel codice in Appendice A, che enumera gli archi secondo l'ordinamento progressivo dei nodi.

Occorre notare un'altra peculiarità del grafo risultante dall'applicazione del modello PLIM. L'ispezione visiva del grafo potrebbe far pensare ad una violazione di quanto detto riguardo alla figura 3.1. Tuttavia, il vincolo 3.9, $\sum_{v \in FS(i)} x_{iv} \leq \sum_{k \in BS(i)} x_{ki} \quad \forall i = 1, 2, \dots, n$, è perfettamente rispettato. Ciò che non viene considerato dal vincolo è l'introduzione delle inversioni di senso di marcia. A conti fatti, in questo modo rilassiamo la condizione che siano sempre di più gli archi entranti in un nodo di quelli che ne escono. Tuttavia, la differenza, in numero, tra i secondi e i primi è sempre minimizzata.

| MAX F | | | | | |
|-------|---|----|---|----|----|
| 0 | 0 | 57 | 0 | 0 | 0 |
| 0 | 0 | 0 | 3 | 0 | 0 |
| 0 | 0 | 0 | 8 | 0 | 50 |
| 0 | 0 | 0 | 0 | 50 | 50 |
| 0 | 0 | 0 | 0 | 0 | 50 |
| 0 | 0 | 0 | 0 | 0 | 0 |

Figura 5.5: Flussi d'arco massimi nel grafo con 6 nodi - approccio di Imposizione Totale.

I nuovi flussi sono generalmente più alti per 2 motivi. In primis, l'algoritmo tende ad addensare il più possibile i flussi, per sfruttare al meglio le capacità degli archi; inoltre, alcuni archi manifestano ora una capacità raddoppiata, a seguito dell'inversione dei loro omologhi inversi.

A questo punto, seguendo la procedura della sottosezione 4.5, è stato utilizzato il nuovo grafo per far circolare gli utenti in maniera più rapida, secondo l'approccio di Prescrizione Parziale.

| MAX F | | | | | | |
|-------|---|----|---|----|----|----|
| 0 | 0 | 57 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 3 | 0 | 0 | 0 |
| 0 | 0 | 0 | 8 | 0 | 50 | 50 |
| 0 | 0 | 0 | 0 | 50 | 50 | 50 |
| 0 | 0 | 0 | 0 | 0 | 0 | 50 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Figura 5.6: Flussi d'arco massimi e tempo d'evacuazione nel grafo con 6 nodi - approccio di Prescrizione Parziale.

Le differenze tra i due approcci non sono sufficientemente significative da far propendere per l'uno per l'altro.

5.1.2. Grafo con 14 nodi

In questo secondo caso, il punto di raccolta è posizionato nel nodo 14.

| Parametro | Valore |
|-------------------|--------|
| zona S_1 | 1-6 |
| zona S_2 | 7-12 |
| zona S_3 | 13-14 |
| U_{ij} | 50 |
| τ | 200 |
| T_{MAX} | 50 |
| $domanda_totale$ | 14000 |
| \tilde{c}_{ij} | 2 |
| β_1 | 0.1 |
| β_2 | 2 |

Tabella 5.2: Parametri grafo di prova con 14 nodi.

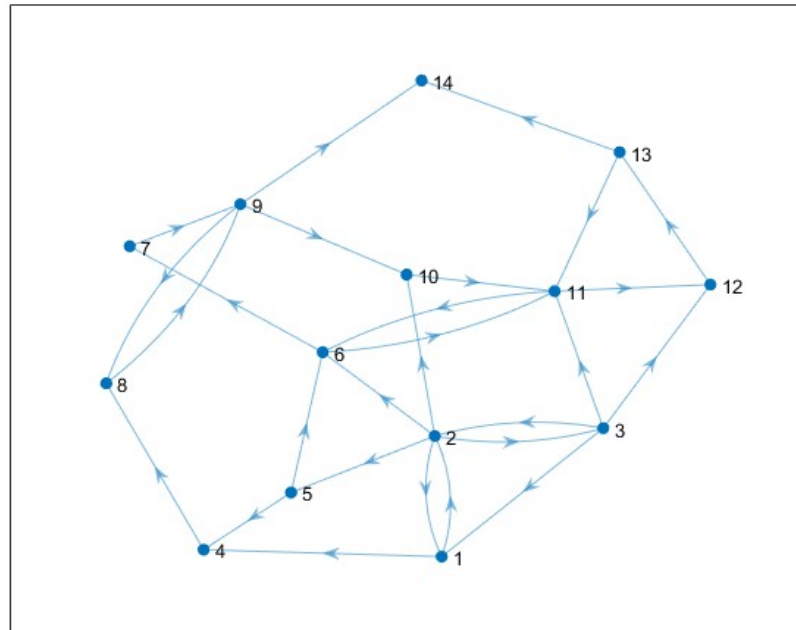


Figura 5.7: Grafo di prova con 14 nodi.

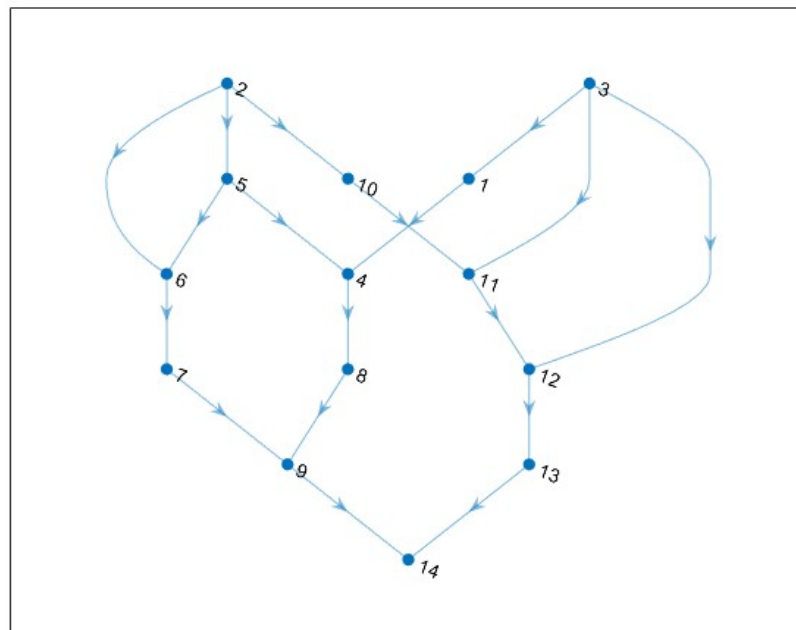


Figura 5.8: Alterazione del grafo con 14 nodi a seguito dell'algoritmo PLIM per $\tau = 200$.

La simulazione per l'Informazione Semplice restituisce un numero di vittime stimato pari a 5405, che diventano 5172 nel caso di Prescrizione Parziale. In nessuno dei 3 approcci il tempo di simulazione, $\tau = 200$, è sufficiente per evacuare del tutto. Anche aumentare τ a 250 non basta per risolvere il problema. Tuttavia, questo rilassamento delle condizioni semplifica ancora di più il grafo ottimale, che diventa il seguente:

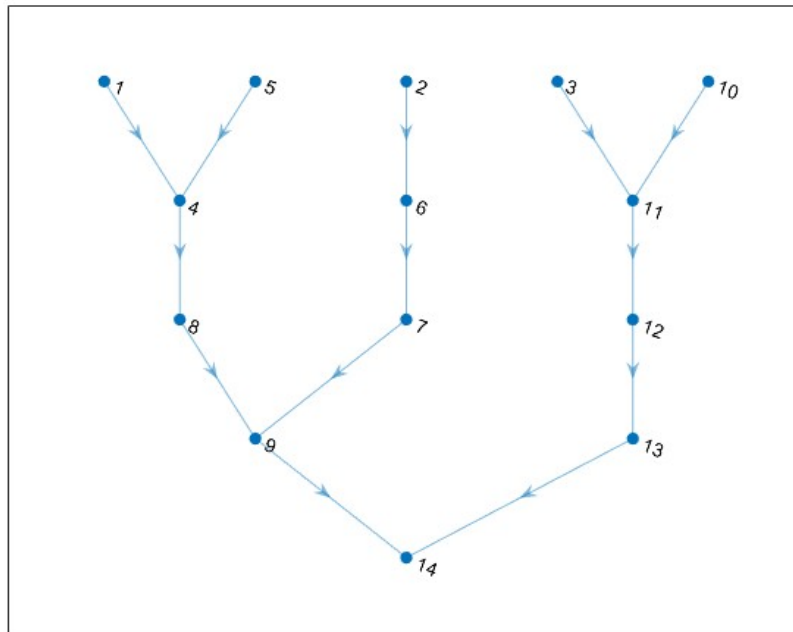


Figura 5.9: Alterazione del grafo con 14 nodi a seguito dell'algoritmo PLIM per $\tau = 250$.

5.2. Applicazione ad un grafo reale: Sioux Falls

5.2.1. Sioux Falls, South Dakota

Sioux Falls è una cittadina di circa 192.517 abitanti (**2020**) in Sud Dakota, nonché il centro più popoloso dello stato. L'area metropolitana della città si estende su 4 contee, per un totale di 276.730 abitanti ([link Wikipedia](#)):

- Lincoln, 65.161;
- McCook, 5.682;
- Minnehaha, 197.214;
- Turner, 8.673.

Dal punto di vista trasportistico, la situazione è riassunta dalla seguente mappa (1998):

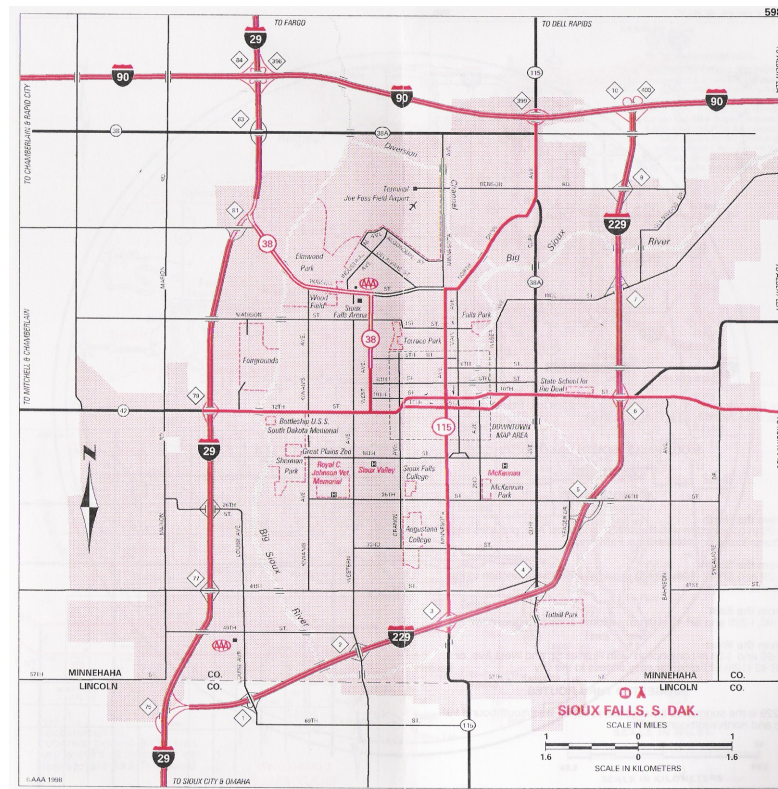


Figura 5.10: Mappa di Sioux Falls. Tratta da **GitHub**.

Per quanto riguarda la morfologia del territorio, Non si è ritenuto fondamentale per questo elaborato affrontare nel dettaglio la morfologia del territorio. L'obiettivo di questa tesi è prevalentemente modellistico; questa località è stata scelta per convenienza, visto che

il grafo della città è comunemente utilizzato in letteratura per scopi di Pianificazione dei Trasporti, per valutare l'efficacia o la correttezza di modelli ed algoritmi.

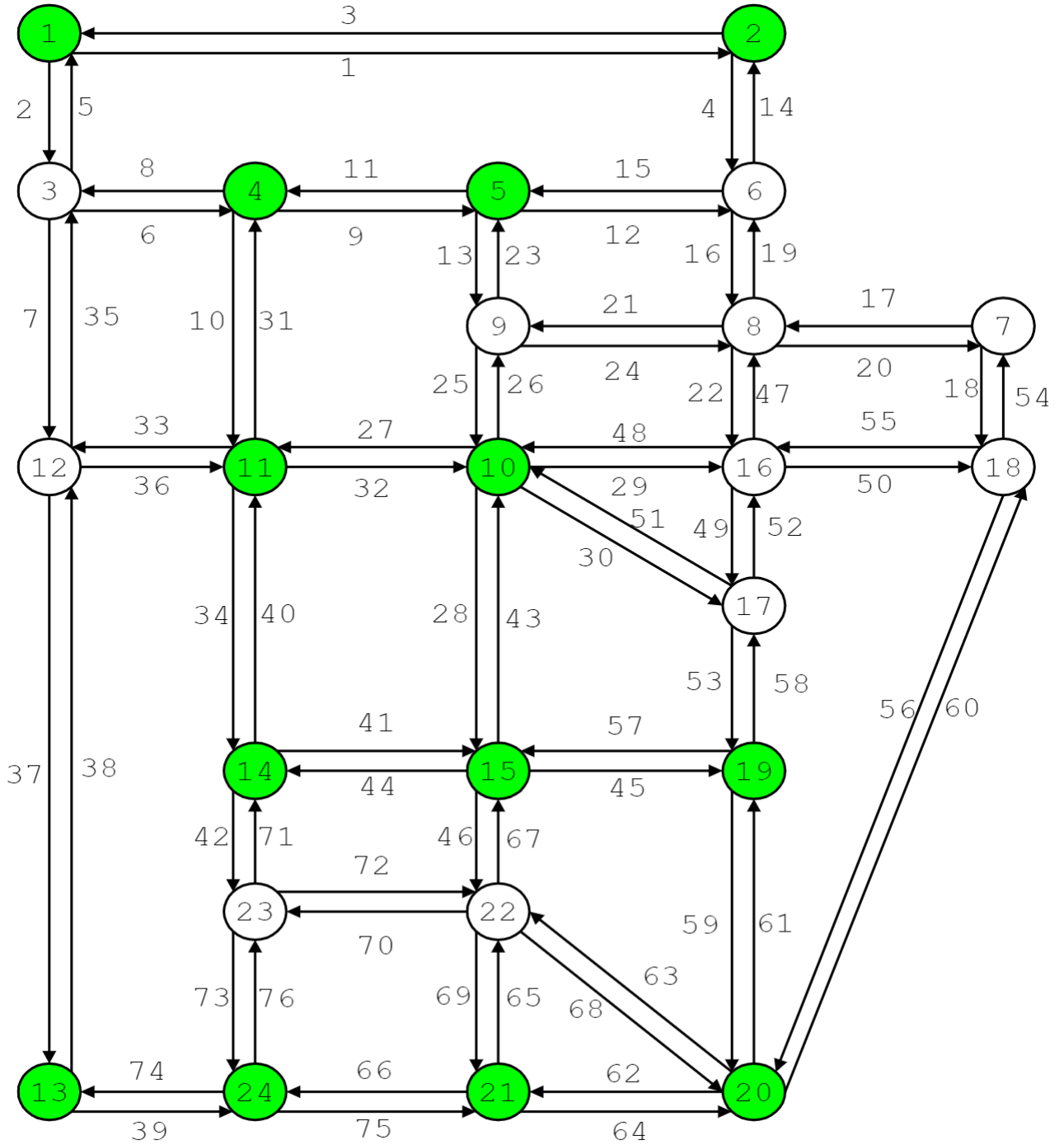


Figura 5.11: Grafo stradale di Sioux Falls. In verde, i centroidi. Tratto da **GitHub**.

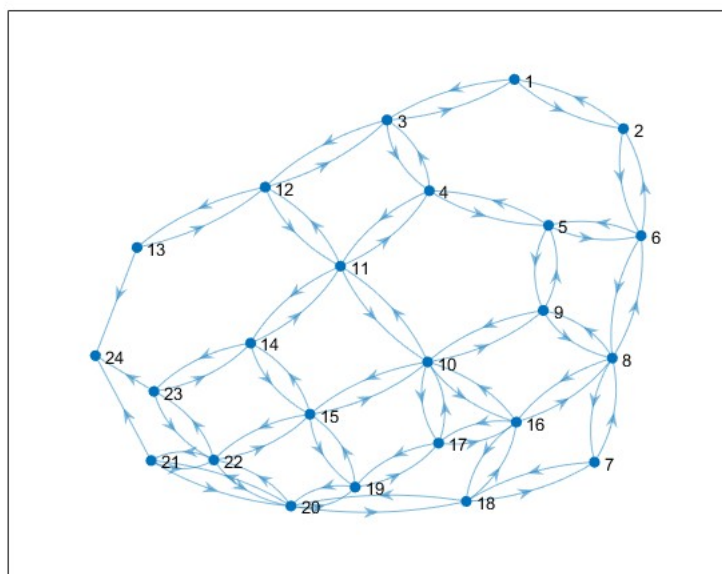


Figura 5.12: Rielaborazione Matlab del grafo stradale di Sioux Falls.

Il grafo in questione è costituito da 24 nodi, di cui 13 centroidi, connessi da 76 tra archi reali e archi connettori. I nodi e gli archi fittizi sono stati accorpati con metodologie classiche di ricalcolo degli attributi, in maniera molto simile a quanto accade in Elettrotecnica per il calcolo di resistenze/capacità/induttanze equivalenti.

Dal punto di vista pratico, il grafo è perfettamente simmetrico e le capacità delle strade sono particolarmente elevate. Queste caratteristiche riflettono le conseguenze delle scelte urbanistiche sulle prime città edificate negli Stati Uniti: la struttura di quartieri e distretti è perfettamente reticolare, come accaduto sin dai tempi delle colonie e degli accampamenti romani, a seguito di una capillare pianificazione a tavolino; inoltre, le strade urbane sono estremamente larghe, paragonabili a sezioni autostradali nel nostro paese, e permettono la mobilitazione di flussi veicolari imponenti.

5.2.2. Simulazioni e risultati

| Parametro | Valore |
|-----------------------|--------|
| zona S_1 | 1-8 |
| zona S_2 | 9-20 |
| zona S_3 | 21-24 |
| τ | 1200 |
| T_{MAX} | 240 |
| <i>domanda_totale</i> | 200000 |
| \tilde{c}_{ij} | 2 |
| β_1 | 0.15 |
| β_2 | 4 |

Tabella 5.3: Parametri grafo di Sioux Falls.

Per quanto riguarda la prima simulazione, relativa all'Informazione Semplice, i risultati sono estremamente promettenti: l'evacuazione viene conclusa perfettamente con abbondante anticipo, in soli 321 minuti, 5 h 21 min, a fronte di una finestra di sopravvivenza τ di 20 ore. La finestra di allontanamento dall'origine T_{MAX} , invece, è di 240 minuti, corrispondenti a 4 ore concesse agli utenti per abbandonare la propria posizione iniziale.

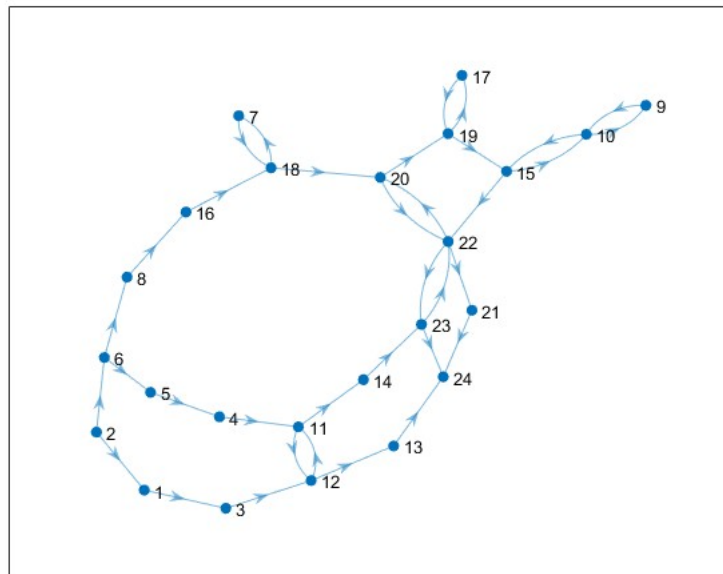


Figura 5.13: Grafo stradale di Sioux Falls a seguito dell'algoritmo PLIM.

La simulazione PLIM per l'Imposizione Totale ha rappresentato senz'altro la componente più problematica di questo elaborato. Per rendere l'idea, le simulazioni sui due grafi di prova hanno richiesto circa 2 e 10 minuti di tempo di compilazione, a fronte dei pochi secondi richiesti dall'algoritmo di assegnazione, che sono aumentati fino a tempi perfettamente accettabili, dell'ordine di un minuto scarso, una volta applicati al grafo reale. La simulazione PLIM sul grafo di Sioux Falls, invece, ha richiesto ben 21 ore consecutive di esecuzione. È stato necessario intervenire *ad hoc* sulla funzione `intlinprog` per modificare i parametri di calcolo, come tempo computazionale massimo, numero massimo di iterazioni del *Branch and Bound* e numero massimo di iterazioni al nodo radice. Le numerose prove hanno richiesto complessivamente un mese. Inizialmente, il problema computazionale principale era la larghezza della finestra τ . Per rendere l'idea, si veda la tabella seguente.

| Variabile | Cardinalità |
|----------------|---------------|
| f_{ij}^t | $l\tau$ |
| g_{ij}^t | $l\tau$ |
| x_{ij} | l |
| y_{ij} | l |
| m_i | $n - 1$ |
| Δq_i^t | $(n - 1)\tau$ |

Tabella 5.4: Dimensione computazionale variabili Imposizione Totale.

l è il numero di archi del grafo. Alle variabili riportate sopra, bisogna aggiungere $n\tau$ variabili di **totale_sicurezza**, generate per motivi di allocazione della funzione obiettivo in MATLAB. Non si riportano i dettagli delle quantità dei vincoli del problema, che in totale ammontano a $l + 3(n - 1) + 2 + n(n - 1) + (2n - 1)\tau + 3l\tau$, vincoli di definizione delle variabili esclusi. Inoltre, all'aumentare del numero di variabili il numero di operazioni eseguite aumenta più che linearmente. Non si entra nel merito della tematica; basti sapere che, euristicamente, dopo numerosi tentativi infruttuosi, errori di compilazione ed esaurimenti di memoria del PC, è stato deciso di inserire come finestra τ il valore di 300 minuti, 5 h, minimamente inferiore ai 321 minuti necessari per evacuare spontaneamente. Passando alla Prescrizione Parziale, il tempo richiesto non varia minimamente rispetto al primo caso, attestandosi esattamente allo stesso 321. Dal punto di vista della gestione dei flussi massimi, $\max_t \frac{F_{ij}^t}{U_{ij}}$, si registrano dei lievi, impercettibili, miglioramenti, ma considerando l'ingente variabilità del modello e la necessità di introdurre misure di gestione del territorio non banali, come le inversioni, le divergenze e le chiusure stradali, non si reputa conveniente l'attuazione del terzo approccio. Dati alla mano, l'approccio di Informazione Semplice sembrerebbe essere il più pratico e conveniente per il grafo di Sioux Falls.

6 | Conclusioni e sviluppi futuri

In conclusione, possiamo tirare le somme dei risultati e delle conseguenze di quanto esposto in questo elaborato.

Procedendo con ordine, il primo aspetto da valutare è l'efficacia dell'albero degli scenari di evacuazione. Si reputa che questa struttura metodologica possa essere sfruttata adeguatamente in quasi ogni contesto emergenziale, se teniamo sempre bene in mente che questi modelli non sono finalizzati all'implementazione dei piani d'emergenza o d'evacuazione, bensì ad essi preliminari. I risultati di nostro interesse, infatti, sono utili per:

1. stimare in via preliminare **l'entità dei flussi sugli archi**, per valutare le sezioni di **massima vulnerabilità della rete**;
2. **analizzare i nodi e gli archi più suscettibili di congestione**, in quanto più caotici ed intrinsecamente più propensi ad ospitare imprevisti e complicazioni. Ad esempio, un arco più affollato registrerà incidenti più facilmente, complicando la gestione dell'evacuazione; inoltre, in un contesto di elevato *stress* come un disastro naturale la probabilità che qualcuno di fragile, come anziani e malati, sia colpito da problematiche fisiche, come un infarto, aumenta considerevolmente;
3. **predisporre le risorse** in maniera adeguata ed efficiente sul territorio, ad esempio con i metodi di Ricerca Operativa Multilivello di [16].

Dal punto di vista scientifico-accademico, questo elaborato si inserisce trasversalmente nella letteratura trasportistica ed emergenziale, traendo fortissima ispirazione da branche diverse della Matematica Applicata; ci si augura che due fattori indirizzino la ricerca di settore verso la commistione tra questi ambiti, nel panorama trasportistico futuro:

- il crescente interesse verso tematiche di Ottimizzazione e Statistica, in primis l'Apprendimento Automatico / *Machine Learning*;
- la progressiva specializzazione dell'Ingegneria dei Trasporti e della Mobilità come branca a sé stante, piuttosto che come derivazione ancillare dell'Ingegneria Civile o Meccanica.

Passiamo ora alla descrizione dei modelli e degli algoritmi. **L'algoritmo PLIM** per l'Imposizione Totale è **computazionalmente molto oneroso**: mentre su grafi piccoli non si raggiungono tempistiche proibitive, le 21 ore di compilazione per simulare l'evacuazione sul grafo reale, di 24 archi e 76 nodi, costituiscono già un intervallo di tempo notevole. Grafi reali più complessi richiederebbero settimane e le simulazioni sarebbero proibitive per dei normali pc; si renderebbe necessario l'utilizzo di supercalcolatori dotati di RAM estremamente elevate.

Un **possibile spunto di miglioramento** potrebbe essere proprio su questo versante, cercando di **ridurre il numero di operazioni da eseguire**, evitando di allocare vincoli e variabili di flusso proporzionalmente all'orizzonte temporale previsto per l'evacuazione. Allo stesso tempo, tuttavia, sarebbe interessante valutare l'introduzione di funzioni di costo d'arco differenti da quelle lineari di questo elaborato, ricadendo così nel dominio dell'Ottimizzazione Non-Lineare, tematica estremamente tecnica, che si lascia agli specialisti della materia, e notoriamente molto onerosa nei calcoli.

Altre possibili variazioni sono comuni a tutti e 3 i modelli e riguardano la scelta dei parametri. Budget, costi d'arco, parametri BPR, ecc. sono stati ipotizzati sui grafi di prova in maniera congrua e ragionevole, **ma non sono stati soggetti a taratura specifica**; in linea di principio, dovremmo farli variare, confrontando di volta in volta i risultati, per verificare le effettive potenzialità dei modelli e redigere un'analisi di sensitività sulla loro influenza sul risultato finale. Lo stesso discorso, ovviamente, vale per i pochi parametri non noti nel caso di Sioux Falls.

Una nota speciale va segnalata per due elementi dei calcoli.

- La **distribuzione della popolazione tra i nodi** è stata generata casualmente con le modalità descritte nella tesi ed è stata mantenuta costante sullo stesso grafo passando da un modello all'altro. Tuttavia, è evidente che tale distribuzione può influenzare in maniera determinante i risultati finali: basti dare un'occhiata alle osservazioni fatte nel Capitolo 5 per il grafo più piccolo. Per omogeneizzare il più possibile i risultati, **occorre ripetere le simulazioni con differenti distribuzioni** ed osservare se sussistono variazioni significative.
- **I parametri comportamentali** legati alla Teoria dei Giochi, relativi alla distribuzione temporale delle immissioni sul grafo e ai fattori di pericolo, sono stati imposti in maniera verosimile e studiata a tavolino; in un contesto applicativo reale, è **consigliabile effettuare una calibrazione** adeguata su base statistica, come ad esempio con il celebre metodo della massima verosimiglianza, usato da [19] e [45].

Come nota ulteriore, si segnalano alcuni possibili spunti non immediati, che richiedono uno studio massiccio per essere implementati.

- **Passare a modelli a tempo continuo.** Nel caso dei modelli di assegnazione, si può pensare di modificare i modelli integrando il Link Transmission Model o il Cell Transmission Model per far propagare diversamente i flussi.
- Variare la **posizione lungo l'albero degli scenari** nei livelli di modellizzazione matematica. Implementare diversamente il gioco matematico o il problema di ottimizzazione potrebbe rivelarsi una scelta vincente, ma richiederebbe senz'altro un notevole sforzo modellistico.

Infine, si conclude questo lavoro con l'auspicio che la ricerca su queste tematiche delicate possa procedere lungo strade innovative e portare a dei nuovi punti di riferimento nella gestione delle emergenze su grande scala.

Bibliografia

- [1] O. Achrekar and C. Vogiatzis. *Evacuation Trees with Contraflow and Divergence Considerations*. Springer, 2018.
- [2] R. Alsnih and P. R. Stopher. *Review of Procedures Associated with Devising Emergency Evacuation Plans*. Transportation Research Record, 2006.
- [3] A. Andreas and J. Smith. Decomposition algorithms for the design of a nonsimultaneous capacitated evacuation tree network. 2009.
- [4] B. Y. K. and V. Verter. Designing a road network for Hazardous Materials Transportation. *Article in Transportation Science* · May 2004.
- [5] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty. *Nonlinear Programming. Theory and Algorithms*. Wiley interscience.
- [6] M. Bell, U. Kanturska, J.-D. Schmöcker, and A. Fonzone. Attacker-defender models and road network vulnerability. 2014.
- [7] M. G. H. Bell. A game theory approach to measuring the performance reliability of transport networks. 1999.
- [8] M. G. H. Bell. Mixed route strategies for the risk-averse shipment of hazardous materials. 2006.
- [9] M. G. H. Bell and C. Cassir. Risk-averse user equilibrium traffic assignment: an application of game theory. 2001.
- [10] A. Bertoni. Trasporto di sostanze pericolose su strada. analisi statistiche e stima degli impatti sulla rete stradale a seguito di incidenti rilevanti. Master's thesis, Politecnico di Milano, 2020.
- [11] D. Bertsimas and M. Sim. The price of robustness. 2004.
- [12] R. H. Bianchi. The 'Last mile two-level delivery problem': identifying key elements and developing a functional heuristic. Master's thesis, Politecnico di Milano, 2015.

- [13] D. F. Bignami. *Protezione civile e riduzione del rischio disastri. Metodi e strumenti di governo della sicurezza territoriale e ambientale*. Maggioli Editore, 2010.
- [14] A. Bonicelli and M. Comella. Sviluppo di indicatori di vulnerabilità sistemica di una rete di trasporto in transitorio e a regime. Master's thesis, Politecnico di Milano, 2010.
- [15] S. Boyd and L. Vanderberghe. *Convex optimization*. Cambridge University press, 2004.
- [16] G. Brown, M. Carlyle, J. Salmerón, and K. Wood. Defending critical infrastructure. *Operations Research Department, Naval Postgraduate School, Monterey, California 93943*.
- [17] V. Calabrese. Infrastrutture critiche. modello quantitativo per la stima della resilienza di infrastrutture stradali. Master's thesis, Politecnico di Milano, 2017.
- [18] G. Cantarella, D. Watling, S. de Luca, and R. D. Pace. *Dynamics and stochasticity in transportation systems*. Elsevier, 2019.
- [19] E. Cascetta. *Transportation Systems Analysis: Models and Applications*. Springer, 2009.
- [20] C. Cassir, M. G. H. Bell, and J. D. Schmoecker. A normative assessment of transport network reliability based on game theory. 2003.
- [21] S. Castelnuovo. Viabilità della sponda orientale del lago di como: definizione di interventi di riduzione della vulnerabilità. Master's thesis, Politecnico di Milano, 2014.
- [22] A. Comi. Dispense del corso di Teoria dei Sistemi di Trasporto.
- [23] M. Conforti, G. Cornuéjols, and G. Zambelli. *Integer programming*. Springer, 2014.
- [24] C. Crespi, G. Fargetta, M. Pavone, R. A. Scollo, and L. Scrimali. *A Game Theory Approach for Crowd Evacuation Modelling*. Springer, 2020.
- [25] L. S. Danna. Applicazione di un modello mesoscopico di traffico per l'analisi di un piano di evacuazione - caso reale. Master's thesis, Politecnico di Milano, 2009.
- [26] A. D. Dot. Problema di consegna dell'ultimo miglio a due livelli. Modello ed euristiche. Master's thesis, Politecnico di Milano, 2016.
- [27] D. Easley and J. Kleinberg. *Networks, Crowds, and Markets: Reasoning about a Highly Connected World*. Cambridge University press, 2010.

- [28] S. Ergün, P. Usta, S. Z. A. Gök, and G. W. Weber. A game theoretical approach to emergency logistics planning in natural disasters. 2021.
- [29] H. Fu, H. J. Pel, and S. P. Hoogendoorn. Optimization of evacuation traffic management with intersection control constraints. *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 1, february 2015.
- [30] S. Ghadimi and M. Wang. Approximation methods for bilevel programming. 2018.
- [31] R. Gibbons. *Teoria dei Giochi*. Il Mulino, 2005.
- [32] A. D. Giudice. Trasporto merci pericolose su strada: un modello per la gerarchizzazione del territorio mediante l'utilizzo di indicatori di sintesi. applicazione alla regione lombardia. Master's thesis, Politecnico di Milano, 2014.
- [33] H. W. Hamacher and S. A. Tjandra. Mathematical modelling of evacuation problems: A state of art. *Berichte des Fraunhofer ITWM, Nr. 24 (2001)*, 2001.
- [34] K. Hausken and J. Zhuang. *Game Theoretic Analysis of Congestion, Safety and Security. Networks, Air Traffic and Emergency Departments*. Springer, 2015.
- [35] Y. Hollander and J. N. Prashker. The applicability of non-cooperative game theory in transport analysis. 2006.
- [36] E. Israeli and R. K. Wood. Shortest-path network interdiction. 2002.
- [37] R. J. Koppa. *Traffic flow theory*. Transportation Research Board, 1975.
- [38] I. S. Kotsireas, A. Nagurney, and P. M. Pardalos. *Dynamics of Disasters—Key Concepts, Models, Algorithms, and Insights*. Springer, 2015.
- [39] A. Laurita. Decomposizione ed Euristiche Lagrangiana per ottimizzare la pianificazione del trasporto di merci pericolose in galleria. Master's thesis, Politecnico di Milano, 2011.
- [40] W. Li, J. Zhu, H. Li, QiongWu, and L. Zhang. A game theory based on monte carlo analysis for optimizing evacuation routing in complex scenes. 2015.
- [41] Y. Lim, B. G. Heydecker, and S. Lee. A continuous network design model in stochastic user equilibrium based on sensitivity analysis. *Journal of Advanced Transportation, Vol. 39, No. 1, pp. 63-79*, 39(1):63–79, 2010.
- [42] S. Lo, H. Huang, P. Wang, and K. Yuen. A game theory based exit selection model for evacuation. 2006.
- [43] R. Lucchetti. A primer in game theory.

- [44] V. G. Lupi and E. Stringhetti. Studio della vulnerabilità di una rete stradale in condizioni di eventi rilevanti: il caso di un'aluvione del fiume Po nel Basso Lodigiano. Master's thesis, Politecnico di Milano, 2007.
- [45] R. Maja. Dispense del corso di Tecnica ed Economia dei Trasporti.
- [46] A. Melloni. Robust Optimization Models for the Hazmat Transport Network Design Problem. Master's thesis, Politecnico di Milano, 2015.
- [47] L. Mussone. Dispense del corso di Dynamic Analysis of Transport Systems.
- [48] H. S. Na. Large-scale Evacuation network model for Transporting evacuees with multiple priorities. Master's thesis, Texas AM University, 2014.
- [49] F. Pan. *Stochastic Network Interdiction: Models and Methods*. PhD thesis, University of Austin, Texas, 5 2005.
- [50] M. C. Sechi. Trasporto di sostanze pericolose su strada. un indice per la valutazione dinamica della robustness della rete a seguito di eventi rilevanti. Master's thesis, Politecnico di Milano, 2021.
- [51] Y. Sheffi. *Urban Transportation Networks: Equilibrium Analysis with mathematical Programming methods*. Prentice Hall, Inc., 1985.
- [52] K. Sigmund. Introduction to evolutionary game theory. In *Proceedings of Symposia in Applied Mathematics*. 2011.
- [53] F. Sironi. La valutazione del rischio nel trasporto merci pericolose: il caso delle gallerie lecchesi. Master's thesis, Politecnico di Milano, 2012.
- [54] J. Trentin. Analisi tramite Teoria dei Giochi della competizione nella distribuzione di contenuti in reti veicolari. Master's thesis, Politecnico di Milano, 2011.
- [55] M. Turicchi. Sicurezza del trasporto ferroviario delle merci pericolose: metodo di analisi ed esempi applicativi. Master's thesis, Politecnico di Milano, 2015.
- [56] V. Ventriglia. Sicurezza nel trasporto di merci pericolose su strada: confronto tra due approcci quantitativi per la valutazione del rischio. Master's thesis, Politecnico di Milano, 2017.
- [57] L. Volpini and O. Redaelli. Il problema della selezione di rete in ambito wireless: analisi dell'interazione tra reti ed utenti tramite un gioco non cooperativo bi-livello. Master's thesis, Politecnico di Milano, 2010.
- [58] E. Zio. *Introduction to the basics of Reliability and Risk analysis*. 2007.

A | Appendice A

Si riportano di seguito i codici utilizzati per il grafo di prova più piccolo con 6 nodi. Per brevità, si omettono i codici relativi al grafo reale. Le uniche differenze sono l'introduzione dei dati, provenienti da **GitHub** e inseriti ad hoc, e i codici (banali) per interpretare le soluzioni restituite dalla funzione **intlinprog**.

A.1. Semplice Informazione - 6 nodi

```

1  %costruzione matrice d'incidenza
   n=6;
3  A=ones(n,n)-eye(n);
   A(1,5)=0; A(1,6)=0; A(2,3)=0; A(2,5)=0;
5  A(2,6)=0; A(3,2)=0; A(4,1)=0; A(5,1)=0;
   A(5,2)=0; A(5,3)=0; A(6,1)=0; A(6,2)=0;
7  A(6,3)=0; A(6,4)=0; A(6,5)=0; A
   B=A; B(n,:)=[];
9  G=digraph(A); plot(G);
   %parametri generali
11 U=50*A; domanda_totale=1000; zone=[2 2]; tau=200; T=50;
   Q=generazione_domanda_distrib_normale(A,domanda_totale);
13 c0 = A; trova_zero = find(c0==0); c0(trova_zero) = Inf; c0=2*c0;

15 alpha=0.8;
   %parametro logit
17 theta = 7;
   %parametri BPR
19 beta1 = 0.1*A; beta2 = 2*A;
   %punto di raccolta
21 dest = n;

23 %parametri modello domanda ai nodi
   aT = alpha*T; a = zeros(1,n-1); b = zeros(1,n-1); c = zeros(1,n-1);
25 for i=1:n-1
       [a(i),b(i),c(i)]=parametri_teorica_dei_giochi(Q(i),alpha,T);
27 end
   dQ = zeros(1,n-1);
29 %parametri costi d'arco stimati
   r3 = 2; r2 = 2; r1 = 2; R = zeros(n,n);
31 X = 2; Y = 1/20; Z = 1/100;
   R(:,1:zone(1)) = r1; R(:,zone(1)+1:zone(1)+zone(2)) = r2;
33 R(:,zone(1)+zone(2)+1:n) = r3; w = R.*c0;

35 nodi = 1:n; successori = cell(1,n); predecessori = cell(1,n);
   for i=1:n-1
37     successori{i} = successori(G,nodi(i));
       predecessori{i} = predecessori(G,nodi(i));
39 end

41 f = cell(T,n-1); F= cell(T,n-1); max_F=zeros(n,n);
   N=zeros(n,n); h=c0;
43
   %primo minuto
45 t=1;
   short_path = {}; tot_cost = {}; P={};

```

```

47  c_arco = cell(T,1); c_arco{1,1} = c0;

49  for i=1:n-1
      dQ(i) = deltaQ(a(i),b(i),c(i),t);
51  Q(i) = Q(i) - dQ(i);
      end
53  %R(:,1:2) = r1;
      R(:,zone(1)+1:zone(1)+zone(2)) = S2(X,Y,t);
55  R(:,1:zone(1)) = S1(X,Y,Z,t);
      %R(:,3:4) = S2(X,Y,t);
57  %R(:,5:6) = S3(X,Y,Z,t);
      w = R.*c0;
59  for i=1:n-1
      numero_successori = length(successori{1,i});
61  for j=1:numero_successori
      save_successori = successori{1,i}(j);
63  [shortestPath, totalCost] = dijkstra(w, save_successori, dest);
      short_path{i,j} = shortestPath;
65  tot_cost{i,j} = totalCost;
      h(i,save_successori) = w(i,save_successori) + tot_cost{i,j};
67  end
      P{1,i}=logit(theta,h(i,successori{1,i}),numero_successori);
69  end

71  for i=1:n-1
      numero_predecessori = length(predecessori{1,i});
73  save_predecessori = predecessori{1,i};
      somma_f(i)=0;
75  conta=0;
      m=1;
77  while (conta<numero_predecessori)&& (t-m>0)
      for k=1:numero_predecessori
79  successori_del_predecessore=successori{1,save_predecessori(k)};
      numero_successori_del_predecessore=length(successori_del_predecessore);
81  flag=0;
      z=0;
83  while (flag==0) && (z<=numero_successori_del_predecessore)
      z=z+1;
85  if (successori_del_predecessore(z)==i)
      flag=1;
87  end

89  end

91  if (floor(c_arco{t-m,1}(save_predecessori(k),i))==m)
      somma_f(i)=somma_f(i)+f{t-m,save_predecessori(k)}(z);
93  conta=conta+1;
      end
95  end
      m=m+1;
97  end
      somma_f(i)= somma_f(i)+dQ(i); %flussi di prossimita
99  f{t,i}=round(somma_f(i)*P{1,i});
      F{t,i}=0*f{t,i}; %inizializzo F per non creare problemi di struttura dati e allocazione memoria
101 end

103 max_passati=0;
      for i=1:n-1
105  numero_successori = length(successori{1,i});
      save_successori = successori{1,i};
107  for j=1:numero_successori

109  max_passati=N(i,save_successori(j))+1;

111

113  while (max_passati>=1) && (c_arco{t-max_passati+1,1}(i,save_successori(j))<max_passati+1)
      max_passati=max_passati-1;
      end

115  for k=1:max_passati
117  F{t,i}(j)=F{t,i}(j)+f{t+k-max_passati,i}(j);

119  end
      N(i,save_successori(j))=max_passati;
121  if (max_F(i,save_successori(j))<F{t,i}(j))

```

```

123         max_F(i,save_successori(j))=F{t,i}(j);
124     end
125 end
126 end
127
128     for i=1:n-1
129         numero_successori = length(successori{1,i});
130         save_successori = successori{1,i};
131         for j=1:numero_successori
132             c_arco{t,1}(i,save_successori(j))=c_arco{1,1}(i,save_successori(j))*
133             BPR(beta1(i,save_successori(j)),beta2(i,save_successori(j)),F{t,i}(j),U(i,save_successori(j)));
134         end
135     end
136
137     t=2;
138     vittime_potenziati=1;
139     while (t<=tau) && (vittime_potenziati>0)
140         vittime_potenziati=0;
141
142         for i=1:n-1
143             dQ(i) = max(0,deltaQ(a(i),b(i),c(i),t));
144             Q(i) = Q(i) - dQ(i);
145         end
146
147         R(:,zone(1)+1:zone(1)+zone(2)) = S2(X,Y,t);
148         R(:,1:zone(1)) = S1(X,Y,Z,t);
149
150         w = R.*c0;
151         for i=1:n-1
152             numero_successori = length(successori{1,i});
153             for j=1:numero_successori
154                 save_successori = successori{1,i}(j);
155                 [shortestPath, totalCost] = dijkstra(w, save_successori, dest);
156                 short_path{i,j} = shortestPath;
157                 tot_cost{i,j} = totalCost;
158                 h(i,save_successori) = w(i,save_successori) + tot_cost{i,j};
159             end
160             P{1,i}=logit(theta,h(i,successori{1,i}),numero_successori);
161         end
162
163         for i=1:n-1
164             numero_predecessori = length(predecessori{1,i});
165             save_predecessori = predecessori{1,i};
166             somma_f(i)=0;
167             conta=0;
168             m=1;
169             while (conta<numero_predecessori)&& (t-m>0)
170                 for k=1:numero_predecessori
171                     successori_del_predecessore=successori{1,save_predecessori(k)};
172                     numero_successori_del_predecessore=length(successori_del_predecessore);
173                     flag=0;
174                     z=0;
175                     while (flag==0) && (z<=numero_successori_del_predecessore)
176                         z=z+1;
177                         if (successori_del_predecessore(z)==i)
178                             flag=1;
179                         end
180                     end
181
182                     if (floor(c_arco{t-m,1}(save_predecessori(k),i))==m)
183                         somma_f(i)=somma_f(i)+f{t-m,save_predecessori(k)}(z);
184                         conta=conta+1;
185                         %si potrebbe riflettere sull'ottimizzazione di questo
186                         %ciclo per non farlo tornare sui nodi gi visitati...
187                     end
188                 end
189                 m=m+1;
190             end
191             somma_f(i)= somma_f(i)+dQ(i); %questi sono quelli che escono
192             f{t,i}=round(somma_f(i)*P{1,i});
193             F{t,i}=0*f{t,i}; %inizializzo F per non creare problemi di struttura dati e allocazione memoria
194         end
195
196     max_passati=0;
197     for i=1:n-1

```

```

199     numero_successori = length(successori{1,i});
      save_successori = successori{1,i};
201     for j=1:numero_successori
203         max_passati=N(i,save_successori(j))+1;
205         while (max_passati>=1) && (t-max_passati<=0)
            max_passati=max_passati-1;
207         end
            while (max_passati>=1) && (c_arco{t-max_passati,1}(i,save_successori(j))<max_passati)
209                 max_passati=max_passati-1;
            end
211         for k=0:max_passati
213             F{t,i}(j)=F{t,i}(j)+f{t+k-max_passati,i}(j);
215         end
            N(i,save_successori(j))=max_passati;
217         if (max_F(i,save_successori(j))<F{t,i}(j))
219             max_F(i,save_successori(j))=F{t,i}(j);
        end
221     end
end
223     for i=1:n-1
225         numero_successori = length(successori{1,i});
        save_successori = successori{1,i};
227         for j=1:numero_successori
            c_arco{t,1}(i,save_successori(j))=c_arco{1,1}(i,save_successori(j))*
229             BPR(beta1(i,save_successori(j)),beta2(i,save_successori(j)),F{t,i}(j),U(i,save_successori(j)));
            vittime_potenziali=vittime_potenziali+F{t,i}(j);
231             %calcolo le potenziali vittime sommando i flussi che non sono ancora arrivati a destinazione
        end
233     end

235 t=t+1;
end
237 vittime=vittime_potenziali;
t=t-1;

```

A.2. Imposizione Totale - 6 nodi

```

n=6;
2 A=ones(n,n)-eye(n);
A(1,5)=0;
4 A(1,6)=0;
A(2,3)=0;
6 A(2,5)=0;
A(2,6)=0;
8 A(3,2)=0;
A(4,1)=0;
10 A(5,1)=0;
A(5,2)=0;
12 A(5,3)=0;
A(6,1)=0;
14 A(6,2)=0;
A(6,3)=0;
16 A(6,4)=0;
A(6,5)=0;
18 A
G=digraph(A)
20 plot(G)
B=A;
22 B(n,:)=[];
G=digraph(A);
24 plot(G);
%%
26 %METTO PER ISCRITTO L'ORDINAMENTO-P DEI NODI
l=nnz(A);
28 [I,J]=find(A);
punti=zeros(l,2);
30 punti(:,1)=I;
punti(:,2)=J;
32 punti_AT=zeros(l,3);
[I,J]=find(A');
34 punti_AT(:,1)=J;
punti_AT(:,2)=I;
36 clear("I")
clear("J")
38 %dest = n;
tau = 200;
40 T = 50;
c0 = ones(1,1);
42 punti(:,3)=zeros(1,1);

44 for i=1:l
j=1;
46 while punti(i,3)==0 && j<l
if (punti(i,1)==punti(j,2) && (punti(i,2)==punti(j,1)))
48 punti(i,3)=j;
end
50 j=j+1;
end
52 end
for i=1:l
54 j=1;
56 while punti_AT(i,3)==0 && j<l
if (punti_AT(i,1)==punti_AT(j,2) && (punti_AT(i,2)==punti_AT(j,1)))
58 punti_AT(i,3)=j;
end
60 j=j+1;
end
62 end
%%
64 %COSTRUISCO L'ORDINAMENTO-P DEI NODI
prog=zeros(n-1,1);
66 nodi = 1:n;
successori = cell(1,n);
68 predecessori = cell(1,n);
for i=1:n-1
70 successori{i} = successors(G,nodi(i));
predecessori{i} = predecessors(G,nodi(i));
72 end

```

```

for i=1:n-1
74     prog(i)=length(successori{i});
end
76     progressiva=zeros(n-1,2);

78     for i=1:n-1
         for j=1:i
80             progressiva(i,2)=progressiva(i,2)+prog(j);
         end
82     end
progressiva(1,1)=1;
84     for i=2:n-1
         progressiva(i,1)=progressiva(i-1,2)+1;
86     end
indici_stella_uscente=progressiva;
88
aux_punti_AT=zeros(1,4);
90     aux_punti_AT(:,1:3)=punti_AT(:,1:3);

92     for i=1:l
         aux_punti_AT(i,4)=i;
94     end
aux_punti=zeros(1,4);
96     aux_punti(:,1:3)=punti(:,1:3);
for i=1:l
98         aux_punti(i,4)=i;
end
100
% capacit
102     U=ones(1,1);
U=50*U;
104     % domanda
domanda_totale=1000;
106     zone=[2 2]
%Q=generazione_domanda_distrib_normale(A,domanda_totale);
108     Q=[268
         4
110     105
         409
112     214
         ]
114
% costi d'arco
116     congestione=3;
w=congestione*c0;
118     % costi inversione
costi_inversione=ones(1,1);
120     costi_inversione=20*costi_inversione;
% budget inversione
122     budget_inversione=100;
%costo divergenze
124     costi_divergenze=ones(n-1,1);
costi_divergenze=20*costi_divergenze;
126     %budget divergenze
budget_divergenze=100;
128     % fattori di pericolo
R = zeros(n,tau);
130     r3 = 0;
r2 = 1;
132     for tt=1:tau
         for ii=1:zone(1)
134             R(ii,tt)=tt;
         end
136         for ii=(zone(1)+1):(zone(1)+zone(2))
             R(ii,tt)=r2;
138         end
         for ii=(zone(1)+zone(2)+1):(n-1)
140             R(ii,tt)=r3;
         end
142     end

144     for tt=1:tau
         R(n,tt)=tt-tau;
146     end

148     matrice_x_in_x_out=zeros(n-1,l);

```

```

150     for j=1:l
151         matrice_x_in_x_out(aux_punti_AT(j,1),j)=1;
152         if (aux_punti_AT(j,3)>0)
153             matrice_x_in_x_out(aux_punti_AT(j,1),aux_punti_AT(j,3))=-1;
154         end
155     end
156     matrice_ingresso=zeros(n-1,1);
157     matrice_uscita=zeros(n,1);
158     for j=1:l
159         matrice_ingresso(aux_punti_AT(j,1),j)=1;
160         if (aux_punti_AT(j,3)>0)
161             matrice_uscita(aux_punti_AT(j,1),aux_punti_AT(j,3))=-1;
162         else
163             if (aux_punti_AT(j,2)==n)
164                 matrice_uscita(n,j)=-1;
165             end
166         end
167     end
168     %%
169     %DICHIAZIONE VARIABILI DI CONTROLLO
170     x = optimvar('x',1,'Type','integer','LowerBound',0,'UpperBound',1);
171     y = optimvar('y',1,'Type','integer','LowerBound',0,'UpperBound',1);
172     f = optimvar('f',1,tau,'LowerBound',0);
173     g = optimvar('g',1,tau,'LowerBound',0);
174     numero_divergenze=optimvar('numero_divergenze',n-1,'Type','integer','LowerBound',0,'UpperBound',n-1);
175     dq= optimvar('dq',n-1,tau,'Type','integer','LowerBound',0);
176     totale_sicurezza=optimvar('totale_sicurezza',n,tau,'Type','integer','LowerBound',0);
177     evacuation_prob = optimproblem;
178     %%
179     %ALLOCAZIONE VINCOLI
180     x_y=optimconstr(1);
181     somma_q_dq=optimconstr(n-1);
182     x_in_x_out=optimconstr(n-1);
183     bilancio_divergenze=optimconstr(n-1);
184     budget_y=optimconstr(1);
185     budget_m=optimconstr(1);
186     % flusso_max=optimconstr(n,n-1);
187     definizione_g=optimconstr(1,tau);
188     flussi_in_transito=optimconstr(n-1,tau);
189     bilancio_flussi_domanda=optimconstr(1,tau);
190     totale_sicurezza_lower=optimconstr(n,tau);
191     saturazione_flusso=optimconstr(1,tau);
192     %%
193     %DICHIAZIONE VINCOLI
194     for p=1:l
195         if aux_punti_AT(p,3)>0
196             x_y(p)= y(p)<=x(aux_punti_AT(p,3));
197         else
198             x_y(p)= y(p)<=0;
199         end
200     end
201     for i=1:n-1
202         somma_q_dq(i) =sum(dq(i,:))=Q(i);
203     end
204     for i=1:n-1
205         x_in_x_out(i)= matrice_x_in_x_out(i,:)*x>=0;
206     end
207     budget_y = costi_inversione'*y<=budget_inversione;
208     budget_m = costi_divergenze'*numero_divergenze<=budget_divergenze;
209     for i=1:n-1
210         bilancio_divergenze(i)= matrice_ingresso(i,:)*x==1+numero_divergenze(i);
211     end
212     for t=1:tau
213         for p=1:l
214             if (t<=w(p))
215                 definizione_g(p,t)= g(p,t)==0;
216             else
217                 definizione_g(p,t)= g(p,t)==f(p,t-w(p));
218             end
219         end
220     end
221     end
222     for t=1:tau
223         for i=1:n-1
224             flussi_in_transito(i,t)= matrice_ingresso(i,:)*f(:,t)+ matrice_uscita(i,:)*g(:,t)-dq(i,t)==0;

```

```

    end
226 end

228 for p=1:l
    for t=1:tau
230         if (aux_punti_AT(p,3)>0)
                saturazione_flusso(p,t)= f(p,t)<=U(p)*x(p)+U(aux_punti_AT(p,3))*y(aux_punti_AT(p,3));
232         else
                saturazione_flusso(p,t)= f(p,t)<=U(p)*x(p);
234         end
    end
236 end
for p=1:l
238     for t=1:tau
            if (aux_punti_AT(p,3)>0)
240                 if (t<=w(p)-1)
                        bilancio_flussi_domanda(p,t)= sum(f(p,1:t))<=U(p)*x(p)+U(aux_punti_AT(p,3))*y(aux_punti_AT(p,3));
242                 elseif ((t>=w(p))&&(t<=tau-w(p)))
                        bilancio_flussi_domanda(p,t)= sum(g(p,t+1:t+w(p)))<=U(p)*x(p)+U(aux_punti_AT(p,3))*y(aux_punti_AT(p,3));
244                 else
                        bilancio_flussi_domanda(p,t)= sum(g(p,t+1:tau))<=U(p)*x(p)+U(aux_punti_AT(p,3))*y(aux_punti_AT(p,3));
246                 end
            else
248                 if (t<=w(p)-1)
                        bilancio_flussi_domanda(p,t)= sum(f(p,1:t))<=U(p)*x(p);
250                 elseif ((t>=w(p))&&(t<=tau-w(p)))
                        bilancio_flussi_domanda(p,t)= sum(g(p,t+1:t+w(p)))<=U(p)*x(p);
252                 else
                        bilancio_flussi_domanda(p,t)= sum(g(p,t+1:tau))<=U(p)*x(p);
254                 end
            end
256     end
end
258
for j=1:n
260     for t=1:tau
            totale_sicurezza_lower(j,t)= totale_sicurezza(j,t)== -matrice_uscita(j,:)*f(:,t);
262     end
end
264
obiettivo=sum(sum(R.*totale_sicurezza));
266 %%
%%ALLOCAZIONE VINCOLI E SOLUZIONE
268 evacuation_prob.Constraints.x_y=x_y;
evacuation_prob.Constraints.somma_q_dq=somma_q_dq;
270 evacuation_prob.Constraints.x_in_x_out=x_in_x_out;
evacuation_prob.Constraints.budget_m=budget_m;
272 evacuation_prob.Constraints.budget_y=budget_y;
evacuation_prob.Constraints.bilancio_divergenze=bilancio_divergenze;
274 evacuation_prob.Constraints.definizione_g=definizione_g;
evacuation_prob.Constraints.flussi_in_transito=flussi_in_transito;
276 evacuation_prob.Constraints.bilancio_flussi_domanda=bilancio_flussi_domanda;
evacuation_prob.Constraints.totale_sicurezza_lower=totale_sicurezza_lower;
278 % evacuation_prob.Constraints.totale_sicurezza_upper=totale_sicurezza_upper;
evacuation_prob.Constraints.saturazione_flusso=saturazione_flusso;
280 evacuation_prob.Objective=obiettivo;

282 [sol,fval]=solve(evacuation_prob);
%%VARIABILI FINALI DI INTERESSE
284 soluzione.DQ=round(sol.dq);
soluzione.F=round(sol.f);
286 soluzione.M=round(sol.numero_divergenze);
soluzione.X=round(sol.x);
288 soluzione.Y=round(sol.y);

290 MAXF=zeros(p,1);
for j=1:p
292     for k=1:tau
            if MAXF(j)<soluzione.F(j,k)
294                 MAXF(j)=soluzione.F(j,k);
            end
296     end
end

```

Come illustrato nella procedura della sezione 4.5, il codice per la Prescrizione Parziale è esattamente coincidente con quello per la Semplice Informazione, a patto di aver eseguito le seguenti righe di codice per inserire la matrice di incidenza e la matrice delle capacità aggiornate.

```
1  %COSTRUZIONE DATI SIMULAZIONE PRESCRIZIONE PARZIALE
   aiuto=aux_punti_AT;
3  Anew=A;
   Unew=zeros(n,n);
5  for k=1:p
       Unew(aiuto(k,1),aiuto(k,2))=U(k);
7  end
   for k=1:p
9     if soluzione.X(k)==0
           Anew(aiuto(k,1),aiuto(k,2))=0;
11        end
       if soluzione.Y(k)==1
13           Unew(aiuto(k,2),aiuto(k,1))=Unew(aiuto(k,2),aiuto(k,1))+Unew(aiuto(k,1),aiuto(k,2));
           Unew(aiuto(k,1),aiuto(k,2))=0;
15        end
   end
```

A.3. Funzioni ausiliarie

Si riportano di seguito le funzioni ausiliarie, piuttosto semplici, definite per accelerare e rendere più comprensibile la stesura del codice per l'assegnazione.

A.3.1. Logit

```

function y = logit(theta,h,n_successori)
2 sumexp = 0;
y=zeros(n_successori,1);
4 for i=1:n_successori
    y(i) = (exp(-h(i)/theta));
6 sumexp = sumexp + y(i);
end
8 y = y./sumexp;
end

```

A.3.2. BPR

```

1 function y = BPR(beta1,beta2,F,C)
y = 1+beta1*(F/C)^beta2;
3 end

```

A.3.3. Generazione domanda distribuzione normale

```

1 function Q = generazione_domanda_distrib_normale(A,domanda_totale)
n=size(A,2);
3 Q=randn(n-1,1);
Q=abs(Q);
5 QTOT=sum(Q);
Q=round(domanda_totale*(1/QTOT)*Q);
7 end

```

A.3.4. Parametri distribuzione uscita

```

1 function [a,b,c]=parametri_teorìa_dei_giochi(Q,alpha,T)
c=Q/(2*alpha*T);
3 a=3*c/(alpha^2*T^2-1);
b=a*alpha*T;
5 end

7 function y = deltaQ(a,b,c,t)
y = c+b.*t-a.*t^2;
9 end

```

A.3.5. Calcolo fattori di pericolo

```

1 function y = S1(a,b,c,t)
y = a+b.*t+c.*t^2;
3 end

5 function y = S2(a,b,t)
y = a+b.*t;
7 end

```

A.3.6. Dijkstra

Di seguito, troviamo una delle possibili formulazioni dell'algorithmo di **Dijkstra**. Tratto dal sito di Mathworks, lavoro di Xiaodong Wang.

```

1     function [shortestPath, totalCost] = dijkstra(netCostMatrix, s, d)
2     %=====
3     % shortestPath: the list of nodes in the shortestPath from source to destination;
4     % totalCost: the total cost of the shortestPath;
5     % farthestNode: the farthest node to reach for each node after performing the routing;
6     % n: the number of nodes in the network;
7     % s: source node index;
8     % d: destination node index;
9     %=====
10    % Code by:
11    % ++by Xiaodong Wang
12    % ++23 Jul 2004 (Updated 29 Jul 2004)
13    % ++http://www.mathworks.com/matlabcentral/fileexchange/5550-dijkstra-shortest-path-routing
14    % Modifications (simplifications) by Meral Shirazipour 9 Dec 2009
15    %=====
16    n = size(netCostMatrix,1);
17    for i = 1:n
18        % initialize the farthest node to be itself;
19        farthestPrevHop(i) = i; % used to compute the RTS/CTS range;
20        farthestNextHop(i) = i;
21    end
22
23    % all the nodes are un-visited;
24    visited(1:n) = false;
25
26    distance(1:n) = inf; % it stores the shortest distance between each node and the source node;
27    parent(1:n) = 0;
28    distance(s) = 0;
29    for i = 1:(n-1),
30        temp = [];
31        for h = 1:n,
32            if ~visited(h) % in the tree;
33                temp=[temp distance(h)];
34            else
35                temp=[temp inf];
36            end
37        end;
38        [t, u] = min(temp); % it starts from node with the shortest distance to the source;
39        visited(u) = true; % mark it as visited;
40        for v = 1:n, % for each neighbors of node u;
41            if ( ( netCostMatrix(u, v) + distance(u) ) < distance(v) )
42                distance(v) = distance(u) + netCostMatrix(u, v); % update the shortest distance when a shorter shortestPath is found;
43                parent(v) = u; % update its parent;
44            end;
45        end;
46    end;
47    shortestPath = [];
48    if parent(d) ~= 0 % if there is a shortestPath!
49        t = d;
50        shortestPath = [d];
51        while t ~= s
52            p = parent(t);
53            shortestPath = [p shortestPath];
54
55            if netCostMatrix(t, farthestPrevHop(t)) < netCostMatrix(t, p)
56                farthestPrevHop(t) = p;
57            end;
58            if netCostMatrix(p, farthestNextHop(p)) < netCostMatrix(p, t)
59                farthestNextHop(p) = t;
60            end;
61
62            t = p;
63        end;
64    end;
65    totalCost = distance(d);
66
67    %return;

```

B | Appendice B

B.1. Parametri comuni

| Parametro | Valore |
|-----------|--------|
| α | 0.8 |
| θ | 7 |
| x | 2 |
| y | 0.05 |
| z | 0.01 |

Tabella B.1: Parametri comuni modello di assegnazione.

B.2. Parametri Sioux Falls

| DATI= | | | | | | | | | | | |
|-------|----|----|-------------|----|----|------|---|---|---|---|---|
| 2 | [1 | 2 | 25900.20064 | 6 | 6 | 0.15 | 4 | 0 | 0 | 1 | ; |
| | 1 | 3 | 23403.47319 | 4 | 4 | 0.15 | 4 | 0 | 0 | 1 | ; |
| 4 | 2 | 1 | 25900.20064 | 6 | 6 | 0.15 | 4 | 0 | 0 | 1 | ; |
| | 2 | 6 | 4958.180928 | 5 | 5 | 0.15 | 4 | 0 | 0 | 1 | ; |
| 6 | 3 | 1 | 23403.47319 | 4 | 4 | 0.15 | 4 | 0 | 0 | 1 | ; |
| | 3 | 4 | 17110.52372 | 4 | 4 | 0.15 | 4 | 0 | 0 | 1 | ; |
| 8 | 3 | 12 | 23403.47319 | 4 | 4 | 0.15 | 4 | 0 | 0 | 1 | ; |
| | 4 | 3 | 17110.52372 | 4 | 4 | 0.15 | 4 | 0 | 0 | 1 | ; |
| 10 | 4 | 5 | 17782.7941 | 2 | 2 | 0.15 | 4 | 0 | 0 | 1 | ; |
| | 4 | 11 | 4908.82673 | 6 | 6 | 0.15 | 4 | 0 | 0 | 1 | ; |
| 12 | 5 | 4 | 17782.7941 | 2 | 2 | 0.15 | 4 | 0 | 0 | 1 | ; |
| | 5 | 6 | 4947.995469 | 4 | 4 | 0.15 | 4 | 0 | 0 | 1 | ; |
| 14 | 5 | 9 | 10000 | 5 | 5 | 0.15 | 4 | 0 | 0 | 1 | ; |
| | 6 | 2 | 4958.180928 | 5 | 5 | 0.15 | 4 | 0 | 0 | 1 | ; |
| 16 | 6 | 5 | 4947.995469 | 4 | 4 | 0.15 | 4 | 0 | 0 | 1 | ; |
| | 6 | 8 | 4898.587646 | 2 | 2 | 0.15 | 4 | 0 | 0 | 1 | ; |
| 18 | 7 | 8 | 7841.81131 | 3 | 3 | 0.15 | 4 | 0 | 0 | 1 | ; |
| | 7 | 18 | 23403.47319 | 2 | 2 | 0.15 | 4 | 0 | 0 | 1 | ; |
| 20 | 8 | 6 | 4898.587646 | 2 | 2 | 0.15 | 4 | 0 | 0 | 1 | ; |
| | 8 | 7 | 7841.81131 | 3 | 3 | 0.15 | 4 | 0 | 0 | 1 | ; |
| 22 | 8 | 9 | 5050.193156 | 10 | 10 | 0.15 | 4 | 0 | 0 | 1 | ; |
| | 8 | 16 | 5045.822583 | 5 | 5 | 0.15 | 4 | 0 | 0 | 1 | ; |
| 24 | 9 | 5 | 10000 | 5 | 5 | 0.15 | 4 | 0 | 0 | 1 | ; |
| | 9 | 8 | 5050.193156 | 10 | 10 | 0.15 | 4 | 0 | 0 | 1 | ; |
| 26 | 9 | 10 | 13915.78842 | 3 | 3 | 0.15 | 4 | 0 | 0 | 1 | ; |
| | 10 | 9 | 13915.78842 | 3 | 3 | 0.15 | 4 | 0 | 0 | 1 | ; |
| 28 | 10 | 11 | 10000 | 5 | 5 | 0.15 | 4 | 0 | 0 | 1 | ; |
| | 10 | 15 | 13512.00155 | 6 | 6 | 0.15 | 4 | 0 | 0 | 1 | ; |
| 30 | 10 | 16 | 4854.917717 | 4 | 4 | 0.15 | 4 | 0 | 0 | 1 | ; |

```

10      17      4993.510694      8      8      0.15      4      0      0      1      ;
32      11      4      4908.82673      6      6      0.15      4      0      0      1      ;
      11      10      10000      5      5      0.15      4      0      0      1      ;
34      11      12      4908.82673      6      6      0.15      4      0      0      1      ;
      11      14      4876.508287      4      4      0.15      4      0      0      1      ;
36      12      3      23403.47319      4      4      0.15      4      0      0      1      ;
      12      11      4908.82673      6      6      0.15      4      0      0      1      ;
38      12      13      25900.20064      3      3      0.15      4      0      0      1      ;
      13      12      25900.20064      3      3      0.15      4      0      0      1      ;
40      13      24      5091.256152      4      4      0.15      4      0      0      1      ;
      14      11      4876.508287      4      4      0.15      4      0      0      1      ;
42      14      15      5127.526119      5      5      0.15      4      0      0      1      ;
      14      23      4924.790605      4      4      0.15      4      0      0      1      ;
44      15      10      13512.00155      6      6      0.15      4      0      0      1      ;
      15      14      5127.526119      5      5      0.15      4      0      0      1      ;
46      15      19      14564.75315      3      3      0.15      4      0      0      1      ;
      15      22      9599.180565      3      3      0.15      4      0      0      1      ;
48      16      8      5045.822583      5      5      0.15      4      0      0      1      ;
      16      10      4854.917717      4      4      0.15      4      0      0      1      ;
50      16      17      5229.910063      2      2      0.15      4      0      0      1      ;
      16      18      19679.89671      3      3      0.15      4      0      0      1      ;
52      17      10      4993.510694      8      8      0.15      4      0      0      1      ;
      17      16      5229.910063      2      2      0.15      4      0      0      1      ;
54      17      19      4823.950831      2      2      0.15      4      0      0      1      ;
      18      7      23403.47319      2      2      0.15      4      0      0      1      ;
56      18      16      19679.89671      3      3      0.15      4      0      0      1      ;
      18      20      23403.47319      4      4      0.15      4      0      0      1      ;
58      19      15      14564.75315      3      3      0.15      4      0      0      1      ;
      19      17      4823.950831      2      2      0.15      4      0      0      1      ;
60      19      20      5002.607563      4      4      0.15      4      0      0      1      ;
      20      18      23403.47319      4      4      0.15      4      0      0      1      ;
62      20      19      5002.607563      4      4      0.15      4      0      0      1      ;
      20      21      5059.91234      6      6      0.15      4      0      0      1      ;
64      20      22      5075.697193      5      5      0.15      4      0      0      1      ;
      21      20      5059.91234      6      6      0.15      4      0      0      1      ;
66      21      22      5229.910063      2      2      0.15      4      0      0      1      ;
      21      24      4885.357564      3      3      0.15      4      0      0      1      ;
68      22      15      9599.180565      3      3      0.15      4      0      0      1      ;
      22      20      5075.697193      5      5      0.15      4      0      0      1      ;
70      22      21      5229.910063      2      2      0.15      4      0      0      1      ;
      22      23      5000      4      4      0.15      4      0      0      1      ;
72      23      14      4924.790605      4      4      0.15      4      0      0      1      ;
      23      22      5000      4      4      0.15      4      0      0      1      ;
74      23      24      5078.508436      2      2      0.15      4      0      0      1      ;
      24      13      5091.256152      4      4      0.15      4      0      0      1      ;
76      24      21      4885.357564      3      3      0.15      4      0      0      1      ;
      24      23      5078.508436      2      2      0.15      4      0      0      1
78 ]

80 n=24;
   A=zeros(n,n);
82 U=zeros(n,n);
   c0=zeros(n,n);
84 for k=1:73
   A(DATI(k,1),DATI(k,2))=1;
86   U(DATI(k,1),DATI(k,2))=round(DATI(k,3));
   c0(DATI(k,1),DATI(k,2))=DATI(k,5);
88 end

```

Gli ultimi 3 archi, dal settantaquattresimo al settantaseiesimo, sono gli archi uscenti dal punto di raccolta; perciò, sono stati esclusi dai calcoli, dato che, nello scenario reale, non troverebbero alcuna utilità (una volta arrivati al nodo, gli utenti vengono di fatto *assorbiti*).

B.3. Parametri d'intervento - Imposizione Totale

| Parametro | Valore |
|-------------------|---------------|
| budget inversioni | 100 |
| budget divergenze | 100 |
| costo inversioni | 20 |
| costo divergenze | 20 |

Tabella B.2: Parametri d'intervento grafo con 6 nodi.

| Parametro | Valore |
|-------------------|---------------|
| budget inversioni | 200 |
| budget divergenze | 200 |
| costo inversioni | 20 |
| costo divergenze | 20 |

Tabella B.3: Parametri d'intervento grafo con 14 nodi.

| Parametro | Valore |
|-------------------|---------------|
| budget inversioni | 12000 |
| budget divergenze | 12000 |
| costo inversioni | 1000 |
| costo divergenze | 1000 |

Tabella B.4: Parametri d'intervento grafo reale.

Elenco delle figure

| | | |
|------|---|----|
| 1.1 | Modello comportamentale in caso di evacuazione di massa. Tratto da [2]. | 4 |
| 1.2 | Ciclo di calcolo delle variabili d'arco e di percorso in un contesto macroscopico. | 8 |
| 2.1 | Procedura di calcolo per l'evacuazione dell'edificio in [40]. | 17 |
| 2.2 | Approccio modellistico all'evacuazione dell'edificio in [42]. | 18 |
| 2.3 | Procedura di calcolo per l'evacuazione dell'edificio in [42]. | 19 |
| 2.4 | Procedura ricorsiva per la gestione ottima di un'evacuazione di massa. Tratto da [29]. | 23 |
| 3.1 | Esempi di nodi in un albero d'evacuazione. Tratto da [1] | 27 |
| 4.1 | Classico grafo a diamante utilizzato a fini didattici. | 39 |
| 4.2 | Matrice di incidenza archi-percorsi per il grafo a diamante. Tratto da [19]. | 39 |
| 4.3 | Ciclo di calcolo delle variabili d'arco e di percorso. | 45 |
| 4.4 | Rappresentazione generica dei modelli di assegnazione all'equilibrio. Tratto da [22]. | 46 |
| 4.5 | Parte 1 della procedura: confronto di tempi e vittime. | 48 |
| 4.6 | Parte 2 della procedura: calcolo delle nuove matrici di incidenza e delle capacità. | 48 |
| 5.1 | Grafo di prova con 6 nodi. | 51 |
| 5.2 | Risultati Semplice Informazione per il grafo con 6 nodi. | 53 |
| 5.3 | Variabili per l'alterazione del grafo con 6 nodi a seguito dell'algoritmo PLIM. | 54 |
| 5.4 | Alterazione del grafo con 6 nodi a seguito dell'algoritmo PLIM. | 54 |
| 5.5 | Flussi d'arco massimi nel grafo con 6 nodi - approccio di Imposizione Totale. | 55 |
| 5.6 | Flussi d'arco massimi e tempo d'evacuazione nel grafo con 6 nodi - approp- cio di Prescrizione Parziale. | 56 |
| 5.7 | Grafo di prova con 14 nodi. | 57 |
| 5.8 | Alterazione del grafo con 14 nodi a seguito dell'algoritmo PLIM per $\tau = 200$ | 57 |
| 5.9 | Alterazione del grafo con 14 nodi a seguito dell'algoritmo PLIM per $\tau = 250$ | 58 |
| 5.10 | Mappa di Sioux Falls. Tratta da Github | 59 |

| | |
|--|----|
| 5.11 Grafo stradale di Sioux Falls. In verde, i centroidi. Tratto da Github . . . | 60 |
| 5.12 Rielaborazione Matlab del grafo stradale di Sioux Falls. | 61 |
| 5.13 Grafo stradale di Sioux Falls a seguito del'algoritmo PLIM. | 63 |

Elenco delle tabelle

| | | |
|-----|--|----|
| 4.1 | Esempi di calcolo di z_{kmi}^t per un nodo con 3 successori. | 42 |
| 5.1 | Parametri grafo di prova con 6 nodi. | 52 |
| 5.2 | Parametri grafo di prova con 14 nodi. | 56 |
| 5.3 | Parametri grafo di Sioux Falls. | 62 |
| 5.4 | Dimensione computazionale variabili Imposizione Totale. | 64 |
| B.1 | Parametri comuni modello di assegnazione. | 85 |
| B.2 | Parametri d'intervento grafo con 6 nodi. | 87 |
| B.3 | Parametri d'intervento grafo con 14 nodi. | 87 |
| B.4 | Parametri d'intervento grafo reale. | 87 |

Ringraziamenti

Chi mi conosce, sa che due appellativi che ben mi si addicono sono *prolisso* e *riconoscete*; perciò, non si stupirà nessuno se la lunghezza di questa sezione dovesse risultare paragonabile a *Guerra e Pace*.

In primis, devo ringraziare i due nomi in copertina. Il Prof. Bruglieri mi ha fornito un supporto tecnico indispensabile, ma soprattutto mi ha dissuaso dall'intraprendere alcune strade scomode che avrebbero reso ancor più lunga la gestazione di questa tesi. Allo stesso modo, ringrazio il Prof. Maja, anch'egli uomo prolisso, ma dal sapere sconfinato (enciclopedico!), nonché estremamente disponibile e professionale. Lei rimarrà sempre nei miei pensieri come miglior professore mai avuto al Politecnico.

Rimanendo in ambito universitario, ringrazio tutti i miei compagni di corso con cui ho condiviso periodi intensi e belle soddisfazioni. Un abbraccio particolare va al *Gruppo FLAG*, trio indispensabile per conseguire questo titolo, a Lorenzo e Isabella, alle prese con l'ATAC insieme a me, e a Lorenzo Falco, amico, concittadino per metà e collaboratore essenziale durante i lunghi mesi passati (talvolta a Via Wildt) a lavorare su quei codici intricatissimi.

Ringrazio i miei capi alla EPA Soluzioni per essermi venuti incontro negli ultimi mesi, aiutandomi a concludere questo percorso e introducendomi al successivo, e le mie colleghe, che mi aiutano con le pratiche e mi strappano sorrisi in pausa caffè.

Un grazie speciale, accorato e sincero, a Thomas, Flavio e Mattia, coinquilini eccellenti, amici affettuosi e compagni di viaggio, a cui devo davvero molti favori. Insieme a loro, ringrazio tutti, ma davvero tutti, i ragazzi e le ragazze della Casa dello Studente, con cui ho passato 5 anni davvero intensi e memorabili della mia vita che ho già avuto modo di celebrare in altre occasioni. Cito, in maniera non esaustiva, Gianfranco, Dario, Riccardo, Vittoria, Enrico, Matilde, Chiara, ma non basterebbero 20 pagine per nominare tutto il Pascolo!

Ringrazio tutti coloro che sono stati, anche per poco tempo o metaforicamente, miei mentori o insegnanti o maestri, come Claudio, Fabrizio, Franco, Maurizio, Marco, Angelo:

è anche grazie a voi se ho percorso certi sentieri o optato per certe scelte. Un maestro insegna sempre qualcosa in più di ciò che spiega.

Analogamente, ringrazio tutte le persone che ho incontrato finora nella mia vita, anche chi mi ha disprezzato o ignorato, perché ogni singolo momento della vita conta e ha un impatto su chi si diventerà.

A questo punto, non mi resta che ringraziare la mia numerosa famiglia (e assimilati)!

Grazie ai Bonucci, ai Pietrini, ai Balletti e a tutte le famiglie che mi hanno sempre stimato e voluto bene.

Grazie alla famiglia Rossi e alla famiglia Becherini, a zii, zie, nonni, nipoti, cugini e a tutti gli altri: siete sempre nei miei pensieri e so bene di essere io nei vostri. Per un aspirante *Stark* come me, nulla conta più della famiglia. Grazie per esserci sempre per me.

Grazie ai miei amici di sempre (Alessio, Luca, Matteo, Giovanni, ...), con cui ci vediamo davvero poco... ma quando succede, sembra di esserci diplomati da neanche un giorno. Un ringraziamento particolare va a Luca e a Francesco. Fratelli, compagni di avventure, filosofi e confidenti: siete tutto questo e anche di più. E a scrivere le nostre storie insieme abbiamo appena cominciato, nonostante la distanza.

Grazie a Chiara per essere una sorella maggiore fortissima e un punto di riferimento a portata di mano in quei momenti quando la famiglia manca di più.

Grazie a Ludovica Sofia, nome che porto letteralmente sulla mia pelle, per amarmi ogni giorno e per la premura costante che mi rivolgi. Io cerco sempre di proteggerti, ma devo ammettere che anche tu te la cavi bene nel guardarmi le spalle.

Grazie a mio Padre, il mio supereroe, che ogni giorno cerca di farmi capire qualcosa di più sulla vita e su come affrontarla. *AUT VIAM INVENIAM, AUT FACIAM* me lo hai insegnato tu. Grazie, Papà!

Infine, grazie a mia Madre, la mia Martha Kent. Sappiamo bene entrambi che per ringraziarti le parole non bastano, né servono. Grazie, Mamma!