



Università Politecnica delle Marche
Facoltà di Ingegneria

Corso di Laurea Magistrale in Ingegneria Gestionale

**UN MODELLO DI OTTIMIZZAZIONE MATEMATICA PER IL
PROBLEMA DI PRELIEVO E CONSEGNA CON FINESTRE
TEMPORALI DELL'AZIENDA TRE ELLE**

A MATHEMATICAL OPTIMIZATION MODEL FOR THE PICK-
UP AND DELIVERY PROBLEM WITH TIME WINDOWS OF
THE TRE ELLE COMPANY

Relatore:
Dott.ssa Ornella Pisacane

Tesi di:
Maia Isidori

Correlatore:
Stefano Luzi

A.A. 2020/2021

Sommario

Abstract.....	5
Abstract.....	6
Capitolo 1.....	7
Introduzione.....	7
1.1 L'importanza della logistica distributiva.....	7
1.2 Motivazioni	10
1.3 Obiettivi	10
1.4 Struttura della tesi	11
Capitolo 2.....	12
Stato dell'arte del Vehicle Routing Problem.....	12
2.1 Introduzione.....	12
2.2 Cenni storici sul VRP	12
2.3 Il VRP classico e le sue varianti	13
2.3.1 Il VRP classico	14
2.3.2 Classificazione dei VRP	16
2.4 Approcci risolutivi per il VRP	21
2.4.1 Algoritmi esatti	23
2.4.2 Metodi Euristici	26
2.4.3 Algoritmi metaeuristici.....	27
2.4.4. Le matheuristics.....	28
Capitolo 3.....	32
Il modello matematico	32
3.1 Introduzione.....	32
3.2 Analisi del problema reale	32
3.3 Il modello.....	33
La rete.....	33
3.4 La formulazione del modello matematico	35
La funzione obiettivo	36
I vincoli	37

Regole di pre-processing	40
Capitolo 4.....	42
La validazione e la sperimentazione	42
4.1 Introduzione.....	42
4.2 Il processo di validazione con istanze di piccole dimensioni	42
4.2.1 Istanza 1	42
4.2.2 Istanza 2	46
4.2.3 Istanza 3	47
4.2.4 Istanza 4	49
4.3 Il caso studio Tre Elle.....	53
4.3.1 Istanza “Tre Elle Ridotto”	53
4.3.2 Istanza “Tre Elle intermedio”	55
4.3.3 Istanza “Tre Elle completo”	61
Capitolo 5.....	69
Conclusioni e sviluppi futuri.....	69
Riferimenti Bibliografici	71

Abstract

La Logistica Distributiva gioca un ruolo fondamentale per rendere un'azienda competitiva sul mercato. Buona parte, infatti, dei costi relativi ad un prodotto è legata al trasporto verso clienti e/o impianti produttivi intermedi o fornitori. I costi in tal senso sono sensibili a fattori come il numero e l'ubicazione di fornitori e clienti, ma anche a parametri più complessi.

Caso di studio di questa tesi è l'azienda Tre Elle che segue i prodotti durante il loro intero ciclo di vita e che, per alcune lavorazioni, si affida ad aziende terze; per questo è costretta ad affrontare numerose attività di consegna/distribuzione. Obiettivo di questo lavoro è quello di risolvere un problema di ottimizzazione dei trasporti per la Tre Elle, minimizzando i costi complessivi. La flotta dei veicoli a disposizione è eterogenea (capacità di carico diverse) e ciascun veicolo, partendo dal deposito ed ivi rientrando, effettua una serie di consegne/prelievi di merce. In particolare, consegna un prodotto finito presso un cliente o un semilavorato presso un fornitore; preleva un prodotto finito presso un fornitore per distribuirlo ad un cliente o riportarlo al deposito. Le attività di distribuzione tengono in conto delle finestre temporali ai clienti/fornitori per la consegna/prelievo della merce. Il problema è formulato mediante un modello di Programmazione Lineare Intera Mista ed afferisce alla classe più generale di problemi di *Vehicle Routing Problem con Finestre Temporali* in cui i nodi della rete distributiva sono il deposito (sede dell'azienda), i clienti ed i fornitori a cui o da cui consegnare/prelevare semilavorati. Il modello è successivamente implementato con il linguaggio *OPL* e risolto utilizzando il solver *IBM ILOG CPLEX*. La fase di validazione è condotta su un insieme di istanze di piccole dimensioni mentre, quella di sperimentazione, sul caso di studio fornito dalla Tre Elle e su un sotto-insieme di istanze da esso ricavate.

Abstract

Distribution Logistics plays a fundamental role in making a company competitive on the market. A large part, in fact, of the costs related to a product is linked to the transport activities towards customers and/or intermediate production facilities or suppliers. The costs in such sense are sensitive to factors like the number and the location of suppliers and customers, but also to more complex parameters.

Case study of this thesis is the company Tre Elle that follows the products during their entire life cycle and that, for some processes, relies on third party companies; for this reason, it has to deal with many delivery/distribution activities.

The aim of this work is to optimize the transport activities of Tre Elle, minimizing its overall costs. The available fleet of vehicles is heterogeneous (such as, they have different cargo capacities) and each vehicle, starting from the depot and returning to it, performs several deliveries/pick-ups of goods. It delivers a finished product to a customer or a semi-finished product to a supplier; it picks a finished product up from a supplier to distribute it to a customer or to the depot. Distribution activities consider time windows at customers/suppliers for deliveries/pickups. The problem is formulated as a Mixed Integer Linear Program and belongs to the more general class of *Vehicle Routing Problems with Time Windows* in which the nodes of the distribution network are the depot, the customers, the suppliers which the semi-finished products are delivered to and the suppliers which the semi-finished products are picked up from. The model is subsequently implemented in *OPL* and solved using the *IBM ILOG CPLEX* solver. The validation phase is performed on a set of small instances. Whereas the experimental campaign is performed on the case of study of Tre Elle and on a sub-set of instances derived from it.

Capitolo 1

Introduzione

1.1 L'importanza della logistica distributiva

Dalla fine degli anni '70, nel mercato, «la concorrenza non solo è divenuta fortissima in tutti i settori ma, a causa del sempre maggiore sviluppo delle telecomunicazioni e dei trasporti ha assunto un carattere planetario» [1]. Le imprese sono chiamate ad evolversi e innovarsi per poter mantenere vivo il proprio business anche in un ambiente fortemente competitivo.

Quello che ora vuole il mercato non è solo un prodotto dalle elevate prestazioni e a costi contenuti, ma vuole la possibilità di un'ampia scelta e di un tempo di attesa minimo.

Per questo, in molti settori, le aziende devono essere più flessibili e capaci di adattarsi ad una domanda particolarmente dinamica e personalizzata. I prodotti sono sempre più variegati e le tempistiche sempre più ristrette.

La qualità e il costo del prodotto non sono più gli unici criteri di valutazione, ma entra in modo deciso in gioco la *customer satisfaction* che ha tra i criteri di valutazione l'*Actual Order Lead Time*. Quest'ultimo rappresenta il tempo che intercorre tra il momento in cui viene fatto l'ordine e il momento dell'effettiva consegna. Questo significa che avere un'elevata rapidità di risposta diventa non solo una caratteristica importante, ma essenziale alla sopravvivenza aziendale.

Bisogna tener conto che quando un cliente decide di chiedere un prodotto all'azienda, quest'ultima deve cercare di svolgere le attività di progettazione, reperimento delle materie prime, produzione e distribuzione nel minor tempo possibile. Per questo la *logistica distributiva* gioca un ruolo fondamentale in due dei criteri che determinano il successo o meno di un'azienda: i costi ed il Lead Time.

L'obiettivo di questo lavoro di tesi è aumentare l'efficienza e la competitività dei servizi di trasporto in un caso reale e, contemporaneamente, anche ridurre il costo complessivo imputabile alle attività di trasporto che tipicamente l'azienda deve sostenere. L'azienda che prenderemo in esame è un'azienda metalmeccanica in cui una percentuale significativa del costo di produzione è proprio rappresentata dai trasporti. Nel mondo delle aziende metalmeccaniche avere un tempo di risposta contenuto non è facile. Ogni fase di lavorazione è fortemente specializzata e richiede impianti specifici. Quindi nell'ottica di dover scegliere tra "make or buy", ovvero se svolgere autonomamente tutte le fasi della lavorazione oppure ricorrere ad un'azienda terza per alcune di queste, si rivela più efficiente la seconda alternativa. Questo però crea la necessità di trasportare il semilavorato da un'azienda all'altra al fine di completarne la produzione.

Il caso di studio preso in esame è quello della Tre Elle, un'azienda metalmeccanica che opera nel territorio fermano dal 1951. L'impresa in questione segue i prodotti durante il loro intero ciclo di vita, dalla progettazione alla consegna al cliente. Le lavorazioni a cui il prodotto è sottoposto possono essere interne all'azienda oppure esterne.

Questo porta alla necessità non solo di una buona collaborazione tra le imprese, ma anche di una Logistica Distributiva efficiente. È qui che si introduce il tema centrale di questo studio: risolvere un particolare problema di instradamento di veicoli, noto in letteratura come *Vehicle Routing Problem* (VRP).

Il VRP è una classe di problemi di ottimizzazione che si pone come obiettivo quello di instradare in maniera efficiente, rispetto ad un determinato criterio scelto, una flotta di veicoli per servire un insieme di clienti geograficamente distribuiti. Ogni veicolo generalmente parte da un deposito ed ivi ritorna dopo aver servito un sotto-insieme di clienti, nel rispetto dei vincoli imposti. Una tale

tratta che il veicolo percorre è indicata in letteratura come *rotta*. Pertanto, una soluzione del VRP è un insieme di rotte, una per ognuno dei veicoli usati.

La Tre Elle, grazie alla sua grande flessibilità, rifornisce settori di varia natura, tra cui quello dell'alta moda. Per questo è importante per l'azienda gestire in autonomia i trasporti al fine di garantire la segretezza del prodotto.

Siccome il trasporto è presente addirittura più volte nel ciclo di produzione diventa evidente la necessità di una Logistica Distributiva ottimizzata. L'azienda deve infatti curarsi di:

- Ritirare materie prime e componenti dai fornitori;
- Consegnare presso i terzisti i semilavorati, ritirati nel deposito dell'azienda Tre Elle, al fine di essere sottoposti ad una specifica lavorazione;
- Ritiro dei semilavorati, presso i terzisti, per poi essere riportati in azienda o consegnati presso altri terzisti per un'ulteriore lavorazione o anche direttamente al cliente finale;
- Consegna del prodotto finito che può essere ritirato dal magazzino dell'azienda o da quello di un terzista che si è occupato dell'ultima fase di lavorazione del prodotto.
- Rispettare le finestre temporali imposte sia dai clienti che dai terzisti.

A tal proposito, si è formulato un modello di programmazione matematica che rispetti i vincoli suddetti e che ottimizzi il costo complessivo del trasporto. Un tale problema afferisce alla classe di VRP con finestre temporali (VRP with Time Windows (VRP-TW)), in cui si ha un insieme di nodi di prelievo (pick-up) ed un insieme di nodi di consegna (delivery) e pertanto, il problema è stato particolarizzato come problema con prelievi e consegne e finestre temporali (Pick-up and Delivery Problem with Time Windows (PDP-TW)).

1.2 Motivazioni

La difficoltà principale dell'azienda sta nello scegliere una rotta per i propri veicoli che possa andare a sfruttare al meglio le capacità del mezzo, rispettando i vincoli imposti e minimizzando i costi. I fattori che influenzano e complicano questa scelta sono numerosi. Prima di tutto i collegamenti che l'azienda ha sono tanti e complessi. In secondo luogo, si è di fronte ad una flotta eterogenea, ovvero formata da mezzi con capacità di carico e consumi diversi. Inoltre, bisogna tener conto di altri vincoli da rispettare come limiti di carico, le finestre temporali dei magazzini delle aziende terze e altro.

Gli effetti che l'uso del modello formulato come supporto alle decisioni può avere sono molteplici e di varia natura. Il più scontato, ma spesso più importante, è la riduzione dei costi. A questo è affiancata anche una gestione più organizzata delle consegne e dei prelievi. Inoltre, garantisce una precisa panoramica sia all'azienda che all'autotrasportatore sui tempi e sul carico del mezzo, aspetti fondamentali.

Infine, bisogna tener conto che per un'azienda avere una logistica dei trasporti ottimizzata e coerente con le sue necessità specifiche, come in questo caso, le permette di garantire non solo un costo minore ma anche un servizio migliore.

1.3 Obiettivi

L'obiettivo, più nello specifico, è quello di formulare un modello matematico che permetta di minimizzare il costo complessivo legato al trasporto, nel rispetto dei vincoli che caratterizzano il funzionamento dell'azienda.

Tra i principali vincoli da rispettare troveremo:

- La capacità di carico dei mezzi;
- Le Time Windows dei magazzini coinvolti;
- Il massimo numero di ore che di lavoro che un autista può svolgere;

Formulato il modello, tramite il linguaggio *OPL*, esso è stato risolto mediante il risolutore *IBM ILOG CPLEX*.

Quindi, si è passati alla sua validazione mediante dei casi di studio molto semplici per poi terminare con la campagna sperimentale condotta sul caso di studio reale fornito dalla Tre Elle e su un insieme di istanze di complessità crescente, ricavate da questo.

1.4 Struttura della tesi

Il lavoro di tesi è organizzato come segue.

Nel secondo capitolo, dopo una breve introduzione storica, si presenta una disamina dello stato dell'arte sul VRP e le sue varianti, focalizzando l'attenzione sul PDP-TW, oggetto di studio di questo lavoro.

Il terzo capitolo descrive il modello matematico formulato per il caso di studio.

Il quarto capitolo descrive in primis la fase di validazione condotta su istanze piccole e termina con la fase di sperimentazione condotta sul caso di studio della Tre Elle e su un insieme di istanze di complessità crescente ricavate da questo.

Infine, nell'ultimo capitolo, verranno presentate delle conclusioni sul lavoro svolto e dei possibili sviluppi futuri.

Capitolo 2

Stato dell'arte del Vehicle Routing Problem

2.1 Introduzione

Il VRP è una classe di problemi di ottimizzazione fortemente studiata nella ricerca operativa. Come già detto, l'obiettivo che ci si pone è quello di instradare in maniera efficiente, rispetto ad un determinato criterio scelto, una flotta di veicoli per servire un insieme di clienti geograficamente distribuiti.

Questo problema ha un ruolo centrale nella gestione della distribuzione ed è affrontato quotidianamente da aziende di tutto il mondo. Negli ultimi 50 anni, il VRP ha attirato l'attenzione di gran parte della comunità della ricerca operativa sia per l'importanza economica del problema sia per le sfide metodologiche che pone [2].

In questo capitolo, analizzeremo la storia di questo problema e come si è evoluto per adattarsi alle esigenze reali delle aziende. Una volta introdotte alcune delle varianti principali del VRP, faremo una panoramica sugli algoritmi risolutivi utilizzati nella letteratura.

2.2 Cenni storici sul VRP

Il Vehicle Routing Problem (VRP) ha origine da un problema più semplice: il Travelling Salesman Problem (TSP), ovvero il problema del commesso viaggiatore, il cui obiettivo è quello di trovare il ciclo hamiltoniano a costo minimo, ovvero il ciclo che deve percorrere il commesso che, partendo da un deposito ed ivi facendo ritorno, serve tutti i clienti una ed una sola volta, a costo minimo (tipicamente a distanza totale minima).

La prima formulazione matematica del TSP venne proposta nel 1954 da Dantzig, Fulkerson e Johnson in [3] in cui gli autori si prefissarono l'obiettivo di trovare una rotta ottimale che collegasse 49 città statunitensi (Fig. 2.1).

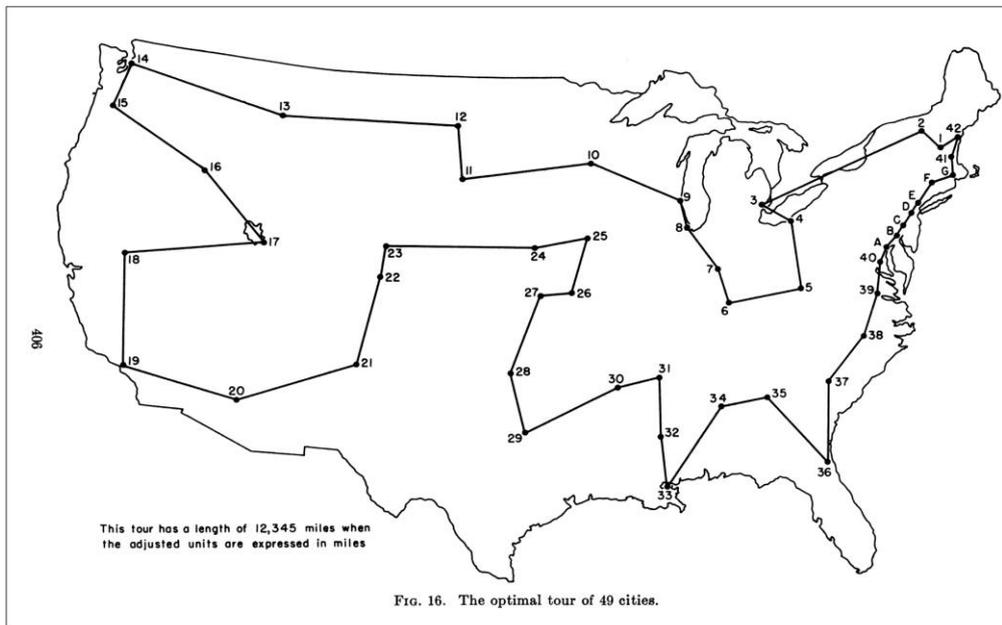


Fig. 2.1. Immagine tratta dall'articolo [3] di Dantzig, Fulkerson e Johnson.

Questo lavoro dette le basi a Dantzig e Ramser, nel 1959, per introdurre il "Truck Dispatching Problem" [16] e formulare un modello che permettesse ad una flotta omogenea di veicoli di soddisfare la domanda di un certo numero di stazioni di rifornimento percorrendo la minima distanza. In particolare, lo studio faceva riferimento al tragitto delle petroliere tra la raffineria di Atlanta e le sue stazioni di servizio subordinate. Quindi si hanno un certo numero di clienti con una diversa domanda di merci. L'obiettivo è soddisfare le esigenze dei clienti e, al contempo, minimizzare la distanza, i costi ed il consumo di tempo.

Cinque anni dopo, Clarke e Wright generalizzarono questo problema a quello denominato Vehicle Routing Problem che diventò uno dei più importanti e studiati nella Ricerca Operativa [4].

Da allora il VRP ha subito e subisce ancora oggi numerose evoluzioni per adattarsi al meglio alle nuove necessità.

2.3 Il VRP classico e le sue varianti

Per arrivare a comprendere al meglio il modello formulato in questo lavoro conviene analizzare prima le varianti più semplici del VRP.

2.3.1 Il VRP classico

In generale, il VRP ha come obiettivo quello di restituire una serie di rotte, una per ogni veicolo utilizzato, che minimizzino la funzione di costo. I veicoli partono da un punto iniziale, rappresentato dal deposito, e vi tornano dopo aver visitato un sotto-insieme di clienti geograficamente sparsi, come nell'esempio mostrato in Fig. 2.2. La tratta che ogni veicolo percorre è appunto denominata rotta.

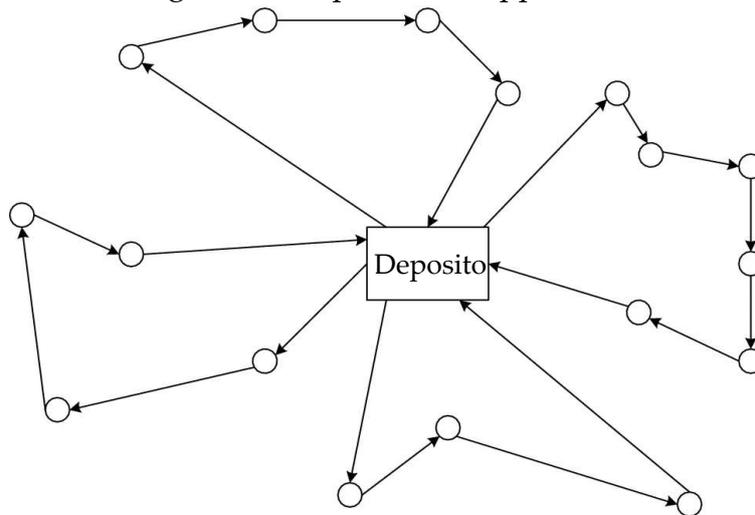


Fig. 2.2. Esempio di un insieme di rotte. Immagine tratta dallo studio [5]

I requisiti principali per determinare una rotta ammissibile, nel VRP, sono i seguenti [5]:

- ogni cliente deve essere visitato una sola volta da un veicolo;
- tutti i veicoli partono dal deposito iniziale e tornano allo stesso deposito dopo aver completato l'attività di distribuzione;
- generalmente, si devono rispettare condizioni sulla capacità di carico dei veicoli e temporali.

Il VRP è caratterizzato dalla combinazione dei seguenti fattori [5]:

1. **Clients:** rappresentano i punti da raggiungere.
2. **Deposito:** punto di partenza e di arrivo di ogni rotta.
3. **La rete stradale:** la rete che collega i clienti tra di loro e con il deposito è generalmente schematizzata tramite un grafo, ovvero un formalismo che

definisce una relazione binaria su una collezione di oggetti. Facendo riferimento ai problemi di Node Routing, in un grafo, i clienti e il deposito sono nodi, mentre gli archi rappresentano collegamenti stradali tra coppie di nodi. Generalmente, ogni arco è pesato con uno o più valori rappresentativi, ad esempio, della distanza, tempo o costo di percorrenza.

I grafi possono essere classificati a seconda che siano (Fig 2.3):

- Orientati o diretti: si definiscono tali se tutti gli archi sono orientati, ovvero se hanno una relazione non simmetrica per cui dati due nodi u e v sia ha che $(u, v) \neq (v, u)$.
- Non orientati: Si definiscono tali se gli archi possono essere percorsi in entrambi i versi e $(u, v) = (v, u)$.
- Misti: Si definiscono misti i grafi in cui troviamo sia archi direzionali che non direzionali.

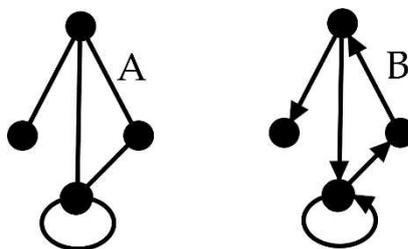


Fig. 2.3. Nell'esempio A possiamo vedere un grafo non orientato. Nell'esempio B possiamo vedere un grafo orientato

Inoltre, è possibile effettuare un'ulteriore classificazione in grafi completi e non completi. Nel primo caso, a differenza del secondo, tutti i nodi del grafo sono tra loro collegati. Nel caso in cui un arco di collegamento tra due nodi non dovesse essere presente è possibile supporre la presenza di un arco a cui viene associato un costo infinito.

4. **Veicolo:** mezzo per svolgere la distribuzione.
5. **Requisiti per l'organizzazione del trasporto:** la rotta dipende dalla natura delle merci trasportate, dal livello di qualità del servizio desiderato e dalle

caratteristiche dei clienti e dei veicoli. Tutte queste caratteristiche si tradurranno nel modello sotto forma di vincoli.

6. **L'obiettivo:** rappresenta il motivo stesso per cui il modello viene formulato.

Esistono diversi obiettivi, ad esempio:

- a. minimizzare il costo totale del trasporto;
- b. minimizzare la distanza percorsa;
- c. minimizzare il numero di veicoli necessari per completare la distribuzione;
- d. minimizzare il tempo in cui il veicolo non contiene merci;
- e. minimizzare le penali dovute al mancato servizio dei clienti.

2.3.2 Classificazione dei VRP

Partendo dal problema di base del VRP sono state elaborate diverse varianti al fine di avvicinarsi il più possibile ai casi reali. Queste varianti differiscono tra loro per funzione obiettivo, vincoli e variabili decisionali.

Analizziamo alcune delle principali varianti:

VRP con vincoli di capacità, CVRP:

Questa variante è molto utilizzata ed è caratterizzata dall'imposizione di un vincolo di carico per il veicolo. A causa dell'aggiunta di questo vincolo il processo di ottimizzazione deve considerare in modo completo la relazione tra il numero di veicoli, la distribuzione del carico e la pianificazione delle rotte per ottenere la distanza di viaggio totale più breve [5] (Fig. 2.4).

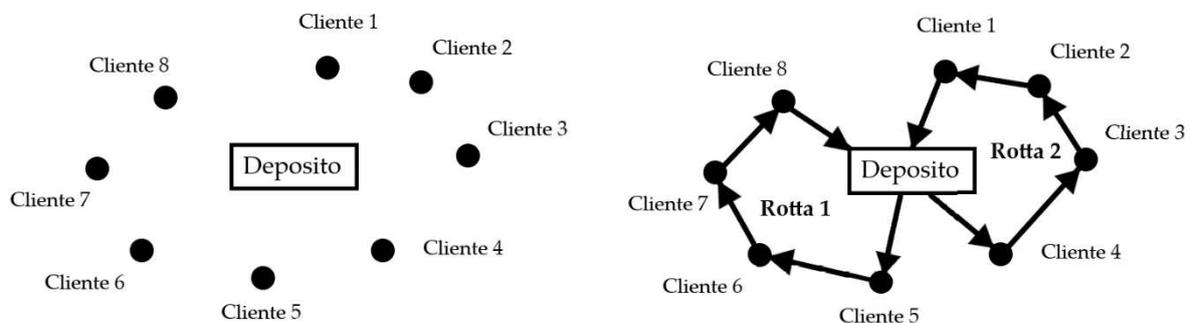


Fig. 2.4 . La limitazione di carico potrebbe rendere impossibile servire tutti i clienti con un'unica tratta.

VRP con finestre temporali, VRP-TW:

In questa tipologia ciascun cliente deve essere visitato in una determinata fascia oraria denominata *time window* (TW). Ogni cliente i è quindi caratterizzato da una coppia di valori temporali $[e_i, l_i]$ che identificano l'intervallo di tempo in cui può essere servito. Si presume che le finestre temporali siano conosciute a priori e che i veicoli partano dal deposito all'istante di tempo $t_0 = 0$.

Le TW, però, sono rappresentate da vincoli diversi a seconda che ci troviamo di fronte a finestre di tipo *hard* o *soft*. Il primo caso prevede che il veicolo possa servire i clienti solamente durante la finestra temporale e che quindi debba aspettare nel caso arrivi in anticipo e venga rifiutato nel caso sia in ritardo. L'altra possibilità prevede che il veicolo possa violare la TW, ma che per fare ciò debba pagare una penale.

Questa tipologia di problema ha a sua volta delle varianti, tra le più rilevanti troviamo:

- Il VRPTW con depositi multipli, MDVSPTW [13]: in questo caso, oltre alla presenza della TW, abbiamo più depositi, ognuno caratterizzato da una flotta che serve i clienti. Questa tipologia di problema di ottimizzazione trova vasto impiego nel campo del trasporto merci o nella pianificazione degli autobus urbani. Hanno dato un importante contributo al MDVSPTW gli studi condotti da Mingozzi et al. e Desaulniers et al. [15]
- Il VRPTW a scomparti multipli, MCVRPTW [14]: in questa variante i veicoli utilizzati sono caratterizzati da scompartimenti interni. Questa variante è importante in quanto sono presenti diverse situazioni in cui è necessario trasportare merce che deve rimanere separata durante il viaggio. Ne è un esempio pratico il trasporto di benzina, tramite camion-cisterne, presso le stazioni di servizio. Questi veicoli sono infatti

caratterizzati da scomparti necessari per separare le varie tipologie di benzina.

VRP con consegna divisa, SDVRP

Finora la maggior parte delle ricerche sul VRP ha posto una condizione a priori, ovvero, che la richiesta di ciascun cliente debba essere completata da un veicolo e con un unico servizio. Dror e Trudeau [6] hanno proposto per la prima volta il VRP a consegna divisa. Successivamente, confrontando la soluzione ottenuta mediante la suddivisione della domanda con quella ottenuta col metodo classico, hanno dimostrato che in alcuni casi il nuovo metodo permette una soluzione migliore.

VRP dinamico, DVRP

Attualmente la maggior parte dei VRP presuppone che fattori come la domanda dei clienti e il tempo di viaggio del veicolo siano noti e statici. In altre parole, la rotta che viene decisa non cambia fino al completamento dello stesso. Tuttavia, potrebbero esserci molti fattori dinamici che possono influenzare la consegna. Per questo si è sviluppato un sistema di trasporto intelligente grazie al quale le consegne possono essere più dinamiche. Per questi cambiamenti dinamici la teoria e il metodo VRP statici non sono più applicabili. Grazie al DVRP, in base alle informazioni note, il sistema pianifica inizialmente la rotta del veicolo ricevendo informazioni sulla domanda del cliente e sulle condizioni del traffico in tempo reale, progettando la migliore rotta possibile in modo da raggiungere l'obiettivo richiesto. In seguito, in base alla strategia di aggiornamento stabilita, il sistema elabora in tempo reale le informazioni che riceve e ricalcola la rotta (Fig 2.5).

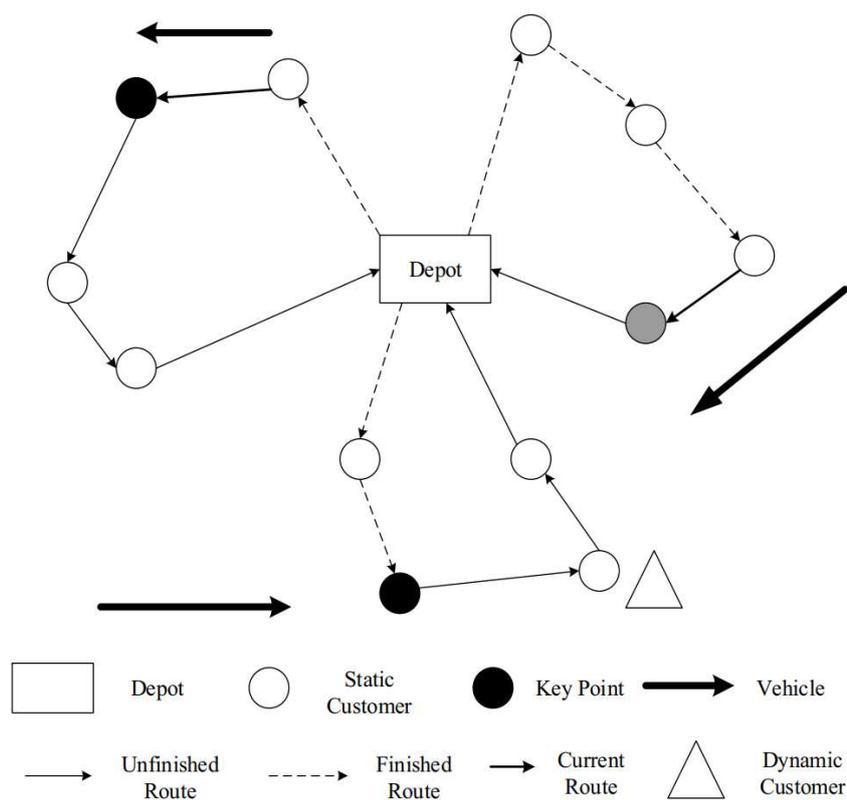


Fig. 2.5. Esempio di DVRP tratto dallo studio [5]

VRP con domanda stocastica, VRPSD.

Al fine di organizzare percorsi di guida adeguati ad una serie di clienti con domanda stocastica, vennero effettuate delle modifiche al modello formulato da Clarck e Wright. La domanda q_i di ciascun cliente può essere rappresentata tramite una distribuzione caratterizzata da un determinato valore atteso e una varianza [7]. Dati questi valori e una rotta r , si va poi ad analizzare la probabilità P che la domanda cumulativa fino al cliente i -esimo sia inferiore o uguale ad un multiplo della capacità del veicolo.

VRP con pickup e delivery, VRPPD

Mentre la versione base del VRP è un puro problema di consegna (pick-up) il VRPPD prevede la possibilità che i clienti restituiscano alcuni prodotti. Poiché è possibile effettuare operazioni diverse, vengono impiegati due parametri distinti:

d_i che rappresenta la domanda del singolo cliente i -esimo e p_i che indica invece la quantità di merce da ritirare presso lo stesso cliente [8].

VRP con flotta “green”, GVRP

In questa tipologia di VRP, l'obiettivo è quello di instradare efficientemente una flotta di veicoli a carburante alternativo (esempio, quelli elettrici). Un tale problema pone aspetti e condizioni aggiuntive da considerare (come l'autonomia tipicamente molto limitata per cui un veicolo deve fermarsi spesso durante la propria tratta per rifornimento).

Multi-trip VRP, MTVRP

Mentre nel VRP classico ogni mezzo aveva la possibilità di effettuare un solo viaggio, in questo caso è ammessa la possibilità del *Multi-Trip*.

VRP con Backhauls, VRPB

Il VRP con Backhauls, ovvero con ritorno, è noto anche come problema linehaul-backhaul ed è un'estensione del CVRP in cui il set di clienti è diviso in due sottoinsiemi. Il primo sottoinsieme contiene i clienti denominati “clienti di linea”, ciascuno dei quali richiede una determinata quantità di prodotto da consegnare. Il secondo sottoinsieme contiene invece i clienti di Backhauls, dove deve essere prelevata una determinata quantità di prodotto in entrata.

Questa partizione dei clienti è estremamente frequente nelle situazioni pratiche. Un esempio comune è quello dell'industria alimentare, dove i supermercati e i negozi sono i clienti di linea, mentre i fornitori di generi alimentari sono i clienti di ritorno. Quindi il VRPB ha l'obiettivo di determinare un insieme di percorsi veicolari che visitano tutti i clienti, in cui:

- ciascun veicolo effettua un solo percorso;
- ogni rotta inizia e finisce al deposito;
- per ciascuna rotta il carico totale associato ai clienti linehaul e backhaul non superi la capacità del veicolo;

- su ciascuna rotta gli eventuali clienti di backhaul vengono visitati dopo tutti i clienti di linehaul;
- la distanza totale percorsa dai veicoli è ridotta al minimo.

Come già detto, le varianti del VRP sono molte e hanno suscitato un forte interesse nella comunità scientifica. Grazie allo studio di Konstantakopoulos, Gayialis e Kechagias [17] è possibile notare quanto ogni variante del VRP è stata analizzata negli ultimi dieci anni (Tabella 2.1).

Varianti VRP	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	Total	Percentuale
CVRP	18	11	12	15	19	34	24	24	18	31	12	218	82.89
VRPTW	7	5	7	7	10	16	12	12	13	22	11	122	46.39
VRPPD	3	1	3	6	4	10	6	5	3	8	4	53	20.15
HFVRP	3	2	1	3	7	4	6	2	4	11	2	45	17.11
MDVRP	1	1	2	3	4	5	3	1	3	3	3	29	11.03
GVRP				1	3	2	7	6	1	8	2	30	11.41
DVRP	1	1	2		3	4	2	1	1	5	1	21	7.98
3D-VRP				2	2	4	3		3	1	2	17	6.46
OVRP	4	1		1	1	1	1		1	1	1	12	4.56
SDVRP		1			2	1	2	1		3		10	3.80
TDVRP		1	1	1			2	4		1		10	3.80
MEVRP				2		1	1		1	4		9	3.42
MTVRP	1	1				1	1	1		3	1	9	3.42
PVRP		2								2	1	5	1.90
ConVRP			1			1			1	1		4	1.52
TTRP	2	1							1			4	1.52

Tabella 2.1: Numero di articoli che ogni anno sono stati redatti aventi come tema una delle varianti del VRP [17].

2.4 Approcci risolutivi per il VRP

Il VRP appartiene ad una classe di problemi di ottimizzazione combinatoria detti *NP-hard*. La caratteristica di questi problemi è che il tempo di risoluzione aumenta esponenzialmente all'aumentare della dimensione dell'istanza. Secondo la teoria computazionale i problemi NP-hard non ammettono algoritmi polinomiali e gli algoritmi in grado di calcolare una soluzione ottima eseguono, nel caso peggiore, un'enumerazione totale e per questo sono algoritmi

esponenziali. Quindi, nell'affrontare un problema NP-hard, ci si trova inevitabilmente di fronte ad un bivio:

- rinunciare alla garanzia di ottimalità a favore dell'efficienza;
- dare priorità all'ottimalità della soluzione accettando tempi computazionali anche molto lunghi.

In questo trade-off tra efficacia ed efficienza entrano in gioco tre tipi di algoritmi risolutivi per approcciarsi al problema: esatti, euristici e metaeuristici. Grazie al lavoro svolto da Laport e Semet (Fig. 2.6) si ha una classificazione di questi metodi facendo un focus su quelli utilizzati nella letteratura sul VRP.

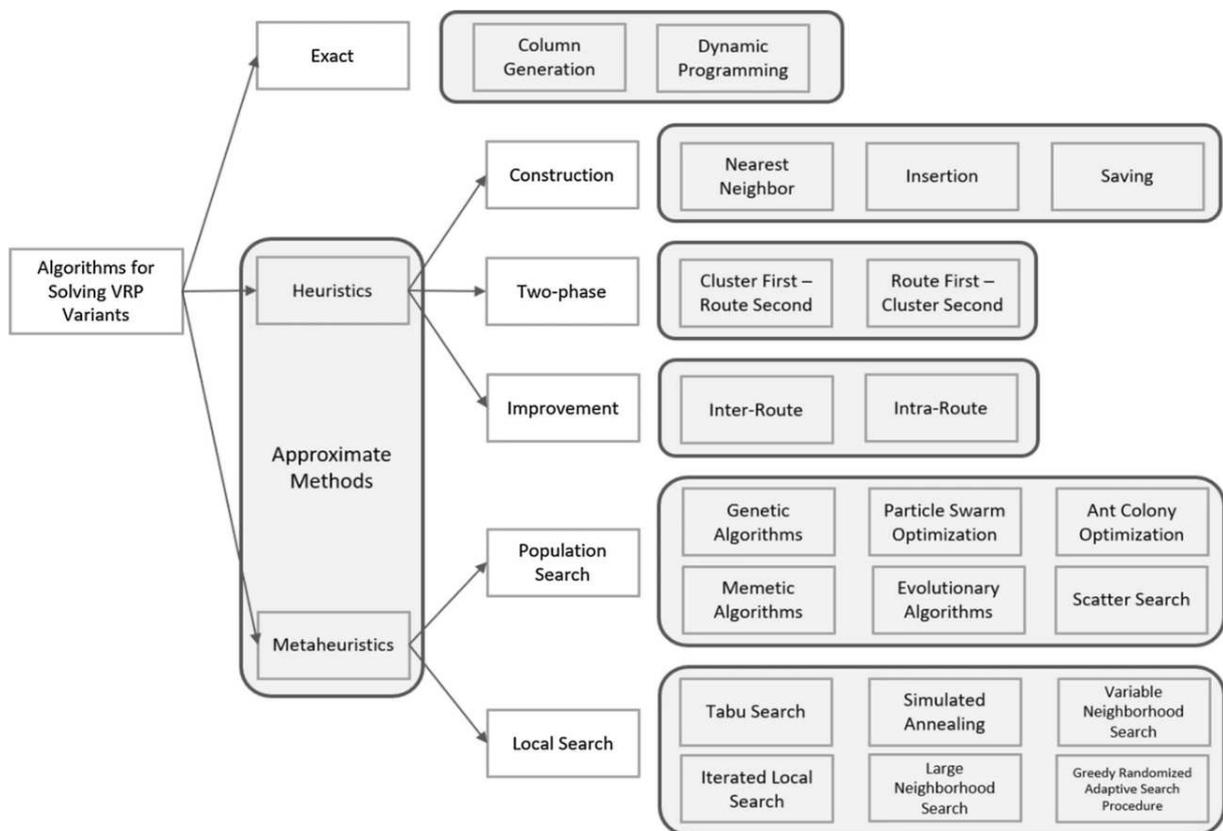


Fig. 2.6. Classificazione degli algoritmi risolutivi proposta da Laport e Semet.

La principale differenza tra questi metodi sta nel livello di profondità che raggiungono nell'esplorazione dello spazio delle soluzioni. Mentre grazie ai metodi euristici classici si ottengono buone soluzioni con tempi limitati,

attraverso i metaeuristici si approfondisce la ricerca della soluzione ottima nelle zone più promettenti dello spazio delle soluzioni.

2.4.1 Algoritmi esatti

Gli approcci esatti che vengono proposti per il VRP consentono di giungere ad una o più soluzioni ottime in un tempo ragionevole. Tuttavia, quando la dimensione delle istanze da elaborare inizia ad essere elevata i tempi di calcolo diventano lunghi e in alcuni casi non convenienti per l'impresa. Le più grandi istanze VRP che possono essere risolte, mediante metodi esatti, arrivano ad una dimensione di 50 clienti, ma sopra a questo valore conviene utilizzare metodi alternativi.

Le tre categorie di modelli di programmazione lineare proposti in letteratura sono:

- formulazioni di tipo *vehicle flow*;
- formulazioni di tipo *commodity flow*;
- formulazioni di tipo *set-partitioning*.

Le prime fanno uso di variabili intere che vengono associate agli archi del grafo e che esprimono il numero di volte in cui un singolo lato viene attraversato da un veicolo. La formulazione commodity flow è caratterizzata dalla presenza di variabili continue, associate agli archi del grafo, che rappresentano l'ammontare di merce trasportata dai veicoli che percorrono gli archi stessi.

La formulazione del set-partitioning viene proposta nel 1964 nello studio [20], dove, per rappresentare un percorso ammissibile viene utilizzata una variabile binaria. Ballinski e Quandt fecero la seguente formulazione [19]:

Dato l'insieme R delle rotte ammissibili in cui r rappresenta una rotta specifica; a_{ir} un coefficiente binario uguale a 1 se e solo se il vertice di i (appartenente all'insieme dei nodi tranne il deposito) appartiene al percorso r ; c_r^* il costo ottimo

del percorso r e y_k una variabile binaria uguale ad 1 se e solo se il percorso r viene utilizzato nella soluzione ottima; il problema si può indicare con l'equazione 2.1.

$$\min \sum c_r^* y_r, \quad r \in R \tag{2.1}$$

Con i vincoli indicati nelle equazioni 2.2 e 2.3:

$$\sum a_{ir} y_r = 1 \quad i \in v \setminus \{0\}, \quad r \in R \tag{2.2}$$

$$y_r = 0,1 \text{ dove } r \in R$$

Errore. Per applicare 0 al testo da visualizzare in questo punto, utilizzare la scheda Home..3)

Tra gli algoritmi esatti utilizzati per risolvere i problemi di VRP, in particolare il CVRP, troviamo il Branch-and-Bound, il Branch-and-Cut e il Branch-and-Price [19]. Per andare a capire su cosa si basano questi algoritmi andiamo ad analizzarne uno.

Il Branch-and-Bound

Con questa metodologia si va a fare una scomposizione ricorsiva dello spazio delle soluzioni andando ad escludere a priori le soluzioni non interessanti. Per questo algoritmo è importante introdurre il concetto di *bound* (limite). Dato il valore ottimo z^* di un problema di ottimizzazione viene detto *lower bound* qualunque valore $z_L \in R$ tale che $z_L \leq z^*$ e *upper bound* qualunque valore $z_U \in R$ tale che $z_L \geq z^*$. In un problema di minimizzazione come lo è il VRP ogni soluzione ammissibile fornisce un *upper bound* che viene chiamato *bound primale*, mentre il *lower bound* viene detto *bound duale*.

Il Branch-and-Bound ha due fasi fondamentali: il branching e il bounding.

Nel branching si scompone (branch) il problema originario P in problemi P^1, \dots, P^k più piccoli e più facili da risolvere. Nel bounding, invece, si eliminano i sotto-problemi non interessanti (Fig. 2.7).

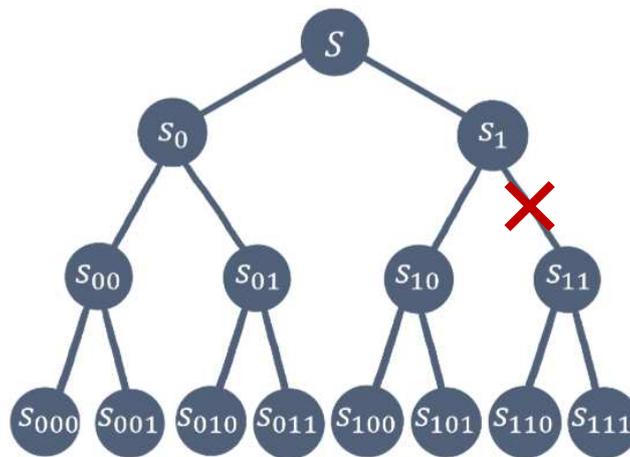


Fig. 2.7: Il bounding permette di escludere alcune soluzioni senza andare obbligatoriamente ad analizzare ogni foglia dell'albero.

Infatti, questo algoritmo permette di non esplorare lo spazio di ricerca in modo totale andando a fare un'enumerazione implicita. I bound primali e duali permettono di capire se conviene, o no, esplorare un determinato ramo. Nella fase di bounding la potatura può essere effettuata per ottimalità, per bound o per inammissibilità.

Tra l'upper bound e il lower bound esiste un *gap* ($Gap = z_U - z_L$) all'interno del quale si trova la soluzione ottima. Col passare del tempo, ovvero con l'esecuzione dell'algoritmo, questo gap tende a rimpicciolirsi (Fig. 2.8) perché si trovano dei lower bound e upper bound migliori. Generalmente l'algoritmo non termina quando $z_L = z^*$, ma quando il gap di ottimalità è 0. Per questo poi l'algoritmo deve certificare l'ottimalità della soluzione.

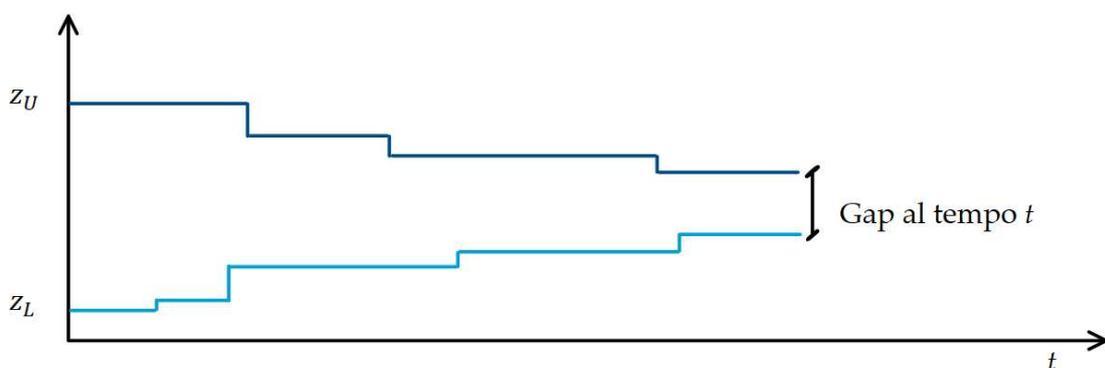


Fig.2.8: Riduzione del Gap nel tempo

2.4.2 Metodi Euristici

Gli algoritmi euristici mirano a fornire una soluzione di buona qualità in un tempo limitato di calcolo. A grandi linee è possibile definire una classificazione degli euristici classici proposti per il VRP [10][11]:

- Metodi euristici costruttivi: operano costruendo gradualmente una soluzione ammissibile, partendo da una che non lo è, cercando di contenere il costo totale della soluzione stessa.
- Metodi euristici a due fasi: scompongono il problema in due. Una fase è caratterizzata dalla divisione dei vertici in gruppi (cluster) e una dalla costruzione delle rotte ammissibili. A loro volta questi metodi si suddividono in due classi:
 - cluster-first & route-second in cui prima vengono raggruppati i vertici in cluster e poi viene calcolato un route per ogni cluster. Ciò si ottiene risolvendo un problema di assegnazione generalizzata (GAP) [19]. Viene quindi utilizzato un percorso su ogni cluster andando a risolvere un problema VRP, ma è difficile effettuare confronti esatti con altri algoritmi a causa del diverso arrotondamento della distanza.
 - route-first & cluster-second in cui viene calcolato un route su tutti i vertici per poi essere suddivisa.
- Metodi euristici migliorativi: a partire da una soluzione ammissibile si cerca di migliorarla analizzandone l'intorno. Questi algoritmi sono di tipo iterativo, ovvero, ad ogni mossa migliorano la loro soluzione. Gli algoritmi di ricerca locale si basano sul concetto di analisi dell'intorno della soluzione corrente e forniscono minimi locali del problema considerato. Una ricerca locale è così strutturata:
 1. Inizializzazione: Si sceglie una soluzione iniziale S_0 da cui partire;
 2. Definizione dell'intorno: Si definisce un intorno $I(S_0)$ della soluzione corrente costituito da soluzioni vicine ad S_0 ;

3. Selezione di una nuova soluzione: Si ispeziona l'intorno di S_0 e si calcola la migliore soluzione S_1 appartenente a questo intorno;
4. Criterio di arresto: se la soluzione data da S_1 è minore di quella data da S_0 (nel caso ci trovassimo in un problema di minimizzazione come negli VRP) si sostituisce S_0 con S_1 e si ritorna al punto 2. Altrimenti si assume S_1 come soluzione finale.

Nelle euristiche di ricerca locale esistono due metodi che permettono di migliorare la qualità delle soluzioni:

- Applicare più volte l'algoritmo a partire da diverse soluzioni iniziali. In questo caso l'ideale sarebbe quello di distribuire le soluzioni iniziali, in modo casuale o guidato, con l'obiettivo che siano ben distribuite lungo tutto lo spazio delle soluzioni. Questo aumenterebbe la probabilità di ottenere una soluzione globalmente ottima piuttosto che un minimo locale.
- Aumentare la dimensione degli intorni di ciascuna soluzione.

2.4.3 Algoritmi metaeuristici

Per ovviare alla forte probabilità degli algoritmi euristici di arrivare ad un minimo locale, nella letteratura si fa un grande uso di approcci metaeuristici.

Le caratteristiche principali degli algoritmi metaeuristici sono: l'esplorazione approfondita delle regioni considerate più promettenti nello spazio delle soluzioni e l'impiego di sofisticate regole di ricerca del neighborhood.

Una particolarità che distingue gli algoritmi metaeuristici da quelli euristici è che il procedimento di ricerca accetta il passaggio attraverso soluzioni non ammissibili e/o fasi non migliorative. Grazie a questo processo i risultati che otteniamo sono solitamente di qualità superiore anche se richiedono un tempo maggiore per giungere ad una soluzione ottima.

La famiglia dei metaeuristici proposti per il problema del vehicle routing problem si può suddividere in due categorie principali: I Local Search, che sono una generalizzazione degli approcci basati sulla ricerca locale, e i Population Search, che invece prendono spunto dal mondo fisico. Fanno, ad esempio, parte del secondo gruppo la Simulated Annealing (SA) e la Tabu Search (TS). In questi due casi la ricerca prende il via da una soluzione iniziale ammissibile x_i , e passa, ad ogni iterazione t , dalla soluzione corrente x_t alla migliore soluzione contenuta nel neighborhood $N(x_t)$, x_{t+1} , finchè non risulta soddisfatto un opportuno criterio di arresto. Per favorire la diversificazione ed evitare fasi di stallo intorno a punti di minimo locale, può essere necessario attraversare sequenze di soluzioni peggiorative.

Fanno invece parte della prima tipologia gli Ant System (AS) e gli Algoritmi Genetici (GA).

Nel primo caso, vengono sfruttate informazioni ottenute dalle iterazioni precedenti per creare numerose nuove soluzioni.

Nel secondo caso, invece, ad ogni iterazione viene considerata una popolazione di soluzioni: ogni popolazione deriva dalla precedente attraverso la combinazione delle soluzioni migliori e l'eliminazione delle peggiori.

Gli algoritmi genetici rientrano nei *weak methods*, sono quindi metodi di risoluzione che possono essere applicati a molti tipi di problemi grazie alla loro versatilità. Questi si distinguono dagli *strong methods* che sono invece progettati per affrontare un tipo specifico di problema (Vessey & Glass, 1998).

2.4.4. Le matheuristics

Boschetti et al. (2010) definì le *matheuristics* come algoritmi euristici che nascono dall'interoperabilità di euristica e tecniche di programmazione matematica.

Una classificazione, proposta nello studio [21], va a dividere le matheuristics in tre classi:

- approcci di decomposizione;
- *euristiche migliorative*;
- approcci basati sul *branch-and-price* e *branch-and-column*.

In questa classificazione rientrano molte delle metodologie utilizzate nel VRP.

Analizziamo ciascuna classe:

Approcci di decomposizione

Questa tipologia di approccio è basata sull'idea di identificare dei sotto-problemi che sono più piccoli e più facili da risolvere del problema originale. I sotto-problemi sono gestiti e risolti in maniera indipendente. Alla fine, si ottiene una soluzione accettabile per il problema originale dalle soluzioni dei suoi sotto-problemi. Nella matheuristic uno o tutti i sotto-problemi sono risolti attraverso metodi risolutivi esatti di una modellazione matematica. A volte, in caso di istanze di grandi dimensioni, viene imposto un limite di tempo perdendo così l'ottimalità della soluzione trovata.

Per loro natura, gli approcci di decomposizione sono particolarmente adatti alla soluzione di problemi complessi come quelli di instradamento dei veicoli. Questo perché i problemi di routing implicano tipicamente le seguenti due decisioni di base (oltre a ulteriori decisioni relative alla particolare applicazione): il raggruppamento dei clienti che sono assegnati a ciascun veicolo e la sequenza dei clienti nei percorsi dei veicoli.

A loro volta, gli approcci di decomposizione possono essere classificati in queste quattro classi:

- *approcci cluster first - route second* in cui si separano le due principali decisioni che caratterizzano i problemi di routing, ovvero, l'assegnazione dei clienti ai veicoli e la sequenza dei clienti visitati da ciascuna rotta;

- *approcci a due fasi* di cui fanno parte tutti i restanti approcci che, come prima, dividono il problema iniziale in due parti ma che non sono del tipo *cluster first - route second*;
- *Approcci di ottimizzazione parziale* in cui vengono utilizzati uno o più modelli di programmazione lineare intera (MILP) per gestire una parte del problema mantenendo costanti tutte le decisioni relative alle parti rimanenti;
- *Approcci Rolling Horizon* che sono metodologie risolutive applicate a problemi in cui le decisioni devono essere prese nel tempo. L'idea è di considerare un breve periodo di tempo e risolvere il corrispondente sotto-problema. La decisione presa viene poi utilizzata per aggiornare le informazioni per il seguente sotto-problema.

Il successo di un approccio di decomposizione dipende fortemente dalla capacità di trovare i giusti sotto-problemi e di integrare adeguatamente le loro soluzioni.

Euristiche migliorative

In questa classe di *matheuristics* si utilizzano metodi esatti per migliorare la soluzione ottenuta tramite l'euristica. Sono stati sviluppati diversi modi per combinare la procedura euristica e la soluzione di un modello MILP. Di seguito, classifichiamo le euristiche di miglioramento in base al tipo di approccio proposto.

- *Approcci one-shot* sono schemi euristici in cui un modello MILP viene risolto una volta per migliorare una soluzione accettabile trovata dall'euristica. Tipicamente, la soluzione del modello MILP viene eseguita alla fine dell'algoritmo per perfezionare la migliore soluzione trovata.
- *Approcci con modelli MILP per l'ottimizzazione locale*: in questa classe si utilizzano i modelli MILP come strumenti per esplorare lo spazio delle soluzioni. La maggior parte degli approcci appartenenti a questa classe

presentati in letteratura per la soluzione di problemi di routing utilizzano modelli MILP come strumenti di intensificazione che permettono una lavorazione più efficiente e sostenibile [22].

Quindi l'euristica del miglioramento si basa sull'idea di utilizzare i modelli MILP per modificare e, auspicabilmente, migliorare una soluzione esistente. La natura dell'approccio influenza fortemente la complessità dei modelli incorporati nell'algoritmo di soluzione. Infatti, mentre negli approcci one-shot i modelli MILP vengono risolti una sola volta, negli approcci appartenenti alla seconda classe vengono risolti ripetutamente e quindi devono dare buone soluzioni in un breve tempo di calcolo. Il successo di questi approcci è dovuto al fatto che solitamente i modelli MILP eseguono una ricerca più esaustiva rispetto agli operatori euristici classici.

Approcci basati sul branch-and-price e il branch-and-column

Gli algoritmi di generazione di branch-and-price/column sono generalmente adottati per risolvere formulazioni di partizionamento di insiemi. Gli algoritmi branch-and-price si sono dimostrati efficaci per la soluzione esatta di un'ampia varietà di problemi di routing, inclusi alcuni dei più famosi e classici, come VRP e VRPTW, e sono al momento la metodologia esatta leader. In letteratura sono stati proposti numerosi schemi che differiscono per come vengono generate le colonne e/o come vengono utilizzate per ottenere una soluzione fattibile. Le metodologie basate sul branch-and-price/column possono essere classificate, in base al tipo di approccio proposto, in: *Restricted master heuristics*, *heuristic branching approaches* e *relaxation-based approaches*.

Le matheuristics basate sul branch-and-price e il branch-and-column stanno diventando sempre più popolari. Ciò è dovuto al successo degli algoritmi branch-and-price sviluppati per la soluzione esatta dei problemi di routing.

Capitolo 3

Il modello matematico

3.1 Introduzione

Come già evidenziato nei capitoli precedenti, il problema affrontato in questo lavoro di tesi rientra nella classe di problemi di Pick-up e Delivery con Time Windows (PDP-TW).

In questo capitolo, esamineremo le necessità dell'azienda in modo da comprendere come possano essere tradotte e, una volta introdotte le notazioni usate, descriveremo il modello formulato.

3.2 Analisi del problema reale

L'obiettivo è quello di formulare un modello matematico in cui gli M veicoli a disposizione si occupino di questi quattro aspetti:

1. Distribuzione di prodotti finiti dall'azienda verso clienti;
2. Distribuzione di semilavorati dall'azienda verso terzi (che si devono occupare di alcune fasi di lavorazione del prodotto). Questi "terzi" generalmente non coincidono con i fornitori;
3. Ritiro presso i fornitori e/o terzi di materie prime e/o semilavorati e/o prodotti finiti e/o componenti da portare in azienda;
4. Distribuzione dal terzista, nel caso sia lui ad ultimare il prodotto, al cliente finale senza il ritorno in azienda.

Per semplicità, riassumiamo le varie possibilità nello schema di Fig. 3.1.



Figura 3.1 Schema riassuntivo delle operazioni richieste ai veicoli

3.3 Il modello

La rete

Date queste premesse, si rappresenta il problema su un grafo orientato e completo, tenendo conto di alcuni aspetti:

- I nodi sono caratterizzati da finestre temporali dovute alle fasce orarie in cui si è disponibile a ricevere e/o consegnare merce;
- I nodi possono essere
 - o tipo *pick-up* nel caso in cui debba consegnare del materiale;
 - o tipo *delivery* nel caso in cui debba ricevere della merce.

Detto questo, si può notare che alcuni attori del problema sono caratterizzati da una duplice natura: possono essere, cioè, visti sia come clienti che come fornitori e quindi, sia come *pick-up* che come *delivery* (ad esempio, un terzista). Questa caratteristica viene affrontata sdoppiando questi nodi e creando, per ciascuno di questi, un nodo per il *pick-up* e uno per il *delivery* aventi le stesse coordinate geografiche. Per questo motivo, nell'ottica di minimizzare la distanza percorsa, se un veicolo visita uno di questi due nodi visiterà anche l'altro. Quindi non si corre il rischio di avere due veicoli che servono la stessa azienda.

L'insieme degli n nodi di *pick-up* comprende i fornitori di materie prime/semilavorati ma anche i terziari che devono distribuire il prodotto finito

direttamente al cliente senza dover rientrare al deposito. Mentre, l'insieme degli m nodi di delivery, contiene sia i clienti che possono essere riforniti direttamente dal deposito, che i terziari presso cui il veicolo deve rilasciare il semilavorato.

Nella formulazione del modello, si è assunto che un nodo di pick-up abbia una quantità positiva di merce da consegnare, infatti, quando il veicolo visiterà quel nodo, la sua capienza verrà ridotta perché avrà caricato della merce. Nel caso, invece, di un nodo di delivery avviene esattamente l'opposto, ovvero, il veicolo incrementerà la sua capacità di carico in quanto ha scaricato della merce. Per questo la quantità verrà assunta negativa.

Anche il deposito è stato clonato per distinguere la partenza dall'arrivo di un dato veicolo ed è stato indicato con id 0 (*DepositoPU*) e $n + m + 1$ (*DepositoD*), rispettivamente.

Descrizione del problema

Pertanto, dato un certo veicolo, la sua rotta può essere così composta:

1. Parte dal deposito DepositoPU e al deposito DepositoD deve rientrare.
2. Può servire un nodo di pick-up.
 - a. Se il nodo di pick-up è un fornitore allora il mezzo preleva la merce e continua la rotta. Il suo delivery è il depositoD.
 - b. Se il nodo di pick-up è un terziario che ha compiuto una lavorazione intermedia di prodotto, il mezzo preleva la merce e continua la rotta. Il suo delivery è il depositoD.
 - c. Se il nodo di pick-up è un terziario che ha finito la lavorazione bisogna trasferire il prodotto finito al cliente. Il mezzo preleva la merce e deve consegnarla al cliente che sarà il delivery di questo nodo.
3. Può servire un nodo di delivery.

- a. Se il nodo di delivery è un cliente, il mezzo lascia la merce e il pick-up di questo delivery potrebbe essere un terziario oppure il depositoPU.
 - b. Se il nodo di delivery è un terziario che deve ricevere la merce da lavorare, il mezzo rilascia la merce e il pick-up associato può essere un altro terziario o il depositoPU.
4. Come già evidenziato, esistono anche situazioni in cui uno stesso terziario può essere di delivery e di pick-up contemporaneamente. In tal caso viene duplicato. Facciamo un esempio: supponiamo che un terziario debba ricevere del semilavorato ma contemporaneamente debba anche restituire della merce (se finita a un cliente, altrimenti al deposito). Questo significa che il terziario sarà nodo di delivery avente il deposito come pick-up associato; sarà invece nodo di pick-up con delivery associato il cliente/deposito per la merce che deve restituire.

3.4 La formulazione del modello matematico

Di seguito, si riporta la notazione usata per formulare matematicamente il problema.

- M*** Numero dei veicoli
n Numero dei nodi di Pick-up
m Numero dei nodi di Delivery

Gli insiemi

- C*** Insieme di tutti i nodi. $C = \{0, \dots, n + m + 1\}$
C0 L'insieme di tutti i nodi esclusi i depositi $C = C \setminus \{0, n + m + 1\}$
C1 L'insieme di tutti i nodi escluso il DepositoPU $C1 = C \setminus 0$
C2 L'insieme di tutti i nodi escluso il DepositoD $C2 = C \setminus (n + m + 1)$
D Insieme dei nodi di delivery il cui pick-up è il deposito
P Insieme dei nodi di pick-up il cui delivery è il deposito

- K Insieme dei veicoli
 R Set di coppie (i, j) in cui i è pick-up di j

I parametri

- q_i Quantità totale che il nodo $i \in C0$ consegna o ritira. $q_i > 0$ se i è un punto di pick-up. $q_i < 0$ se i è un punto di delivery.
 p_j Quantità che un nodo $i \in D$ deve ricevere dal deposito
 s_i Tempo di servizio al nodo i
 e_i Orario di apertura della TW del nodo i
 l_i Orario di chiusura della TW del nodo i
 d_{ij} Distanza, espressa in Km, tra il nodo i ed il nodo j
 $Q_{max k}$ Capacità di carico, espressa in Kg, del veicolo k
 G_k Consumo, espresso in €/Km, del veicolo k
 t_{max} Durata massima ammissibile della rotta espressa in minuti
 t_{ij} Tempo di viaggio, espresso in minuti, dal nodo i al nodo j

Le variabili decisionali

- x_{ij}^k Variabile binaria con $k \in K, i \in C2, j \in C1$.
 $x_{ij}^k = 1$ se il veicolo k viaggia direttamente dal nodo i al nodo j ; 0 altrimenti.
 T_i^k $k \in K, i \in C$. Tempo, espresso in minuti, di inizio del servizio di k al nodo i .
 L_i^k $k \in K, i \in C$. Carico, espresso in Kg, presente sul veicolo k al nodo i .

Vista la natura delle variabili decisionali in gioco, il modello matematico proposto è di Programmazione Lineare Mista Intera.

La funzione obiettivo

L'obiettivo di questo modello è quello di minimizzare il costo totale della soluzione, ovvero il costo ottenuto sommando il costo di ciascuna rotta che fa parte della soluzione (3.1).

$$\min \sum_{\substack{k \in K \\ i, j \in C}} x_{ij}^k d_{ij} G_k \quad (3.1)$$

Quindi se l'arco (i, j) è attraversato dal k -esimo veicolo, ovvero $x_{ij}^k = 1$, si va a sommare il costo di quel tratto dato dal prodotto della distanza tra i due nodi ed il consumo al km del k -esimo veicolo che lo attraversa.

I vincoli

I vincoli 3.2 impongono che ogni nodo, ad esclusione del deposito, sia visitato esattamente una volta.

$$\sum_{\substack{k \in K \\ j \in C1}} x_{ij}^k = 1, \forall i \in C0 \quad (3.2)$$

Nel caso in cui il nodo i , diverso dal deposito, sia il pick-up del nodo j , anch'esso diverso dal deposito, è importante che questi due nodi siano serviti dallo stesso veicolo k (vincoli 3.3).

$$\sum_{h \in C2} x_{hi}^k - \sum_{m \in C2} x_{mj}^k = 0, \quad \forall k \in K, \forall (i, j) \in R \quad (3.3)$$

I vincoli 3.4 e 3.5 impongono che un veicolo può come non può essere usato, e quindi uscire da 0 e rientrare in $n + m + 1$. Inoltre, non è possibile usare un veicolo per attraversare l'arco diretto che da 0 va ad $n + m + 1$.

$$\sum_{j \in C0} x_{0j}^k \leq 1, \quad \forall k \in K \quad (3.4)$$

$$\sum_{j \in C0} x_{j, n+m+1}^k \leq 1, \forall k \in K \quad (3.5)$$

È necessario imporre i vincoli di conservazione del flusso, ovvero, se un veicolo visita un nodo j deve anche da questo ripartire (3.6).

$$\sum_{i \in C2, i \neq j} x_{ij}^k = \sum_{h \in C1, h \neq j} x_{jh}^k, \quad \forall k \in K, \forall j \in C0 \quad (3.6)$$

I vincoli 3.7 e 3.8 servono ad imporre che nessun veicolo vada da un nodo $j \in C2$ al nodo 0 (depositPU). Viceversa, nessun veicolo può andare dal nodo $n + m + 1$ (depositoD) a un qualunque nodo $j \in C1$

$$x_{i0}^k == 0, \quad k \in K, i \in C2 \quad (3.7)$$

$$x_{n+m+1j}^k == 0, \quad k \in K, j \in C1 \quad (3.8)$$

Infine, il vincolo 3.9 serve ad impedire che si utilizzino più di M veicoli.

$$\sum_{\substack{k \in K \\ j \in C1}} x_{0j}^k \leq M \quad (3.9)$$

Vincoli temporali

A seconda degli accordi tra azienda e lavoratore, il tempo che ogni mezzo impiega per effettuare la rotta non può superare un determinato tempo di viaggio (t_{max}). Questo vincolo può essere così espresso (3.10):

$$T_{n+m+1}^k - T_0^k \leq t_{max}, \quad \forall k \in K \quad (3.10)$$

La presenza delle finestre temporali obbliga che vengano imposti i vincoli (3.11), ovvero che ciascun nodo venga servito all'interno della sua finestra temporale.

$$e_i \sum_{j \in C1} x_{ij}^k \leq T_i^k \leq l_i \sum_{j \in C1} x_{ij}^k, \quad \forall i \in C0, k \in K \quad (3.11)$$

La presenza di $\sum_{j \in C1} x_{ij}^k$ serve a far sì che, nel caso in cui il veicolo k non visitasse il nodo i , la variabile T_i^k assuma valore 0. Come già accennato, le finestre temporali possono essere vincoli di tipo *hard* o *soft*. In questo caso, ci troviamo di fronte alla prima tipologia. Infatti, è possibile servire i clienti solamente durante la finestra temporale e quindi il veicolo deve aspettare nel caso arrivi in anticipo e non può arrivare in ritardo.

Abbiamo già imposto che quando un nodo $i \in C0$ è pick-up di un nodo $j \in C0$ questi due nodi devono essere serviti dallo stesso mezzo, ma è importante che venga servito prima il nodo di pick-up e poi il nodo di delivery (3.12).

$$T_i^k + t_{ij} + s_i \leq T_j^k, \quad \forall k \in K, (i, j) \in R \quad (3.12)$$

Infine, i vincoli 3.13 e 3.14 permettono di aggiornare la variabile T_i^k . Questi sono due vincoli condizionali che si attivano solo se il veicolo k attraversa l'arco (i, j) in cui è stato impostato come *bigM* il valore t_{max} .

$$T_j^k \geq T_i^k + (t_{ij} + s_i)x_{ij}^k - t_{max}(1 - x_{ij}^k), \quad \forall k \in K, i \in C2, j \in C1 \quad (3.13)$$

$$T_j^k \leq T_i^k + (t_{ij} + s_i)x_{ij}^k + t_{max}(1 - x_{ij}^k), \quad \forall k \in K, i \in C2, j \in C1 \quad (3.14)$$

Vincoli di carico

Per quanto riguarda la variabile L^k , il primo vincolo da rispettare è che, per ogni mezzo k , il carico non possa superare la capacità del mezzo stesso (3.15).

$$L_i^k \leq Q_{max}^k \sum_{j \in C1} x_{ij}^k, \quad \forall k \in K, \forall i \in C2 \quad (3.15)$$

Siccome alcuni nodi di delivery hanno come nodo di pick-up il deposito PU, è necessario che ogni veicolo, al nodo 0, carichi esclusivamente la merce relativa ai nodi di delivery che visita nella sua rotta (3.16).

$$L_0^k = \sum_{j \in D} p_j \sum_{i \in C0} x_{ij}^k \quad \forall k \in K \quad (3.16)$$

Il peso della merce di ogni mezzo al nodo 0, però, non deve superare i limiti di carico del veicolo (3.17).

$$0 \leq L_0^k \leq Q_{max}^k, \quad \forall k \in K \quad (3.17)$$

Infine, i vincoli 3.18 e 3.19 permettono di aggiornare la variabile L . Come nella variabile T , ci troviamo davanti a vincoli condizionali, ma in questo caso, il valore di *bigM* è rappresentato dalla capacità massima del mezzo Q_{max}^k .

$$L_j^k \geq L_i^k + q_j x_{ij}^k - Q_{max}^k (1 - x_{ij}^k), \quad \forall k \in K, i \in C2, j \in C1, i \neq j \quad (3.18)$$

$$L_j^k \leq L_i^k + q_j x_{ij}^k + Q_{max}^k (1 - x_{ij}^k), \quad \forall k \in K, i \in C2, j \in C1, i \neq j \quad (3.19)$$

Regole di pre-processing

È possibile applicare delle regole di pre-processing che mirano a ridurre il numero di variabili in gioco, scartando, a priori, alcuni archi che, per alcuni dei vincoli imposti, non potrebbero essere serviti.

In particolare, queste regole sono legate alle finestre temporali, al massimo tempo di rientro al deposito e alla capacità di carico del veicolo.

1. Dati due nodi $(i, j) \in P$ ($i \neq j$) nel caso in cui la somma delle quantità da prelevare ai due nodi ($q_i + q_j$) superasse la capacità del k -esimo veicolo allora la variabile x_{ij}^k può essere posta uguale a zero. In altre parole:

$$x_{ij}^k = 0 \quad \forall k \in K, i \in P, j \in P, i \neq j: q_i + q_j > Q_{max}^k$$

2. Supponendo il caso in cui il veicolo k arrivi al nodo i all'estremo sinistro della sua finestra temporale (e_i). Se sommando a questo il tempo di servizio a quel nodo (s_i) e il tempo per andare da i a j (t_{ij}) si supera l'estremo destro della finestra temporale di j (l_j), allora si può impostare $x_{ij}^k (\forall k \in K)$ uguale a zero. In altre parole:

$$x_{ij}^k = 0 \forall k \in K, i \in C0, j \in C0, i \neq j: e_i + s_i + t_{ij} > l_j$$

3. Nel caso in cui sommando il tempo per andare dal deposito al nodo i , quello per andare da i a j , quello per tornare da j al deposito e i relativi tempi di servizio (s_i e s_j) si superasse il tempo massimo a disposizione per una rotta, allora si può impostare la variabile $x_{ij}^k (\forall k \in K)$ uguale a zero. In altre parole:

$$x_{ij}^k = 0 \forall k \in K, i \in C0, j \in C0, i \neq j: t_{0i} + s_i + t_{ij} + s_j > T_{max}$$

Capitolo 4

La validazione e la sperimentazione

4.1 Introduzione

In questo capitolo, come prima cosa affronteremo il processo di validazione del modello proposto. A tal proposito, si procederà con istanze di dimensione e complessità crescenti e si verificheranno le soluzioni ottenute. In seguito, analizzeremo il caso di studio reale.

Il modello è stato implementato tramite il linguaggio OPL e risolto con IBM ILOG CPLEX. Il computer utilizzato è caratterizzato da:

- Processore Intel® Core™ i5-6500 CPU@ 3.20 GHz 3.19 Ghz
- RAM da 8,00 GB

4.2 Il processo di validazione con istanze di piccole dimensioni

Di seguito, verranno descritte istanze che sono state pensate specificatamente per validare il modello e per analizzarne la soluzione.

4.2.1 Istanza 1

I dati

La prima istanza utilizzata è particolarmente semplice ed è caratterizzata da:

- un solo veicolo ($M = 1$);
- tre punti di pick-up escluso il deposito ($n = 3$);
- un punto di delivery escluso il deposito ($m = 1$).

Chiameremo i tre nodi di pick-up:

- Nodo 1;
- Nodo 2;
- Nodo 3.

Chiameremo il punto di delivery:

- Nodo 4.

Essendo il deposito clonato avrà due nomi, uno per il pick-up e uno per il delivery:

- Nodo 0 (depositoPU);
- Nodo 5 (depositoD).

Ad ogni nodo sono associate delle coordinate, la quantità di merce da caricare o scaricare espressa in Kg, il tempo necessario ad eseguire l'operazione di carico/scarico e la relativa Time Window. Quest'ultime sono molto lasche, vanno infatti dal tempo 0 a 480.

Il veicolo deve eseguire i seguenti compiti:

- trasportare merce pari a 400Kg dal deposito (Nodo 0) al Nodo 4;
- trasportare merce pari a 100Kg dal Nodo 1 al Nodo 4;
- trasportare merce pari a 200Kg dal Nodo 2 al deposito (Nodo 5);
- trasportare merce pari a 100Kg dal Nodo 3 al deposito (Nodo 5);

Da qui è possibile ricavare le quantità che l'autotrasportatore deve caricare o lasciare ad ogni nodo i (q_i). Queste vengono riassunte nella tabella 4.1.

Sono note a priori, per ogni nodo i , le informazioni relative a:

- coordinate geografiche del nodo i ;
- tempo di servizio al nodo i (s_i);
- apertura della TW al nodo i (e_i);
- chiusura della TW al nodo i (l_i).

Queste informazioni possono essere riassunte nella seguente tabella (4.1):

	q_i (Kg)	Coordinata x	Coordinata y	e_i	s_i	e_i	l_i
Nodo 0	400	30	0	400	5	0	480
Nodo 1	100	0	20	100	10	0	480
Nodo 2	200	10	50	200	20	0	480
Nodo 3	100	50	50	100	30	0	480

Nodo 4	-500	60	20	-500	40	0	480
Nodo 5	-300	30	0	-300	/	0	480

Tabella 4.1: Informazioni relative ai nodi dell'istanza 1

La distribuzione dei nodi (fornitori e non, incluso il deposito) è rappresentata attraverso la Fig. 4.1.

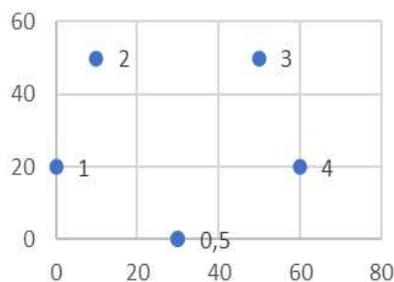


Fig. 4.1: Disposizione spaziale dei nodi dell'istanza 1.

Di seguito vengono riportate la matrice delle distanze (tabella 4.2) e quella dei tempi (tabella 4.3).

	Nodo 0	Nodo 1	Nodo 2	Nodo 3	Nodo 4	Nodo 5
Nodo 0	0,00	36,06	53,85	53,85	36,06	0,00
Nodo 1	36,06	0,00	31,62	58,31	60,00	36,06
Nodo 2	53,85	31,62	0,00	40,00	58,31	53,85
Nodo 3	53,85	58,31	40,00	0,00	31,62	53,85
Nodo 4	36,06	60,00	58,31	31,62	0,00	36,06
Nodo 5	0,00	36,06	53,85	53,85	36,06	0,00

Tabella 4.2: Matrice delle distanze. Distanza d_{ij} che intercorre tra il nodo i e il nodo j .

	Nodo 0	Nodo 1	Nodo 2	Nodo 3	Nodo 4	Nodo 5
Nodo 0	0	20	60	60	20	0
Nodo 1	20	0	10	40	50	20
Nodo 2	60	10	0	10	40	60
Nodo 3	60	40	10	0	10	60
Nodo 4	20	50	40	10	0	20
Nodo 5	0	20	60	60	20	0

Tabella 4.3: Matrice dei tempi. Tempo t_{ij} necessario per arrivare dal nodo i al nodo j .

Il veicolo ha una capacità massima di 1000Kg, un costo di viaggio pari a 2€/Km e il tempo massimo di viaggio accettabile (t_{max}) è di 480 minuti.

La soluzione

In pochi secondi, il solver ha trovato soluzione ottima. In particolare, è stata trovata la seguente rotta:

Nodo 0, Nodo 1, Nodo 2, Nodo 3, Nodo 4 e Nodo 5.

Questa rotta è graficamente rappresentata in Fig. 4.2.

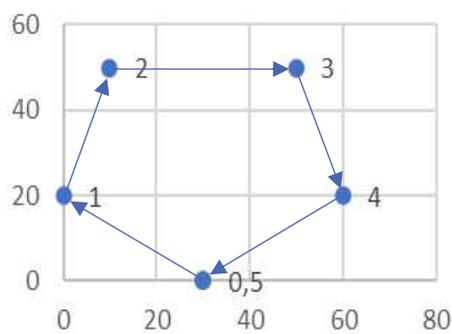


Fig. 4.2: Rotta fornita dal risolutore per l'istanza 1.

La variabile L_i

I valori della variabile L_i , che rappresenta il carico del veicolo dopo aver attraversato l'i-esimo nodo, sono indicati nella tabella 4.4.

	Nodo 0	Nodo 1	Nodo 2	Nodo 3	Nodo 4	Nodo 5
Veicolo 1	400	500	700	800	300	0

Tabella 4.4: Valore della variabile L per ogni nodo.

La variabile T_i

I valori della variabile T_i , che rappresenta il tempo in minuti dell'inizio del servizio all'i-esimo nodo, sono indicati nella tabella 4.5.

	Nodo 0	Nodo 1	Nodo 2	Nodo 3	Nodo 4	Nodo 5
Veicolo 1	0	25	45	75	115	175

Tabella 4.5: Valore della variabile T per ogni nodo.

Come si può osservare, i valori delle variabili L_i e T_i sono coerenti con la rotta individuata. Inoltre, i vincoli di precedenza tra coppie di nodi sono rispettati.

4.2.2 Istanza 2

I dati

La seconda istanza ha gli stessi dati iniziali della prima ad eccezione delle finestre temporali. In particolare, per mettere alla prova il modello, la TW del Nodo 3 è stata ristretta molto. Questo ha permesso di mettere sotto analisi due cose:

- 1) che il modello rispettasse la limitazione temporale imposta dalla TW;
- 2) che il modello rispettasse le precedenze (il Nodo 1 è pick-up del Nodo 4).

Le modifiche fatte sono riassunte nella tabella 4.6.

	e_i	l_i
Nodo 0	0	480
Nodo 1	40	400
Nodo 2	100	400
Nodo 3	80	85
Nodo 4	150	400
Nodo 5	0	480

Tabella 4.6: TW dell'istanza 2.

La soluzione

In pochi secondi, il solver è riuscito a trovare una soluzione ottima. In particolare, è stata trovata una rotta in cui i nodi vengono serviti in questo ordine:

Nodo 0, Nodo 3, Nodo 2, Nodo 1, Nodo 4 e Nodo 5.

Il vincolo di precedenza è stato rispettato e la rotta è graficamente rappresentata in Fig. 4.3.

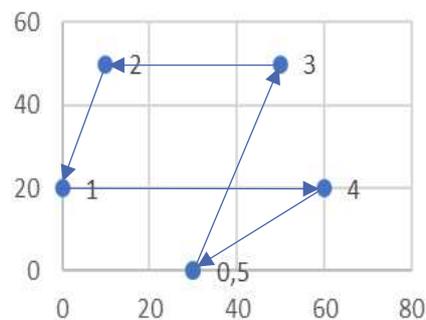


Fig. 4.3. Rappresentazione grafica della rotta fornite dal risolutore nell'istanza 2.

La variabile L_i

I valori della variabile L_i rispettano i limiti di carico e sono indicati in tabella 4.7.

	Nodo0	Nodo1	Nodo2	Nodo3	Nodo4	Nodo5
Veicolo 1	400	800	700	500	300	0

Tabella 4.7

La variabile T_i

I valori della variabile T_i rispettano le nuove TW, anche quella relativa al Nodo 3, e sono indicati in tabella 4.8.

	Nodo0	Nodo1	Nodo2	Nodo3	Nodo4	Nodo5
Veicolo 1	15	150	120	80	210	270

Tabella 4.8

Anche in questo caso, si osserva che i valori di queste due variabili sono coerenti con la rotta individuata e che i vincoli di TW e di precedenza sono rispettati.

4.2.3 Istanza 3

I dati

La terza istanza continua ad utilizzare la maggior parte dei dati iniziali della prima. Le modifiche fatte riguardano il numero di veicoli utilizzati ($M = 2$) e l'ampiezza delle TW (tabella 4.9). Questa istanza è servita a verificare che, al fine di rispettare i vincoli, il modello utilizzasse più di un veicolo nonostante non fosse economicamente conveniente. Le TW del Nodo 1 e del Nodo 3 sono stringenti e praticamente sovrapposte in modo da rendere impossibile che un solo veicolo riesca a servire entrambi i nodi. Un altro contributo di questa istanza è quello di verificare l'equazione 3.3, ovvero, se un nodo i è pick-up di un nodo j (con $i, j \in C0$ quindi diversi dal deposito) devono essere serviti dallo stesso veicolo.

	e_i	l_i
Nodo 0	0	200
Nodo 1	30	50

Nodo 2	55	200
Nodo 3	30	70
Nodo 4	0	250
Nodo 5	0	300

Tabella 4.9: TW dell'istanza 3.

Le caratteristiche tecniche dei due veicoli sono invece riassunte nella tabella 4.10.

	Q_k Capacità veicolo Kg	G_k Consumo €/km
Veicolo 1	1000	2
Veicolo 2	500	2.5

Tabella 4.10: Caratteristiche tecniche dei veicoli dell'istanza 3

La soluzione

In pochi secondi, il solver è riuscito a trovare una soluzione ottima. In particolare, la soluzione è formata dalle seguenti due rotte:

Rotta veicolo 1: Nodo 0, Nodo 1, Nodo 2, Nodo 4 e Nodo 5.

Rotta veicolo 2: Nodo 0, Nodo 3, Nodo 5.

Queste due rotte rispettano il vincolo di precedenza e sono graficamente rappresentata in Fig. 4.4.

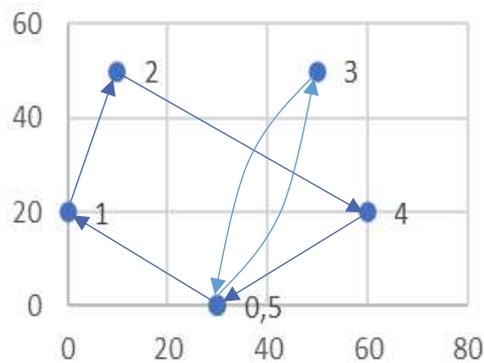


Fig. 4.4. rappresentazione grafica delle rotte fornite dal risolutore nell'istanza 3.

La variabile L_i

I valori della variabile L_i sono indicati nella tabella 4.11.

	Nodo 0	Nodo 1	Nodo 2	Nodo 3	Nodo 4	Nodo 5
Veicolo 1	400	500	700	0	200	0
Veicolo 2	0	0	0	100	0	0

Tabella 4.11. Valori della variabile L forniti dal solver.

Questa istanza ci ha permesso anche di verificare che, come descritto nel modello, venisse restituito $L_i^k = 0$ nel caso il veicolo k non attraversasse l' i -esimo nodo.

La variabile T_i

I valori della variabile T_i sono indicati nella tabella 4.12.

	Nodo0	Nodo1	Nodo2	Nodo3	Nodo4	Nodo5
Veicolo 1	10	35	55	0	115	175
Veicolo 2	5	0	0	70	0	160

Tabella 4.12: Valori della variabile T forniti dal solver.

Questa istanza ci ha permesso anche di verificare che, come descritto nel modello, venisse restituito $T_i^k = 0$ nel caso il veicolo k non attraversasse l' i -esimo nodo.

4.2.4 Istanza 4

Questa istanza è di complessità maggiore rispetto alle precedenti essendo composta da 9 nodi più i due relativi al deposito.

I dati

Il numero di veicoli utilizzati è pari a due ($M = 2$) e le loro caratteristiche tecniche sono riportate nella tabella 4.13.

	Q[k] Capacità veicolo	G[k] Costo al km
Veicolo 1	1000	2
Veicolo 2	500	4

Tabella 4.13: Caratteristiche tecniche dei veicoli dell'istanza 3

In questa istanza sono presenti:

- 4 nodi che fungono esclusivamente da pick-up (Nodo 1, Nodo 2, Nodo 3, Nodo 4);
- 5 nodi che fungono esclusivamente da delivery (Nodo 6, Nodo 8, Nodo 9, Nodo 10, Nodo 11);
- Un nodo che funge sia da pick-up che da delivery e che è stato quindi duplicato. (Nodo 5 per il pick-up e Nodo 7 per il delivery).

L'obiettivo è trovare le rotte che, al minor costo possibile, permettano di:

- Consegnare, prelevando dal deposito (Nodo 0), 10 Kg al nodo8, 20Kg al Nodo 11, 50Kg al Nodo 6 e 30Kg al Nodo 7;
- Ritirare 200Kg dal Nodo 1, 20Kg dal Nodo 5 e 100Kg dal Nodo 4 da versare al deposito (Nodo 12);
- Trasferire 50Kg dal Nodo 2 al Nodo 9;
- Trasferire 50Kg dal Nodo 2 al Nodo 8;
- Trasferire 10Kg dal Nodo 3 al Nodo 10.

Queste informazioni possono essere riassunte nella tabella 4.14.

Nodo Pick-Up	Nodo Delivery	Quantità
Nodo 2	Nodo 9	50
Nodo 2	Nodo 8	50
Nodo 3	Nodo 10	10
Nodo 1	Nodo 12	200
Nodo 5	Nodo 12	20
Nodo 4	Nodo 12	100
Nodo 0	Nodo 8	10
Nodo 0	Nodo 11	20
Nodo 0	Nodo 6	50
Nodo 0	Nodo 7	30

Tabella 4.14: Riassunto delle relazioni tra i nodi

Le restanti informazioni note a priori sui nodi sono riassunte nella tabella 4.15.

	q_i (Kg)	Coordinata x	Coordinata y	e_i	s_i	e_i
Nodo 0	110	5,3	5,3	3	0	480
Nodo 1	200	0	3	4	10	450
Nodo 2	100	2,4	9,7	5	65	70
Nodo 3	10	4,7	6,4	6	65	70
Nodo 4	100	7,4	6,3	7	40	300
Nodo 5	20	3,5	2,5	8	70	300
Nodo 6	-50	5,2	0	5	40	200
Nodo 7	-30	3,5	2,5	6	70	300

Nodo 8	-60	1,5	4,3	3	0	480
Nodo 9	-50	0,3	7,5	4	50	100
Nodo 10	-10	6,6	8,5	2	10	350
Nodo 11	-20	6,6	7	1	80	200
Nodo 12	-320	5,3	5,3	7	0	480

Tabella 4.15: Informazioni relative ai nodi dell'istanza 4.

Infine, viene riportata la matrice delle distanze (tabella 4.16). Assumendo che i veicoli vadano ad una velocità media di 60Km/h è possibile ricavare la matrice dei tempi.

	Nodo 0	Nodo 1	Nodo 2	Nodo 3	Nodo 4	Nodo 5	Nodo 6	Nodo 7	Nodo 8	Nodo 9	Nodo 10	Nodo 11	Nodo 12
Nodo 0	0,0	5,8	5,3	1,3	2,3	3,3	5,3	3,3	3,9	5,5	3,5	2,1	0,0
Nodo 1	5,8	0,0	7,1	5,8	8,1	3,5	6,0	3,5	2,0	4,5	8,6	7,7	5,8
Nodo 2	5,3	7,1	0,0	4,0	6,0	7,3	10,1	7,3	5,5	3,0	4,4	5,0	5,3
Nodo 3	1,3	5,8	4,0	0,0	2,7	4,1	6,4	4,1	3,8	4,5	2,8	2,0	1,3
Nodo 4	2,3	8,1	6,0	2,7	0,0	5,4	6,7	5,4	6,2	7,2	2,3	1,1	2,3
Nodo 5	3,3	3,5	7,3	4,1	5,4	0,0	3,0	0,0	2,7	5,9	6,8	5,5	3,3
Nodo 6	5,3	6,0	10,1	6,4	6,7	3,0	0,0	3,0	5,7	9,0	8,6	7,1	5,3
Nodo 7	3,3	3,5	7,3	4,1	5,4	0,0	3,0	0,0	2,7	5,9	6,8	5,5	3,3
Nodo 8	3,9	2,0	5,5	3,8	6,2	2,7	5,7	2,7	0,0	3,4	6,6	5,8	3,9
Nodo 9	5,5	4,5	3,0	4,5	7,2	5,9	9,0	5,9	3,4	0,0	6,4	6,3	5,5
Nodo 10	3,5	8,6	4,4	2,8	2,3	6,8	8,6	6,8	6,6	6,4	0,0	1,5	3,5
Nodo 11	2,1	7,7	5,0	2,0	1,1	5,5	7,1	5,5	5,8	6,3	1,5	0,0	2,1
Nodo 12	0,0	5,8	5,3	1,3	2,3	3,3	5,3	3,3	3,9	5,5	3,5	2,1	0,0

Tabella 4.16: Matrice dei tempi dell'istanza 4.

La soluzione

In pochi secondi, il solver è riuscito a trovare una soluzione ottima. In particolare, la soluzione è formata dalle seguenti due rotte:

Rotta veicolo 1: Nodo 0, Nodo 3, Nodo 10, Nodo 11, Nodo 4, Nodo 12.

Rotta veicolo 2: Nodo 0, Nodo 2, Nodo 9, Nodo 8, Nodo 1, Nodo 5, Nodo 7, Nodo 6, Nodo 12.

Queste due rotte sono graficamente rappresentata in Fig. 4.5.

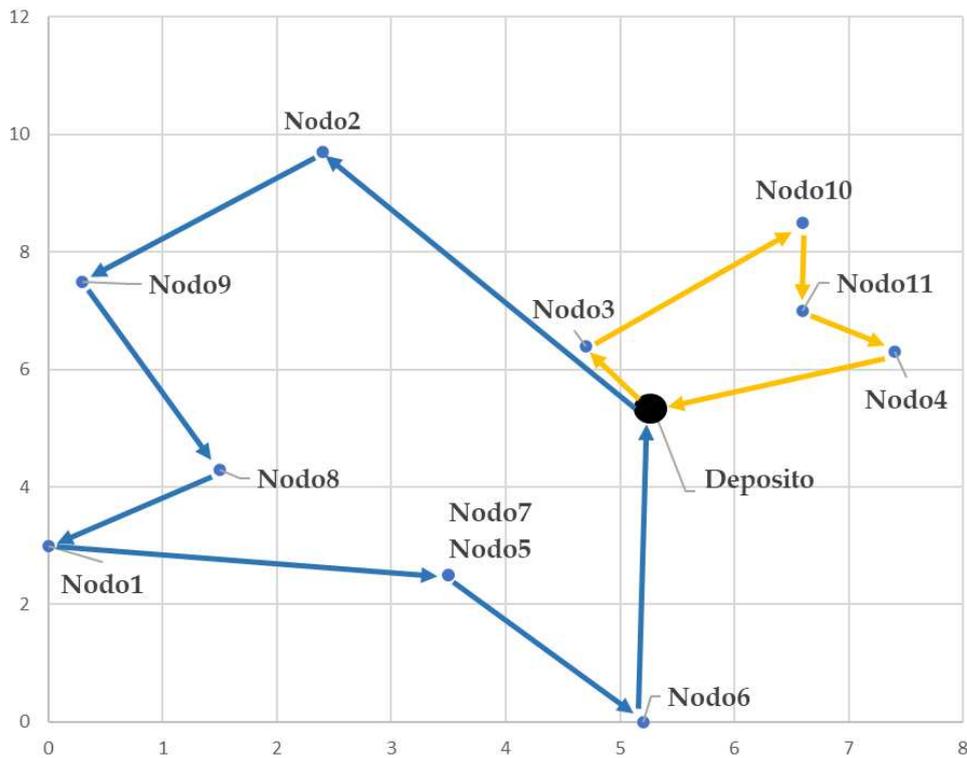


Fig. 4.5. rappresentazione grafica delle rotte fornite dal risolutore nell'istanza 4.

La variabile L_i

I valori della variabile L_i sono indicati nella tabella 4.17.

	Nodo 0	Nodo 1	Nodo 2	Nodo 3	Nodo 4	Nodo 5	Nodo 6	Nodo 7	Nodo 8	Nodo 9	Nodo 10	Nodo 11	Nodo 12
V0	20	0	0	30	1	0	0	0	0	0	20	0	0
V1	90	280	190	0	0	300	220	270	080	140	0	0	0

Tabella 4.17: Valori della variabile L dati dal solver.

La variabile T_i

I valori della variabile T_i sono indicati nella tabella 4.18.

	Nodo 0	Nodo 1	Nodo 2	Nodo 3	Nodo 4	Nodo 5	Nodo 6	Nodo 7	Nodo 8	Nodo 9	Nodo 10	Nodo 11	Nodo 12
V0	63	0	0	67	0	0	0	0	0	0	76	80	91
V1	57	85	65	0	0	93	110	101	80	73	0	0	120

Tabella 4.18: Valori della variabile L dati dal solver

4.3 Il caso studio Tre Elle

Una volta validato con delle istanze di piccole dimensioni, il modello è stato testato sul caso reale. Quest'ultimo però, essendo particolarmente complesso (33 nodi più due del deposito), è stato analizzato per step. Sono state create tre diverse istanze che verranno chiamate "Tre Elle Ridotto", "Tre Elle Intermedio" e "Tre Elle Completo". Inoltre, per motivi di privacy, saranno omessi sia i nomi delle aziende che le loro coordinate geografiche.

Sono stati presi in considerazione i seguenti tre veicoli già in uso da parte dell'azienda:

- Fiorino con capacità 600Kg e costo di utilizzo pari a 0,6€/Km;
- Furgone IVECO con capacità di 1300Kg e costo di utilizzo pari a 1,10€/Km;
- Motrice con capacità 8000Kg e costo di utilizzo pari 2€/km.

4.3.1 Istanza "Tre Elle Ridotto"

I dati

Questa istanza è composta in totale da nove nodi più i due relativi al deposito (Nodo 0 per il pick-up e Nodo 10 per il delivery). In particolare, sono presenti:

- Due nodi di pick-up (Nodo 4 e Nodo 9);
- Sette nodi di delivery.

Oltre agli scambi con il deposito sono presenti due set $i, j \in R$, in quanto:

- Il Nodo 4 deve portare al Nodo 1 120 Kg;
- Il Nodo 4 deve portare al Nodo 2 110 Kg.

Le informazioni possono essere riassunte nella tabella 4.19.

	q_i	e_i	s_i	e_i
Nodo 0	2240	15	0	480
Nodo 1	-200	10	0	360
Nodo 2	-180	10	240	480
Nodo 3	-390	10	240	480
Nodo 4	330	20	60	80
Nodo 5	-800	15	0	240
Nodo 6	-120	10	0	240
Nodo 7	-680	10	0	480
Nodo 8	-100	10	60	420
Nodo 9	510	30	300	480
Nodo 10	-610	0	0	480

Tabella 4.19: Informazioni relative ai nodi.

La soluzione

La soluzione restituita è ottima ed è stata trovata in 7,35 secondi con

Costo totale	1410,509
Distanza totale percorsa	1293,706

Le rotte trovate sono:

Rotta veicolo 1: Nodo 0, Nodo 9, Nodo 10.

Rotta veicolo 2: Nodo 0, Nodo 4, Nodo 1, Nodo 8, Nodo 6, Nodo 3, Nodo 2, Nodo 10.

Rotta veicolo 3: Nodo 0, Nodo 5, Nodo 7, Nodo 10.

La variabile L_i

I valori della variabile L_i sono indicati nella tabella 4.20.

	Nodo 0	Nodo 1	Nodo 2	Nodo 3	Nodo 4	Nodo 5	Nodo 6	Nodo 7	Nodo 8	Nodo 9	Nodo 10
Veicolo 1	0	0	0	0	0	0	0	0	0	510	0
Veicolo 2	760	890	100	280	1090	0	670	0	790	0	0
Veicolo 3	1480	0	0	0	0	680	0	0	0	0	0

Tabella 4.20: Valori della variabile L restituiti dal solver.

La variabile T_i

I valori della variabile T_i sono indicati nella tabella 4.21.

	Nodo 0	Nodo 1	Nodo 2	Nodo 3	Nodo 4	Nodo 5	Nodo 6	Nodo 7	Nodo 8	Nodo 9	Nodo 10
Veicolo1	232	0	0	0	0	0	0	0	0	300	367
Veicolo2	72	91	326	242	80	0	149	0	112	0	332
Veicolo3	1480	0	0	0	0	680	0	0	0	0	0

Tabella 4.21: Valori della variabile T restituiti dal solver.

4.3.2 Istanza “Tre Elle intermedio”

Questa istanza è composta da 17 nodi più i due relativi al deposito (Nodo 0 per il pick-up e Nodo 18 per il delivery) per un totale di 19 nodi. Rispetto al caso ridotto, il tempo necessario al risolutore per trovare una soluzione ottima è cresciuto notevolmente.

Per questo risulta importante analizzare, non solo la soluzione ottima ottenuta lasciando al risolutore un tempo molto grande, ma anche lo status delle soluzioni restituite nel momento in cui impostiamo una limitazione temporale.

Studieremo le soluzioni ottenute impostando time-limit di 1', 5', 10' e senza limite

I dati

I nodi da 0 a 9 rimangono invariati, quindi verranno riportate le informazioni relative ai nodi da 10 a 18.

Le informazioni possono essere riassunte nella tabella 4.22.

	q_i	s_i	e_i	l_i
Nodo 10	-200	5	0	480
Nodo 11	-50	5	0	480
Nodo 12	-50	10	120	450
Nodo 13	-50	10	120	450
Nodo 14	300	15	60	480
Nodo 15	-100	5	0	240
Nodo 16	450	10	60	120

Nodo 17	-70	5	0	420
Nodo 18	-1590	10	0	480

Tabella 4.22: Informazioni relative ai nodi.

La soluzione col tempo limite di 1 minuto

La soluzione restituita è ammissibile, ma con un gap di ottimalità del 28,33%.

Costo totale 1410,51

Distanza totale percorsa 1293,71

Le rotte trovate, rappresentate graficamente nella Fig. 4.6, sono:

Rotta veicolo 1: Nodo 0, Nodo 8, Nodo 10, Nodo 9, Nodo 13, Nodo 18.

Rotta veicolo 2: Nodo 0, Nodo 4, Nodo 2, Nodo 14, Nodo 1, Nodo 15, Nodo 6, Nodo 12, Nodo 18.

Rotta veicolo 3: Nodo 0, Nodo 17, Nodo 16, Nodo 3, Nodo 11, Nodo 7, Nodo 5, Nodo 18.

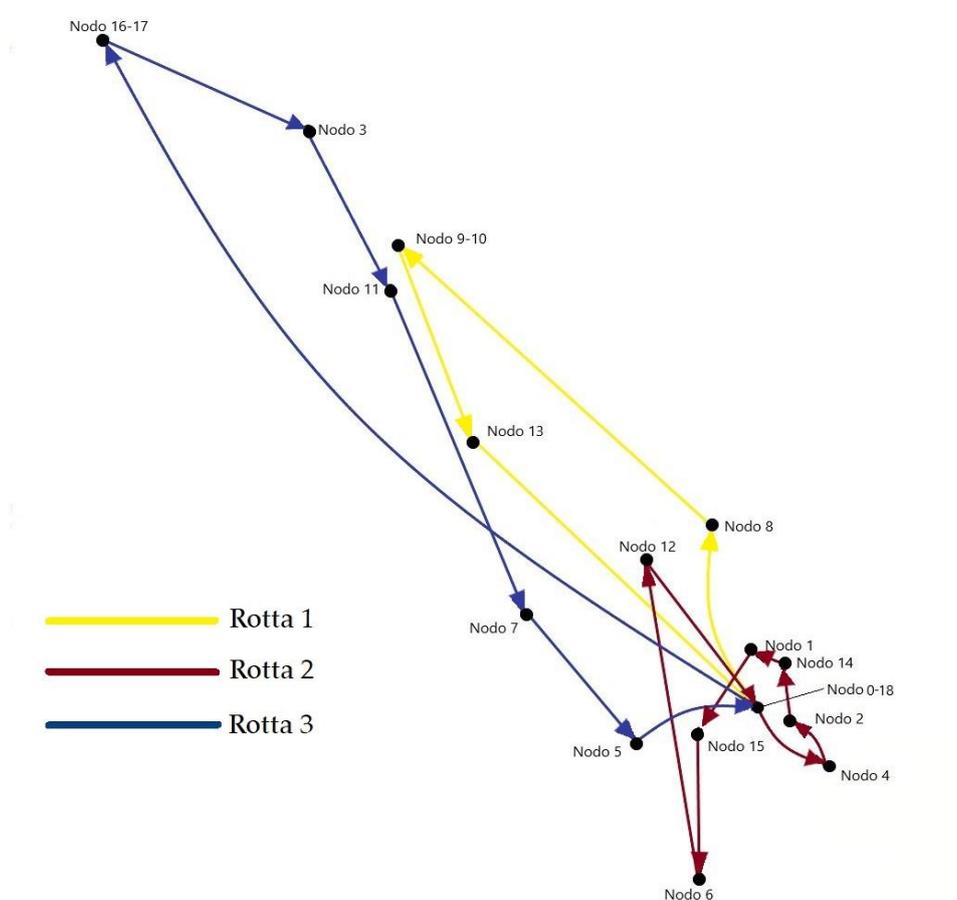


Fig. 4.6. Rotte dell'istanza Tre Elle Intermedio con tempo limite di un minuto

La soluzione col tempo limite di 5 minuti

La soluzione restituita è ammissibile, ma con un gap di ottimalità del 15,02%.

Costo totale	1264,2
Distanza totale percorsa	1228,3

Le rotte trovate, rappresentate in Fig. 4.7, sono:

Rotta veicolo 1: Nodo 0, Nodo 17, Nodo 16, Nodo 13, Nodo 8, Nodo 18.

Rotta veicolo 2: Nodo 0, Nodo 12, Nodo 6, Nodo 4, Nodo 2, Nodo 14, Nodo 1, Nodo 18.

Rotta veicolo 3: Nodo 0, Nodo 9, Nodo 10, Nodo 3, Nodo 11, Nodo 7, Nodo 5, Nodo 15, Nodo 18.

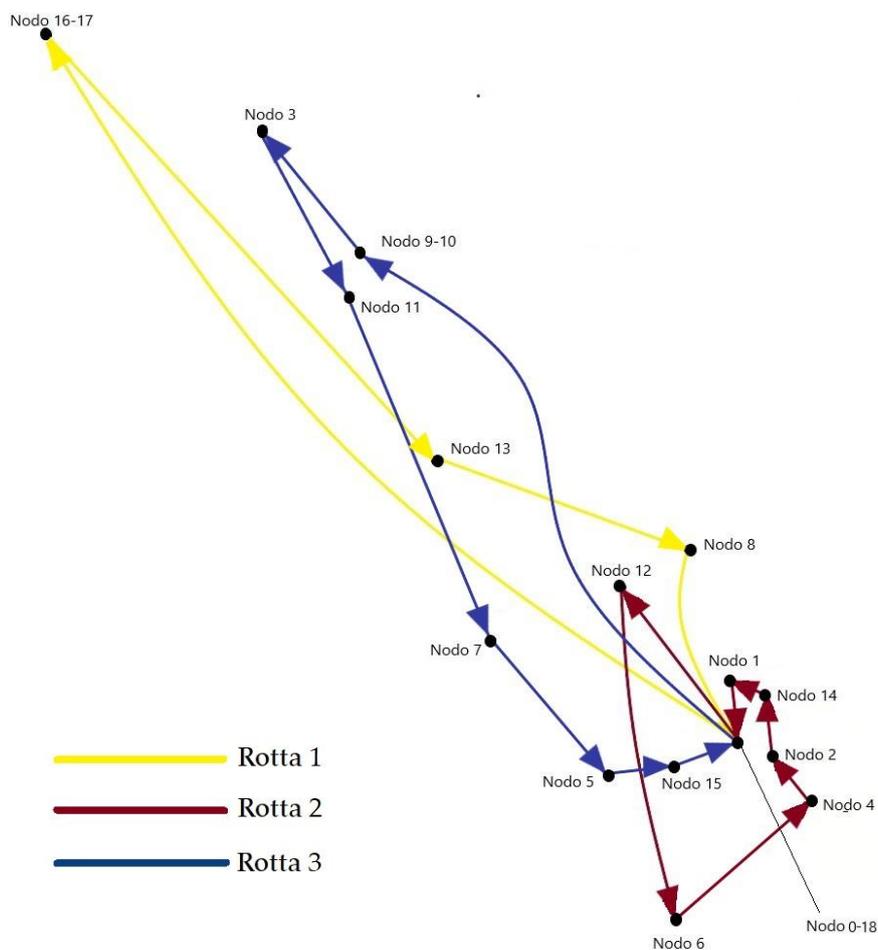


Fig. 4.7. Rotte dell'istanza Tre Elle Intermedio con tempo limite di cinque minuti

La soluzione col tempo limite di 10 minuti

La soluzione restituita secondo il solver è sicuramente ammissibile, ma non ha garanzie di ottimalità. Osservando il caso seguente, ovvero quello senza limiti di tempo, noteremo che in realtà questa è una soluzione ottima.

Costo totale	1099,10
Distanza totale percorsa	1230,60

Le rotte trovate, rappresentate in Fig. 4.8, sono:

Rotta veicolo 1: Nodo 0, Nodo 11, Nodo 16, Nodo 17, Nodo 13, Nodo 18.

Rotta veicolo 2: Nodo 0, Nodo 8, Nodo 10, Nodo 9, Nodo 3, Nodo 12, Nodo 6, Nodo 4, Nodo 2, Nodo 14, Nodo 1, Nodo 18.

Rotta veicolo 3: Nodo 0, Nodo 15, Nodo 5, Nodo 7, Nodo 18.

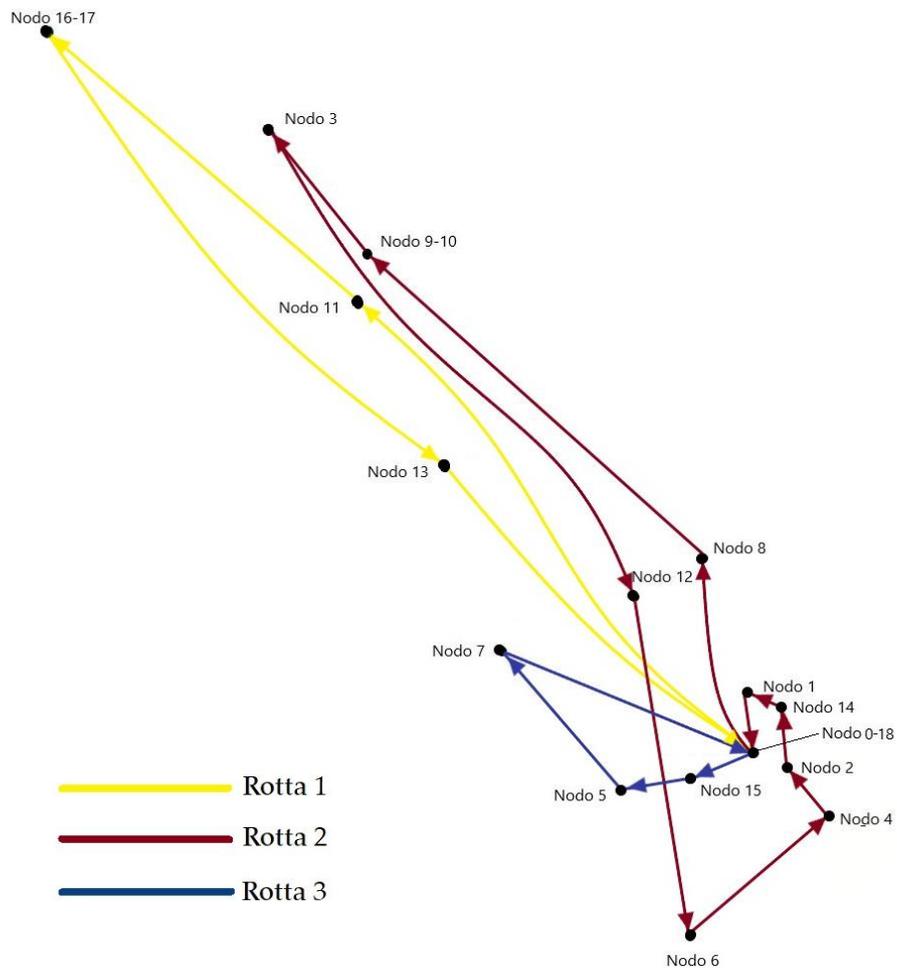


Fig. 4.8. Rotte dell'istanza Tre Elle Intermedio con tempo limite di cinque minuti

La soluzione ottima senza limiti di tempo

Quando non vengono impostati limiti di tempo il risolutore si ferma solo quando ha verificato di aver trovato una soluzione esatta.

Costo totale	1099,1
Distanza totale percorsa	1230,6

Si può notare che in realtà la soluzione ottima era stata già ottenuta dopo 10 minuti, ma che CPLEX non sia riuscito, in questo tempo, a certificarne l'ottimalità.

Questa soluzione è stata restituita dopo 37' e 55''.

La variabile L_i

I valori della variabile L_i sono indicati nella tabella 4.23

La variabile T_i

I valori della variabile T_i sono indicati nella tabella 4.23.

	L_i^1	L_i^2	L_i^3	T_i^1	T_i^2	T_i^3
Nodo 0	170	1010	1580	0	0	0
Nodo 1	0	910	0	0	450	0
Nodo 2	0	810	0	0	425	0
Nodo 3	0	830	0	0	117	0
Nodo 4	0	990	0	0	410	0
Nodo 5	0	0	680	0	0	23
Nodo 6	0	660	0	0	2390	0
Nodo 7	0	0	0	0	0	42
Nodo 8	0	910	0	0	30	0
Nodo 9	0	1220	0	0	91	0
Nodo 10	0	710	0	0	86	0
Nodo 11	120	0	0	72	0	0
Nodo 12	0	780	0	0	283	0
Nodo 13	450	0	0	247	0	0
Nodo 14	0	1110	0	0	433	0

Nodo 15	0	0	1480	0	0	16
Nodo 16	570	0	0	120	0	0
Nodo 17	500	0	0	130	0	0
Nodo 18	0	0	0	456	457	72

Tabella 4.23. Valori delle variabili L e T trovate dal risolutore nella soluzione ottima.

Confronto delle soluzioni

Mettendo a confronto le soluzioni trovate con i diversi tempi di esecuzione si può analizzare come, nonostante la distanza totale percorsa sia diminuita di poco, il costo è variato notevolmente (Fig. 4.9). Infatti, ad esempio, confrontando il primo ed il secondo caso si nota che il costo diminuisce del 10,4% mentre la distanza del 5,0%. La differenza diventa ancora più evidente tra il primo e il terzo caso in quanto il costo diminuisce del 22,0%, mentre la distanza percorsa del 4,8%. Come si può vedere, tra il secondo caso ed il terzo c'è una riduzione del costo ma un aumento della distanza percorsa. Questo perché trovandoci davanti ad una flotta eterogenea il modello tiene conto delle caratteristiche tecniche di ogni veicolo.

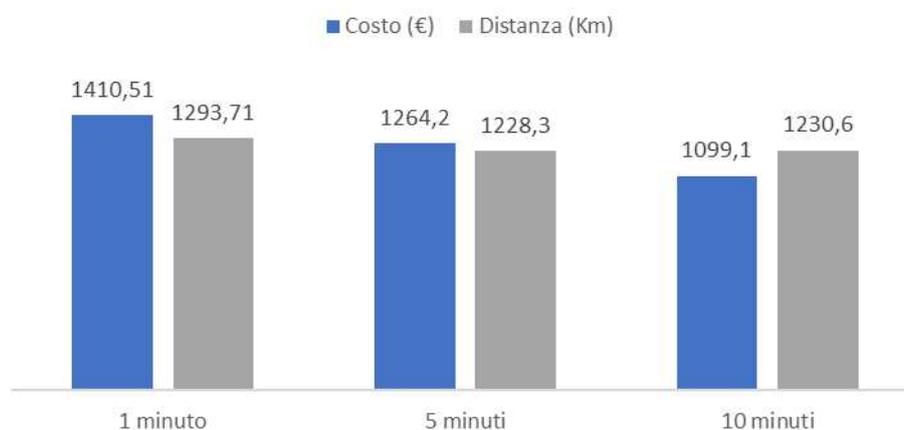


Fig. 4.9. Confronto tra le soluzioni ottenute in un minuto, cinque minuti e dieci minuti

Infatti, analizzando le figure 4.6, 4.7 e 4.8, vediamo che inizialmente il modello sfrutta molto il veicolo 3 (rotta 3) che tra tutti è quello con capacità di carico maggiore. Quest'ultimo però, è anche il veicolo con i consumi maggiori, quindi il modello cerca di ridurne l'utilizzo.

4.3.3 Istanza “Tre Elle completo”

Il caso studio fornito dall'azienda è molto grande e per questo, a causa delle capacità computazionali limitate della macchina, è risultato difficile trovare una soluzione ottima.

I dati

Questa istanza, come già detto, è composta da 33 nodi più i due relativi al deposito (Nodo 0 per il pick-up e Nodo 34 per il delivery) per un totale di 35 nodi. Essendo alcune informazioni cambiate, rispetto alle due istanze precedenti, è riportata una tabella con i relativi dati (tabella 4.25). Rimangono però le relazioni tra il Nodo 4 ed il Nodo 2 e tra il Nodo 4 ed il Nodo 1.

Come già evidenziato nel capitolo 3, questo modello è stato formulato introducendo le seguenti variabili decisionali: x_{ij}^k , L_i^k e T_i^k .

La variabile binaria x_{ij}^k dipende dalla tripletta i , j e k . Nell'istanza che stiamo considerando le variabili i e j possono assumere 35 valori diversi, mentre k può assumerne 3. Questo porta ad avere $35 * 35 * 3 = 3675$ variabili x_{ij}^k .

La variabile L_i^k , invece, dipende dalla coppia i e k . Nell'istanza che stiamo considerando la variabile i può assumere 35 valori diversi, mentre k può assumerne 3. Questo porta ad avere $35 * 3 = 105$ variabili L_i^k .

Anche la variabile T_i^k dipende dalla coppia i e k . Questo porta ad avere $35 * 3 = 105$ variabili T_i^k .

Quindi in totale, come ha mostrato anche CPLEX (fig. 4.6), questa istanza ha 3885 variabili.

Analizziamo ora il numero di vincoli. Va svolta un'analisi, simile alla precedente, su tutti e diciotto i vincoli del modello. Per rendere la lettura più immediata questo calcolo è stato riassunto nella tabella 4.24.

Vincolo	Riferimento e relativi range		Numero vincoli
Vincolo 1	4.2	$\forall i \in C0$	33
Vincolo 2	4.3	$\forall k \in K, \quad \forall (i,j) \in R$	$3 * 2 = 6$
Vincolo 3	4.4	$\forall k \in K$	3
Vincolo 4	4.5	$\forall k \in K$	3
Vincolo 5	4.6	$\forall k \in K, \quad \forall j \in C1$	$3 * 33 = 102$
Vincolo 6	4.7	$k \in K, \quad i \in C2$	$3 * 34 = 102$
Vincolo 7	4.8	$k \in K, \quad j \in C1$	$3 * 34 = 102$
Vincolo 8	4.9	x	1
Vincolo 9	4.10	$\forall k \in K$	3
Vincolo 10	4.11	$\forall i \in C0, \quad k \in K$	$3 * 33 = 99$
Vincolo 11	4.12	$\forall k \in K, \quad (i,j) \in R$	$3 * 2 = 6$
Vincolo 12	4.13	$\forall k \in K, \quad i \in C2, \quad j \in C1$	$3 * 34 * 34 = 3468$
Vincolo 13	4.14	$\forall k \in K, \quad i \in C2, \quad j \in C1$	$3 * 34 * 34 = 3468$
Vincolo 14	4.15	$\forall k \in K, \quad \forall i \in C2$	$3 * 34 = 102$
Vincolo 15	4.16	$\forall k \in k$	3
Vincolo 16	4.17	$\forall k \in k$	3
Vincolo 17	4.18	$k \in K, \quad i \in C2, \quad j \in C1$	$3 * 34 * 34 = 3468$
Vincolo 18	4.19	$k \in K, \quad i \in C2, \quad j \in C1$	$3 * 34 * 34 = 3468$

Tabella 4.24. Calcolo del numero di vincoli del modello

Andando a sommare i valori trovati possiamo determinare che i vincoli di questo modello sono 14443. I dati relativi al numero di variabili e vincoli sono riportati anche dallo stesso CPLEX nel settore *statistics* (Fig. 4.10).

Risulta comprensibile che, in un'istanza con 3385 variabili e 14443 vincoli, il tempo per raggiungere una soluzione ottima possa essere elevato, soprattutto se la macchina utilizzata non ha alte prestazioni.

Statistic	Value
▼ Cplex	
Constraints	14443
▼ Variables	3885
Binary	3675
Other	210

Fig. 4.10. Numero di vincoli e variabili dell'istanza Tre Elle completo

La soluzione col tempo limite di 5 minuti

La soluzione restituita è ammissibile, ma ha un gap di ottimalità del 66,31%.

Costo totale 799.89

Distanza totale percorsa 696.5

Le rotte, rappresentate graficamente in Fig. 4.11, sono:

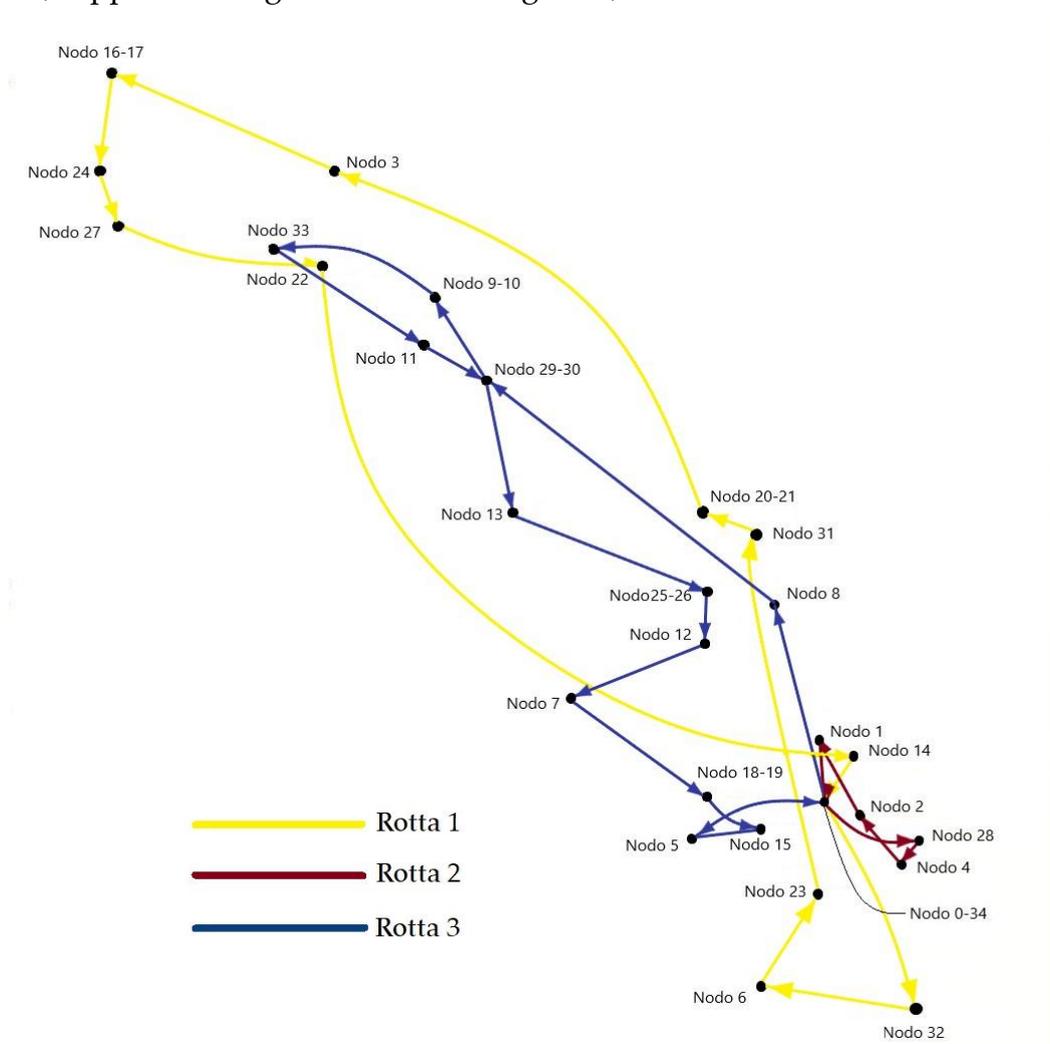


Fig. 4.11. Rotte dell'istanza Tre Elle Completo trovate in cinque minuti

Rotta veicolo 1: Nodo 0, Nodo 32, Nodo 6, Nodo 23, Nodo 31, Nodo 21, Nodo 20, Nodo 3, Nodo 17, Nodo 16, Nodo 24, Nodo 27, Nodo 22, Nodo 14, Nodo 34.

Rotta veicolo 2: Nodo 0, Nodo 28, Nodo 4, Nodo 2, Nodo 1, Nodo 34.

Rotta veicolo 3: Nodo 0, Nodo 8, Nodo 30, Nodo 9, Nodo 10, Nodo 33, Nodo 11, Nodo 29, Nodo 13, Nodo 25, Nodo 26, Nodo 12, Nodo 7, Nodo 18, Nodo 19, Nodo 15, Nodo 5, Nodo 34.

La soluzione col tempo limite di 10 minuti

La soluzione restituita è ammissibile, ma con un gap di ottimalità del 65,46%.

Costo totale	788,84
Distanza totale percorsa	699,29

Le rotte, rappresentate graficamente in Fig. 4.12, sono:

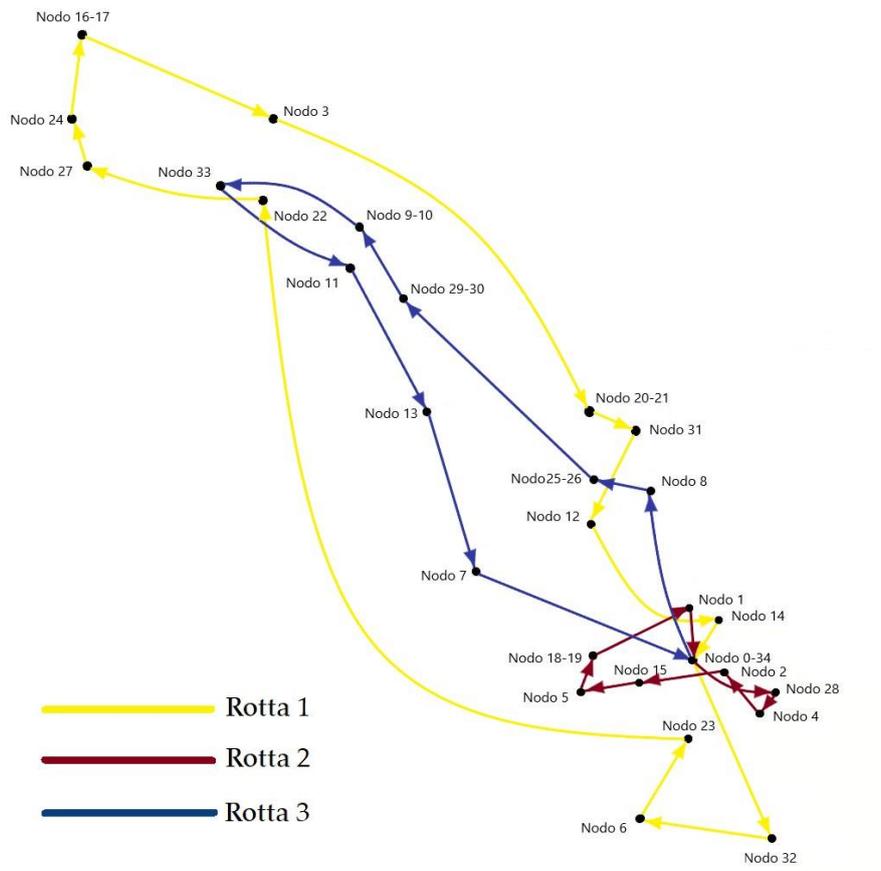


Fig. 4.12. Rotte dell'istanza Tre Elle Completo trovate in dieci minuti

Rotta veicolo 1: Nodo 0, Nodo 32, Nodo 6, Nodo 23, Nodo 22, Nodo 27, Nodo 24, Nodo 16, Nodo 17, Nodo 3, Nodo 20, Nodo 21, Nodo 31, Nodo 12, Nodo 14, Nodo 34.

Rotta veicolo 2: Nodo 0, Nodo 28, Nodo 4, Nodo 2, Nodo 15, Nodo 5, Nodo 19, Nodo 18, Nodo 1, Nodo 34.

Rotta veicolo 3: Nodo 0, Nodo 8, Nodo 25, Nodo 26, Nodo 29, Nodo 30, Nodo 9, Nodo 10, Nodo 33, Nodo 11, Nodo 13, Nodo 7, Nodo 34.

La soluzione col tempo limite di un'ora

La soluzione restituita è ammissibile, ma con un gap di ottimalità del 44,79%.

Costo totale	507,75
Distanza totale percorsa	463,69

Le rotte, rappresentate graficamente in Fig. 4.13, sono:

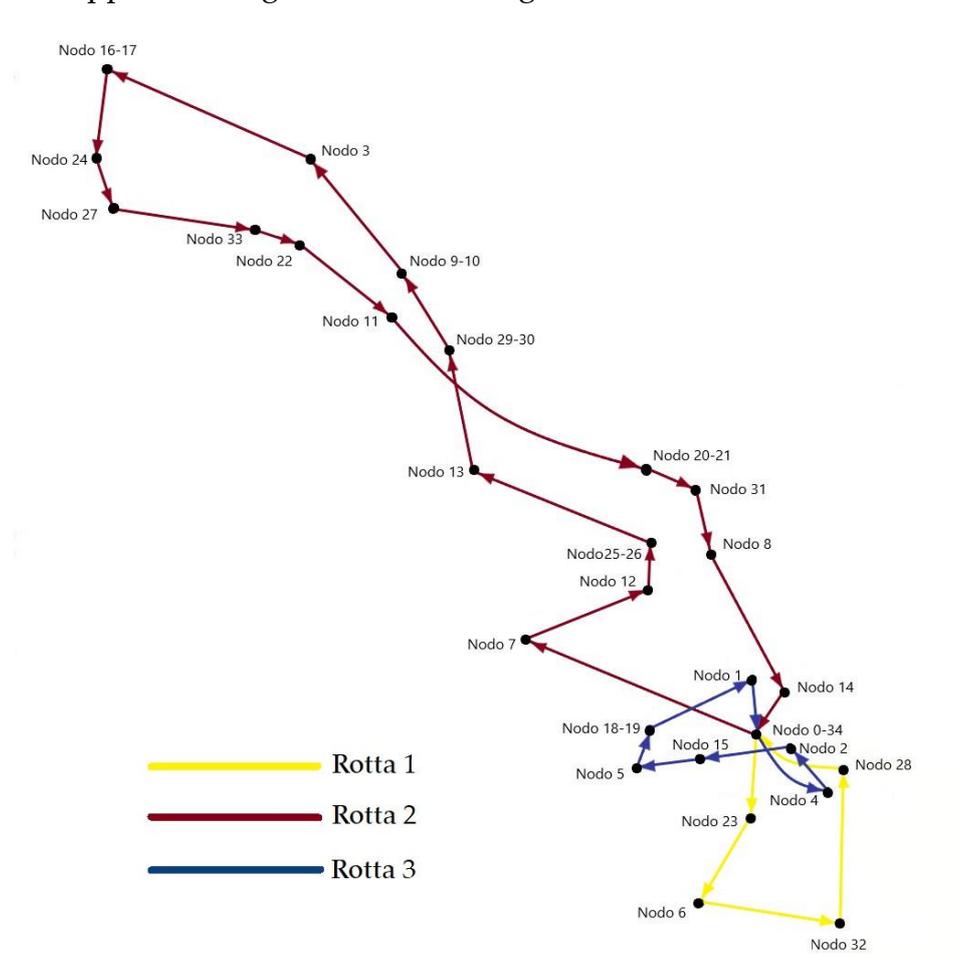


Fig. 4.13. Rotte dell'istanza Tre Elle Completo trovate in un'ora

Rotta veicolo 1: Nodo 0, Nodo 23, Nodo 6, Nodo 32, Nodo 28, Nodo 34.

Rotta veicolo 2: Nodo 0, Nodo 7, Nodo 12, Nodo 26, Nodo 25, Nodo 13, Nodo 30, Nodo 29, Nodo 10, Nodo 9, Nodo 3, Nodo 17, Nodo 16, Nodo 24, Nodo 27, Nodo 33, Nodo 22, Nodo 11, Nodo 21, Nodo 20, Nodo 31, Nodo 8, Nodo 14, Nodo 34.

Rotta veicolo 3: Nodo 0, Nodo 4, Nodo 2, Nodo 15, Nodo 5, Nodo 18, Nodo 19, Nodo 1, Nodo 34.

La soluzione col tempo limite di due ore

La soluzione restituita è ammissibile, ma ha un gap di ottimalità del 43,89%. A causa dell'elevato costo computazionale e delle caratteristiche tecniche della macchina il solver non riesce ad elaborare il modello per più di due ore.

Costo totale	506.8
Distanza totale percorsa	462.82

Le rotte trovate dal solver sono:

Rotta veicolo 1: Nodo 0, Nodo 23, Nodo 6, Nodo 32, Nodo 28, Nodo 34.

Rotta veicolo 2: Nodo 0, Nodo 7, Nodo 12, Nodo 25, Nodo 26, Nodo 13, Nodo 11, Nodo 22, Nodo 33, Nodo 27, Nodo 24, Nodo 16, Nodo 17, Nodo 3, Nodo 9, Nodo 10, Nodo 29, Nodo 30, Nodo 21, Nodo 20, Nodo 31, Nodo 8, Nodo 14, Nodo 34.

Rotta veicolo 3: Nodo 0, Nodo 4, Nodo 2, Nodo 15, Nodo 5, Nodo 19, Nodo 18, Nodo 1, Nodo 34.

Queste rotte, che rispettano i vincoli di precedenza (il nodo 4 è pick-up dei nodi 2 ed 1), sono graficamente rappresentata in fig. 4.14.

La variabile L_i

I valori della variabile L_i sono indicati nella tabella 4.25

La variabile T_i

I valori della variabile T_i sono indicati nella tabella 4.25.

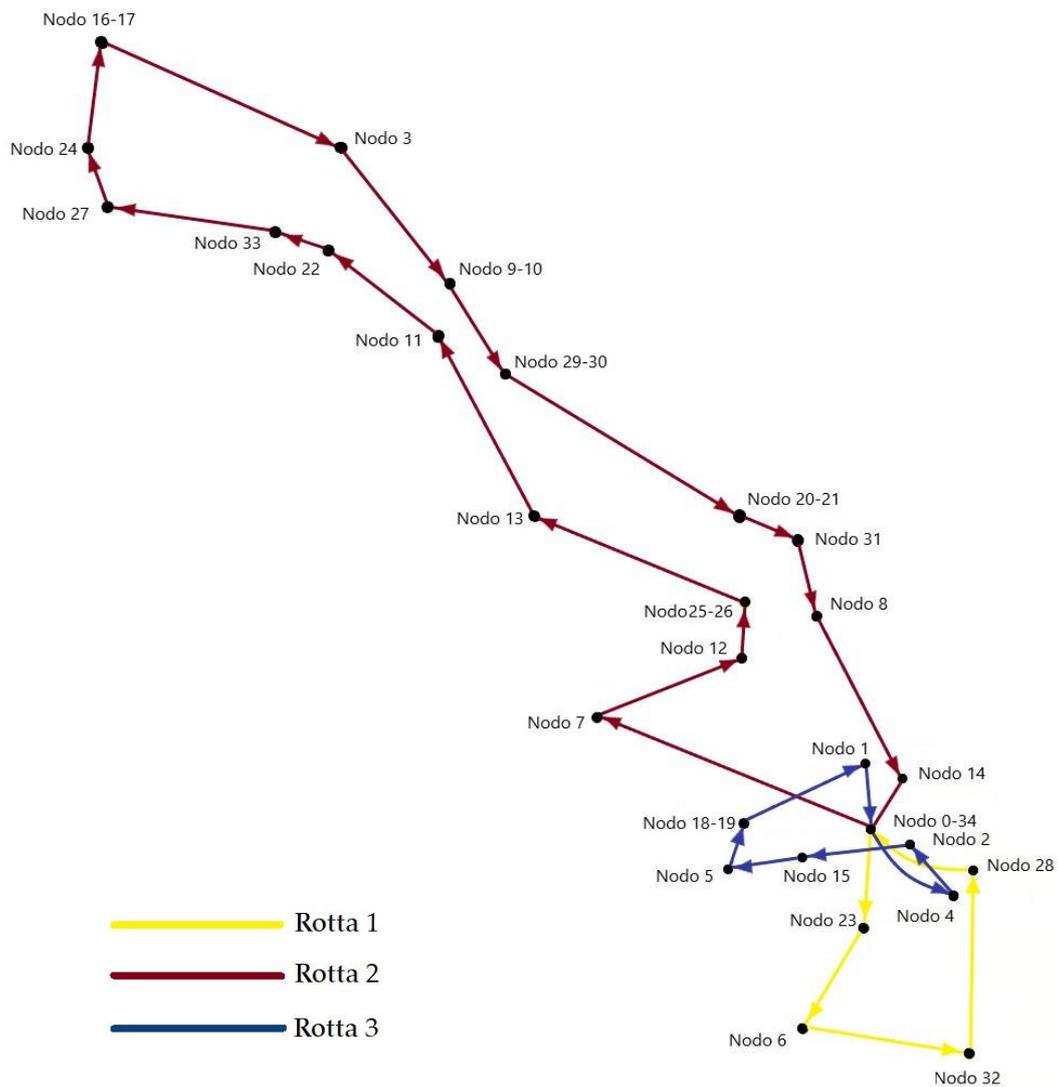


Fig. 4.14. Rappresentazione grafica delle rotte trovate nell'istanza Tre elle completa con time-limit di due ore.

	q_i	s_i	T_i^1	T_i^2	T_i^3	L_i^1	L_i^2	L_i^3
Nodo 0	2455	0	0	0	0	440	1135	880
Nodo 1	-200	15	0	0	73,45	0	0	480
Nodo 2	-180	5	0	0	26,83	0	0	1030
Nodo 3	-90	10	0	278	0	0	325	0
Nodo 4	330	15	0	0	6,69	0	0	1210
Nodo 5	-300	5	0	0	46,49	0	0	630
Nodo 6	-120	10	23,722	0	0	420	0	0
Nodo 7	-200	10	0	22,39	0	0	935	0
Nodo 8	-100	10	0	413,37	0	0	445	0

Nodo 9	200	5	0	305,77	0	0	525	0
Nodo 10	-200	15	0	310,77	0	0	325	0
Nodo 11	-50	20	0	143,3	0	0	765	0
Nodo 12	-50	10	0	40,51	0	0	885	0
Nodo 13	-50	30	0	86,96	0	0	815	0
Nodo 14	50	10	0	444,41	0	0	495	0
Nodo 15	-100	5	0	0	39,68	0	0	930
Nodo 16	50	5	0	239,98	0	0	485	0
Nodo 17	-70	5	0	244,98	0	0	415	0
Nodo 18	150	5	0	0	63,21	0	0	680
Nodo 19	-100	10	0	0	53,21	0	0	530
Nodo 20	100	10	0	390,5	0	0	425	0
Nodo 21	-50	5	0	385,5	0	0	325	0
Nodo 22	70	5	0	177,6	0	0	835	0
Nodo 23	100	5	6,63	0	0	540	0	0
Nodo 24	-100	5	0	218,73	0	0	0	0
Nodo 25	50	10	0	53,071	0	0	435	0
Nodo 26	-70	5	0	63,07	0	0	935	0
Nodo 27	-150	5	0	212,74	0	0	865	0
Nodo 28	-120	3	73	0	0	100	535	0
Nodo 29	100	5	0	336,52	0	0	425	0
Nodo 30	-50	10	0	341,52	0	0	375	0
Nodo 31	120	5	0	401,09	0	0	545	0
Nodo 32	-200	5	50,978	0	0	220	0	0
Nodo 33	-150	5	0	188,15	0	0	685	0
Nodo 34	-	5	82,93	455,82	91,37	0	0	0

Tabella 4.25. Valori delle variabili L e T trovate dal risolutore nella soluzione ottima.

Capitolo 5

Conclusioni e sviluppi futuri

In questa tesi, si è voluto aumentare l'efficienza e la competitività dei servizi di trasporto in un caso reale e, contemporaneamente, anche ridurre il costo complessivo imputabile alle attività di trasporto che tipicamente l'azienda in esame, la Tre Elle, deve sostenere. Questo è stato possibile studiando un problema di ottimizzazione appartenente alla famiglia dei problemi di Vehicle Routing ed in particolare, un problema di Pick-up e Delivery con Finestre Temporali (TW). Le ipotesi alla base di questa tipologia di problemi sono che:

1. Ogni veicolo venga usato una sola volta nell'orizzonte di pianificazione;
2. Ogni veicolo parte dal deposito e ivi ritorna;
3. La rete è composta da nodi di pick-up e nodi di delivery;
4. Ogni nodo è caratterizzato da una TW;
5. Ogni nodo può essere servito una sola volta,
6. La durata delle rotte non può superare il tempo limite concordato con l'autista.

Con questo studio, oltre alla possibilità di trasferire la merce dal deposito ad un nodo di delivery o da un nodo di pick-up al deposito si possono effettuare trasporti tra due nodi diversi dal deposito dell'azienda stessa. Questa condizione è necessaria perché la Tre Elle è un'azienda metalmeccanica che segue un prodotto nel suo intero ciclo di vita e che per alcune lavorazioni si appoggia anche ad aziende terze. Per questo non è sufficiente considerare il trasporto dall'azienda verso i clienti e dai fornitori verso la Tre Elle, ma bisogna occuparsi anche di altri trasporti come:

- Trasporto da un terzista ad un altro;
- Trasporto da un terzista ad un cliente finale.

Il modello di Programmazione Lineare Mista Intera così formulato è stato validato su un insieme di istanze di piccole dimensioni e successivamente sperimentato sul caso di studio reale e su un insieme di istanze di medie dimensioni ricavate da questo. Solo sul caso di studio reale, che presenta oltre 30 nodi, il risolutore non è stato in grado di trovare la soluzione ottima in due ore di computazione ma ha restituito solo una soluzione ammissibile.

In tal senso, un primo importante sviluppo futuro di questo lavoro di testi potrebbe riguardare la progettazione di approcci euristici o meta-euristici che siano in grado di ottenere una buona soluzione ammissibile (anche migliore di quella ottenuta dal risolutore) su un caso come quello reale. Infine, sfruttando la formulazione matematica proposta, si potrebbero progettare anche degli approcci matheuristici.

Riferimenti Bibliografici

- [1] Bertoni G., Penati M.E, 1992. L'automazione industriale. Dal mulino da seta alla fabbrica automatica. Esculapio, Bologna, 240.
- [2] Laporte G., Toth P. & Vigo D. Vehicle routing: historical perspective and recent contributions. *EURO J Transp Logist* 2, 1–4 (2013).
- [3] Dantzig G., Fulkerson R. e Johnson S. (1954). Solution of a Large-Scale Traveling-Salesman Problem.
- [4] Braekers K., Ramaekers K., & Van Nieuwenhuysse I. (2016). The vehicle routing problem: State of the art classification and review. *Computers & Industrial Engineering*, 99, 300-313.
- [5] Zhang H., Ge H., Yang J. et al. Review of Vehicle Routing Problems: Models, Classification and Solving Algorithms. *Arch Computat Methods Eng* (2021).
- [6] Dror M, Trudeau P (1989) Savings by split delivery routing. *Transp Sci* 23(2):141–145
- [7] Gutierrez A., Dieulle L., Labadie, N., & Velasco, N. (2018). A Hybrid metaheuristic algorithm for the vehicle routing problem with stochastic demands. *Computers & Operations Research*, 99, 135-147.
- [8] Mosheiov G. (1998). Vehicle routing with pick-up and delivery: tour-partitioning
- [9] Toth P. e Vigo D. VRP with Backhauls. *The Vehicle Routing Problem*. 2002, 195-224 heuristics. *Computers & Industrial Engineering*, 34(3), 669-684.
- [10] Labadie N., Prins C. e Prodhon C. (2016) *Metaheuristics for vehicle routing problems*. Wiley, Hoboken.
- [11] Lahyani R., Khemakhem M., Semet F. (2015) Rich vehicle routing problems: from a taxonomy to a definition. *Eur J Oper Res* 241:1–14.

- [12] Elshaer R., & Awad H. (2020). A taxonomic review of metaheuristic algorithms for solving the vehicle routing problem and its variants. *Computers & Industrial Engineering*, 140, 106242.
- [13] Desaulniers G., Lavigne J., e Soumis F. (1998). Multi-depot vehicle scheduling problems with time windows and waiting costs. *European Journal of Operational Research*, 111(3), 479-494.
- [14] Li C. L., Simchi-Levi, D. e Desrochers M. (1992). On the distance constrained vehicle routing problem. *Operations research*, 40(4), 790-799.
- [15] Hadjar A. e Soumis F. (2009). Dynamic window reduction for the multiple depot vehicle scheduling problem with time windows. *Computers & Operations Research*, 36(7), 2160-2172.
- [16] G. B. Dantzig and J. H. Ramser. (1959) The Truck Dispatching Problem. *Management Science*, vol. 6, issue 1, 80-91.
- [17] Konstantakopoulos G.D., Gayialis S.P. e Kechagias E.P. (2020). Vehicle routing problem and related algorithms for logistics distribution: a literature review and classification. *Oper Res Int J*.
- [18] Kallehauge B. Larsen J., Madsen O. B. e Solomon M. M. (2005). Vehicle routing problem with time windows. In *Column generation* (pp. 67-98). Springer, Boston, MA.
- [19] S. Kumar and R. Panneerselvam, "A Survey on the Vehicle Routing Problem and Its Variants," *Intelligent Information Management*, Vol. 4 No. 3, 2012, pp. 66-74.
- [20] Y. Agarwal, K. Mathur and H. M. Salkin, "A set-partitioning-based algorithm for the vehicle routing problem," *Networks*, Vol. 19, No. 7, 1989, pp. 731-749.

[21] Archetti, C., Speranza, M.G. A survey on matheuristics for routing problems. *EURO J Comput Optim* 2, 223–246 (2014).

[22] Krist V. Gernaey, Jakob K. Huusom, Rafiqul Gani, *Computer Aided Chemical Engineering*. Elsevier, Volume 37, 2015, Pages 35-40.