Sommario

[Le 3 alternative 2](#_Toc130459652)

[Caso applicativo 3](#_Toc130459653)

[Stand Alone (SA) 4](#_Toc130459654)

[Neighbors Savings (NS) 6](#_Toc130459655)

[Extreme Neighbors Savings (ENS) 11](#_Toc130459656)

[Conclusione 13](#_Toc130459657)

[Considerazioni su altri esempi 14](#_Toc130459658)

# Le 3 alternative

Una volta risolto il problema di allocazione della capacità, è necessario definire una regola che possa ripartire il costo per il servizio (in questo caso di trasporto) tra tutti i clienti in modo equo. Una delle principali problematiche nella logistica dei trasporti è definire quanto del costo totale del percorso complessivo di un determinato veicolo è giusto imputare ad ogni cliente da servire.

Trovare una soluzione che avalli tutte le caratteristiche che possano influenzare la ripartizione del costo non è di facile risoluzione. Lo stato dell’arte, se così si può intendere, in letteratura consiste nell’applicazione dello Shapley value, teorizzato dall’omonimo nel 1953. Lo Shapley Value si presenta come il contributo marginale medio di un giocatore i in una specifica coalizione, rispetto le possibili permutazioni dei giocatori e permette di calcolare la convenienza che questi può avere nell’entrare in un “gioco”. Nell’ambito del commesso viaggiatore (TSP), lo Shapley value permette di calcolare una giusta ripartizione dei costi, per ogni giocatore i, basandosi appunto sul loro contributo marginale. Nonostante questa sia la soluzione migliore dal punto di vista della qualità, è invece la peggiore per quanto riguarda la complessità. Lo Shapley Value infatti opera su una complessità computazionale dell’ordine di O(n!), il che vuol dire che se già il numero di clienti da servire è composto da dieci o più elementi, questo rende possibile calcolarlo in tempo polinomiale, ma solo nei casi in cui il numero di giocatori non è elevato. Sono numerosi gli autori che hanno elaborato soluzioni alternative che si propongono di cercare un’approssimazione che sia un buon compromesso tra qualità e complessità, ma non saranno oggetto di trattazione in questo articolo.

Questo lavoro di ricerca propone una strategia di allocazione dei costi al di fuori del paradigma dello Shapley value. Infatti è stato necessario trovare una strategia che si applicasse ad ogni singolo problema di TSP, la cui la totalità delle soluzioni viene generata dal singolo problema di VRP iniziale. Nel singolo TSP ogni cliente è composto da un punto di ritiro Pi e da uno di consegna Di. Inoltre come fattore ulteriore da considerare la quantità di clienti da servire essendo elevata rende necessaria una semplificazione.

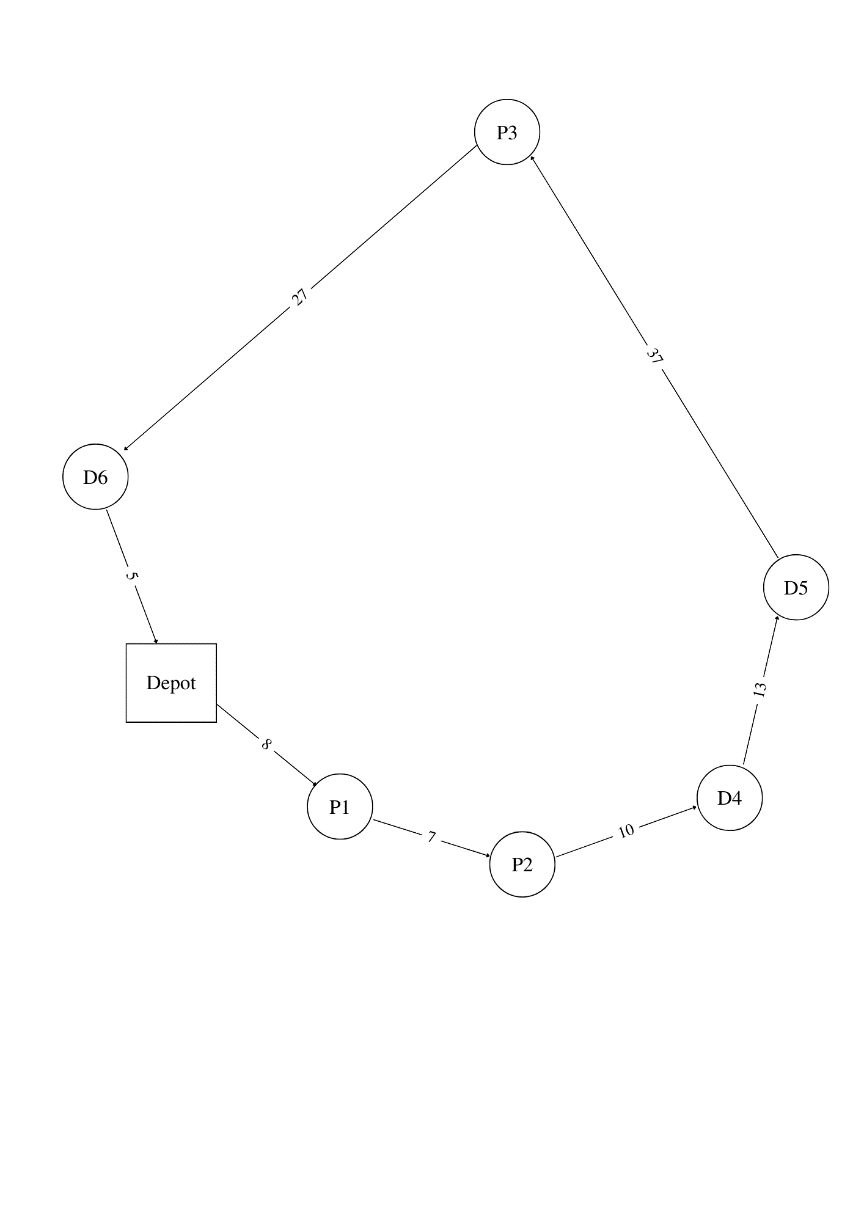
Per l’allocazione dei costi sono state individuate 3 regole distinte, ciascuna con aspetti positivi e negativi, in grado di compensarsi tra di loro. È stata individuata come soluzione finale una media ponderata dei risultati:

* Regola *Stand Alone*
* Regola *Neighbors Savings*
* Regola *Extreme Neighbors Savings*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Dep (o/d)** | **P1** | **P2** | **P3** | **D4** | **D5** | **D6** |
| **Dep**  **(o/d)** | / | 8 | 13 | 31 | 19 | 28 | 5 |
| **P1** | 8 | / | 7 | 22 | 14 | 15 | 11 |
| **P2** | 13 | 7 | / | 25 | 10 | 27 | 14 |
| **P3** | 31 | 22 | 25 | / | 41 | 37 | 27 |
| **D4** | 19 | 14 | 10 | 41 | / | 13 | 28 |
| **D5** | 28 | 15 | 27 | 37 | 13 | / | 19 |
| **D6** | 5 | 11 | 14 | 27 | 28 | 19 | / |

## Caso applicativo

Nei paragrafi successivi, una volta argomentate le tre regole di allocazione dei costi, queste verranno applicate al seguente esempio. Il problema del VRP ha generato un sotto problema di TSP, definendo il percorso minimo per visitare tutti i clienti. Si considerino quindi n clienti (n = 1, 2, 3), un deposito, nodi pickup Pi (i = 1, 2, 3), nodi delivery Di (i = 4, 5, 6) e una matrice delle distanze che esprime il costo come somma di distanza e durata tra i nodi.

La soluzione che l’algoritmo descritto nell’*appendice A* propone di servire ogni cliente i dell’istanza con un unico veicolo k, e risulta essere quella rappresentata in figura, con un costo totale per servire tutti i clienti i, pari a:

* : uguale a 1 se il veicolo k attraversa arco (i, j) e 0 altrimenti.
* : costo percorrenza arco (i, j)

L’obiettivo è quindi ripartire in maniera equa il costo totale tra i 3 clienti i. Per farlo useremo le 3 regole di allocazione, ma prima definiamo:

* n = numero di clienti da servire
* P = {1,...,n} : set di nodi pickup
* D = {n+1 ,..., 2n} : set di nodi delivery
* o = deposito come partenza
* d = deposito come arrivo
* Ci,n+i = costo dell’arco tra il nodo Pi e il nodo Dn+i

## Stand Alone (SA)

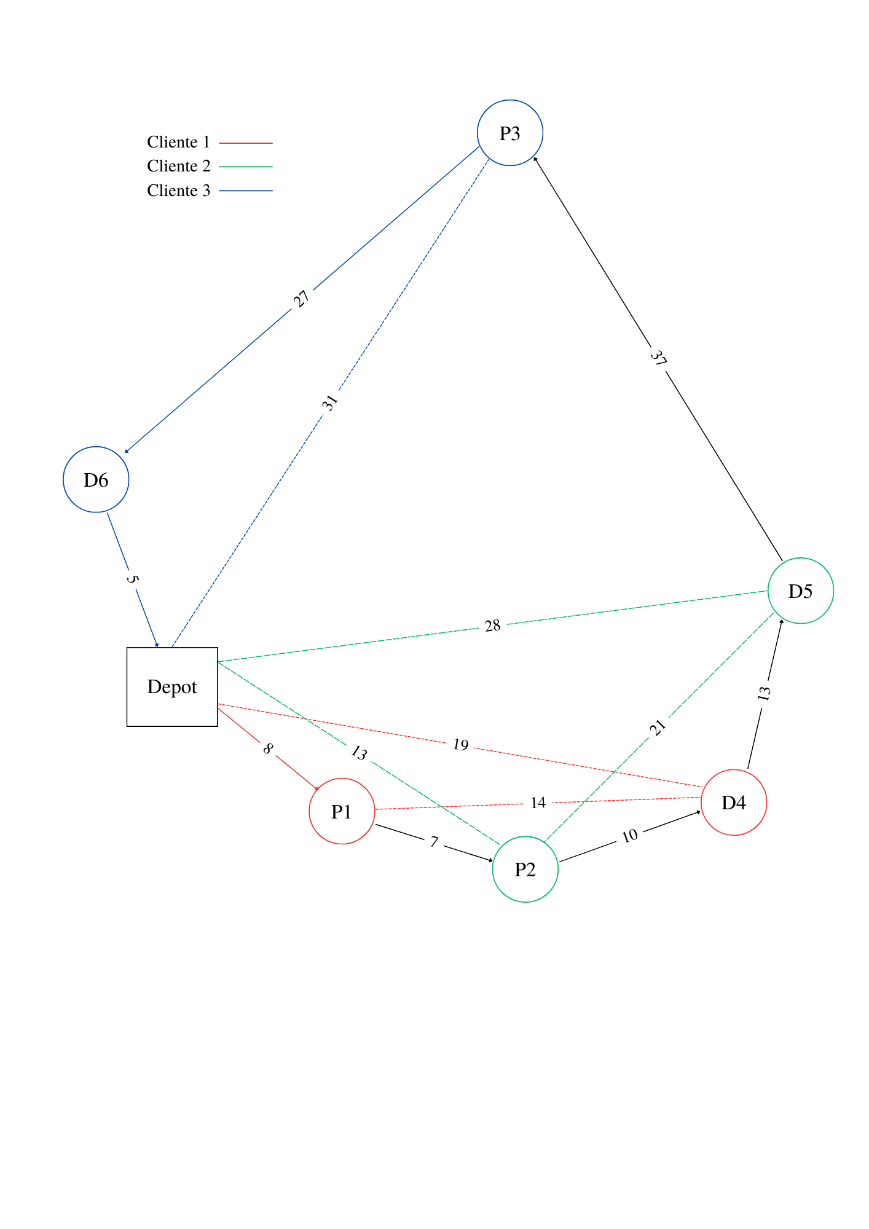
Immagine che contiene diagramma

Descrizione generata automaticamenteCome punto di partenza è stato preso come benchmark di riferimento una regola che potesse garantire una prima soluzione ammissibile, da migliorare in seguito. La regola *Stand Alone* assegna i costi agli n clienti in proporzione al percorso che origina dal deposito verso il nodo di prelievo Pi, consegna al nodo Di, quindi rientra al deposito. Si è optato per uno scenario non cooperativo, nel quale ogni singolo tour venisse considerato a sé stante, senza considerare eventuali riduzioni di costi imputabili alla vicinanza tra i clienti da servire. L’ipotesi alla base consiste nella normalizzazione di tutte le distanze percorse, alle quali è associato un costo proporzionale sia alla distanza che alla durata.

È possibile definire il costo associato ad ogni cliente i (con i che vada 1 a n), come distanza di ogni cliente (composto da un nodo pickup e un nodo delivery) dal deposito.

Una volta calcolati tutti i costi associati alle distanze per ogni cliente i, questi vanno normalizzati al fine di individuare la porzione del costo totale da imputare a ciascuno di loro.

Per ogni cliente i, la porzione di costo del costo totale, secondo la regola *Stand alone* risulta:

* **Esempio:**

Si riprenda l’esempio riportato nel “*Caso applicativo*”, nel quale l’obiettivo è ripartire il costo totale per servire tutti i clienti i:

Si utilizza la regola *Stand Alone* per imputare a ciascun cliente i, una quota del costo totale, in funzione della distanza di ogni cliente dal deposito.

Mentre il costo imputabile a ciascun cliente risulta essere

|  |  |  |
| --- | --- | --- |
| **Cliente 1** | **Cliente 2** | **Cliente 3** |
|  |  |  |
|  |  |  |
|  |  |  |

In questo scenario, basato unicamente sulle distanze tra il deposito e i punti di prelievo e consegna di ciascun cliente, sono i clienti 2 e 3 quelli che si vedono imputare un costo maggiore. In particolare, spostando il focus sul cliente 2, si evince come a questo venga imputato il costo maggiore, senza considerare la sua distanza tra i nodi prelievo e consegna del cliente 1. Per cui è pure vero che la sua distanza totale dal deposito (espressa in termini di costo) è 68, che è la maggiore tra le tre, ma è vero anche che questo si trovi a metà strada tra P1 e D4. Non viene percepito il beneficio derivante dal fatto di essere vicini ad altri clienti da servire.

Come si evince, la regola *Stand Alone* ha il grande vantaggio della semplicità di utilizzo e della bassa complessità computazionale, ma questo rappresenta anche il suo principale svantaggio, data la troppa semplicità nel non considerare la riduzione dei costi che si potrebbe ottenere per i clienti vicini tra loro. Questo aspetto viene considerato nelle altre due regole proposte in seguito.

## Neighbors Savings (NS)

Per compensare la mancanza della regola *Stand Alone*, che non considera la vicinanza tra i clienti, l’idea alla base è caratterizzata, partendo da un cliente i, dal calcolo del suo contributo rispetto al costo totale, escludendolo dalla distanza totale percorsa “*tagliando*” e “*ricucendo*” il passaggio. Dato il cliente i, vengono rimossi tutti gli archi adiacenti ai nodi Pi e Di del tour considerato, e aggiunti tutti gli archi che permettono al tour di bypassare questi nodi. Così facendo, maggiore sarà la differenza tra la vicinanza dei nodi e l’effettiva distanza dal deposito, minore risulterà il costo marginale nel visitare Pi e Di.

* : rappresenta la distanza totale percorsa da un veicolo k che visita il cliente i nel suo tour.
* : rappresenta la somma di tutte le distanze degli archi percorsi per visitare sia Pi che Di, come deviazione dal tour del veicolo k, senza passare per il cliente i. In sintesi è la somma di tutte le distanze associate agli spigoli del grafo, attraversati da k, che hanno Pi o Di come nodo in partenza o in arrivo.

Vengono presi, per ogni cliente i, , per calcolare per ogni cliente i. Si faccia riferimento alla variabile booleana del generico arco

Nota: consente di risolvere il caso in cui si abbia il nodo pickup e delivery dello stesso cliente, uno di seguito all’altro in soluzione.

* : rappresenta la somma di tutte le distanze associate ai bordi del grafo che il veicolo k dovrebbe attraversare per completare il suo tour senza visitare Pi e Di e rispettando l’ordine di visita dei nodi della soluzione originale. Riprendendo la notazione precedente, ma con l che rappresenta i nodi antecessori, mentre h i successori:

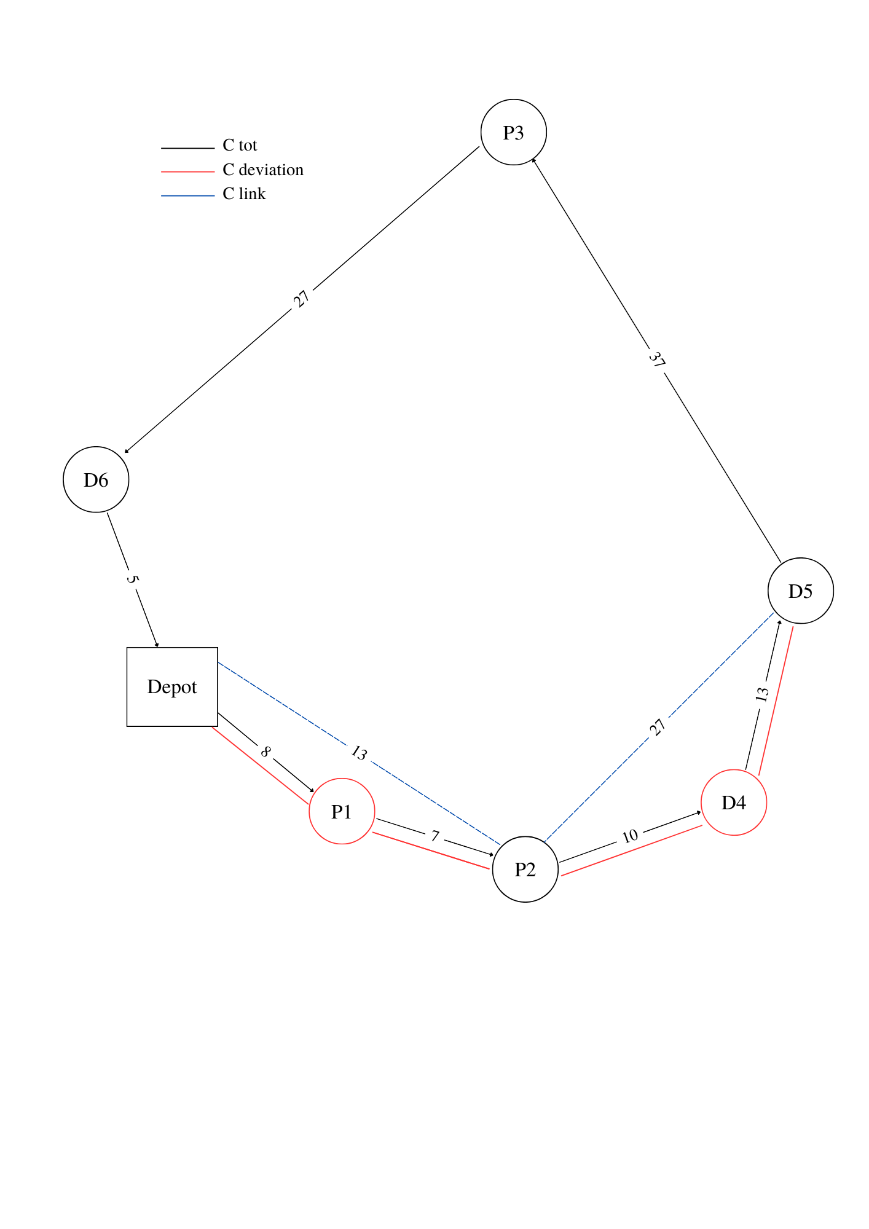
Nota: e consentono di risolvere il caso in cui si abbia il nodo pickup e delivery dello stesso cliente, uno di seguito all’altro in soluzione.

* : rappresenta il contributo della distanza marginale di i, essendo .

Una volta calcolati i contributi della distanza marginale di tutti i clienti, questi devono essere normalizzati e in base a questa normalizzazione, viene assegnato a ciascuno una porzione di costo.

* **Esempio:**

Viene ripreso lo stesso esempio utilizzato per spiegare la regola *Stand Alone* e sono applicate le formule descritte in precedenza per ognuno dei 3 clienti.

**Cliente 1**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Dep (o/d)** | **P1** | **P2** | **P3** | **D4** | **D5** | **D6** |
| **Dep**  **(o/d)** | / | 8 | 13 | 31 | 19 | 28 | 5 |
| **P1** | 8 | / | 7 | 22 | 14 | 15 | 11 |
| **P2** | 13 | 7 | / | 25 | 10 | 27 | 14 |
| **P3** | 31 | 22 | 25 | / | 41 | 37 | 27 |
| **D4** | 19 | 14 | 10 | 41 | / | 13 | 28 |
| **D5** | 28 | 15 | 27 | 37 | 13 | / | 19 |
| **D6** | 5 | 11 | 14 | 27 | 28 | 19 | / |

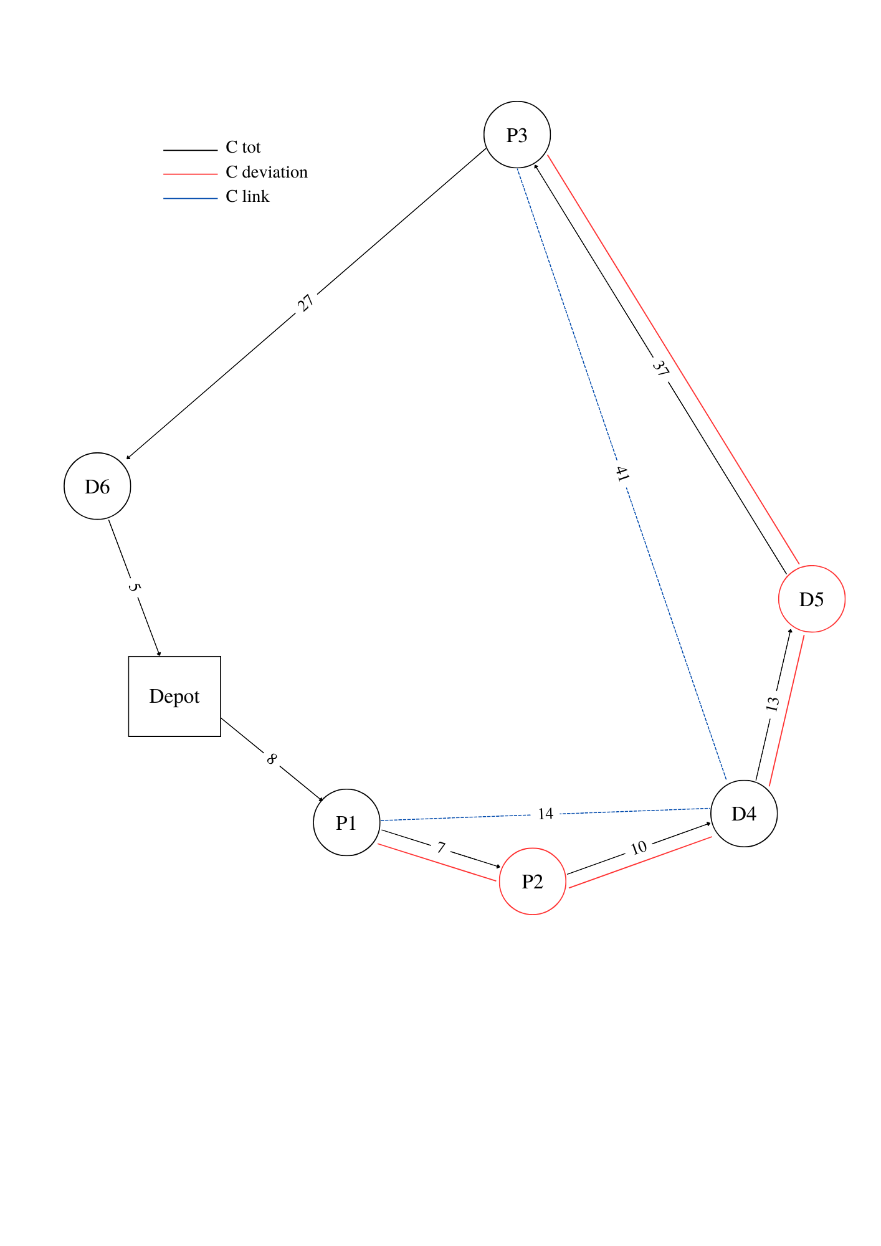
Prendiamo il cliente i = 1 ( e

*Costo totale (uguale per tutti)*

*Costo deviation*

*Costo link*

*Costo marginale*

**Cliente 2**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Dep (o/d)** | **P1** | **P2** | **P3** | **D4** | **D5** | **D6** |
| **Dep**  **(o/d)** | / | 8 | 13 | 31 | 19 | 28 | 5 |
| **P1** | 8 | / | 7 | 22 | 14 | 15 | 11 |
| **P2** | 13 | 7 | / | 25 | 10 | 27 | 14 |
| **P3** | 31 | 22 | 25 | / | 41 | 37 | 27 |
| **D4** | 19 | 14 | 10 | 41 | / | 13 | 28 |
| **D5** | 28 | 15 | 27 | 37 | 13 | / | 19 |
| **D6** | 5 | 11 | 14 | 27 | 28 | 19 | / |

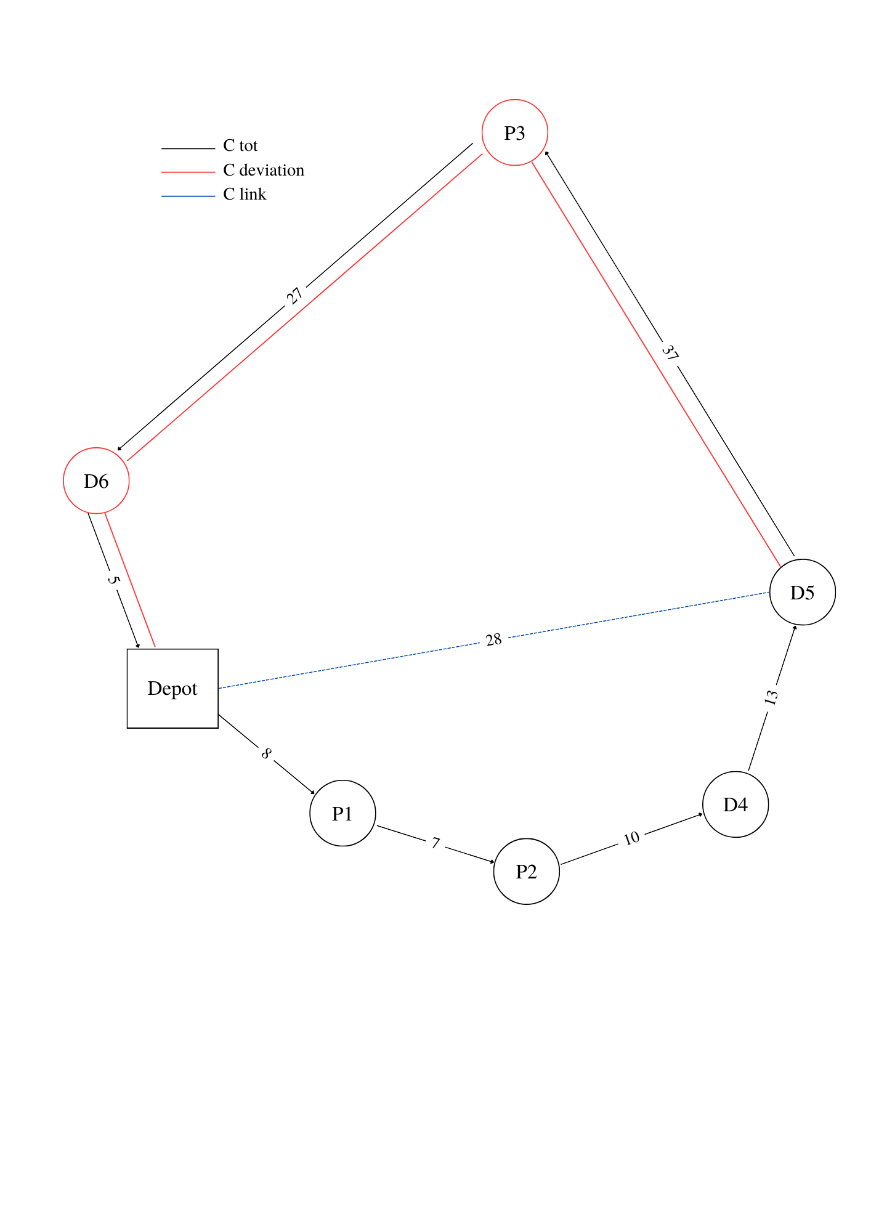
Prendiamo il cliente i = 2 ( e

*Costo totale (uguale per tutti)*

*Costo deviation*

*Costo link*

*Costo marginale*

**Cliente 3**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Dep (o/d)** | **P1** | **P2** | **P3** | **D4** | **D5** | **D6** |
| **Dep**  **(o/d)** | / | 8 | 13 | 31 | 19 | 28 | 5 |
| **P1** | 8 | / | 7 | 22 | 14 | 15 | 11 |
| **P2** | 13 | 7 | / | 25 | 10 | 27 | 14 |
| **P3** | 31 | 22 | 25 | / | 41 | 37 | 27 |
| **D4** | 19 | 14 | 10 | 41 | / | 13 | 28 |
| **D5** | 28 | 15 | 27 | 37 | 13 | / | 19 |
| **D6** | 5 | 11 | 14 | 27 | 28 | 19 | / |

Prendiamo il cliente i = 3 ( e

*Costo totale (uguale per tutti)*

*Costo deviation*

*Costo link*

*Costo marginale*

Una volta calcolati tutti i contributi della distanza marginale per ogni cliente i, questi vanno normalizzati al fine di individuare i costi da imputare a ciascuno di loro.

per poi calcolare il vero e proprio

|  |  |  |
| --- | --- | --- |
| **Cliente 1** | **Cliente 2** | **Cliente 3** |
|  |  |  |
|  |  |  |

In questo scenario, sono i clienti 1 e 2 quelli che si vedono imputare un costo maggiore, mentre a differenza del caso precedente, il cliente più lontano risente di una diminuzione di costo, in funzione della sua vicinanza relativa agli altri clienti da servire.

La problematica maggiore nella regola *Neighbors savings*, come si evince anche dall’esempio, consiste nel rischio di gravare troppo su quei clienti che si trovano più vicini al deposito. Questi, infatti, più si troveranno ad essere serviti all’interno di un percorso con una durata di percorrenza maggiore, più verranno svantaggiati in termini di allocazione di costi. Per bilanciare questo svantaggio, viene introdotta una terza regola. L’*Extreme Neighbors savings*.

## Extreme Neighbors Savings (ENS)

Come visto in precedenza, maggiore sarà il percorso che il veicolo k dovrà affrontare, maggiore sarà anche la quota di costo imputabile ai nodi più vicini. Per bilanciare l’aspetto negativo in questione, dato che il percorso effettuato da k corrisponde proprio a , questo viene tolto dall’equazione. Potrebbe verificarsi il caso in cui la somma delle distanze degli archi percorsi per visitare sia Pi che Di come deviazioni dal tour del veicolo k sia inferiore alla somma di tutte le distanze associate ai bordi del grafico. In altre parole, onde evitare che vi sia un costo del contributo marginale negativo e quindi impossibile, questo diventa:

* **Esempio:**

Viene ripreso lo stesso esempio precedente. Viene applicata la formula descritta per tutti e 3 i clienti.

**Cliente 1**

**Cliente 2**

**Cliente 3**

Una volta calcolati tutti i contributi della distanza marginale per ogni cliente i, questi vanno normalizzati al fine di individuare i costi da imputare a ciascuno di loro.

per poi calcolare il vero e proprio

|  |  |  |
| --- | --- | --- |
| **Cliente 1** | **Cliente 2** | **Cliente 3** |
|  |  |  |
|  |  |  |

Come è facile notare, l’aspetto negativo di questo metodo consiste nel considerare nulli tutti quei risultati che potrebbero far uscire un valore negativo. Secondo questa logica infatti il cliente 1 non dovrebbe pagare nulla per il servizio di prelievo e consegna, ma come già anticipato in precedenza, questa regola ha lo scopo di controbilanciare il difetto della *Neighbors savings* e l’effetto è la compensazione reciproca.

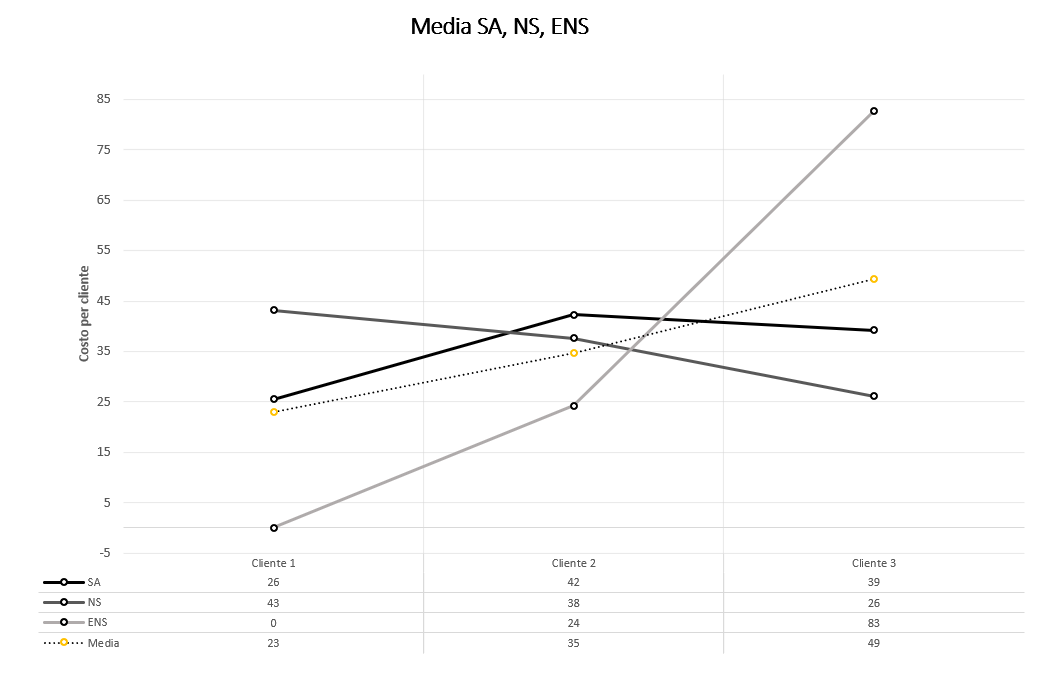
# Conclusione

Ognuna delle soluzioni proposte ha vantaggi e svantaggi. L’obiettivo prefissato è stato quello cercare un set di regole la cui unione potesse generare una soluzione ammissibile, di complessità computazionale non elevata. Infatti, questa soluzione risiede al di fuori del paradigma di Shapley che sarebbe stato uno strumento molto più preciso, ma ad un costo computazionale maggiore. Per giungere ad una conclusione unica, l’output della strategia proposta consiste in una media ponderata dei risultati delle tre regole.

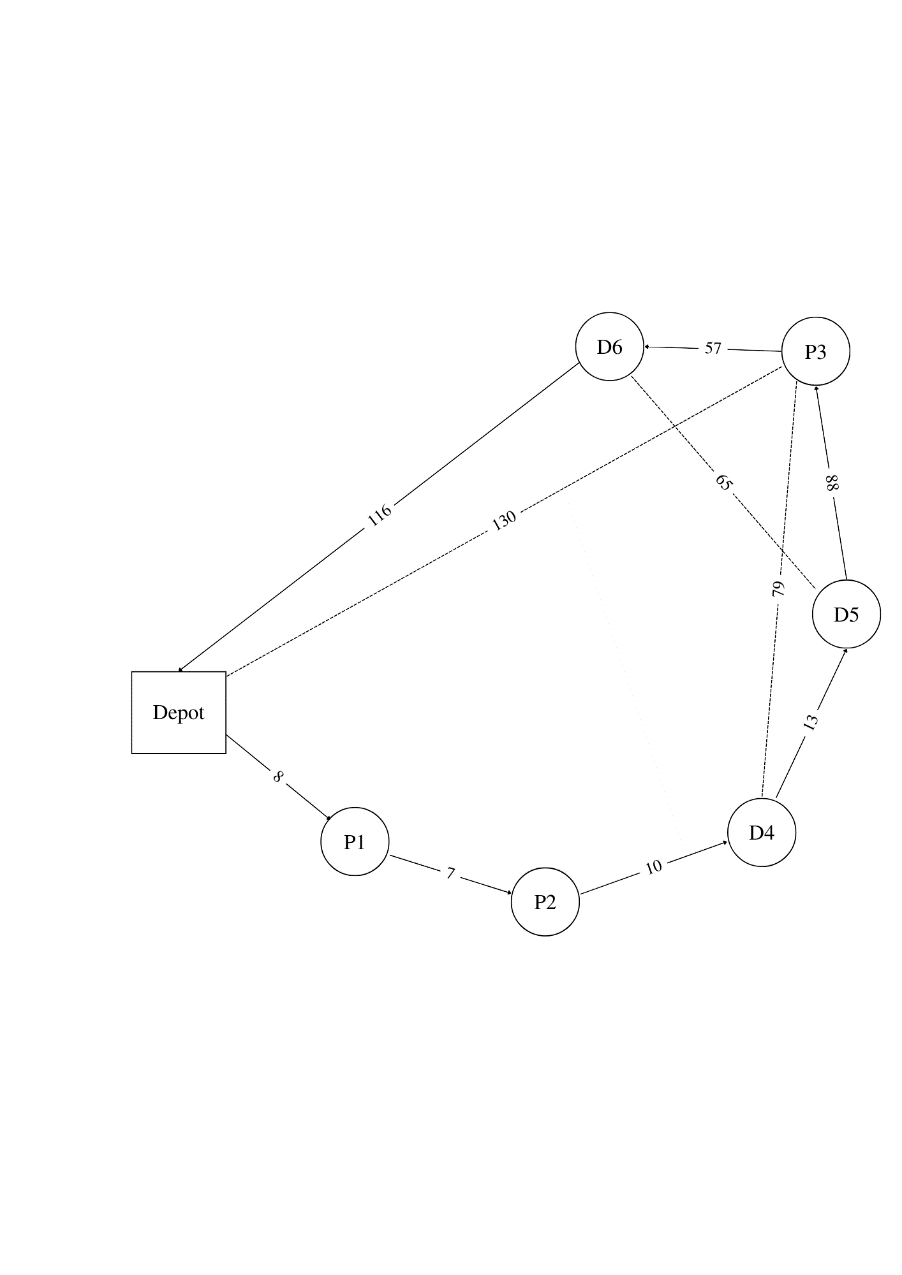
Riprendendo i tre diversi risultati dell’esempio e riassumendoli in una tabella è possibile trovare facilmente una media delle alternative proposte:

|  |  |  |  |
| --- | --- | --- | --- |
| **RULE** | **Cliente 1** | **Cliente 2** | **Cliente 3** |
| SA | 26 | 42 | 39 |
| NS | 43 | 38 | 26 |
| ENS | 0 | 24 | 83 |
| **Media** | **23** | **35** | **49** |

Per una maggiore comprensione, si riportano di seguito i risultati ottenuti sotto forma di grafico:

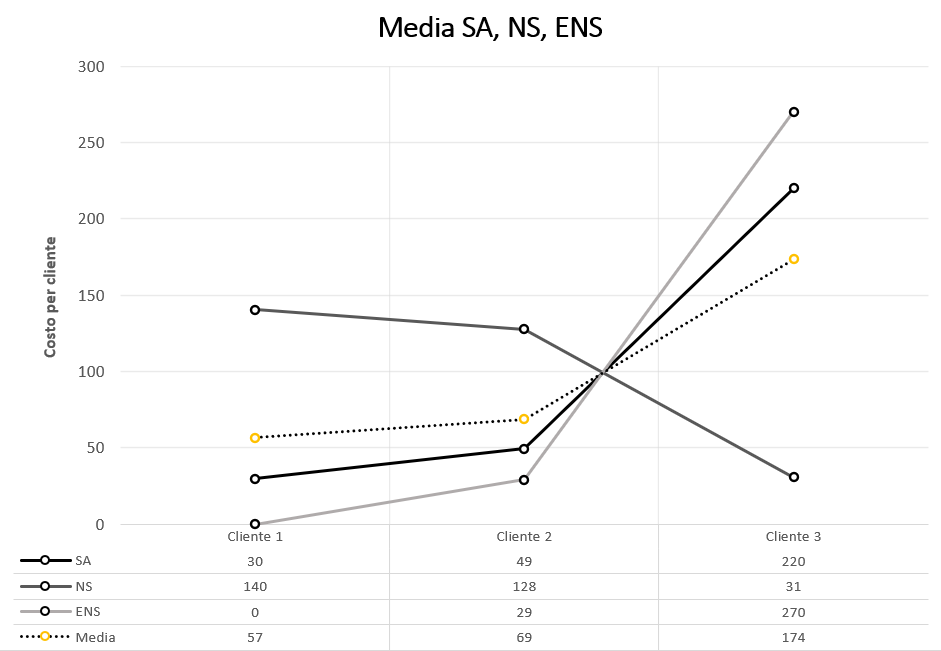


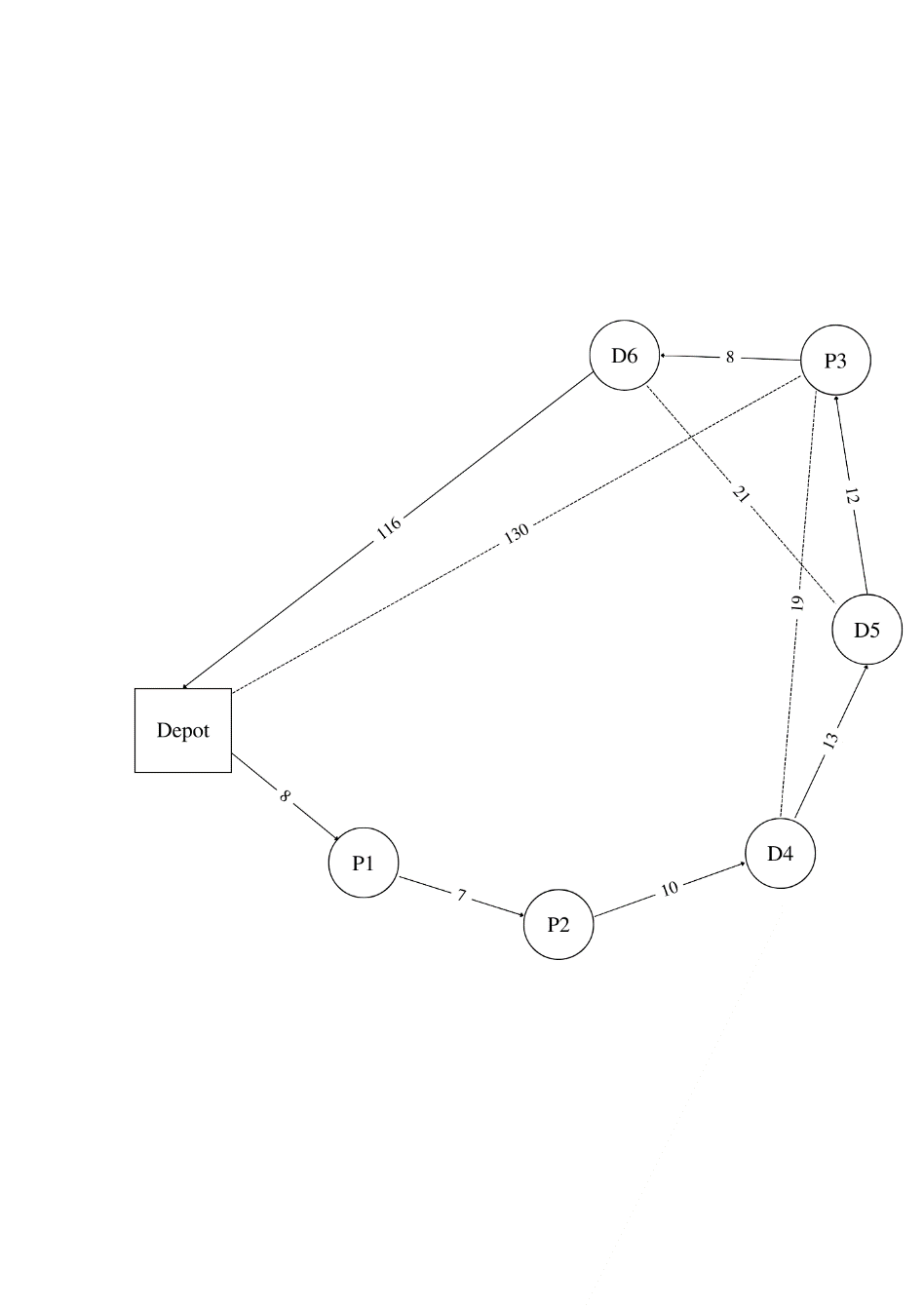
# Considerazioni su altri esempi

Per valutare la bontà dell’algoritmo, si prenda un caso simile al precedente, ma con il cliente 3 () posto ad una distanza di un ordine di grandezza maggiore rispetto alle altre.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Dep (o/d)** | **P1** | **P2** | **P3** | **D4** | **D5** | **D6** |
| **Dep**  **(o/d)** | / | 8 | 13 | 130 | 19 | 28 | 116 |
| **P1** | 8 | / | 7 | 22 | 14 | 15 | 11 |
| **P2** | 13 | 7 | / | 25 | 10 | 27 | 14 |
| **P3** | 31 | 22 | 25 | / | 79 | 88 | 57 |
| **D4** | 19 | 14 | 10 | 41 | / | 13 | 28 |
| **D5** | 28 | 15 | 27 | 37 | 13 | / | 65 |
| **D6** | 5 | 11 | 14 | 27 | 28 | 19 | / |

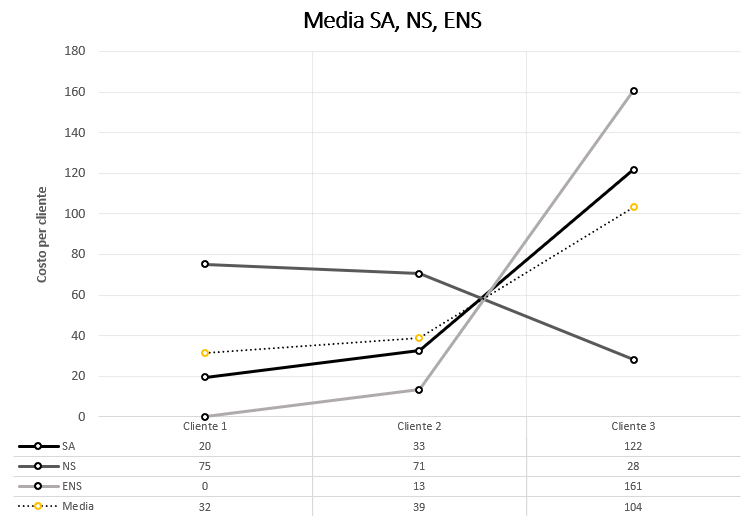
|  |  |  |  |
| --- | --- | --- | --- |
| **RULE** | **Cliente 1** | **Cliente 2** | **Cliente 3** |
| % SA | 10% | 16% | 74% |
| % NS | 47% | 43% | 10% |
| % ENS | 0% | 10% | 90% |
| **% Media** | **19%** | **23%** | **58%** |



Partendo dallo stesso esempio di partenza, si consideri il caso in cui, il cliente 3 è sempre posto a distanza maggiore dal deposito, ma è anche più vicino agli altri clienti.

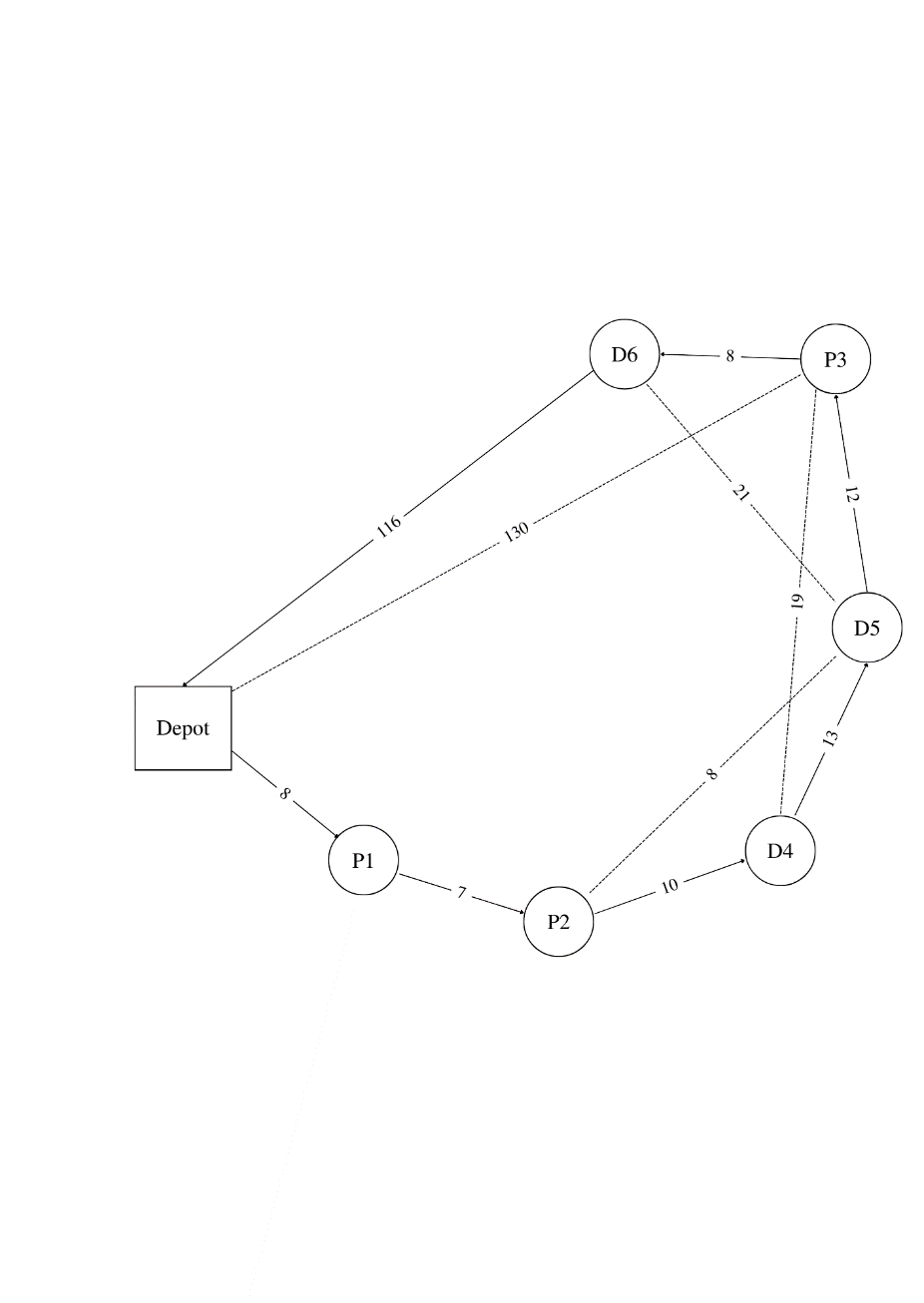
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Dep (o/d)** | **P1** | **P2** | **P3** | **D4** | **D5** | **D6** |
| **Dep**  **(o/d)** | / | 8 | 13 | 130 | 19 | 28 | 116 |
| **P1** | 8 | / | 7 | 22 | 14 | 15 | 11 |
| **P2** | 13 | 7 | / | 25 | 10 | 27 | 14 |
| **P3** | 31 | 22 | 25 | / | 19 | 12 | 8 |
| **D4** | 19 | 14 | 10 | 41 | / | 13 | 28 |
| **D5** | 28 | 15 | 27 | 37 | 13 | / | 21 |
| **D6** | 5 | 11 | 14 | 27 | 28 | 19 | / |

|  |  |  |  |
| --- | --- | --- | --- |
| **RULE** | **Cliente 1** | **Cliente 2** | **Cliente 3** |
| % SA | 11% | 19% | 70% |
| % NS | 43% | 41% | 16% |
| % ENS | 0% | 8% | 92% |
| **% Media** | **18%** | **22%** | **60%** |

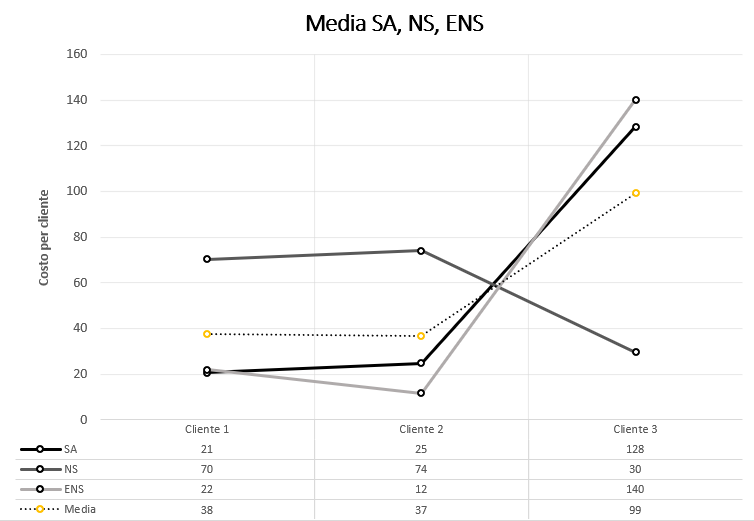


Da notare come, anche nel caso in cui la vicinanza tra il cliente 3 e gli altri, sia minore, il risultato rimane quasi costante. La regola *Extreme* *Neighbors savings*, “tira” l’aumento dei costi verso il cliente 3, insieme alla regola *Stand Alone*. Si prova a ridistribuire la situazione, facendo in modo che non vada a 0.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Dep (o/d)** | **P1** | **P2** | **P3** | **D4** | **D5** | **D6** |
| **Dep**  **(o/d)** | / | 8 | 13 | 130 | 19 | 28 | 116 |
| **P1** | 8 | / | 7 | 22 | 14 | 15 | 11 |
| **P2** | 13 | 7 | / | 25 | 10 | 8 | 14 |
| **P3** | 31 | 22 | 25 | / | 19 | 12 | 8 |
| **D4** | 19 | 14 | 10 | 41 | / | 13 | 28 |
| **D5** | 28 | 15 | 27 | 37 | 13 | / | 21 |
| **D6** | 5 | 11 | 14 | 27 | 28 | 19 | / |



|  |  |  |  |
| --- | --- | --- | --- |
| **RULE** | **Cliente 1** | **Cliente 2** | **Cliente 3** |
| % SA | 12% | 14% | 74% |
| % NS | 40% | 43% | 17% |
| % ENS | 13% | 7% | 80% |
| **% Media** | **22%** | **21%** | **57%** |



Per concludere queste considerazioni, anche in quest’ultimo caso non vi è una variazione troppo significativa della ripartizione del costo. In tutti i casi trattati, la media finale assume un andamento simile alla regola *Stand Alone*, indice del fatto la distanza tra il deposito e i clienti da servire è il parametro con peso maggiore. Ma questo viene smorzato dalle altre due regole, mantenendo il risultato finale sullo stesso ordine di grandezza e riuscendo nell’obiettivo trovare di una strategia di allocazione dei costi al di fuori del paradigma dello Shapley value che potesse essere consistente.