Traffic Engineering con Simplesso su Rete

Samuele Pilleri Jacopo Fidelio

Luglio 2021

Indice

Abstract

Il problema del Traffic Engineering consiste nel distribuire il traffico all'interno di una rete IP in maniera ottimale. Per far ciò occorre tener conto sia del protocollo di routing scelto, sia del meccanismo di forwarding usato dalla rete. Obiettivo di un buon protocollo di routing è quello di determinare un percorso (sequenza di nodi) attraverso la rete dalla sorgente alla destinazione.

Nelle reti che utilizzano Open Shortest Path First (OSPF, Dijkstra), l'instradamento avviene tramite shortest paths e il meccanismo di forwarding è vincolato a distribuire il traffico in maniera uniforme lungo i cammini minimi. Questi vincoli spesso rendono l'ottimizzazione impossibile. In questa tesina introdurremo la natura dei problemi ed illustreremo un meccanismo per risolverne uno in particolare, il multi-commodity min-cost flow, attraverso metodi basati sulla programmazione lineare.

1 Multi-commodity Flow problems

Col termine multi-commodity flow si identifica una famiglia di problemi di network flow che riguardano l'instradamento su rete di più flussi (commodity), aventi ciascuno una specifica domanda di traffico da un nodo sorgente ad un nodo destinazione. Nella fattispecie, il problema preso in esame richiede di minimizzare l'utilizzo massimo di ogni link attraversato da uno o più flussi, quindi è un problema di multi-commodity min-cost flow. Il fatto che i flussi non siano splittabili, come vedremo, lo rende $\mathcal{NP}$-completo.

2 Shortest Path

Poiché questi algoritmi si basano sul concetto di shortest path è innanzitutto necessario fornire una definizione appropriata di questo problema. Sia data una rete con topologia $G = (V, E)$, con $m = ||V||$ nodi e $n = ||E||$ archi diretti. Assumiamo l'esistenza della matrice di traffico T dove le entry $T(s_k, t_k) = d_k$ denotano la domanda di traffico dal nodo sorgente s al nodo destinazione t per le commodity $k \in \mathcal{K}$. Supponiamo che un'allocazione ottimale basata su alcune funzioni di costo a livello di rete produca un insieme di percorsi $\mathcal{P}_k$ per ogni commodity k, in modo che la larghezza di banda totale consumata da questi cammini sul link (i, j) sia c_{ij}. Si può notare che si possono ottenere le stesse performance in termini di larghezza di banda consumata da ogni link con un insieme di shortest paths formulati attraverso la risoluzione di un programma lineare e del suo duale.

Il programma lineare primale è così formulato:

$$
\begin{aligned}
\min \sum_{(i,j)\in E} \sum_{k\in\mathcal{K}} & d_k X_{ij}^k \\
& \text{soggetto a} \\
& \sum_{j:(j,i)\in E} X_{ij}^k - \sum_{j:(i,j)\in E} X_{ij}^k = 0 && i \neq s_k, t_k \quad k \in \mathcal{K} \\
& \sum_{j:(j,i)\in E} X_{ij}^k - \sum_{j:(i,j)\in E} X_{ij}^k = 1 && i = t_k \quad k \in \mathcal{K} \\
& \sum_{j:(j,i)\in E} X_{ij}^k - \sum_{j:(i,j)\in E} X_{ij}^k = -1 && i = s_k \quad k \in \mathcal{K} \\
& \sum_{k\in\mathcal{K}} d_k X_{ij}^k \leq c_{ij} && (i,j) \in E \\
& 0 \leq X_{ij}^k \leq 1 && (i,j) \in E \quad k \in \mathcal{K}
\end{aligned}
\tag{1}
$$

Dove X_{ij}^k è la frazione di traffico per commodity k che attraversa l'arco (i,j). Risolvere il programma lineare restituisce come risultato un'allocazione del traffico $\{X_{ij}^k\}$ che consuma al massimo c_{ij} larghezza di banda su ogni link (i,j). Per ottenere i pesi dei link per lo shortest path, bisogna risolvere il duale del programma lineare sopra discusso:

$$\begin{aligned}
\max \sum_{\substack{k\in\mathcal{K}\\ t\in V}} & U_{t_k}^k - \sum_{(i,j)\in E} c_{ij}W_{ij} \\
& \text{soggetto a} \\
& U_j^k - U_i^k \leq W_{ij} + 1 \qquad \forall k \in \mathcal{K}, (i,j) \in E \\
& W_{ij} \geq 0 \\
& U_{s_k}^k = 0
\end{aligned} \tag{2}$$

La variabile U_i^k viene ricavata sostituendo le due variabili duali $y_{s_k}^k$ e $y_{i_k}^k$:

$$U_i^k = y_{s_k}^k - y_{i_k}^k$$

Il duale ritorna un insieme di link pesati $\{W_{ij}\}$ dal quale può essere costruito un set di shortest paths consistente con le variabili di allocazione del traffico $\{X_{ij}^k\}$.

3 Multicommodity Min-Cost Flow

L'obiettivo è minimizzare l'utilizzazione massima dei link. Si hanno quindi:

- topologia di rete $G = (V, E)$
- capacità c_{ij} dei link $(i,j) \in E$
- insieme $\mathcal{K}$ di flussi sorgente/destinazione
  - d_k: domanda di traffico del flusso
  - s_k: sorgente del flusso
  - t_k: destinazione del flusso
- α: massima utilizzazione dei link

3.1 Formulazione PL

$$\min \alpha$$

soggetto a

$$\sum_{j:(i,j)\in E} d_k X_{ij}^k - \sum_{j:(j,i)\in E} d_k X_{ij}^k = \begin{cases} 0 & i \neq s_k, t_k \quad k \in \mathcal{K} \\ d_k & i = s_k \quad k \in \mathcal{K} \\ -d_k & i = t_k \quad k \in \mathcal{K} \end{cases} \tag{3}$$

$$\sum_{k\in\mathcal{K}} d_k X_{ij}^k \leq c_{ij}\alpha,$$

$$X_{ij}^k \geq 0$$

$$\alpha \geq 0$$

Con questa formulazione del problema possono esserci molte soluzioni con lo stesso valore di α, e a parità di α si vogliono scegliere le soluzioni che privilegiano l'uso di shortest paths.

Introducendo il parametro r si può aggiungere alla funzione di costo il termine $r \sum_{k\in\mathcal{K}} \sum_{(i,j)\in E} X_{ij}^k$ che esprime la lunghezza totale di tutti i cammini, con r piccolo a piacere per rendere questa metrica poco influente.

$$\min \alpha + r \sum_{k\in\mathcal{K}} \sum_{(i,j)\in E} d_k X_{ij}^k$$

soggetto a

$$\sum_{j:(i,j)\in E} d_k X_{ij}^k - \sum_{j:(j,i)\in E} d_k X_{ij}^k = \begin{cases} 0 & i \neq s_k, t_k \quad k \in \mathcal{K} \\ d_k & i = s_k \quad k \in \mathcal{K} \\ -d_k & i = t_k \quad k \in \mathcal{K} \end{cases} \tag{4}$$

$$c_{ij}\alpha - \sum_{k\in\mathcal{K}} d_k X_{ij}^k \geq 0$$

$$X_{ij}^k \geq 0$$

$$\alpha \geq 0$$

La formulazione duale dunque diventa la seguente:

$$
\begin{aligned}
\max \sum_{k \in \mathcal{K}} & d_k U_{t_k}^k \\
& \text{soggetto a} \\
& U_j^k - U_i^k \leq W_{ij} + r \qquad k \in \mathcal{K}, (i,j) \in E \\
& \sum_{(i,j) \in E} c_{ij} W_{ij} = 1, \\
& W_{ij} \geq 0 \\
& U_{s_k}^k = 0
\end{aligned} \tag{5}
$$

Si noti che la soluzione trovata è specifica per problemi formulati a partire dai flussi, mentre IP considera le destinazioni. Questo problema non è bloccante in quanto si può facilmente riformulare il modello per tenere conto di questo nuovo vincolo.

L'impedimento principale risiede nel fatto che non tutte le soluzioni potrebbero essere compatibili con la ripartizione equa sui camminimi minimi equivalenti utilizzata da OSPF. Tenere conto di questo vincolo e imporre la non-divisibilità dei flussi porta il problema ad essere $\mathcal{NP}$-completo.

4 Unsplittable MMCF

Per ottenere la non-divisibilità dei flussi occorre aggiungere un vincolo di interezza sulla variabile X_{ij}^k.

Non potendo dividere i singoli flussi con OSPF, è pratica diffusa nelle reti IP considerare piccole sottoreti come unità di indirizzamento. In questo modo è anche più semplice effettuare una stima dei flussi e calcolare un instradamento quasi-ottimale.

Nell'implementazione proposta ci avvaliamo di un solver che usa la tecnica branch and cut per ottenere l'interezza delle soluzioni.

Riferimenti bibliografici

[1] Ashwin Sridharan, R. Guerin e C. Diot. "Achieving near-optimal traffic engineering solutions for current OSPF/IS-IS networks". In: *IEEE INFOCOM 2003. Twenty-second Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE Cat. No.03CH37428)*. Vol. 2. 2003, 1167–1177 vol.2. DOI: 10.1109/INFCOM.2003.1208953.

[2] B. Fortz e M. Thorup. "Internet traffic engineering by optimizing OSPF weights". In: *Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (Cat. No.00CH37064)*. Vol. 2. 2000, 519–528 vol.2. DOI: 10.1109/INFCOM.2000.832225.

[3] Fabien Geyer. "Routing Optimization for SDN Networks Based on Pivoting Rules for the Simplex Algorithm". In: *DRCN 2017 - Design of Reliable Communication Networks; 13th International Conference*. 2017, pp. 1–8.

Si ringrazia inoltre il Prof. Michele Garetto le cui lezioni sono servite come incipit per questo lavoro.