“Traffic engineering”

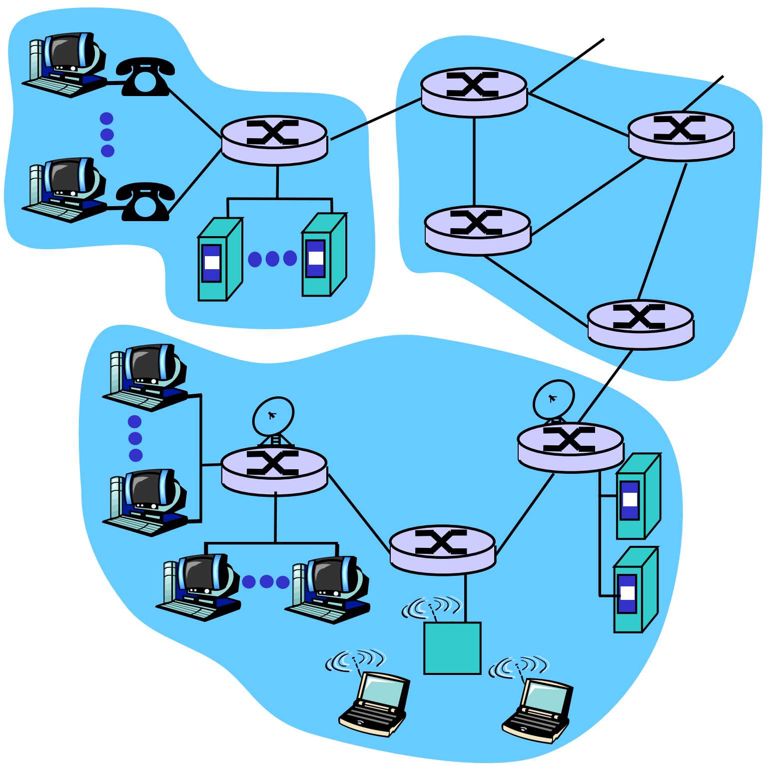
- premesse: instradamento del traffico
- il problema di ingegneria del traffico
- metodi basati su programmazione lineare
- implementazione in reti reali

Funzioni del livello di rete

- Trasportare pacchetti da host sorgente a host destinazione
- Protocolli di livello rete in *ogni* host, router

Tre funzioni importanti:

- *determinazione del cammino:* percorso seguito dai pacchetti da sorgente a destinazione. *Algoritmi di routing*
- *switching:* spostare pacchetti entro router da interfaccia di input a opportuna interfaccia di output
- *call setup:* alcune architetture di rete richiedono di impostare la connessione prima di tx dati

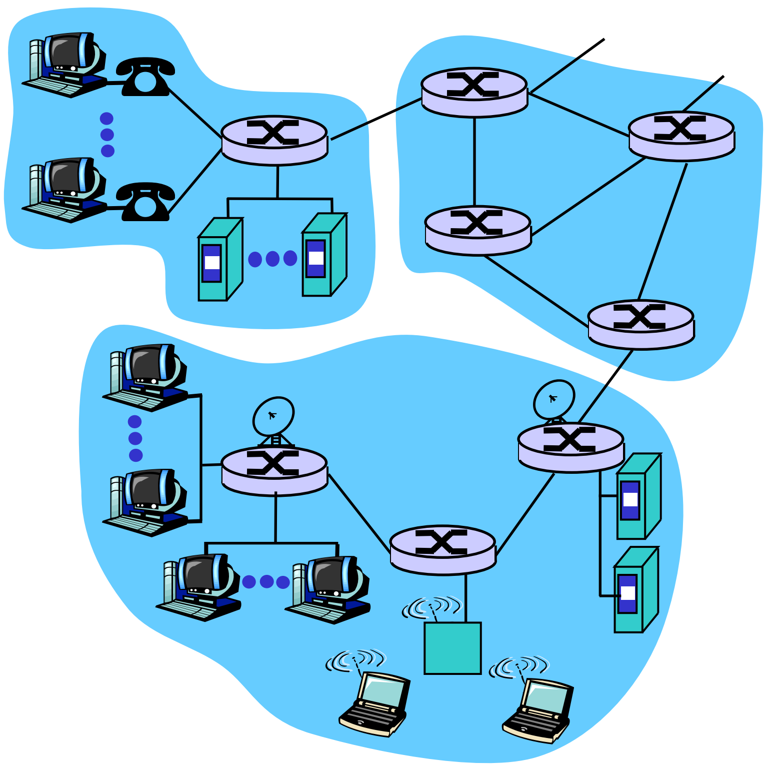


Funzioni del livello di rete

- Trasportare pacchetti da host sorgente a host destinazione
- Protocolli di livello rete in *ogni* host, router

Tre funzioni importanti:

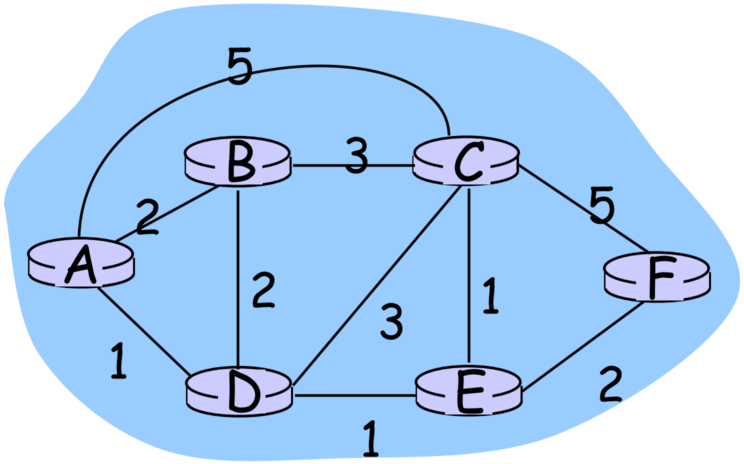
- ***determinazione del cammino:*** **percorso seguito dai pacchetti da sorgente a destinazione. *Algoritmi di routing***
- *switching:* spostare pacchetti entro router da interfaccia di input a opportuna interfaccia di output
- *call setup:* alcune architetture di rete richiedono di impostare la connessione prima di tx dati



Instradamento (routing)

Protocollo di routing

Obiettivo: determinare un "buon" percorso (sequenza di router) attraverso la rete dalla sorgente alla destinazione



Astrazione sotto forma di grafo per algoritmi di instradamento:

- i nodi del grafo sono router
- gli archi del grafo sono i canali fisici
  - peso degli archi: ritardo, costo monetario, livello di congestione

- "buon" percorso:
  - tipicamente significa cammino di costo minimo
  - altre definizioni sono possibili

Classificazione degli algoritmi di routing: informazione globale o decentralizzata?

Globale:

- tutti i router conoscono la topologia completa della rete (inclusi i pesi dei link)
- algoritmi "link state": usano l'algoritmo di Dijkstra per trovare i cammini più a costo minimo da un certo router a tutte le destinazioni

Decentralizzata:

- un router conosce solo i vicini direttamente collegati a livello fisico, e il costo per raggiungerli
- processo computazionale iterativo, tramite scambio di informazione tra nodi vicini
- algoritmi "distance vector" (Bellman-Ford)

Ingegneria del traffico: il contesto

- conoscenza della topologia
- conoscenza della matrice di traffico
  - K – insieme dei flussi sorg/dest
  - per ogni $k \in K$:
    - d_k – domanda di traffico del flusso
    - s_k – sorgente del flusso
    - t_k – destinazione del flusso
- criteri di ottimizzazione
  - minimizzazione della utilizzazione massima di un link
  - mantenere utilizzazione di ogni link sotto una soglia prefissata (es: 60%)

Il problema chiave:
come settare i pesi associati ai link?

Digressione:
programmazione lineare (PL)

Forma standard di un problema di programmazione lineare

$$\text{minimize} \quad \sum_{i=1}^{n} c_i x_i$$

$$\text{subject to} \sum_{j=1}^{n} a_{i,j} x_j = b_i, \quad i = 1, \ldots, m$$

$$x_j \geq 0, \quad j = 1, \ldots, n$$

- si assume che n ≥ m (più variabili che vincoli) e che i vincoli siano linearmente indipendenti
- risolubile in tempo polinomiale in (n,m)
  - algoritmo del simplesso

Riduzione alla forma standard

- Variabili di surplus (per vincoli di tipo $\geq$)
- Variabili di slack (per vincoli di tipo $\leq$)
- Variabili libere

cfr. capitolo 2 di "Linear and Nonlinear Programming", D. Luenberger

Esempio: percorsi ottimi con pesi assegnati (shortest-paths)

- topologia $G = (V,E)$
- K – insieme dei flussi da instradare
  - $k \in K$, s_k - sorgente, t_k - destinazione
- pesi dei link pre-assegnati $\{w_{ij} : (i,j) \in E\}$
- X_{ij}^{k} frazione del flusso k instradato sul link $(i,j) \in E$

Esempio: percorsi ottimi con pesi assegnati (shortest paths)

- si decompone in K sotto-problemi indipendenti, uno per ciascun flusso $k \in K$

$$\min \sum_{(i,j)\in E} w_{ij} X_{ij}^{k}$$

$$s.t. \quad \sum_{j:(i,j)\in E} X_{ij}^{k} - \sum_{j:(j,i)\in E} X_{ji}^{k} = 0, \qquad i \neq s_k, t_k$$

$$\sum_{j:(s_k,j)\in E} X_{ij}^{k} - \sum_{j:(j,s_k)\in E} X_{ji}^{k} = 1,$$

$$\sum_{j:(t_k,j)\in E} X_{ij}^{k} - \sum_{j:(j,t_k)\in E} X_{ji}^{k} = -1,$$

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

Interpretazione della soluzione

- siano $\left\{ \overline{X}_{ij}^{k} \right\}$ le soluzioni ottime del problema
- se $\left\{ \overline{X}_{ij}^{k} \right\}$ assumono i valori 0 and 1, si hanno singoli percorsi di costo minimo.
- se $\left\{ \overline{X}_{ij}^{k} \right\}$ assumono altri valori, esistono molteplici percorsi di costo minimo da utilizzare.

Forma matriciale

$$\begin{aligned} &\text{minimize} \quad \sum_{i} c_i x_j \\ &\text{subject to} \ \sum_{j} a_{i,j} x_j = b_i, \ i = 1, \ldots, m \\ &\qquad\qquad x_j \geq 0, \qquad j = 1, \ldots, n \end{aligned} \quad \longrightarrow \quad \begin{aligned} &\text{minimize} \quad c^T x \\ &\text{subject to} \ \ Ax = b \\ &\qquad\qquad\quad x \ \geq 0, \end{aligned}$$

- x,c colonna di dimensione n
- b vettore colonna di dimensione m ($n \geq m$)
- A matrice di m righe e n colonne
- x^0 si dice ammissibile se $Ax^0 = b$, $x^0 \geq 0$

Soluzioni di base e teorema fondamentale della PL

- Soluzione di base: soluzione ammissibile in cui almeno n-m variabili valgono zero
- teorema fondamentale della PL
  - se esiste una soluzione ammissibile, esiste una soluzione ammissibile di base
  - se esiste una soluzione ottima, esiste una soluzione ottima di base
- algoritmo del simplesso: esplora le soluzioni di base finchè trova quella ottima

Problema duale (forma simmetrica)

Primale

$$\begin{aligned} \text{minimize} \quad & c^T x \\ \text{subject to} \quad & Ax \geq b \\ & x \geq 0, \end{aligned}$$

Duale

$$\begin{aligned} \text{maximize} \quad & y^T b \\ \text{subject to} \quad & y^T A \leq c^T \\ & y \geq 0 \end{aligned}$$

cfr. capitolo 4 di "Linear and Nonlinear Programming", D. Luenberger

Problema duale (forma asimmetrica)

Primale	Duale
minimize $c^T x$	maximize $y^T b$
subject to $Ax = b$	subject to $y^T A \leq c^T$
$x \geq 0,$	

Dualità: proprietà

- se x, y ammissibili, allora $c^T x \geq y^T b$
- se x', y' ammissibili, e se $c^T x' = y'^T b$, allora x' e y' sono ottime
- se uno dei due problemi ha una soluzione con costo finito, lo stesso vale per il duale. Se uno dei due problemi ha soluzione con costo infinito, il duale non ammette soluzioni ammissibili

slackness complementare

Siano x e y soluzioni ammissibili. Condizione necessaria e sufficiente affinchè siano soluzioni ottime e che per ogni i valgano le seguenti due proprietà:

1. $x_i > 0 \Rightarrow y^T A_i = c_i$
2. $x_i = 0 \Leftarrow y^T A_i < c_i$

Dove A_i è la i-esima colonna della matrice A

Esempio: problema Primale Shortest Path (P-SP)

$$\min \sum_{(i,j)\in E} w_{ij} X_{ij}^{k}$$

$$s.t. \quad \sum_{j:(i,j)\in E} X_{ij}^{k} - \sum_{j:(j,i)\in E} X_{ji}^{k} = 0, \quad i \neq s_k, t_k$$

$$\sum_{j:(s_k,j)\in E} X_{ij}^{k} - \sum_{j:(j,s_k)\in E} X_{ji}^{k} = 1,$$

$$\sum_{j:(t_k,j)\in E} X_{ij}^{k} - \sum_{j:(j,t_k)\in E} X_{ji}^{k} = -1,$$

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

Problema duale shortest path (D-SP)

- introduco le variabili duali $y_i^k, i \in V$
- problema duale, per ogni k$\in$K

$$\max\ y_{s_k}^k - y_{t_k}^k$$
$$s.t.\quad y_i^k - y_j^k \le w_{ij},\ (i,j) \in E$$

Problema duale shortest path (D-SP)

- il cambio di variabili

$$U_i^k = y_{s_k}^k - y_i^k, \quad i \in V$$

- porta a:

$$\max U_{t_k}^k, \quad k \in K$$
$$s.t. \quad U_j^k - U_i^k \le w_{ij}, (i,j) \in E, k \in K$$
$$U_{s_k}^k = 0, \qquad k \in K$$

Soluzioni primali-duali di SP

$$\max U_{t_k}^k, \qquad k \in K$$

$$s.t. \quad U_j^k - U_i^k \le w_{ij}, (i,j) \in E, k \in K$$

$$U_{s_k}^k = 0, \qquad k \in K$$

- $\left\{\overline{U}_i^k\right\}$ soluzione ottima del problema duale
- $\overline{X}_{ij}^k > 0 \Rightarrow \overline{U}_j^k - \overline{U}_i^k = w_{ij}, \quad \overline{U}_j^k$ è la lunghezza dello shortest path da s_k a j
- $\overline{U}_{t_k}^k$ lunghezza dello shortest path da s_k a t_k

Come impostare i pesi degli archi per OSPF?

Problema di ingegneria del traffico (TE): minimizzare l'utilizzazione massima dei link

- topologia $G = (V,E)$
- c_{ij} - capacità del link $(i,j) \in E$
- K - insieme di flussi sorg/dest
  - $k \in K$, d_k - domanda, s_k - sorg, t_k - destinazione
- α - massima utilizzazione dei link

Formulazione PL del problema TE

$$\text{minimize } \alpha$$

$$s.t. \sum_{j:(i,j)\in E} X_{ij}^{k} - \sum_{j:(j,i)\in E} X_{ji}^{k} = \begin{cases} 0, & i \neq s_k, t_k, k \in K \\ d_k, & i = s_k, k \in K \\ -d_k, & i = t_k, k \in K \end{cases}$$

$$\sum_{k\in K} X_{ij}^{k} \leq c_{ij}\alpha, \qquad (i,j) \in E$$

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

X_{ij}^{k} representa ora il data-rate del k-esimo flusso sul link (i,j)

formulazione PL di TE modificata

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

$$s.t. \sum_{j:(i,j)\in E} X_{ij}^{k} - \sum_{j:(j,i)\in E} X_{ji}^{k} = \begin{cases} 0, & i \neq s_k, t_k, k \in K \\ d_k, & i = s_k k \in K \\ -d_k, & i = t_k k \in K \end{cases}$$

$$c_{ij}\alpha - \sum_{k\in K} X_{ij}^{k} \geq 0, \qquad (i,j) \in E$$

$$X_{ij}^{k} \geq 0, \alpha > 0$$

r : parametro di controllo (pre-fissato)

Formulazione PL di TE modificata

- possono esserci molte soluzioni con lo stesso valore di α
- a parità di α, si vogliono soluzioni che privilegiano l'uso di shortest paths

$\Rightarrow$ si aggiunge il termine $r \sum_{k \in K} \sum_{(i,j) \in E} X_{ij}^{k}$ al costo, con r piccolo

- da risolvere con algoritmi standard (es: simplesso)

D: possiamo trovare dei pesi sui link tali che la soluzione ottima di TE coincida con quella risultante dal problema di SP (Dijkstra)?

Dualità con vincoli eterogenei

Primal	Dual
minimize $c^T x$	maximize $y_1^T b_1 + y_2^T b_2$
subject to $Ax = b_1$	subject to $y_1^T A + y_2^T A' \leq c^T$
$A' x \geq b_2$	$y_2 \geq 0$
$x \geq 0,$	

- vincoli con uguaglianza nel primale $\Rightarrow$ variabili libere nel duale

Formulazione duale di TE

- variabili di decisione $\{U_i^k\}, \quad \{W_{ij}\}$

$$\max \sum_{k\in K} d_k U_{t_k}^k$$

$$s.t. U_j^k - U_i^k \le W_{ij} + r, \quad k \in K, (i,j) \in E$$

$$\sum_{(i,j)\in E} c_{ij} W_{ij} = 1,$$

$$W_{ij} \ge 0, U_{s_k}^k = 0$$

Interpretazione delle soluzioni

- soluzione ottima problema primale $\left\{\overline{X}_{ij}^{k}\right\}, \overline{\alpha}$

  problema duale $\left\{\overline{U}_{i}^{k}\right\}, \left\{\overline{W}_{ij}\right\}$
- se $\overline{X}_{ij}^{k} > 0$, allora $\overline{U}_{j}^{k} - \overline{U}_{i}^{k} = \overline{W}_{ij} + r$
- ovvero $\overline{U}_{j}^{k}$ può essere vista come la distanza minima da s_k to j quando si usano come pesi sui link le quantità $\left\{\overline{W}_{ij} + r\right\}$

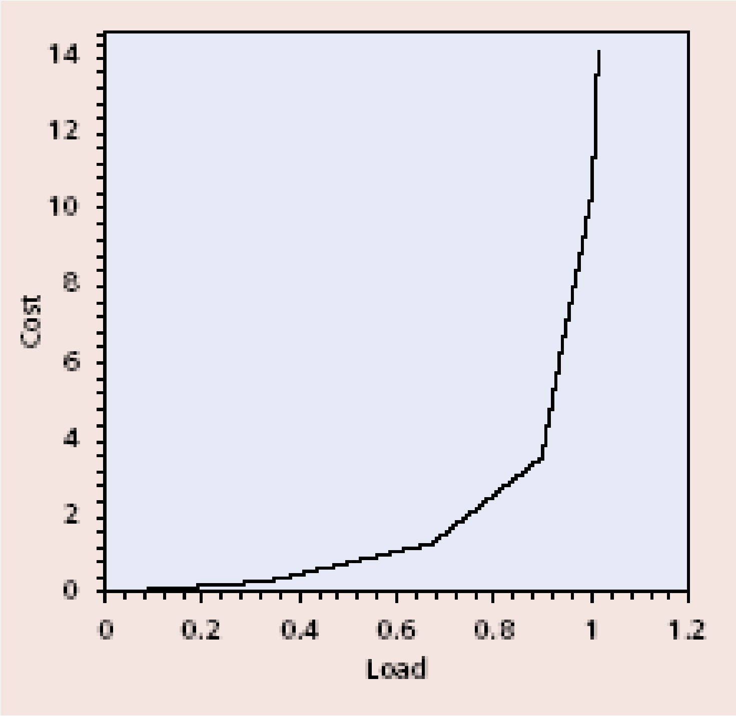
Perciò: la soluzione del problema di traffic engineering fornisce anche la soluzione al problema della scelta ottima dei pesi sui link, ponendo $w_{ij} = \overline{W}_{ij} + r$

Altre funzioni di costo

- il metodo descritto funziona per un vasto insieme di funzioni di costo
- ad esempio:

$$\Phi = \sum_{(i,j)\in E} \Phi_{ij}\left(\sum_{k\in K} d_k X_{ij}^k\right)$$

- dove Φ_{ij} è lineare a tratti
- porta ad una formulazione PL simile alla precedente



Problemi

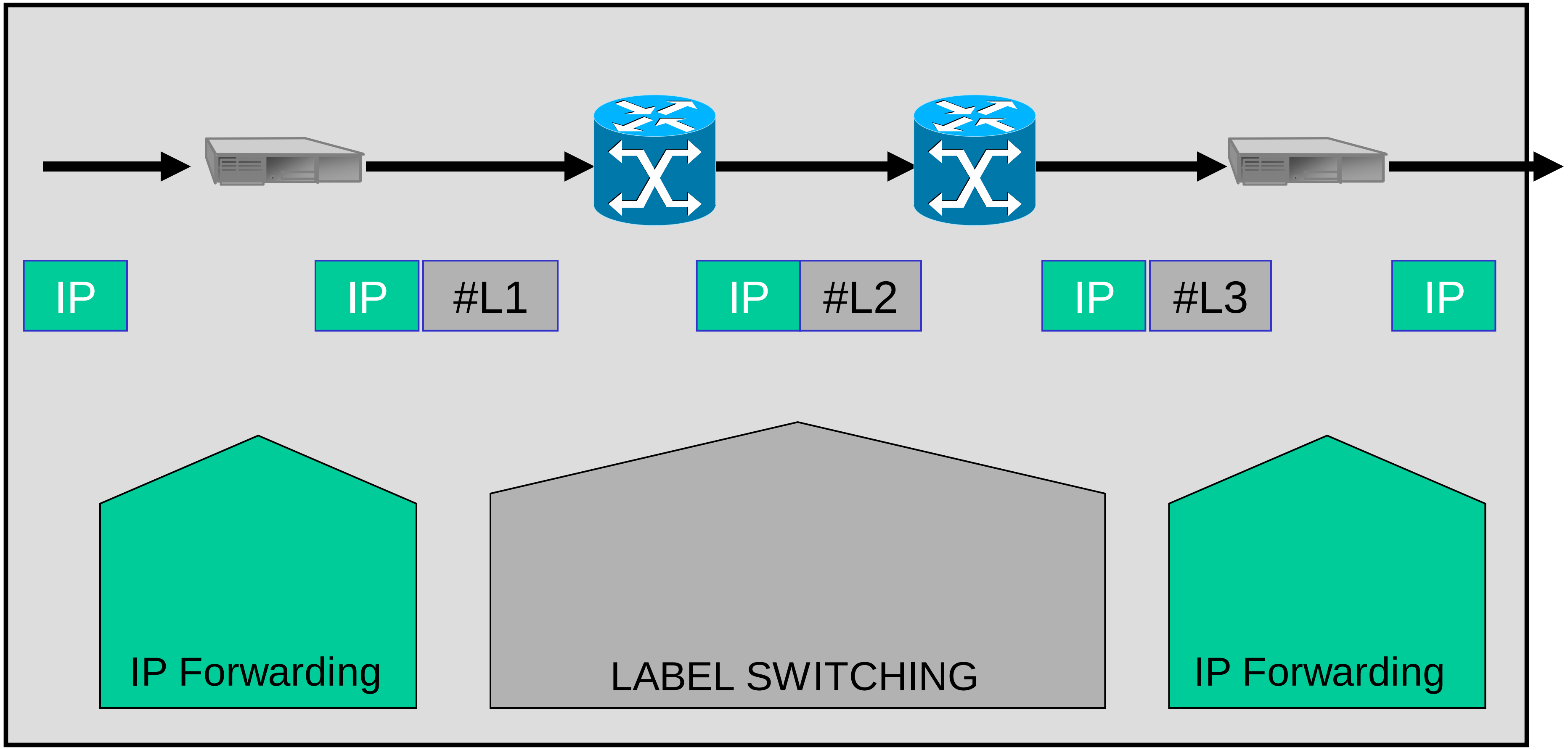
- soluzioni sono specifiche per flusso – occorrono soluzioni specifiche per destinazione (stile IP)
  - non critico, si può riformulare il problema per tener conto di questo vincolo
- soluzioni potrebbero non essere compatibili con la ripartizioni equa sui cammini minimi equivalenti operata da OSPF
  - tener conto di questo vincolo porta a problema NP-hard
    - sono state proposte soluzioni euristiche a questo problema
  - oppure occorre modificare routing IP (es: MPLS)

Routing vs switching

- routing: basato su address lookup, max prefix match
  - richiede operazione di ricerca nella routing table
  - complessità $O(\log n)$ - $O(n)$
- switching: basato su identificativi di circuito
  - identificativi indicizzati
  - complessità $O(1)$
  - scalabile a reti di grandi dimensioni

$\Rightarrow$ MPLS

Idea MPLS: instradamento al bordo, commutazione nel nucleo



MPLS (Multi-Protocol Label Switching)

- Si assegnano "etichette" (label) ai flussi, instradamento su percorso "label-switched"
- permette granularità fine all'instradamento
  - consente ripartizione sbilanciata del traffico su più percorsi
- non vincolato ad alcun algoritmo di instradamento

OSPF (Open Shortest Path First)

- protocollo di tipo link-state
  - costi dei link compresi tra 0 e 65535
  - raccomandazione Cisco: costo $\propto$ 1/(link capacity)
  - rapida convergenza senza loop, scala bene
  - topologia completa in ogni nodo, algo di Dijkstra
  - messaggi "OSPF advertisement", con una entry per ogni router adiacente, mandati in flooding nell'AS
- permette percorsi multipli a pari costo:
  - traffico diviso su tutte le interfacce di uscita appartenenti a percorsi di costo minimo
- IS-IS (Intermediate system-Intermediate System)
  - protocollo simile, proprietario Cisco

Un possibile approccio per risolvere lo “splitting problem“

- le tabelle di routing attuali contengono migliaia di prefissi di instradamento
- invece di instradare ogni prefisso su tutti i percorsi di pari costo, si assegnano selettivamente i next-hop a (ogni) prefisso
  - ovvero, alcuni next-hop vengono rimossi dall'insieme di quelli utilizzabili da un certo prefisso
- obiettivo: approssimare i carichi ottimali sui link

A. Sridharan, R. Guerin and C. Diot, "Achieving Near-Optimal Traffic Engineering Solutions for Current OSPF/IS-IS Networks", *IEEE/ACM Transaction on Networking*, VOL. 13, April 2005.

Esempio: min-max load heuristic

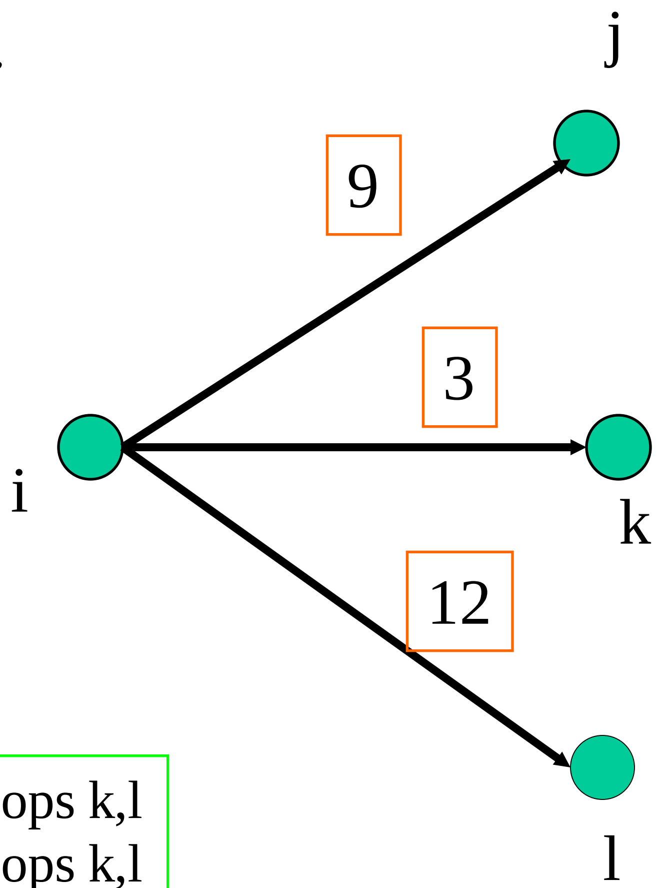
Situazione iniziale

Prefix A : 5

Prefix B : 1

Prefix C : 8

Prefix D : 10



Prefixes: C D

$\boxed{4 + 5 = 9}$

Prefixes: A B

$\boxed{2.5 + 0.5 = 3}$

Prefixes: A B C D

$\boxed{2.5 + 0.5 + 4 + 5 = 12}$

Prefix A: Hops k,l
Prefix B : Hops k,l
Prefix C: Hops j,l
Prefix D: Hops j,l

Esempio: min-max load heuristic

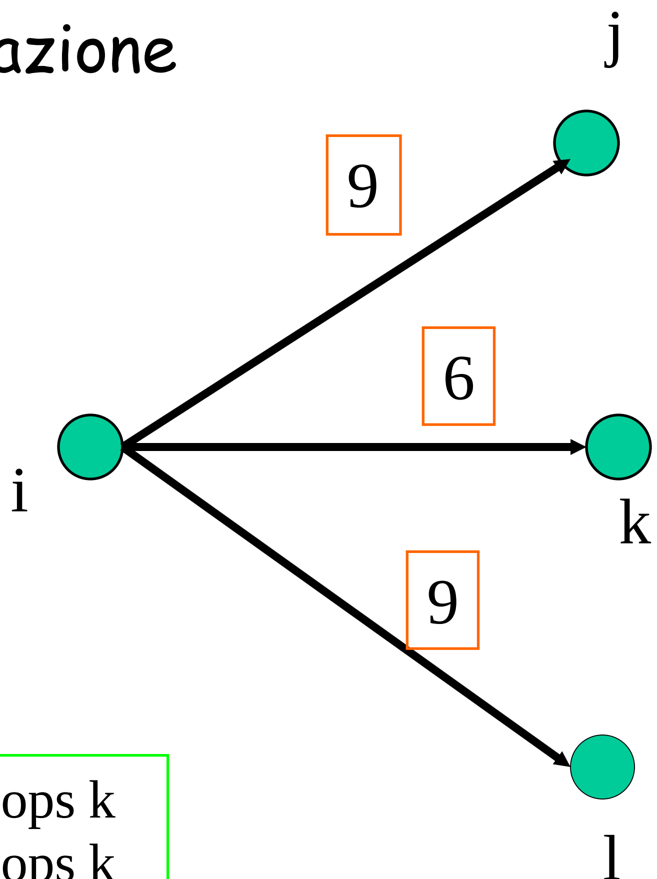
Dopo allocazione selettiva

Prefix A : 5

Prefix B : 1

Prefix C : 8

Prefix D : 10



Prefixes: C D

$\boxed{4 + 5 = 9}$

Prefixes: A B

$\boxed{5 + 1 = 6}$

Prefixes: C D

$\boxed{4 + 5 = 9}$

Prefix A: Hops k
Prefix B : Hops k
Prefix C: Hops j,l
Prefix D: Hops j,l

Vantaggi

- non necessita di cambiamenti sul piano dati
- lascia immodificati i protocolli di routing esistenti
- i router attuali hanno decine di migliaia di prefissi nella tabella di instradamento
  - ciò fornisce grande flessibilità nella allocazione dei next-hop per approssimare il bilanciamento ottimale del carico

Conclusioni

- si può usare OSPF/ISIS per ottenere obiettivi di ingegneria del traffico
- basta impostare opportunamente i pesi sui link, calcolati con metodi di PL
- Lo "splitting problem" complica le cose
  - tecniche euristiche ottengono buone prestazioni
  - piccoli cambiamenti nelle tabelle di instradamento forniscono prestazioni ancora migliori
- MPLS non soffre di questi problemi (permette di implementare direttamente la soluzione ottima)