



METODOLOGIE DI OTTIMIZZAZIONE ORIENTATE ALLA QoS E APPLICATE ALLE RETI NGN, MULTISERVIZIO E MULTIDOMINIO

Ing. Maria Teresa Carta

*Tesi per il conseguimento del Dottorato di Ricerca in
Ingegneria Elettronica ed Informatica*

Relatori: *Prof. Ing. D.D. Giusto e Prof. Ing. Luigi Atzori*

**XX Ciclo di Dottorato di Ingegneria Elettronica e Informatica
Università degli Studi di Cagliari
Febbraio 2008**

A zio Giampaolo

Ringraziamenti personali

Se dovessi esprimere in parole il sentimento di gratitudine nei confronti delle persone a me care che, non solo in questi ultimi tre anni, ma in tutta la mia vita mi hanno sempre aiutato e sostenuto non basterebbe un libro intero, mi limiterò, perciò, a dire un semplice grazie.

Grazie a mamma, babbo, kati, matti, nonna e al resto della mia famiglia, per ogni suo componente vicino o lontano, presente o che, purtroppo, non c'è più, esisterà sempre un posto nel mio cuore.

Grazie a Michele, che fa parte anch'egli della mia famiglia, ma che merita un ulteriore ringraziamento per l'amore con cui mi sostiene nei momenti duri della vita e perché mi sopporta anche quando divento insopportabile.

Grazie alle amiche vere A. e A. (non voglio seguire nessun ordine perché siete alla pari nella scala degli affetti per cui vi cito solo con le iniziali che, fortunatamente, sono uguali!), perché quando ho bisogno accorrono in mio aiuto.

Sommario

RINGRAZIAMENTI PERSONALI.....	III
SOMMARIO	IV
INDICE DELLE FIGURE	VI
ABSTRACT.....	VII
CAPITOLO 1 PRESENTAZIONE DELLA TESI	9
1.1 Gestione della qualità nelle reti NGN.....	9
1.2 Problema dell'ottimizzazione	10
CAPITOLO 2 GESTIONE DELLA QUALITÀ NELLE RETI NGN.....	11
2.1 Tipologie di applicazioni.....	13
2.1.1 Applicazioni elastiche.....	14
2.1.2 Applicazioni Streaming o Real-Time	14
2.2 Qualità del servizio.....	16
2.2.1 Approcci relativi alla definizione di QoS	16
2.2.2.1 Approccio ITU/ETSI.....	17
2.2.2.2 Approccio IETF.....	18
2.2.2.3 Sistemi di gestione della qualità.....	19
2.2.2 Parametri di QoS	20
2.2.3 Parametri di performance relativi al trasferimento di pacchetti IP.....	21
2.2.4 Parametri di performance relativi all'affidabilità	25
2.2.5 Parametri di performance relativi all'accesso al servizio.....	26
2.3 Classificazione dei servizi e definizione dei requisiti ottimali	27
2.3.1 Classi di QoS.....	28
2.3.2 Gli SLA	28
CAPITOLO 3 OTTIMIZZAZIONE DI UNA RETE NGN	30
3.1 Architettura di riferimento	32
3.1.1 Il DiffServ.....	32
3.1.2 L'MPLS.....	33
3.1.3 Il TE.....	33
3.2.1.1 Obiettivi TE.....	33
3.1.4 Perché MPLS per il TE.....	34
3.1.5 La scelta: il DS-TE	34
3.2 Elementi chiave	35
3.2.1 Attributi relativi al traffico.....	36
3.2.2 Attributi relativi alle risorse.....	39
3.2.3 Allocazione delle risorse	39
3.2.4 Parametri da ottimizzare	41
3.2.5 Parametri che realizzano l'ottimizzazione.....	42
PARTE I- TECNICHE DI OTTIMIZZAZIONE: ROUTING E REROUTING LSP.....	43
CAPITOLO 4 PROBLEMATICHE DI ROUTING E STATO DELL'ARTE.....	44
4.1 Modellamento del problema di instradamento	44
4.1.1 Dati in ingresso al problema e assunzioni.....	45
4.1.2 Caratteristiche dei tunnel.....	46
4.2 Stato dell'arte.....	48
CAPITOLO 5 TECNICHE PER L' INSTRADAMENTO OTTIMO	51
5.1 Introduzione.....	51
5.2 Il Multicommodity Flow Problem.....	52
5.3 Adattamento dell'algoritmo al caso di studio	56
5.4 Ottimizzazione del routing online-dinamico	61
5.4.1 Uno schema generale per il routing dinamico.....	62
5.2.4.1 Meccanismi di preemption	63
5.2.4.2 Definizione di una procedura di routing dinamico preemption based	66

5.4.2 Individuazione degli LSP da abbattere	68
5.4.3 Tecnica per il processo di rerouting degli LSP abbattuti	74
5.5 Seconda tecnica: approccio globale.....	83
5.6 Tecnica Ibrida.....	89
PARTE II- TECNICHE DI OTTIMIZZAZIONE: VIDEO STREAMING.....	90
CAPITOLO 6 PROBLEMATICHE SERVIZIO VIDEO STRAMING SU WIRELESS LAN	91
6.1 Introduzione.....	91
6.2 Stato dell'arte.....	92
CAPITOLO 7 METODO DI OTTIMIZZAZIONE PROPOSTO	94
7.1 Framework e obiettivi.....	94
7.1.1 Architettura e assunzioni base	94
7.1.2 Scopo.....	97
7.2 Controllo della Probabilità di Starvation di Playback.....	97
7.3 Processamento dinamico dei Delay Statistical Trend.....	100
7.3.1 Tecniche di Aging	100
7.4 Workflow dell'algoritmo proposto	101
CAPITOLO 8 RISULTATI SPERIMENTALI.....	104
8.1 Configurazioni Generali	104
8.2 Valutazioni sulla stima del ritardo di trasmissione dei pacchetti.....	105
8.2.1 Chi-quadro test	108
8.3 Validazione dell'ipotesi semplificativa per il calcolo di R_s	113
8.4 Valutazioni finali sul metodo proposto.....	115
8.4.1 Valutazione della qualità video.....	115
8.4.2 Valutazioni sul rate sorgente e occupazione del playback buffer	115
8.4.3 Valutazioni sulla complessità computazionale	117
8.5 Effetti delle tecniche di aging sul sistema di controllo	117
CONCLUSIONI.....	122
AKNOWLEDGMENTS	124
REPORT E ARTICOLI PRODOTTI.....	125
RIFERIMENTI.....	126
APPENDICE A CLASSI DI COMPLESSITÀ DEI PROBLEMI DI OTTIMIZZAZIONE.....	129

Indice delle figure

Figura 1 Classificazione delle applicazioni.....	13
Figura 2 Il modello generale di QoS e gli approcci ITU/ETSI e IETF.....	18
Figura 3 Campo ToS nel protocollo IPv4.....	19
Figura 4 Valutazione della qualità intrinseca.....	21
Figura 5 Categorie di applicazioni in funzione della reazione alla perdita di pacchetti.....	24
Figura 6 Skew.....	25
Figura 7 Fasi necessarie al processo di ottimizzazione.....	30
Figura 8 Allocations della banda secondo l'RDM.....	41
Figura 9 Schema del MCFP.....	51
Figura 10 Esempio di preemption con rerouting di LSP.....	66
Figura 11 Architettura di riferimento per il sistema di video streaming.....	96
Figura 12 Pdf per il processo $Y_m(\tau)$	99
Figura 13 Calcolo della fd	101
Figura 14 Diagramma di flusso dello schema proposto.....	102
Figura 15 Calcolo di Y_τ^*	103
Figura 16 Pdf del delay di rete.....	106
Figura 17 Autocorrelazione del delay di rete.....	107
Figura 18 Determinazione dei campioni da confrontare nel Chi-Test.....	108
Figura 19 Valore medio del chiquadro contro c ed f (in secondi). Prima tecnica di aging....	110
Figura 20 Valore medio del chiquadro contro c ed f (in secondi). Seconda tecnica di aging.....	111
Figura 21 Valore medio del chiquadro contro c ed f (in secondi). Terza tecnica di aging....	112
Figura 22 Source bit rate versus T per differenti valori iniziali dell'occupazione del buffer.....	114
Figura 23 Source bit rate versus tempo.....	116
Figura 24 Occupazione del Playback buffer versus tempo.....	116
Figura 25 Andamento della deviazione standard del rate sorgente per Aging 1.....	118
Figura 26 Andamento della Y_τ^* per Aging 1.....	119
Figura 27 Andamento della occupazione media del buffer per Aging 1.....	120

Abstract

Il mondo delle telecomunicazioni, a fronte di una crescente evoluzione tecnologica, ha subito dei forti mutamenti rispetto agli esordi e, ancora, possiede ampi margini di crescita in un prossimo futuro dove la centralità dell'utente si farà più forte. I servizi offerti sono sempre più numerosi, personalizzati e necessitano una classificazione opportuna, mediante dei parametri rappresentativi e oggettivi che consentono la definizione di vincoli, rispettati i quali, si ottengono le prestazioni richieste in termini di qualità. Inoltre, le reti di accesso sono sempre più differenti tra loro ma, allo stesso tempo, interconnesse da una comune infrastruttura di backbone. Si parla, quindi, di reti di nuova generazione (NGN) multiservizio e multidominio.

Il rispetto dei parametri di qualità e dei relativi Service Level Agreement concordati con l'utente, si ottiene mediante il miglioramento delle performance della rete che, in un contesto così complesso e articolato, può essere ottenuto attraverso strategie diverse e a vari livelli d'intervento.

La presente tesi tratta lo studio di tecniche di ottimizzazione della rete con un approccio orientato sia al traffico che alle risorse di rete. Un'attenta analisi dello scenario e delle tipologie di traffico, ha consentito la definizione e la scelta dei parametri di qualità, i quali permettono di agire attivamente sull'efficienza della rete e sulla possibilità di implementare una politica di gestione della QoS. Inoltre, sono stati analizzati vari approcci architetturali relativi alla rete di trasporto che hanno condotto alla definizione di alcune linee guida utili, sin nella fase di progettazione, per predisporre al meglio la fruibilità di servizi diversificati e di alta qualità. La soluzione architetturale che meglio sposa gli obiettivi di performance è data dall'integrazione di MPLS e Diffserv con applicazione del Traffic Engineering (DS-TE), in cui, all'approccio di differenziazione del trattamento del traffico sulla base dell'appartenenza a classi di servizio (CoS), si affianca il meccanismo dell'MPLS, che offre garanzie di qualità end-to-end, per la costruzione dei percorsi, Label Switch Path (LSP), su cui inoltrare gli aggregati di traffico; infine, il TE permette una gestione efficiente delle risorse.

Relativamente alla rete di backbone DS-TE, il focus è andato principalmente sulle problematiche di ottimizzazione del routing degli LSP. In particolare, si definiscono delle metodologie che consentono un miglior bilanciamento del carico sulla rete, ma soprattutto, che permettono agli aggregati classificati con una maggiore priorità di essere sempre serviti. Oltre all'approccio di riottimizzazione globale degli LSP sulla rete, è stata proposta una procedura che, ad ogni richiesta di un nuovo LSP, individua un percorso ottimo di instradamento, ricorrendo, ove necessario, ad una tecnica di preemption ottimizzata per rimuovere LSP a priorità inferiore, e nel contempo, cerca per questi un percorso alternativo, riducendo il service disruption.

Per quanto riguarda l'ottimizzazione della rete a garanzia della qualità per alcune tipologie di servizio specifiche, è stato preso in esame il caso delle applicazioni video-streaming che riscuotono, oggi, grande interesse. In particolare, lo scenario considerato è quello di una rete di tipo wireless che, per la sua natura, produce un throughput di tipo bursty e, quindi, rende molto difficile l'alta qualità dello streaming video. L'algoritmo proposto è di tipo window-based e prevede un Source Rate Control che modifica il rate di sorgente compensando i periodi alternanti tra basso ed alto throughput in modo tale da evitare il tipico "saw" effect. Il rate viene modificato periodicamente all'inizio di ogni finestra temporale, il suo valore si ottiene imponendo che il valore della probabilità di starvation, del buffer nell'end-system sia inferiore ad una certa soglia per tutta la durata della finestra. Tale scopo è perseguito facendo una previsione a breve-termine del ritardo che il traffico subirà nella rete per la durata della finestra, a partire dalle informazioni raccolte nelle finestre precedenti.

Capitolo 1

Presentazione della tesi

Il lavoro proposto in questa tesi nasce e si sviluppa nell'ambito del progetto IKNOS, Gestione della Qualità attraverso KPI in Reti IP di Nuova Generazione Multiservizio, Multidominio e Multivendor Orientata alla Standardizzazione. Il progetto, finanziato dal Ministero Italiano dell'Università e Ricerca MIUR con fondi FAR secondo la legge 297/1999 Art. 5 protocollo n. 13246, ha avuto inizio nel gennaio 2005 ed è, attualmente, in fase di completamento.

I partner del progetto, Tiscali (Sede di Cagliari), Consorzio Nazionale Interuniversitario per le Telecomunicazioni (CNIT) (Sede di Napoli), Università degli studi di Cagliari (Sede Cagliari), perseguono come obiettivi principali lo studio e definizione di nuove metodologie per il supporto nell'ambito delle reti NGN delle seguenti fasi di gestione della QoS:

- Progettazione
- Impatto delle nuove applicazioni
- Ottimizzazione
- Monitoraggio.

L'argomento principale oggetto della tesi è la fase di ottimizzazione, comunque per maggiori informazioni relative al progetto si rimanda al sito <http://mclab.diee.unica.it/iknos>.

1.1 Gestione della qualità nelle reti NGN

La prima parte della presente tesi affronta il tema della gestione della qualità del servizio nelle reti di telecomunicazioni multiservizio e multidominio. Dopo aver definito le tipologie di servizio e introdotto la qualità, vengono analizzate caratteristiche dei servizi, identificandone: gli indicatori di performance e la suddivisione in classi di servizio, contraddistinte da stringenti vincoli dei parametri di qualità. Inoltre, si presentano vari approcci architetturali per la gestione della qualità. Questa fase iniziale di definizione dei requisiti e di classificazione è particolarmente delicata, poiché costituisce la base del un sistema di gestione della qualità teso a perseguire quattro obiettivi fondamentali: adattabilità, scalabilità, capillarità e qualità end-to-end.

1.2 Problema dell'ottimizzazione

Il secondo capitolo è dedicato alle problematiche dell'ottimizzazione della rete di nuova generazione (NGN), in particolare viene descritta l'architettura di riferimento per il backbone e vengono presentati gli attributi relativi sia al traffico che alle risorse con cui si ottimizza la gestione delle performance della rete. Si stabiliscono, anche, i parametri principali da ottimizzare e quelli su cui, principalmente, si può agire con opportune metodologie di ottimizzazione.

Data la vastità del problema e i numerosi piani d'intervento, si sono scelti due casi di studio specifici per le metodiche di ottimizzazione che sono discussi nelle due sezioni in cui si articola tesi. La prima sezione propone delle tecniche di ottimizzazione del routing e rerouting degli LSP sulla rete di backbone, mentre nella seconda sezione si presenta un sistema di controllo che ottimizza la qualità del video streaming in una wireless LAN.

Gli ultimi due capitoli saranno dedicati alla presentazione e discussione dei risultati della tecnica specifica per il video streaming su wireless LAN e alle conclusioni.

Capitolo 2

Gestione della qualità nelle reti

NGN

Lo sviluppo di reti IP di nuova generazione, multiservizio, multidominio e multivendor è orientato alla fornitura di servizi Quadruple-Play: informazioni di tipo voce, dati e video (Triple-Play) vengono offerte su un'unica infrastruttura di rete basata su IP unitamente ad applicazioni multimediali su piattaforme mobili di terza generazione. Dal punto di vista dell'utente, questo scenario di convergenza offre notevoli vantaggi in termini di comodità, semplicità e costi. Diventa, infatti, possibile accedere da un qualunque terminale, fisso o mobile, ad un ventaglio sempre più ampio di servizi, con un alto livello di personalizzazione e disponibilità always on, utilizzando un'opportuna piattaforma di servizio che ne consenta la fruizione (set-top-box, videotelefono, ecc.). Dal punto di vista degli operatori di rete, la capacità di includere in offerte quadruple play tutti i servizi di comunicazione, significa l'opportunità di ridurre i costi relativamente alti legati alla gestione dei singoli servizi, sfruttando un'unica piattaforma altamente flessibile ed innovativa, basata sulla preesistente infrastruttura di trasporto. Il controllo della qualità e la fornitura di servizi differenziati sulla base delle esigenze di ogni singolo utente rappresentano un notevole valore aggiunto, che permette agli operatori di conquistare nuove quote di mercato.

Per contro, questa notevole varietà di servizi e di informazioni trasportate implica l'esigenza di definire delle politiche di gestione tese a garantire specifici requisiti di performance per ciascun flusso dati. È evidente, infatti, che applicazioni di video-conferenza, che comportano la generazione di traffico dati, audio e video con vincoli di real-time, hanno esigenze molto diverse da applicazioni come il servizio di posta elettronica, che possono accettare un ritardo nella consegna dei dati, ma assolutamente non tollerano perdite. L'individuazione di un modello flessibile per la gestione della qualità del servizio diventa quindi condizione indispensabile per lo sviluppo di reti di nuova generazione, al fine di diversificare il trattamento dei singoli flussi di dati e di garantire lo sfruttamento ottimale delle risorse.

Nelle reti IP tradizionali l'unico servizio offerto è di tipo best effort, privo cioè della capacità di fornire agli utenti un livello di qualità adeguato alle diverse applicazioni. Il protocollo IP, infatti, non garantisce nulla riguardo l'effettiva ricezione, l'ordinamento e la gestione delle priorità dei pacchetti; è connectionless e non prevede nessun meccanismo per il controllo e il recupero degli errori. La rete ha l'unico obiettivo di consegnare i pacchetti e lascia ai terminali alle estremità il compito di recuperare eventuali situazioni d'errore. Alcune funzionalità quali la numerazione dei pacchetti e la ritrasmissione in caso di perdita o errori, possono essere garantite dal livello di trasporto qualora si utilizzi il protocollo TCP (Transfer Control Protocol). Questo modello implementativo è stato concepito per un tipo di traffico (File Transfert Protocol, TELNET, Hyper Text Transfert Protocol etc.) che non necessita di requisiti specifici per quanto riguarda i ritardi, la banda disponibile e gli errori di trasmissione e si è dimostrato sufficientemente valido fino alla diffusione di servizi triple-play e quadruple-play. Applicazioni di streaming audio-video, Voice over IP (VoIP), videoconferenza o IP Television (IPTV) impongono, infatti, vincoli stringenti sui tempi di consegna dei pacchetti e sulla percentuale delle risorse di rete utilizzate da ogni flusso dati e sono maggiormente sensibili agli errori di trasmissione. Per queste nuove classi di applicazioni, l'approccio best effort non è più sufficiente. Occorre definire un nuovo modello di rete che consenta di garantire ad ogni singola applicazione il livello di qualità ottimale, calcolato sulla base delle caratteristiche specifiche e dei parametri di performance richiesti.

Per andare in contro a questa necessità, sono state proposte diverse soluzioni. In particolare, l'Internet Engineering Task Force (IETF) ha ideato due architetture per la gestione della qualità del servizio: l'Integrated Service (IntServ) e il Differentiated Service (DiffServ). La prima è basata sul protocollo RSVP (Resource reSerVation Protocol) per l'instaurazione di un ambiente connection oriented, in cui si riservano risorse per ogni singolo flusso dati con la necessità di utilizzare opportuni algoritmi di segnalazione e di conservare informazioni di stato sul flusso in ogni nodo attraversato. La seconda, invece, raggruppa in classi flussi differenti con delle caratteristiche comuni, come ad esempio la rete di origine o vincoli particolari legati ai ritardi o al bit-rate. In tal modo, i pacchetti vengono elaborati nei nodi di rete a seconda della classe di appartenenza e non in modo individuale come nel caso precedente. Questo approccio orientato alle classi sacrifica la capillarità del servizio a vantaggio di una maggiore scalabilità del sistema all'aumentare del numero di nodi e di utenti. Una terza alternativa nata in ambito IETF è quella di utilizzare il protocollo MPLS (MultiProtocol Label Switching) per fornire il supporto ad una completa gestione end-to-end della qualità mediante applicazioni di traffic engineering. Queste soluzioni possono essere adattate a seconda della tipologia della rete, eventualmente sviluppando architetture ibride che consentano di soddisfare le esigenze di qualunque utente con un servizio puntuale, scalabile allo stesso tempo e con garanzie di qualità sia hop-by-hop che end-to-end.

Indipendentemente dalla scelta delle politiche di gestione, lo sviluppo di un sistema per il controllo della QoS presuppone la definizione di un quadro completo delle applicazioni di riferimento e dei relativi requisiti di qualità. La progettazione dell'architettura di rete deve tener conto delle caratteristiche e delle esigenze di tutti i servizi, tradizionali e innovativi, che

dovranno essere supportati. Ovviamente, la soluzione architetturale dovrà essere flessibile e predisposta all'introduzione di nuovi servizi. È necessaria una stima delle prestazioni attese dagli utenti finali, relativamente alla disponibilità del servizio, alla velocità di trasmissione e alla qualità percepita. Sulla base di tale stima è possibile definire un modello di classificazione delle applicazioni e calcolare i valori ottimali per i parametri di performance relativi al trasferimento dei pacchetti o all'affidabilità della rete.

2.1 Tipologie di applicazioni

La realizzazione degli obiettivi di qualità del servizio nelle reti NGN richiede una caratterizzazione delle diverse applicazioni sulla base dei requisiti di performance richiesti. A tale scopo può essere utile partire dall'analisi di alcune caratteristiche comuni che serviranno da guida per la definizione del modello delle applicazioni di riferimento per reti multiservizio con controllo della qualità.

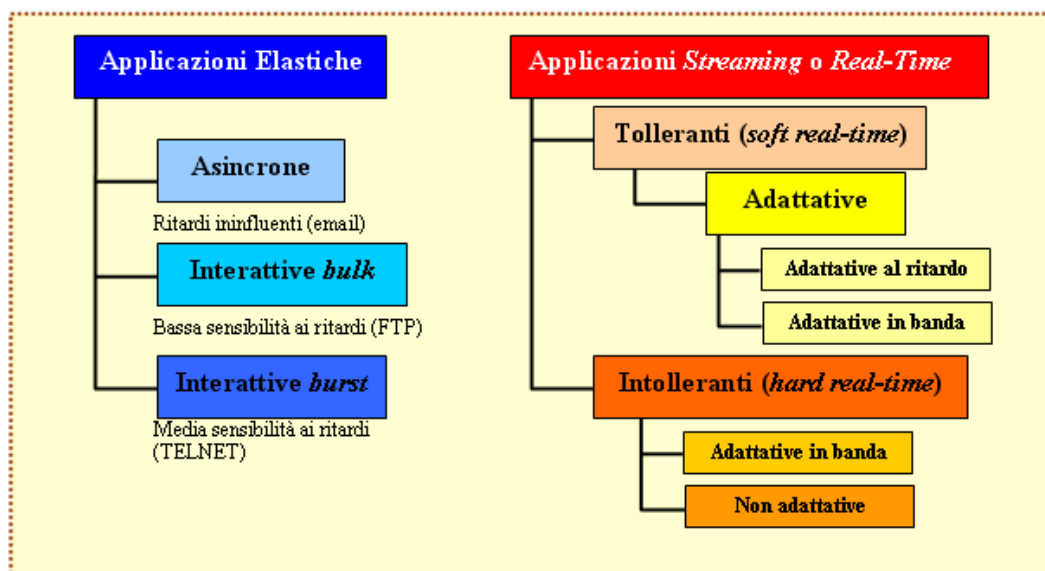


Figura 1 Classificazione delle applicazioni.

In linea di principio, è possibile dare diverse classificazioni dei servizi di comunicazione che attualmente possono essere forniti su base IP. Alcuni esempi si possono trovare in [1] e [2]. In questa sezione, si è scelto di fare riferimento alla classificazione definita dall'ITU nella RFC1633 [3]. Tale classificazione è definita sulla base del ritardo che ogni pacchetto dati subisce nel transitare attraverso la rete e nasce dalla considerazione che il ritardo di trasmissione è uno dei parametri che maggiormente pregiudica le prestazioni di un'applicazione. I servizi di riferimento possono essere classificati qualitativamente in due categorie differenti in funzione del grado in cui risentono degli effetti di questo ritardo. In particolare, è possibile raggruppare in un'unica classe tutte le applicazioni che elaborano i dati senza vincoli di real-time e che quindi sono in grado di svolgere le proprie funzioni, anche se alcuni dati arrivano in ritardo. Un'altra classe di applicazione richiede, invece, che ogni pacchetto arrivi a destinazione entro un dato intervallo di tempo, oltre il quale verrebbe

scartato. Nel primo caso si parla di **Applicazioni Elastiche** nel secondo di **Applicazioni Streaming** o **Real-Time** [Figura 1].

2.1.1 Applicazioni elastiche

Le applicazioni elastiche sono quelle che comunemente vengono offerte dal servizio di tipo best effort delle reti IP: e-mail, *web browsing*, FTP, TELNET, etc. Queste applicazioni elaborano i dati nell'istante in cui arrivano, senza necessità di buffer in ricezione o di una caratterizzazione a priori del livello di servizio richiesto, e sono pertanto in grado di tollerare eventuali ritardi. Questo non significa che il traffico elastico è insensibile agli effetti dei ritardi di trasmissione, piuttosto, il ritardo riduce le prestazioni delle applicazioni. In particolare, questa riduzione è dovuta al ritardo medio più che alla distribuzione statistica dei ritardi, perciò alcuni pacchetti possono arrivare con un ritardo notevole purché il valore medio rimanga entro certi limiti.

Come indicato in Figura 1, le applicazioni elastiche possono essere classificate in tre sottocategorie:

- *applicazioni asincrone*: sono le applicazioni meno sensibili ai ritardi di trasmissione (es. email, FAX). In questi casi, la rete riesce solitamente a garantire un livello di prestazioni sufficiente a soddisfare le attese dell'utente.
- *Applicazioni interattive bulk*: risentono in misura limitata degli effetti dei ritardi (es. FTP, o il traffico http). Trasferiscono grandi quantità di dati che non necessitano di essere utilizzati a destinazione con precisi limiti temporali e che vengono trasmessi con velocità pressoché costante.
- *Applicazioni interattive burst*: sono caratterizzate da picchi istantanei nel bitrate che varia sensibilmente rispetto al valore medio (es. TELNET). Risentono mediamente dei ritardi.

Il modello di servizio più indicato a garantire i requisiti di performance richiesti alle applicazioni elastiche è il servizio di tipo ASAP (as-soon-as-possible) o *best effort*: i dati vengono elaborati quanto prima possibile in funzione dell'istante in cui arrivano a destinazione e della tipologia dell'applicazione. Tale approccio consente di minimizzare il ritardo più nel caso di traffico interattivo *bulk* o *burst* che nel caso di applicazioni asincrone. Inoltre, a differenza del modello di servizio real-time, non richiede politiche di *admission control*.

2.1.2 Applicazioni Streaming o Real-Time

Si definiscono applicazioni *streaming* o *real-time* tutte le applicazioni che impongono un limite massimo al ritardo subito dai pacchetti: se un pacchetto arriva con un ritardo superiore ad una data soglia non è più utilizzabile e viene conseguentemente scartato perché privo di valore.

Un'importante categoria di applicazioni *real-time* sono le applicazioni di "*signal playback*" che riproducono in ricezione segnali generati da una sorgente, segmentati in pacchetti e trasportati dalla rete fino a destinazione con un ritardo aleatorio tra pacchetto e pacchetto. Il

ricevitore, per ricostruire correttamente il segnale originale, deve rispettare dei vincoli temporali ben precisi, evitando il rimescolamento dei pacchetti e gli errori di temporizzazione. Questo comporta la necessità di un *buffer* in ricezione che consenta di compensare la variazione di ritardo tra i diversi pacchetti (*jitter*), ritardando la riproduzione di ogni pacchetto della stessa quantità (*offset delay*): viene definito il "punto di playback" che determina l'istante temporale oltre il quale i dati in ingresso non sono più utilizzabili per la ricostruzione real-time del segnale originale. Il punto di playback può essere stabilito a priori, se è disponibile una conoscenza della rete, oppure ricavato dall'analisi dei ritardi dei singoli pacchetti, in modo da poter variare l'*offset delay* dinamicamente.

Le applicazioni di "*signal playback*" si possono suddividere in *soft real-time* (o tolleranti) e *hard real-time* (o intolleranti): le prime accettano una variazione del ritardo massimo accumulabile da ciascun pacchetto; le seconde richiedono che il ritardo massimo sia fissato a priori per evitare qualsiasi forma di distorsione del segnale originale. Generalmente, le applicazioni *real-time* di tipo tollerante cercano di ricavare il valore dell'*offset delay* dall'analisi dei ritardi dei singoli pacchetti per ridurre il ritardo complessivo con cui il segnale viene riprodotto. In tal caso si parla di applicazioni di *Adaptive Playback*, che hanno esigenze meno stringenti rispetto alle altre applicazioni: riducendo i vincoli sul ritardo è possibile incrementare il livello di utilizzazione della rete, senza che l'applicazione ne risenta in modo pesante. Per questo motivo, per applicazione di *Adaptive Playback* è sufficiente un servizio di trasporto analogo al classico *Best Effort* riferito però ad una rete priva di carico. Esempi di applicazioni adattative sono il VoD (Video on Demand) o la televisione diffusa.

Le performance di un'applicazione real-time sono valutate sulla base di due parametri: *latenza* e *fedeltà*. Per *latenza* si intende il periodo di permanenza di un pacchetto dati nella rete, cioè il ritardo che subisce nel percorso tra nodo sorgente e destinazione, mentre per *fedeltà* si intende la qualità del segnale riprodotto dall'applicazione. La sensibilità alla *latenza* è tanto maggiore quanto più si richiede l'interazione tra i due estremi della connessione; le tradizionali chiamate telefoniche, le videoconferenze o il tele-lavoro sono esempi di applicazioni che risentono in modo particolare degli effetti della *latenza*. Le riproduzioni audio o video, invece, sono maggiormente sensibili alla variazione della *fedeltà*. Con riferimento a quest'ultimo parametro si può riformulare la definizione di applicazione intollerante come un'applicazione che deve riprodurre esattamente il segnale trasmesso, a meno di un inevitabile ritardo temporale, senza alcuna perdita di *fedeltà*. Quindi un'applicazione *real-time* intollerante deve appoggiarsi ad un servizio di trasporto dati che garantisca un ritardo massimo per i pacchetti ed una assoluta mancanza di perdite.

Il ritardo può influire sulle prestazioni di un'applicazione di "*signal playback*" in due modi. Da un lato, il valore dell'*offset delay*, determinato sulla base della predizione dei ritardi dei pacchetti, determina il valore della *latenza* di un'applicazione. Dall'altro, i ritardi dei singoli pacchetti possono ridurre la *fedeltà* del playback qualora eccedano il valore dell'*offset delay*; in questi casi, infatti, l'applicazione può scegliere se introdurre distorsione del segnale, modificando l'*offset delay* in modo da riprodurre tutti i pacchetti anche se in ritardo, oppure riprodurre un segnale incompleto, scartando i pacchetti che arrivano oltre l'*offset*. La scelta

dipenderà dal tipo di applicazione, ma in entrambi i casi la fedeltà del segnale riprodotto risulterà compromessa.

2.2 Qualità del servizio

La qualità, nel senso più lato del termine, indica "la totalità delle caratteristiche di un'entità che supportano la sua abilità a soddisfare le necessità di stato e d'impiego". Nel campo delle telecomunicazioni il termine qualità è associato comunemente alla capacità di un'infrastruttura nel differenziare il trattamento di flussi di traffico presenti contemporaneamente sulla rete al fine di soddisfare le attese dell'utente [4].

Gli obiettivi di un sistema per la gestione della QoS (Qualità of Service) sono rivolti a garantire la disponibilità del servizio e la qualità della trasmissione: monitorare la rete per ottenere tempi di risposta predicibili, riuscire a stabilire a priori il livello di servizio che la rete è in grado di offrire, controllare gli accessi per privilegiare eventuali servizi "premium", massimizzare l'uso delle risorse e garantire che siano equamente distribuite. Per realizzare un controllo completo della qualità non si può contare sul fatto che la banda disponibile sia sempre sufficientemente elevata né risulta economico sovradimensionare le risorse di rete (overprovisioning). È necessario intervenire con tecniche di monitoring sull'intero percorso tra end-system, per fornire: garanzie end-to-end, controllo e recupero d'errore in grado mantenere elevate le prestazioni del sistema.

2.2.1 Approcci relativi alla definizione di QoS

Secondo il modello definito nella [5] esistono tre modi di intendere la qualità: **intrinseca**, **percepita**, **valutata**.

- La QoS intrinseca è dovuta a stime di aspetti tecnici, perciò determinata dal progetto della rete di trasporto e delle reti di accesso, terminazioni, e connessioni; è slegata dalla percezione che l'utente ha nei confronti del servizio, mentre è il risultato della comparazione delle caratteristiche di performance attese e i valori misurati. La realizzazione dei requisiti di qualità coinvolge la selezione di appropriati protocolli di trasporto, di meccanismi di assicurazione della qualità e relativi valori dei parametri.
- La QoS percepita riguarda, invece, l'esperienza dell'utente nell'uso di un servizio. Cioè dal confronto delle attese dell'utente e le prestazioni relative al servizio che egli osserva. A causa della differente percezione degli utenti si potrebbe verificare che il solo fatto di garantire particolari parametri di qualità del servizio potrebbe non essere sufficiente a soddisfare le attese degli utenti.
- La QoS valutata è determinante nella scelta dell'utente di continuare o meno ad utilizzare i servizi. La decisione dipende dalla qualità percepita, dal prezzo del servizio e dalla risposta del fornitore ai problemi che gli vengono presentati dagli utenti. Gli enti che lavorano nel campo delle telecomunicazioni, quali l'ITU (International Telecommunication Union) o l'ETSI o l'IETF, non se ne occupano.

I livelli di QoS intrinseca, percepita e valutata potrebbero essere assicurati indipendentemente gli uni dagli altri. Il primo è responsabilità del provider della rete e dipende dalla architettura, planning e management della rete e coinvolge principalmente aspetti tecnici. Un appropriato uso delle possibilità di qualità intrinseca adattate ai particolari servizi offerti, insieme all'analisi di mercato, sono elementi indispensabili per assicurare un elevato livello di QoS percepita; anche in questo caso la responsabilità è del fornitore di servizio; sulla percezione della qualità intervengono, inoltre, la pubblicità e gli sforzi di commercializzazione. La QoS valutata dipende principalmente dalle politiche di tariffazione di un provider così come dalla assistenza all'utente e dal supporto tecnico.

2.2.2.1 Approccio ITU/ETSI

La definizione di QoS nel metodo ITU/ETSI è principalmente orientata alla qualità percepita piuttosto che quella intrinseca. Nonostante ciò, viene introdotta la nozione di prestazioni della rete (NP, Network Performance) riguardanti le caratteristiche tecniche. Viene fatta una chiara distinzione fra QoS, riferita agli effetti percettibili dall'utente, e NP, comprendente tutte le funzioni della rete essenziali per fornire un servizio. I parametri di QoS sono orientati verso l'utilizzatore e non sono traducibili direttamente in parametri di rete. D'altra parte, i parametri di prestazioni della rete determinano la qualità osservata dai clienti che potrebbero eventualmente non ritenerli significativi [6]. È indispensabile, però, che esista una corrispondenza costante fra i parametri di QoS e di NP.

Le prestazioni della rete menzionate sopra, corrispondono alla QoS intrinseca. Sono definite nella [8] come "la capacità di una rete o di una parte della rete di fornire le funzioni relative alle comunicazioni fra gli utenti". NP è definito e misurato in termini di parametri di particolari componenti della rete addetti a fornire un servizio. Questi parametri giocano un ruolo fondamentale in relazione all'efficienza della rete e all'efficacia nel fornire un servizio. Un livello elevato di NP è realizzato da un'adatta progettazione, configurazione, funzionamento e manutenzione del sistema [8]. Alcuni parametri di prestazioni della rete sono definiti da ITU nelle [9],[11].

Affrontando il problema della QoS da diversi punti di vista, ITU ed ETSI distinguono quattro definizioni particolari (Figura 2):

- Requisiti di QoS richiesti dall'utente: rispecchiano le sue preferenze di qualità rispetto ad un particolare servizio. La progettazione del servizio offerto al cliente è fatta in base a tali requisiti, anche se il fornitore di servizio non sempre può soddisfare le aspettative del cliente.
- QoS offerta dal fornitore: può essere influenzata dalle considerazioni di strategia del fornitore di servizio, dal costo sostenuto per quel servizio e da altri fattori [12]. È espresso dai valori assegnati ai parametri comprensibili al cliente (per esempio, "la disponibilità di servizio durante un anno è 99,95 per cento").
- QoS realizzata dal fornitore: è espressa solitamente dallo stesso insieme di parametri. Il confronto della qualità offerta e quella realizzata dà al fornitore di servizio una valutazione preliminare delle prestazioni percepite dall'utente.

- QoS percepita dal cliente: valuta la qualità di servizio confrontando la qualità data dall'esperienza e i suoi requisiti. L'ITU definisce un insieme dei parametri di QoS nella [8]. I parametri relativi alla QoS e alle prestazioni della rete (NP) sono correlati. Accertare le alte prestazioni della rete è cruciale per la riuscita della fornitura del servizio

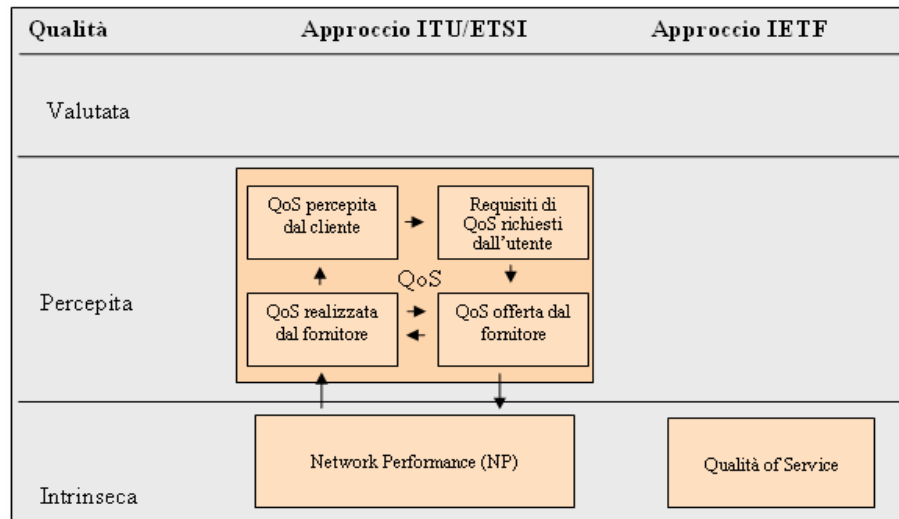


Figura 2 Il modello generale di QoS e gli approcci ITU/ETSI e IETF.

I parametri della QoS offerti possono essere suddivisi come parametri relativi alla rete e parametri non relativi alla rete. L'ultima categoria può essere messa in relazione ai parametri di NP. Per questi parametri esistono, in letteratura, una serie di valori specificati. I network provider ottengono informazioni utili sulle prestazioni della rete in base alle misure dei parametri.

2.2.2.2 Approccio IETF

Il metodo dell'IETF si focalizza sulla QoS intrinseca e non si occupa di QoS percepita. Tale approccio nasce dagli obiettivi principali dell'IETF, interessati all'architettura Internet e relativi sviluppi, all'affidabilità ed efficacia. La QoS è intesa da IETF come "insieme dei requisiti di servizio che si devono incontrare nella rete durante il trasporto di un flusso" [13]. È equivalente alla nozione di NP ITU/ETSI ed è definita in termini di parametri. L'IETF, negli anni scorsi, ha dedicato molta attenzione ad assicurare la QoS nelle reti IP, per la quale ha sviluppato vari meccanismi di QoS. In particolare, ha proposto due architetture significative di rete: Integrated Service (IntServ) [14] e Differentiated Service (DiffServ) [15], e ha standardizzato il protocollo di segnalazione relativo alla prenotazione delle risorse (RSVP, Resource Reservation Protocol), originariamente destinato alla realizzazione dell'IntServ e, successivamente, esteso per altri scopi.

L'IETF definisce alcuni parametri di QoS indipendenti dall'architettura così come i parametri specifici dei componenti della rete, quali le metriche di traffico, gli indicatori del pacchetto, dropper o scheduler, costituenti un'architettura di rete particolare.

2.2.2.3 Sistemi di gestione della qualità

Nelle reti IP tradizionali, la gestione della qualità e la differenziazione nel servizio offerto ai singoli pacchetti è realizzata attraverso l'impostazione di alcuni campi dell'header IP. In particolare, la versione 4 del protocollo IP prevede un campo di un byte, all'interno dell'header dei pacchetti, che specifica il tipo di servizio (ToS - *Type of Service*), come mostrato in Figura 3.

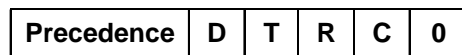


Figura 3 Campo ToS nel protocollo IPv4.

I primi tre bit ("Precedence") vengono utilizzati per specificare la priorità del pacchetto. I quattro successivi (DTRC) contengono informazioni relative all'instradamento e specificano come la rete deve realizzare il compromesso tra ritardo (delay), throughput, affidabilità (reliability) e costo (cost) e l'ultimo bit rimane generalmente inutilizzato e viene settato a zero:

0000 = normal service

0001 = minimize monetary cost

0010 = maximize reliability

0100 = maximize throughput

1000 = minimize delay

Con la diffusione di reti multiservizio, l'utilizzo del campo TOS dell'IPv4 si è rivelato inadeguato. Infatti, oltre che comportare un notevole sovraccarico sui router, offre un modello statico e limitato per la differenziazione dei servizi.

Per rimediare, almeno in parte, a questi problemi, negli ultimi anni sono stati sviluppati diversi meccanismi per la gestione della QoS, riconducibili a due tipologie principali:

- *flow-based*: offrono una specifica qualità di servizio ad ogni singola istanza di applicazione sulla base del flusso di appartenenza. Questo approccio consente di soddisfare le esigenze di ogni singolo utente con una notevole granularità.
- *class-based*: raggruppano in classi pacchetti appartenenti a vari tipi di flussi sulla base di requisiti comuni di QoS, privilegiando la scalabilità alla puntualità del servizio.

Nella versione 6 del protocollo IP si è quindi cercato di migliorare il supporto alla qualità con l'introduzione, sempre all'interno dell'header, di due nuovi campi: *flow label* e *traffic class*. Il primo consente la gestione dei pacchetti a seconda del flusso e viene utilizzato da sistemi *flow-based* quali IntServ o Multi-Protocol Label Switching (MPLS) [16]; il secondo associa ad ogni pacchetto una priorità introducendo il concetto di classe di servizio che sta alla base di architetture DiffServ. Attualmente, si stanno diffondendo soluzioni ibride che consentono di soddisfare entrambi i requisiti di granularità e scalabilità della qualità dei servizi offerti, così come richiesto nello sviluppo di reti di nuova generazione.

2.2.2 Parametri di QoS

La definizione dei parametri relativi alla qualità del servizio e l'individuazione dei loro valori ottimali è un problema di fondamentale importanza e di difficile soluzione. Si pensi, per esempio all'identificazione di parametri oggettivi e misurabili che verifichino la qualità percepita.

L'utente, infatti, è interessato a confrontare lo stesso servizio offerto da fornitori differenti in termini di parametri di prestazioni universali e orientati verso l'utilizzatore. Ciò implica che le prestazioni dovrebbero essere espresse da parametri che :

- considerino tutte le funzioni del servizio dal punto di vista dell'utente;
- si focalizzino sugli effetti percettibili dall'utente, piuttosto che sulle loro cause all'interno della rete;
- siano indipendenti dall'architettura di rete specifica o dalla tecnologia;
- si possano misurare obiettivamente o soggettivamente al punto di accesso di servizio;
- si possano collegare facilmente con i parametri di prestazioni della rete;
- possano essere assicurati ad un utente dal provider di servizio.

Tali parametri non sono definiti univocamente e si può trovare qualche esempio di essi nella [10] in cui l'ITU fa una distinzione in quattro gruppi principali a seconda che gli aspetti a cui si riferiscono siano: il supporto del servizio; l'operabilità del servizio; la servibilità del servizio; la sicurezza del servizio. Il supporto del servizio, in generale, indica la capacità del provider di fornire un servizio e di assistere l'utente nella relativa utilizzazione. I parametri relativi all'operabilità del servizio determinano la misura in cui un servizio può essere agevolmente utilizzato dall'utente. I parametri riferiti alla servibilità determinano la capacità del servizio di essere ottenuto una volta richiesto dall'utente e continuare ad essere fornito senza danno eccessivo per tutta la durata della richiesta. La sicurezza di servizio specifica il livello di protezione del servizio contro gli accessi non autorizzati, l'uso fraudolento, il disastro naturale ed altri danni [10]. Molto più semplice è la valutazione della QoS intrinseca espressa, fondamentalmente, dal seguente insieme di parametri ritenuti significativi per la maggior parte dei servizi basati su IP:

- *Bit rate* di trasferimento dei dati dell'utente disponibile per il servizio;
- *Delay* sperimentato sui pacchetti che passano attraverso la rete; tale ritardo può essere considerato sia in relazione al percorso end-to-end sia ad un particolare elemento di rete;
- *Jitter (variazioni del ritardo dei pacchetti)* anch'esso può essere considerato in una relazione end-to-end o rispetto ad un elemento particolare della rete;
- *Packet loss rate* definito solitamente come il rapporto del numero di pacchetti non consegnati e quelli trasmessi.

Questi parametri vengono misurati sperimentalmente sulla base del trattamento che i pacchetti subiscono durante la trasmissione nella rete. Essi possono essere tradotti in particolari parametri dei componenti dell'architettura di rete e usati per verificare la QoS. La QoS

intrinseca può avere i seguenti attributi dipendenti sia dall'architettura di rete che dalle richieste dell'applicazione:

- end-to-end (per esempio, nel modello di IntServ) oppure limitato ad un particolare dominio o ad un insieme di domini (per esempio, nel modello di DiffServ)
- applicato a tutto il traffico o solo per una sessione particolare o per un numero limitato di sessioni
- unidirezionale o bidirezionale
- QoS garantita o statistica

La qualità del servizio è normalmente una caratteristica della comunicazione end-to-end tra host. Essa dovrebbe essere assicurata per l'intero percorso tra le due estremità ma il percorso potrebbe attraversare diversi sistemi autonomi appartenenti a vari provider di rete e quindi la qualità relativa al singolo sistema autonomo contribuisce alla qualità finale del servizio.

I parametri principali attraverso i quali è possibile valutare la qualità del servizio intrinseca possono essere suddivisi in tre macrocategorie (Figura 4): parametri di performance relativi al trasferimento di pacchetti IP; parametri di performance relativi all'affidabilità della rete; parametri di performance relativi all'accesso al servizio.

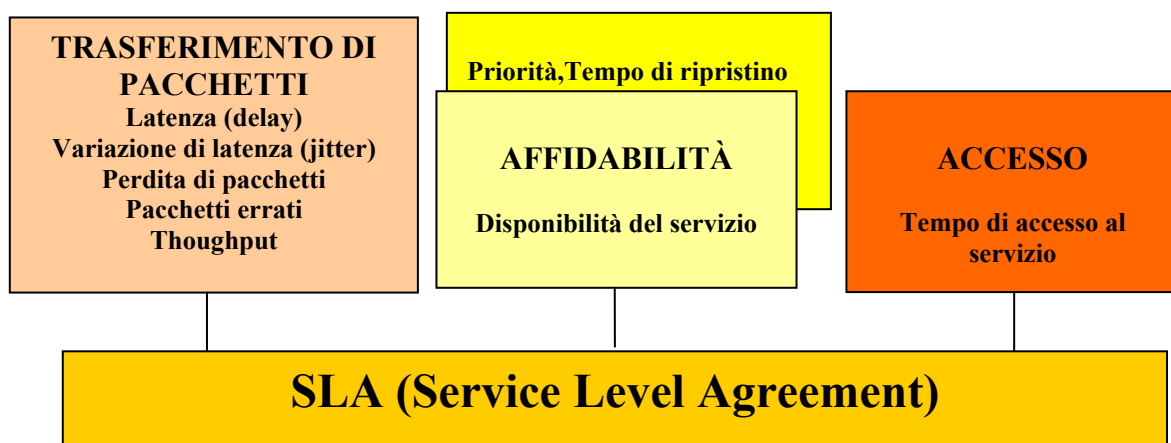


Figura 4 Valutazione della qualità intrinseca.

2.2.3 Parametri di performance relativi al trasferimento di pacchetti IP

Come si è sottolineato più volte, possono essere presi in considerazione differenti parametri. Per garantire un approccio ad alto livello, facilmente riconducibile alle classificazioni solitamente utilizzate nei sistemi per la gestione della qualità, si è deciso di fare riferimento a fonti autorevoli come l'International Telecommunication Union – Telecommunication Standardization Sector (ITU-T) che ha definito due standard internazionali (recommendation) relativi alla QoS. La prima recommendation, Y.1540, definisce i parametri di QoS per il trasferimento di pacchetti su reti basate sul protocollo IP[17]. La seconda, Y.1541, introduce il concetto di Classe di Servizio (CoS) e definisce 6 differenti classi; per

ogni CoS la recommendation indica il valore massimo che i parametri di QoS non devono superare[18].

L'ITU-T modella la rete di riferimento come una concatenazione di sotto reti (*section network*) e interconnessioni di link di scambio (*exchange link*). Le interfacce di queste sottoreti, che prendono il nome di punti di misura (MP: *Measure Point*), sono i perimetri nei quali si possono osservare i protocolli di comunicazione standardizzati. Gli eventi di performance significativi che possono essere contati, cronometrati, o confrontati negli MP prendono il nome di eventi di riferimento (RE). Gli RE specifici sono definiti dal protocollo d'interfaccia.

La recommendation Y.1540 definisce cinque parametri di performance per valutare la qualità della trasmissione tra due terminali utente:

1. **IPTD** (IP Packet Transfer Delay);
2. **IPDV** (IP Packet Delay Variation);
3. **IPLR** (IP Packet Loss Ratio);
4. **IPER** (IP Packet Error Ratio);
5. **SIPR** (Spurious IP Packet Rate).

Nel caso di applicazioni triple-play, può essere utile considerare altri parametri, come lo skew.

Per quanto riguarda i parametri relativi al flusso o al throughput, due indicatori sono definiti nell'appendice dei un'altra recommendation ITU, la I.380 [11]:

- **IPTT** (IP Packet Throughput);
- **IPOT** (Octet based IP Packet Throughput)

Inoltre, la recommendation Y.1541 rimanda alla recommendation Y.1221 per la definizione di quattro descrittori di traffico:

- **RP** (Peak Rate);
- **BP** (Peak Bucket size);
- **BS** (Sustainable Bucket size);
- Maximum allowed packet size.

IPTD (IP Packet Transfer Delay)

"Tempo ($t_2 - t_1$) tra due eventi di riferimento: un evento di ingresso RE_1 al tempo t_1 e un evento di uscita al tempo t_2 dove ($t_2 > t_1$) e ($t_2 - t_1$) $\leq T_{Max}$. Se il pacchetto è frammentato, t_2 è il tempo corrispondente all'ultimo evento di uscita."

L'IPTD rappresenta tutti i vari tipi di ritardi che i pacchetti subiscono nel percorso fra sorgente e destinazione. Include il tempo necessario a stabilire un particolare servizio, dalla richiesta iniziale dell'utente alla ricezione della conferma da parte del server. Il ritardo può essere espresso in vari modi, come tempo di risposta, come ritardo *end-to-end* o *hop-by-hop*; può essere introdotto dalle applicazioni o dai componenti di rete (router, hub, switch, firewall, etc). E' un parametro particolarmente importante perchè ha un impatto molto diretto sulla

soddisfazione dell'utente che talvolta percepisce come ritardi anche gli effetti prodotti da altri parametri di rete, come ad esempio il throughput.

L'IPTD viene misurato sia nel caso in cui il trasferimento del pacchetto abbia successo, sia nel caso in cui si verifichino degli errori di trasmissione. È un ritardo *one-way*, ovvero esprime il tempo di attraversamento in una direzione e corrisponde alla somma di due contributi:

1. il tempo necessario al primo bit per attraversare la rete. Questa componente è funzione della distanza fisica, del numero di apparecchiature attive e passive lungo il percorso e del carico istantaneo di rete;
2. il tempo necessario a trasmettere tutti i bit del pacchetto, che varia in funzione della velocità di trasmissione della linea.

Alternativamente, si può calcolare il tempo di andata e ritorno, ovvero il *Round Trip Time* (RTT), definito come:

$$\text{RTT} = 2 * (\text{one-way delay}) + \text{tempo di switching}$$

dove per tempo di switching si intende il tempo che intercorre tra la ricezione del pacchetto e la sua ritrasmissione.

Assieme all'IPTD, la recommendation Y.1540 definisce il *Mean IP Packet Transfer Delay*, ovvero la media aritmetica dei valori di IPTD per una popolazione di interesse.

IPDV (IP Packet Delay Variation)

"La variazione del ritardo (v_k) per il trasferimento di un pacchetto k tra due MP è la differenza tra il ritardo assoluto (x_k) per il trasferimento del pacchetto e il ritardo $d_{1,2}$ per il trasferimento di un pacchetto di riferimento: $v_k = x_k - d_{1,2}$. Il ritardo di riferimento, $d_{1,2}$, tra sorgente e destinazione è il ritardo di trasferimento assoluto del primo pacchetto tra i due MP."

L'IPDV definisce la variazione del ritardo che subiscono i vari pacchetti (*jitter*) nel percorso fra sorgente e destinazione. Varia soprattutto in funzione del carico di rete istantaneo, di fallimenti nella rete e di non linearità nelle apparecchiature attive. Può essere causato da problemi di routing, congestioni o da instabilità ed errori nei componenti di rete (router o switch). Valori elevati di jitter possono causare rimescolamento dei pacchetti ed errori di temporizzazione. Nel caso di applicazioni fortemente sensibili agli effetti di questo parametro (*streaming audio-video*) è possibile limitare l'influenza del jitter sulla qualità del servizio utilizzando un buffer in ricezione. Questa soluzione comporta un aumento del ritardo (IPTD) percepito dall'utente finale ed è efficace, a seconda delle dimensioni del buffer, solo entro un certo range di jitter.

IPLR (IP Packet Loss Ratio)

"Rapporto tra il numero complessivo di pacchetti persi ed il numero complessivo di pacchetti trasmessi in una popolazione di interesse".

$$IPLR = \frac{N_{Persi}}{N_{Trasmessi}}$$

L'IPLR rappresenta il numero di pacchetti persi durante la trasmissione e viene generalmente espresso in termini di percentuale. I pacchetti possono andare perduti perchè scartati dai router in caso di congestione, perchè male indirizzati oppure a causa di errori di trasmissione.

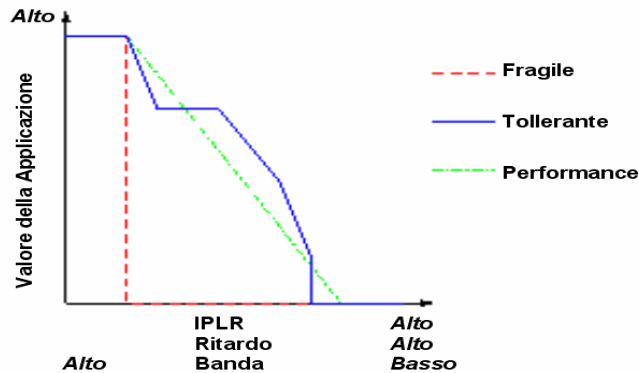


Figura 5 Categorie di applicazioni in funzione della reazione alla perdita di pacchetti.

Le applicazioni possono reagire in maniera differente alla perdita di pacchetti. Tipicamente si individuano tre categorie (Figura 5), definite sulla base dei valori dell'IPLR, della latenza e della banda richiesta:

- Fragile: se la perdita di pacchetti supera una data soglia il valore dell'applicazione diventa nullo.
- Tollerante: la perdita di pacchetti può essere tollerata, ma al crescere dell'IPLR diminuisce il valore dell'applicazione.
- Performance: l'applicazione può tollerare valori molto elevati di IPLR, ma con bassi livelli di performance.

IPER (IP Packet Error Ratio)

"Rapporto tra il numero complessivo di pacchetti errati ed la somma totale dei pacchetti trasferiti con successo e dei pacchetti errati in una popolazione di interesse".

$$IPER = \frac{N_{Errati}}{N_{Corretti} + N_{Errati}}$$

Per servizi non real-time, è possibile utilizzare dei meccanismi di identificazione e recupero di errori. Il protocollo IP non prevede nessuna funzionalità per la correzione o la ritrasmissione di pacchetti corrotti, che però possono essere gestite dal livello di trasporto qualora si utilizzi il protocollo TCP. Per quanto riguarda invece i servizi real-time, la ritrasmissione dei pacchetti errati comporterebbe un aumento del IPDT. Pertanto, per questo tipo di applicazioni è necessario monitorare la rete per ridurre quanto più possibile il valore dell'IPER.

SIPR (Spurious IP Packet Rate)

"Il SIPR in un ingresso MP è il rapporto tra il numero totale di pacchetti IP spuri osservati nell'ingresso MP durante uno specifico intervallo di tempo e l'intervallo di tempo considerato (i.e., il numero di pacchetti IP spuri per secondo)".

Skew

Lo skew è un tipo di ritardo che si può manifestare tra due flussi di dati che vengono trasmessi contemporaneamente da un terminale. Per esempio, nel caso di applicazioni di IPTV è necessaria la sincronizzazione perfetta tra stream audio e video

Nel caso di due flussi di pacchetti trasmessi su due canali diversi A e B, lo skew misura la differenza tra il ritardo dei due canali (Figura 6).

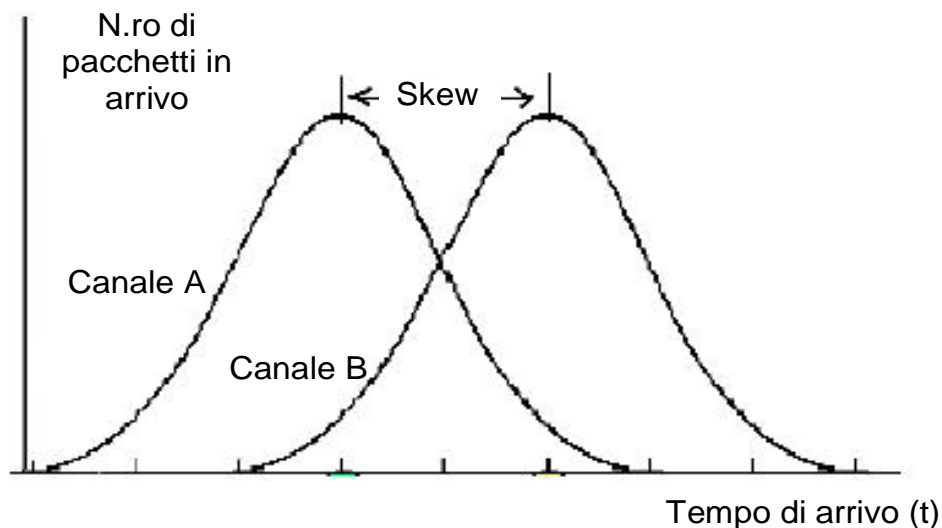


Figura 6 Skew

IPPT (IP Packet Throughput)

"Per una popolazione di interesse, il throughput di un pacchetto IP basato sull'ottetto in un'uscita MP è il rapporto tra il numero totale di pacchetti IP trasferiti con successo durante uno specifico intervallo di tempo e l'intervallo di tempo considerato (i.e. il numero di pacchetti IP trasferiti con successo per secondi di servizio)".

L'IPPT misura la velocità con cui i pacchetti IP vengono trasferiti con successo.

IPOT (Octet based IP Packet Throughput)

"Per una popolazione di interesse, il throughput di un pacchetto IP in un'uscita MP è il rapporto tra il numero totale di ottetti trasmessi per pacchetti IP trasferiti con successo durante uno specifico intervallo di tempo e l'intervallo di tempo considerato (i.e. il numero di ottetti contenuto nei pacchetti IP trasferiti con successo per secondi di servizio)".

L'IPOT misura il throughput in termini del numero di ottetti contenuti nei pacchetti che vengono trasmessi con successo.

2.2.4 Parametri di performance relativi all'affidabilità

L'affidabilità della rete è la percentuale di tempo in cui il sistema funziona correttamente, particolarmente importante nel caso di applicazioni mission-critical. I parametri che notoriamente vengono utilizzati per valutare l'affidabilità di un sistema sono:

- **MTTF o MTFF (Mean Time to First Failures):** tempo medio prima che si verifichi il primo guasto.
- **MTBF (Mean Time Between Failures):** tempo medio che intercorre tra due guasti successive in un sistema riparabile.
- **MDT (Mean Down Time):** tempo medio tra l'istante in cui avviene il guasto e il ripristino totale del sistema. Include il tempo di individuazione dei guasti, il tempo di riparazione e il tempo di ripristino.
- **MTTR (Mean Time Ripare):** tempo medio reale per riportare il sistema ad una condizione di normale funzionamento.
- **MUT (Mean Up Time):** tempo medio del guasto libero.
- **MTU = MTBF – MDT**

Nella recommendation Y.1540, l'ITU-T valuta l'affidabilità di una rete in termini di parametri relativi alla disponibilità del servizio, definita in funzione del parametro IPLR:

"Considerando uno specifico flusso, una porzione di rete è detta disponibile in un dato periodo di osservazione se il valore dell'IPLR misurato per quel flusso è minore di una data soglia, c1. In caso contrario, la porzione di rete è detta indisponibile. Il valore specificato dalla Y.1540 per c1 è 0.75."

Sulla base di questa definizione, la recommendation Y.1540 definisce due parametri di performance relativi alla disponibilità della rete:

- **PIA (Percent IP service Availability):** *"Per una data porzione di rete e un dato flusso, la PIA è la percentuale di tempo disponibile programmato durante il quale la porzione di rete mantiene il flusso in uno stato di disponibilità."*
- **PIU (Percent IP service Unavailability):** *"La PIU è il complemento della PIA, ovvero la percentuale di tempo disponibile programmato durante il quale il flusso è in uno stato di indisponibilità." E data da $PIU = 100 - PIA$*

2.2.5 Parametri di performance relativi all'accesso al servizio

Tra i parametri che maggiormente influenzano la qualità dal punto di vista dell'utente, particolarmente importante è il tempo di accesso al servizio, ovvero il tempo che intercorre tra la richiesta dell'utente e la conferma da parte del server. Le prestazioni di un sistema relativamente all'accesso ad un servizio sono fortemente correlate al problema della servibilità. Tutte le tipologie di applicazione sono sensibili a eventuali ritardi durante la fase di accesso.

Gli utenti, quando fanno richiesta ad un ISP di un particolare servizio, ad esempio la ricerca di un'informazione o l'acquisto di un prodotto sul web, desiderano poterlo ottenere con facilità, seguendo indicazioni chiare e in tempi quanto più veloci possibile. Nel caso del web-browsing, prove sperimentali hanno dimostrato che il tempo medio entro il quale un utente è disposto ad attendere il caricamento di una pagina è non più di 8 secondi. Eccessivi ritardi creano non soltanto impazienza, ma anche diffidenza e confusione. Nel caso in cui il costo di un servizio sia proporzionale alla durata della connessione, per esempio applicazioni mobili, questo problema diventa ancora più rilevante.

Per poter stimare i tempi d'accesso riferiti a ciascun servizio è necessario disporre di prove sperimentali che diano conto della disponibilità dell'utente di attendere di fronte a tempi di connessione e di risposta più o meno lunghi.

2.3 Classificazione dei servizi e definizione dei requisiti ottimali

In questa sezione sono definiti i valori numerici dei parametri chiave di QoS che dovrebbero essere garantiti dalle reti IP di nuova generazione in un percorso tra terminali utente. In particolare, per offrire un supporto pratico alla gestione della QoS nelle comunicazioni tra end-user e ISP, o tra due ISP, i servizi di riferimento sono stati raggruppati in Classi di Servizio (CoS) ciascuna delle quali necessita di specifici requisiti di qualità. Per semplificare la trattazione del problema e minimizzare la complessità delle tecnologie di rete necessarie all'implementazione della soluzione proposta, si è scelto di definire un numero preciso e limitato di classi di QoS.

Le applicazioni possono essere classificate in vario modo a seconda dei parametri che vengono presi in considerazione (priorità, sensibilità al ritardo, interattività, ecc.) e delle metriche (assolute o relative) che vengono utilizzate per definire le prestazioni della rete in termini di qualità del servizio. Ad esempio, la gestione della qualità del servizio nelle reti ATM si basa sulla definizione di 5 classi di applicazioni stabilita sulla base di altrettanti parametri di traffico: Constant Bit Rate (CBR); Real-time Variable Bit Rate (rt-VBR); Non-real-time Variable Bit Rate (nrt-VBR); Available Bit Rate (ABR); Unspecified Bit Rate (UBR); e Guarantees Frame Rate (GFR): garantisce la disponibilità di banda per frame di celle multiple. Anche i modelli *flow-based* e *class-based* definiti dall'IETF per il supporto alla qualità del servizio nelle reti IP di nuova generazione si basano entrambi su una classificazione delle applicazioni definita in funzione delle specifiche esigenze. L'IntServ stabilisce tre possibili livelli di servizio: Guaranteed Service, Controlled Load Service e best effort. Il DiffServ utilizza, invece, tre classi principali, corrispondenti a tre diversi trattamenti sui vari nodi della rete (PHB), all'interno delle quale definisce delle sottoclassi sulla base di parametri specifici:

1. *Expedited Forwarding Behaviour* (EF): appartengono a questa classe, detta anche *Premium Service*, tutte le applicazioni che necessitano di un servizio di rete con caratteristiche minime garantite, come ad esempio servizi mission-critical e real-time. Per traffico EF, la rete assicura banda minima con bassa latenza, basso jitter e poche perdite. In altre parole, questo modello di servizio emula una linea dedicata e dovrebbe impegnare solo una piccola percentuale della capacità totale della rete.
2. *Assured Forwarding Behaviours* (AF): comprende tutte le applicazioni che richiedono un livello di affidabilità maggiore rispetto al servizio best effort, senza però stretti vincoli sui parametri della QoS. A sua volta, AF è suddivisa in quattro sottoclassi indipendenti, ognuna con le proprie risorse di rete: AF1, AF2, AF3 e AF4. Queste classi hanno priorità crescente da AF1 a AF4. All'interno di ogni sottoclasse è definita una ulteriore suddivisione relativa a tre diversi livelli di probabilità nello scartare un pacchetto in caso di congestione (*drop precedence*). In

questo modo si ottengono 12 differenti classi AF indicate con la notazione AF_{IJ}, dove I indica la sottoclasse e J le regole di discarding. Valori più bassi degli indici significano qualità più alta.

3. *Default Behaviour*: corrisponde al servizio best effort offerto dal protocollo IP.

La varietà delle possibili soluzioni alla gestione dei servizi in reti eterogenee impone, quindi, la necessità di definire un modello di classificazione che possa essere flessibile e allo stesso tempo coerente con le diverse architetture di QoS. Lo schema che meglio si presta alla realizzazione di questo obiettivo e al quale si è scelto di fare riferimento è quello definito dall'ITU-T nella recommendation Y.1541, nella quale vengono introdotte sei classi di servizio e per ciascuna delle vengono specificati i valori dei parametri di performance definiti nella recommendation Y.1540.

2.3.1 Classi di QoS

Nella recommendation Y.1541 dell'ITU-T ogni classe di servizio specifica una combinazione di valori limite per un sottoinsieme di parametri di performance. In TABELLA 1 sono specificate le corrispondenze fra i servizi di riferimento e le sei classi di QoS definite dall'ITU.

Le classi e i relativi requisiti di QoS vengono riferiti a ogni singolo *flusso di pacchetti IP* tra due MP che delimitano un percorso end-to-end in una rete IP. A titolo esplicativo, per *flusso di pacchetti IP* si intende il traffico associato ad una connessione, o stream connectionless, avente gli stessi host sorgente (SRC), host di destinazione (DST), classe di servizio e identificativo di sessione.

Le classi 0 e 1 sono caratterizzate da valori molto specifici e restrittivi di parametri quali: ritardo di trasferimento dei pacchetti e perdita dei pacchetti, oltre che variazione del ritardo (jitter).

Le classi 2 e 3 invece presentano dei vincoli rispetto ai primi due parametri mentre sono del tutto insensibili alla variazione del ritardo dei pacchetti, che quindi risulterà non specificato. Le classi 0 e 2 differiscono da quelle 1 e 3 essenzialmente per le restrizioni relative al ritardo dei pacchetti che nel primo caso sono più forti. Per quanto riguarda la classe 4, essa è caratterizzata da un limitato valore di packet loss e una restrizione molto blanda relativa al ritardo. Infine l'ultima classe, la 5, è quella che non fornisce nessuna garanzia sulla performance del servizio e che corrisponde al servizio best effort tradizionale.

2.3.2 Gli SLA

Per andare in contro alle esigenze degli utenti e stabilire le caratteristiche specifiche relative alla qualità del servizio offerto, solitamente viene stipulato un accordo tra utente e provider di rete, detto *Service Level Agreement (SLA)*. Secondo la definizione dell'ITU-T, un *SLA* è "l'insieme degli accordi negoziati tra l'utente e il fornitore di servizi, relativi al livello di

caratteristiche del servizio e all'insieme delle metriche associate". Il contenuto degli SLA dipende dal tipo di servizio offerto e include gli attributi richiesti per gli accordi negoziati.

Nella accezione più generale del termine, gli SLA comprendono caratteristiche tecniche e parametri del servizio così come aspetti di tipo legale e aspetti relativi alla tariffazione del servizio. Nello specifico, vengono definiti i valori che possono assumere i parametri di QoS, quali la banda massima utilizzabile, il prezzo del servizio, ecc. Inoltre, vengono presi accordi sul comportamento da tenere se i vincoli concordati non possono essere soddisfatti.

TABELLA 1 SERVIZI DI RIFERIMENTO E CLASSI DI QoS

Servizio	Classe di QoS
Voce	0,1
Videotelefonia	0,1
Videoconferenza	0,1
Gaming interattivo real-time	1
Gaming interattivo non-real-time	2,3
Telemetria	4
Telemetria bidirezionale	1
TELNET	4
Audio streaming	4
IPTV: Video on Demand	4
IPTV: broadcast TV	1
FTP	4, 5
Messagistica vocale	2,3
Web-browsing	4, 5
Transazioni on-line (es. E-	4
Posta elettronica con server in rete	4, 5
FAX	5
Servizi a bassa priorità (es. SMS)	5
E-mail	5
DNS	5
Server-to-server	5
Segnalazione	2
Radio IP	1, 2
Telemedicina	0

Capitolo 3

Ottimizzazione di una rete NGN

Il problema dell'ottimizzazione di una rete di telecomunicazione, di qualsiasi natura essa sia, coinvolge una moltitudine di aspetti che vanno dall'architettura di trasporto alle tipologie di servizi offerti piuttosto che alle particolari tecnologie implementate per il management della rete. Si pensi alle complicazioni che si verificano in una rete multiservizio multidomino e multivendor in cui si hanno problematiche legate ai servizi offerti, alla comunicazione tra domini differenti e alle diverse apparecchiature costituenti l'architettura di rete. In questo tipo di rete complessa l'incapacità di prevedere le curve di traffico dei vari servizi profondamente diverse tra loro e tutt'altro che uniformi, rende il problema dell'ottimizzazione ancora più importante. Per questo motivo il processo di ottimizzazione può essere fortemente influenzato anche dalle decisioni prese in fase di progetto ma, soprattutto, sarà frutto di una attenta analisi e di uno studio ad hoc eseguito sulla rete da ottimizzare.

Le fasi principali che consentono di approdare ad un buon processo di ottimizzazione sono rappresentate nella Figura 7 e descritte di seguito.

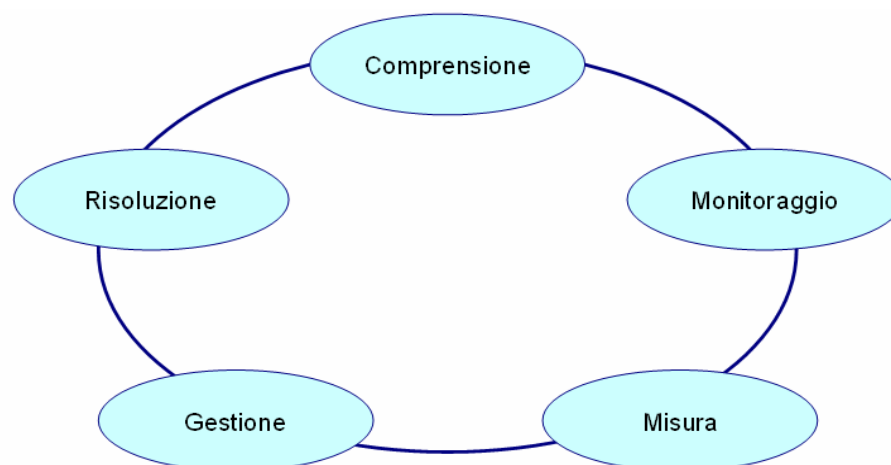


Figura 7 Fasi necessarie al processo di ottimizzazione.

Fase 1: Comprensione e identificazione

Conoscere completamente la rete e l'ambiente d'applicazione è il primo passo da compiere. In particolare, si devono conoscere le risorse disponibili a soddisfare le richieste degli utenti ma vi è anche la necessità di strumenti di rete in grado di identificare quali applicazioni stanno usando le risorse di rete. Da queste informazioni, allo scopo di migliorare la distribuzione delle risorse, si possono mettere in pratica delle strategie di mercato che promuovano dei particolari servizi in fasce orarie in cui si hanno più disponibilità.

Fase 2: Monitoraggio

Una volta ottenute le informazioni relative alle risorse e alla rete nel suo complesso, si passa alla fase di monitoraggio oculato delle performance relative a risorse chiave che guidano il business value. Attraverso l'uso di opportuni tool che forniscono un monitoraggio della rete sia real time che nel tempo (storico degli eventi), si può osservare quanto le risorse di rete siano sfruttate. Queste informazioni verranno utilizzate per ottenere il massimo delle performance e dell'affidabilità.

Fase 3: Misura

Attraverso la misura è possibile avere un'informazione oggettiva delle performance di rete nell'istante considerato oltre che il loro andamento nel tempo. Inoltre la misura consente il confronto tra i valori di alcuni parametri di rete misurati e i rispettivi range previsti o accettabili in relazione ai servizi offerti.

Fase 4: Management – Gestione

Questo step utilizza le informazioni ottenute grazie alle fasi precedenti per determinare quali componenti e applicazioni della rete necessitano di una gestione più accurata. La gestione consiste nel setting degli allarmi o notifiche relative a metriche di performance di elementi critici della rete o delle applicazioni. Le violazioni delle soglie di allarme si basano sulla disponibilità delle risorse, sulle variazioni e consistenza delle performance nel tempo, inoltre, consentono di individuare i componenti che hanno maggior impatto sulla produttività.

Fase 5: Risoluzione - Miglioramento

Attraverso i dati raccolti si attiva un meccanismo a retroazione secondo il quale si apportano i doverosi assestamenti nella rete, si valutano le variazioni, si introducono altri accorgimenti e, per affinamenti successivi, si dovrebbe raggiungere un livello di ottimizzazione piuttosto elevato. L'ottimizzazione delle performance di rete può essere correttiva o perfettiva. Nel primo caso, lo scopo è quello di risolvere o rimediare ai problemi che si sono manifestati, individua per esempio da uno o più allarmi, o che si stanno manifestando. Nel secondo caso, invece, lo scopo è quello di migliorare le performance della rete perfino quando non esistono espliciti problemi che si devono risolvere. Uno degli aspetti più importanti di questa fase, però, è quello di eseguire, per quanto possibile, un'azione preventiva in grado di reagire agli eventi futuri che vengono previsti mediante analisi dagli eventi precedenti.

Come si è detto l'ottimizzazione può riguardare diversi aspetti e in relazione agli aspetti di ingegneria del traffico, secondo la RFC3272, l'ottimizzazione può essere realizzata attraverso la gestione delle capacità della rete e la gestione del traffico. Nel capacity management si includono: capacity planning, controllo del routing e gestione delle risorse. Per quanto riguarda le risorse di rete di maggior interesse si considerano: la *banda* dei link, lo *spazio nei buffer* e le *risorse computazionali*. Mentre, la gestione del traffico include:

- funzioni di controllo del traffico nei nodi (condizionamento del traffico, gestione delle code, scheduling)
- altre funzioni che regolano il flusso del traffico nella rete o che regolano e controllano l'accesso alle risorse della rete dei diversi pacchetti o diversi flussi di traffico

Nei paragrafi successivi, una volta presentata l'architettura di riferimento, si passerà all'analisi degli attributi della rete. Tali attributi consentono l'individuazione dei parametri da ottimizzare in una rete NGN. Seguiranno le sezioni dedicate alle metodologie di ottimizzazione studiate per due particolari contesti: l'architettura di backbone e il dominio wireless relativamente al servizio di video streaming.

3.1 Architettura di riferimento

L'architettura di riferimento sarà l'MPLS-TE con supporto di DiffServ (DS-TE) [19], in quanto risulta quella che più efficientemente sia in grado di distribuire le risorse di rete tra le varie tipologie di traffico in maniera puntuale. Introduciamo le più importanti caratteristiche di queste tre componenti essenziali dell'architettura di rete, dando particolar risalto agli aspetti legati al Traffic Engineering.

3.1.1 Il DiffServ

Nell'architettura Differentiated Service (DiffServ) [20] vengono definite delle classi di servizio (Classes of Service o CoS) che consentono di trattare il flusso di dati attraverso degli aggregati di traffico. Ciascun aggregato subirà in ciascun nodo della rete un trattamento differenziato utilizzando un meccanismo di tipo Per-Hop, il DiffServ, infatti, è un approccio connectionless e non consente di garantire l'allocazione della capacità trasmissiva nei link attraversati. Negli edge-router, i pacchetti vengono marcati come appartenenti ad una classe ben definita in modo da determinare il trattamento Per Hop Behavior (PHB) che subiranno nei successivi hop. Il PHB è realizzato mediante una combinazione di schema scheduling e management nelle code che tengano conto dei diversi requisiti caratterizzanti le tipologie di traffico considerate.

Il DiffServ attraverso questa differenziazione del traffico realizza una qualità di servizio molto alta ma la sua efficacia complessiva è limitata dal tradizionale shortest path routing impiegato dal protocollo IP. Tale limite può essere superato attraverso le capacità del traffic engineering offerte dal MPLS.

3.1.2 L'MPLS

L'elemento di base del Multi Protocol Label Switching (MPLS) [21],[22],[23],[24] è quello di attaccare una short label di lunghezza fissa ai pacchetti in ingresso al dominio MPLS. Tale operazione viene eseguita dagli edge router LER (label edge router), mentre i router in grado di inoltrare i pacchetti (forwarding) all'interno del dominio sono chiamati label switching router (LSR). I pacchetti vengono instradati sulla base dell'etichetta che gli è stata assegnata trascurando le indicazioni presenti nel header IP. L'assegnamento delle label viene effettuato sulla base del forwarding equivalent class (FEC), ovvero quando i pacchetti presentano la stessa FEC allora saranno etichettati allo stesso modo e seguiranno lo stesso percorso nella rete MPLS. Un percorso seguito da una FEC prende il nome di Label Switched Path (LSP). Per stabilire, mantenere, demolire un LSP vengono usati due protocolli: il label distribution protocol (LDP) e un'estensione del resource reservation protocol (RSVP).

3.1.3 Il TE

Lo scopo del Traffic Engineering [25] è quello di ottimizzare l'uso delle risorse disponibili migliorando le performance della rete mediante il controllo del traffico nella rete stessa. In linea generale implica l'applicazione di tecnologie e principi scientifici concernenti la misura, la modellizzazione, la caratterizzazione e il controllo del traffico IP per raggiungere degli obiettivi di performance predefiniti. Per questo motivo il TE gioca un ruolo fondamentale nel processo di ottimizzazione della rete.

Il miglioramento delle performance si deve realizzare sia a livello di risorse che a livello del traffico della rete. In relazione alle risorse, si deve migliorare il loro impiego in termini sia economici che di affidabilità. Mentre, le misure relative al traffico includono aspetti quali diminuzione del delay, del jitter e del packet loss e massimizzazione del throughput.

3.2.1.1 Obiettivi TE

Gli obiettivi del traffic engineering si possono suddividere in due insiemi principali:

- Obiettivi orientati al traffico - Si tratta principalmente di migliorare la QoS dei flussi di traffico. Nel caso Best Effort principalmente la perdita, nel caso di più classi di servizio anche il ritardo.
- Obiettivi orientati alle risorse - Si tratta di ottimizzare l'uso delle risorse, in particolare effettuare una gestione efficiente della banda evitando di trovarsi in condizioni in cui parte delle risorse sono sovraccariche e parte sotto-utilizzate.

Dal punto di vista del traffico, l'obiettivo globale potrebbe essere riassunto come: minimizzazione della congestione "prolungata" ovvero se non è possibile in alcun modo evitare la congestione fare in modo che si verifichi per il periodo più breve possibile. La congestione si presenta, in genere, quando: le risorse risultano inadeguate rispetto al carico offerto, oppure, i flussi di traffico sono distribuiti in modo inefficiente nella rete. Il primo caso può essere affrontato usando due metodi (applicabili anche contemporaneamente) quali:

espansione della capacità e meccanismi di controllo di congestione su traffici elastici (controllo dei tassi, controllo di flusso a finestra, gestione delle code, controllo dello scheduling, ecc.). Il secondo caso può essere affrontato e gestito dal TE, il quale dovrebbe fornire le funzionalità di un meccanismo di controllo retroazionato adattativo composto da: un insieme di elementi di rete interconnessi; un sistema di monitoring delle prestazioni della rete; un insieme di strumenti di gestione delle configurazioni.

L'ingegneria del traffico stabilisce una politica di controllo, osserva il sistema tramite il monitoring, applica delle azioni di controllo per guidare la rete nello stato desiderato in accordo con la politica definita. Le azioni possono essere intraprese in risposta ad uno stato corrente o sulla base di previsioni ottenute tramite modelli revisionali. Per quanto concerne le azioni di controllo, esse dovrebbero comprendere sia la modifica dei parametri di gestione del traffico (parametri di scheduling, di queue management, ...), sia la modifica di parametri associati al routing (metriche, costi, percorsi) che la modifica di attributi e vincoli associati alle risorse.

3.1.4 Perché MPLS per il TE

In linea di principio gli IGP (Interior Gateway Protocol) potrebbero essere lo strumento più naturale per l'applicazione delle tecniche di TE, ma nella pratica non sono adatti a fornire un supporto adeguato ed anzi sono spesso fonte di inefficienze perché si basano su meccanismi shortest path che generano spesso condizioni di uso inefficiente delle risorse, inoltre usano metriche semplici ed additive che non tengono conto della disponibilità di banda e delle caratteristiche del traffico. Questi difetti generano condizioni di congestione.

La soluzione "tradizionale" utilizzata sino a qualche tempo fa era l'IP over ATM o IP over Frame-relay, utilizzando i VP (Virtual Circuit) ATM per creare delle topologie virtuali fra router indipendenti dalla rete fisica effettivamente presente. Questo modo di operare tramite un overlay model ha diversi vantaggi:

- permette di realizzare tramite VC un routing vincolato;
- permette la configurazione di percorsi su base amministrativa;
- permette la aggregazione di percorsi;
- da un supporto per il CAC, policing e shaping.

In questo senso MPLS può essere applicato al posto di ATM e Frame-relay, con ulteriori vantaggi quali ad esempio: il livello di integrazione con IP, la possibilità di automatizzare gli interventi di TE e il minor impegno in termini di complessità e segnalazione.

3.1.5 La scelta: il DS-TE

Appurato che l'architettura MPLS-TE con supporto per il DiffServ sia il criterio ad oggi più appropriato per implementare la qualità del servizio nelle reti IP, l'architettura di riferimento proposta sarà appunto la DiffServ-aware Traffic Engineering (DS-TE), il cui scopo

primario sarà è quello di garantire le risorse di rete separatamente per ogni tipo di traffico. Il TE viene effettuato sulla Class of Service, mediante procedure di *admission control* e *resource reservation* calibrate sulle classi di traffico. Si introducono in questo ambito i concetti di *Class Type* (CT), un raggruppamento di TT basato sulla loro CoS, e di *Bandwith Constraint* (BC), un limite sulla percentuale di risorse trasmissive di un link che un CT può utilizzare. Le relazioni tra CT e BC sono definite nei cosiddetti *Bandwidth Constraint Models*, che sanciscono le regole per la suddivisione delle risorse trasmissive tra i vari CT.

3.2 Elementi chiave

Sulla base delle caratteristiche dell'architettura di riferimento appena descritta possiamo delineare i problemi o aspetti fondamentali relativi al TE su MPLS che possono entrare in gioco nel processo di ottimizzazione. Il primo problema concerne come mappare i pacchetti sulle forwarding equivalent class; il secondo problema riguarda come mappare le FEC sui traffic trunk e il terzo, invece, è relativo a come devono essere mappati i traffic trunk sulla topologia fisica della rete attraverso gli LSP. Lo studio che viene presentato in questo documento non è focalizzato sui primi due problemi che, seppur molto importanti nell'ottica dell'ottimizzazione della rete, vengono dati per già risolti. Invece ci concentriamo sull'ultimo aspetto cercando di fornire un metodo che consente di dare una soluzione che ottimizzi le performance della rete.

Gli obiettivi chiave di performance possono essere suddivisi in due macrocategorie: quelli orientati al traffico e quelli orientati alle risorse. Nel primo caso si tiene conto degli aspetti che migliorano la qualità di servizio del flusso di traffico quali: minimizzazione del packet loss, minimizzazione del ritardo (peak to peak packet delay, loss ratio, maximum transfer delay), massimizzazione del throughput e soddisfazione dei requisiti espressi dai Service Level Agreement (SLA). Per ciascuna classe di servizio esistono dei limiti accettabili di questi valori così come è stato evidenziato dai lavori precedenti sulla determinazione delle caratteristiche e dei vincoli per ciascun servizio è pertanto necessario ottimizzare il sistema per soddisfare tali vincoli. Nel secondo caso si considerano gli aspetti legati alla corretta ed efficiente utilizzazione delle risorse. Lo scopo principale è che un sottoinsieme delle risorse non divenga sovrautilizzato provocando delle congestioni, mentre altri sottoinsiemi costituiti da percorsi alternativi siano sotto utilizzati. Un ruolo fondamentale è giocato, quindi, dalla gestione delle risorse di banda.

Relativamente alle congestioni, ribadendo i concetti esposti precedentemente, non solo si deve cercare di minimizzarle al massimo ma, qualora si presentassero, è necessario che abbiano una durata minima, per cui vanno previste delle tecniche di recupero, protezione nonché di controllo.

Le azioni per il controllo del traffico e delle risorse consistono nella modifica dei parametri di management del traffico, e/o nella modifica dei parametri associati al routing e/o la modifica degli attributi e dei limiti associati alle risorse; in ogni caso, deve essere ridotto al minimo

possibile il livello di intervento manuale, in quanto sebbene possa essere efficace per la gestione di reti di piccolissime dimensioni è del tutto inadatto alla gestione su larga scala.

Nella sua accezione più generale in termini matematici ottimizzare significa trovare il miglior valore di una funzione F all'interno di un dominio specificato di parametri modificabili (x_1, \dots, x_n) e per certi vincoli fissati. La funzione F prende il nome di funzione di costo. Lo scopo di questo lavoro sarà quindi quello di determinare delle funzione di costo che mettono in relazione i parametri attraverso cui si attua l'ottimizzazione.

Lo studio delle caratteristiche e quindi degli attributi relativi al traffico e alle risorse [27] consentirà la determinazione sia dei parametri da ottimizzare che quelli attraverso cui l'ottimizzazione può essere concretizzata, ovviamente tale studio va adattato al caso specifico che si sta analizzando. Per esempio, nella prima parte della tesi, si studia l'architettura di backbone NGN e si presentano delle tecniche che possono ottimizzarne le prestazioni relativamente alla gestione del traffico, mentre nella seconda parte vengono determinati dei fattori chiave su cui si può intervenire a partire dalle caratteristiche dell'architettura di rete wireless e della tipologia del servizio considerato e si presenta una tecnica ad hoc che ottimizza la qualità del servizio.

3.2.1 Attributi relativi al traffico

Gli attributi di base dei traffic trunk (aggregati di flussi di traffico con caratteristiche simili) relativi al Traffic Engineering sono:

- Attributi dei parametri di traffico
- Attributi di selezione e mantenimento di percorso generico
- Attributi di priorità
- Attributi di preemption
- Attributi di resilienza
- Attributi di policing

Attributi dei parametri di traffico

Possono essere utilizzati per descrivere le caratteristiche dei flussi di traffico (più precisamente delle FEC) in modo da rendere possibile l'aggregazione in traffic trunk. Ogni caratteristica potrebbe includere: peak rate, mean rate, dimensione dei burst consentita ecc. Da tali attributi si evince quale sarà l'allocazione di risorse richiesta (es. banda da allocare sui vari link).

Attributi di selezione e mantenimento di percorso generico

Definiscono le regole per la selezione delle strade che i traffic trunk devono prendere, così come le regole per il mantenimento dei percorsi appena stabiliti. Si rende necessario l'uso di uno schema di routing basato su dei vincoli di banda (**Constraint Band Routing**) per la selezione del percorso; inoltre l'implementazione dell'MPLS deve riconfigurarsi

dinamicamente adattandosi ai cambiamenti degli “stati del sistema”. Questi attributi riguardano:

- Percorsi espliciti specificati amministrativamente. Supponendo che l’amministratore di rete specifichi un percorso predefinito, sia esso un percorso realizzabile o no, si ha un attributo di tipo binario che rende quel percorso “vincolante” o “non vincolante”
- Gerarchie e regole di preferenza per i percorsi multipli. In questo caso si permette all’amministratore di selezionare un set di percorsi alternativi aventi ciascuno una preferenza, per esempio in previsione di failure.
- Distribuzione del carico attraverso TT (Traffic Trunk) paralleli.
- Attributi di affinità delle classi di risorse. Tali attributi associati ad un TT possono essere usati per includere o escludere una classe di risorse dal percorso di quello specifico TT.
- Attributi di adattabilità (riottimizzazione). Tale parametro specifica se quel traffic trunk è soggetto a riottimizzazione. Disabilitato quando nonostante i cambiamenti della rete quel TT non può subire variazioni di percorso.

L’aspetto più delicato in relazione alla riottimizzazione è la **stabilità**. Infatti il sistema non dovrebbe essere troppo reattivo nei confronti di un’elevata dinamicità della rete ma allo stesso tempo si dovrebbe adattare abbastanza velocemente per consentire un uso ottimo delle risorse. Ciò implica che la **frequenza di riottimizzazione** non sia troppo elevata ma neanche troppo lenta.

Attributi di priorità

Questo attributo definisce l’importanza relativa al traffic trunk. Nel CBR con MPLS diventa molto importante perché potrebbe essere usato per determinare l’ordine con cui vengono associati i percorsi ai vari TT e implementare politiche di Preemption.

Attributi di preemption

Tale attributo determina se un TT può “rimpiazzare” un altro TT di un percorso specificato e quando un TT può rimpiazzare uno specifico TT. La PREEMPTION è utilizzabile per raggiungere gli obiettivi di miglioramento delle performance sia orientati al traffico che alle risorse; infatti, in un ambiente di servizi differenziati, la preemption può essere usata per assicurare un TT con alta priorità possa sempre essere instradato nel percorso relativamente più favorevole.

La Preemption può anche essere impiegata per implementare varie politiche di ripristino prioritizzate a seguito di eventi di guasto della rete.

Attributi di resilienza

Nell’ottica più generale dell’ottimizzazione si rende necessario considerare gli aspetti relativi alle tecniche di protezione della rete. Infatti, nonostante la scelta di meccanismi dinamici in grado di adattarsi alle variazioni del traffico e di tecniche preventive contro i fenomeni di congestione, potrebbero verificarsi eventi difficilmente prevedibili come per esempio guasti tecnici di apparecchiature o nei link. Questi attributi vincolano strettamente le

interazioni tra MPLS e routing, e quindi è necessario tenerne conto nella procedura di ottimizzazione.

La **resilienza** è la capacità di un sistema di adattarsi alle condizioni d'uso e di resistere all'usura in modo da garantire la disponibilità dei servizi erogati. Questi attributi quindi determinano il comportamento di un TT sotto condizioni di guasto. L'MPLS è dotata di meccanismi mirati a: rilevare i guasti, notificare i guasti, recuperare e ripristinare i servizi. Esistono numerose politiche di recupero per i TT i cui percorsi hanno subito delle situazioni di guasto:

- il TT non viene reinstradato. Questa politica si può usare quando tra due nodi esistono degli LSP paralleli multipli e in condizioni di guasto su un LSP il TT viene mappato su uno degli LSP restanti in accordo con quanto definito dalla politica.
- il TT viene reinstradato in un possibile percorso con abbastanza risorse disponibili. Se non ne esistesse nessuna, allora non verrebbe reinstradato.
- il TT viene reinstradato in un percorso disponibile senza considerare la sua disponibilità di risorse.

esistono altri possibili schemi compresi quelli che nascono dalla combinazione dei tre appena descritti.

L'attributo di resilienza "base" indica la procedura di recupero che può essere applicata al TT il cui percorso ha subito un guasto. Nello specifico è un variabile binaria che indica se il TT può essere reinstradato quando si verifica una situazione di guasto. L'attributo in forma estesa può specificare dettagliate azioni in caso di scenari di guasto (es. un set di percorsi alternativi che possano sostituire quello momentaneamente indisponibile).

A questo punto si rende necessario precisare che la *riottimizzazione* (definita precedentemente) è distinta dalla "*resilienza*". Infatti, l'adattabilità alle variazioni di stato della rete attraverso la riottimizzazione implica la resilienza in caso di guasti, mentre non è vero il contrario.

Attributi di policing

Essi determinano le azioni che dovrebbero essere compiute dal protocollo quando un TT diventa insoddisfacente. Cioè quando il TT eccede rispetto a quanto specificato nei parametri di traffico del contratto sottoscritto. In generale, tali attributi possono indicare se un TT non conforme debba essere limitato nel rate, etichettato, o semplicemente inoltrato senza applicare nessuna particolare politica.

Applicare una politica di traffico è indispensabile in molti scenari, ma potrebbe essere indesiderabile in altri. In generale è auspicabile che si applichi la politica sul traffico nei nodi di ingresso alla rete (per assicurare che i parametri di traffico siano conformi a quanto stipulato negli SLA) e invece minimizzare l'uso di tali politiche nei nodi interni (core-router).

3.2.2 Attributi relativi alle risorse

Tali attributi sono parte dei parametri di stato della topologia che sono usati per vincolare il routing dei TT attraverso risorse specifiche.

Il MAM

Il Maximum Allocation Multiplier di una risorsa è un attributo configurabile dall'amministratore di rete che determina la porzione di risorse allocabile per i TT. Questo attributo viene usato soprattutto in relazione alla banda dei link, ma può essere utilizzato relativamente alle risorse del buffer sull'LSR. Tale valore può essere scelto in modo che le risorse siano sotto-allocate o sovrallocate. Nel primo caso la domanda complessiva di tutti i TT prevede un'allocazione inferiore alla capacità di tale risorsa, mentre, nel secondo caso la richiesta di allocazione di tutti i TT risulta essere eccedente rispetto alla capacità della risorsa. Una sotto-allocazione può essere utile per limitare l'uso delle risorse, anche se il problema è più complesso nel caso dell'MPLS, invece una sovrallocazione potrebbe dare benefici in termini di efficiente impegno di risorse quando il picco di domanda dei vari TT non coincidono temporalmente.

Attributi delle classi di risorse

Tali attributi sono parametri assegnati amministrativamente che esprimono la nozione di classe per le risorse. Possono essere visti come dei colori da assegnare alle risorse tali per cui il set di risorse con lo medesimo colore costituisce una classe. Gli attributi delle classi di risorse rendono possibile l'implementazione delle politiche di ottimizzazione in termini sia di traffico che di risorse. In generale le risorse chiave di maggior interesse sono i link.

3.2.3 Allocazione delle risorse

Nella definizione dell'architettura di riferimento è emersa l'importanza di avere uno schema di routing basato su dei vincoli di banda per la selezione del percorso migliore nel rispetto delle disponibilità delle risorse di rete. Il **constraint band routing (CBR)** usando come input gli attributi associati ai TT, attributi associati alle risorse, altre informazioni di stato sulla topologia e sulla base delle politiche impostate, consente di determinare per ogni TT un LSP che soddisfi le richieste. Ovviamente utilizzando questa tecnica automatizzata il livello di intervento e di configurazione manuale è ridotto al minimo. Per cui giocando sugli attributi dei TT e delle risorse si può guidare la rete verso uno stato di configurazione ottimale.

Per quanto riguarda il DS-TE, la tecnica si basa sulla definizione di classi di traffico (class type) su cui sono mappati i servizi offerti nella architettura di rete. A ciascuna classe viene associato un trattamento o politica di traffico che garantisce il rispetto dei requisiti specificati nei Service Level Agreement. Inoltre si utilizzano degli attributi di preemption che consentono di definire delle TEclass date dalla coppia CT e priorità di preemption. Se consideriamo 8 CT e 8 priorità di preemption si ottengono un numero massimo di 64 TEclass su cui posso mappare opportunamente i flussi di traffico.

Per definizione ad ogni ClassType CT_c ($0 \leq c \leq MaxCT - 1$) si può associare un limite di banda BC_b ($0 \leq b \leq MaxBC - 1$) o un set di limiti di banda. Le regole che riguardano i limiti di banda sono: il massimo numero di vincoli di banda, a quali CT applicare questi vincoli e come. Per ogni CT possiamo definire la banda riservata (reserved), come la somma delle bande riservate da tutti gli LSP che appartengono a CT_c .

I modelli che consentono l'assegnazione della banda possono essere diversi e perseguono in maniera più o meno efficace i seguenti obiettivi:

- Efficienza nell'utilizzo della banda sia in condizioni normali che in sovraccarico
- Bandwidth isolation, che si verifica quando una CT non può sottrarre banda ad un'altra CT
- Protezione contro degradazioni della QoS, almeno per le CT ad alta priorità
- Semplicità di utilizzazione

Le tecniche normalmente utilizzate sono tre:

1. Maximum Allocation Model - MAM; (rfc 4125)
2. Maximum Allocation with Reservation - MAR; (rfc 4126)
3. Russian Doll Model - RDM; (rfc 4127)

Il metodo MAM è estremamente semplice. Comporta la definizione per ogni CT del proprio vincolo di banda massima. Nel caso in cui la somma delle bande dei CT sia inferiore o uguale al massimo della banda disponibile nel link ogni CT ha la sua porzione di banda assegnata (isolation), ma con tale metodologia non è consentita la condivisione di banda tra le CT (minor efficienza).

Il metodo MAR è simile al MAM, introduce, però, una certa dinamicità, permettendo di variare l'allocazione di banda per ogni CT in base all'effettivo consumo operato dalle singole CT. L'allocazione di banda su un link si basa su tre elementi quali la stima della banda necessaria ad ogni CT, la banda effettivamente utilizzata da ogni CT e, infine, lo stato di occupazione della banda del link. È compito degli LSR monitorare il valore dei tre elementi e l'intervallo di tempo in cui si effettuano le variazioni può essere qualsiasi (secondi, minuti, ore, giorni, settimane,...).

Il metodo RDM (Figura 8) prevede che ogni CT abbia a disposizione una banda uguale alla somma delle bande allocabili per le classi inferiori più una propria componente. Quindi se le CT delle classi inferiori utilizzano poca banda, la restante parte sarà a disposizione della CT considerata. Tale tecnica presume che il massimo numero dei BCs (limiti di banda) sia uguale al massimo numero di CTs=8; tutti gli LSPs da CT_c devono usare non più di BC_b (con $b \leq c \leq 7$ e $BC_b \leq BC_{b-1}$, per $b=1 \dots 7$). Consentendo la condivisione della banda, l'RDM, diversamente dai due modelli precedenti, assicura un'elevata efficienza e allo stesso tempo consente un elevato grado di isolamento tra le classi oltre che protezione contro la degradazione della QoS in tutte le classi. L'isolamento tra le classi, però, è garantito solo se si utilizza la preemption.

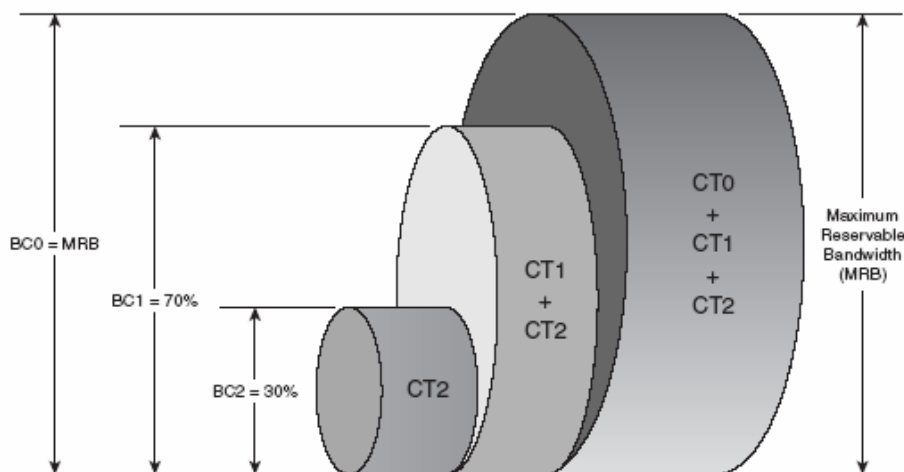


Figura 8 Allocamento della banda secondo l'RDM.

Secondo un confronto con gli altri due modelli [26] basato sulle caratteristiche appena descritte e sugli obiettivi del DS-TE, l'RDM risulta essere la scelta migliore per la nostra architettura di riferimento, quella cioè che si avvicina maggiormente agli obiettivi di cui sopra.

3.2.4 Parametri da ottimizzare

I parametri più importanti che devono essere ottimizzati relativi all'affidabilità e all'economicità nell'uso delle risorse sono:

- occupazione della banda nei link (parametro puntuale)
- carico computazionale nei nodi (parametro puntuale)
- occupazione dei buffer e/o delle code nei nodi (parametro puntuale)
- stabilità della rete (parametro globale)
- complessità della rete (parametro globale)

Analizzando le caratteristiche dei servizi offerti da una architettura di rete multiservizio, si evince che ciascun servizio necessita di opportuni requisiti che garantiscano la qualità. Tali requisiti consentono di identificare i principali parametri relativi al traffico da ottimizzare che possiamo così riassumere:

- ritardo medio dei pacchetti (Delay);
- variazione del ritardo dei pacchetti (Jitter);
- probabilità di perdita di pacchetti (Packet Loss);
- IPER ;
- Throughput;
- occupazione della banda.

3.2.5 Parametri che realizzano l'ottimizzazione

Come è stato precisato precedentemente, il problema che ci apprestiamo a risolvere relativamente alla rete di backbone consiste nell'individuazione della mappatura ottima dei Traffic Trunk sulla topologia fisica della rete attraverso la definizione degli LSP, ovvero assegnare le risorse di rete e definire i percorsi su cui viaggeranno i TT in modo che:

- la rete lavori bene, cioè garantisca adeguati livelli di performance, sia in condizioni normali che in condizioni di sovraccarico qualora si presentassero;
- si minimizzi il carico dovuto ai processi di segnalazione, esempio set up nuovi LSP, preemption degli LSP ecc.;
- si massimizzi l'efficienza nell'uso delle risorse;
- si minimizzi la complessità.

L'individuazione dei parametri che realizzano l'ottimizzazione è una fase particolarmente delicata. Inoltre, non si ha una corrispondenza biunivoca tra parametro da ottimizzare e parametro che realizza l'ottimizzazione. Potrebbe risultare che, al fine di ottimizzare differenti parametri relativi alle risorse o al traffico si debba agire sulla stessa variabile. Per esempio, diminuendo la banda assegnata al servizio, si potrebbe avere una perdita di pacchetti oppure se si applica un sistema di controllo del rate un rallentamento dell'arrivo dei pacchetti (maggiore delay). Quindi, agendo sulla banda (variabile che realizza l'ottimizzazione) si potranno rispettare i vincoli dei servizi in termini di ritardo o perdita di pacchetti (parametri di traffico da ottimizzare).

Nel caso della *stabilità della rete*, parametro globale relativo alle risorse, sulla base degli elementi raccolti nella parte di analisi degli attributi è evidente che la variabile su cui agire per ottenere l'ottimizzazione risulta essere la *frequenza di riottimizzazione*. L'individuazione del parametro su cui agire non sarà altrettanto semplice in altri casi.

**Parte I- Tecniche di
Ottimizzazione: Routing e
rerouting LSP**

Capitolo 4

Problematiche di Routing e stato dell'arte

4.1 Modellamento del problema di instradamento

Una rete di telecomunicazioni costituisce un complesso di mezzi che consente, tramite il ricorso ad un insieme di risorse tecniche e organizzative, il trasferimento dell'informazione generata da un'utenza nell'ambito della fruizione di una serie di servizi erogati. La sua somiglianza con altri tipi di strutture preposte al trasferimento di beni materiali è forte: si pensi ad una rete stradale o ferroviaria o ad una per la diffusione dell'energia elettrica. In tutti questi casi, l'idea che guida la costituzione di questi sistemi è quella di permettere il trasferimento di un bene di interesse in un modo che risulti il più efficace ed efficiente possibile. [29]

La trattazione teorica di una rete implica l'introduzione di un modello che ne consenta la rappresentazione. Il ricorso ad un grafo appare la scelta più naturale e logica. Il passo successivo consiste nello stabilire quale sia il significato che viene attribuito ai nodi e rami che lo compongono. Prima di poter stabilire una topologia è però necessario precisare il tipo di funzioni - di natura logica o di natura fisica - che si stanno prendendo in considerazione, poiché entrambe concorrono al trasferimento dell'informazione però con finalità e rappresentazioni distinte.

Nel caso delle funzioni di natura logica, viene trattata informazione intesa come bene immateriale composto da un insieme di stati logici che assumono significato in un processo di comunicazione. Un ramo rappresenta allora l'ideale percorso diretto che l'informazione segue nel passare dal suo punto di origine a quello di destinazione, mentre un nodo costituisce la rappresentazione di un punto di origine/destinazione. L'insieme dei rami e nodi con questo significato definisce il grafo logico della rete, che descrive il modello identificato dalle funzioni di natura logica, la rete logica. In particolare, i rami sono in corrispondenza con gli apparati di rete che svolgono funzioni di moltiplicazione, mentre i nodi sono in corrispondenza con gli apparati che operano la commutazione.

Nel caso delle funzioni di natura fisica viene sempre trattata informazione, sotto forma però di segnali fisici di natura elettromagnetica, delle entità materiali. Un ramo assume allora il significato di via per il trasferimento dei segnali che trasportano l'informazione, mentre un nodo costituisce il punto di ricezione/trasmissione dei segnali. Si parla di conseguenza di grafo fisico della rete, che individua la topologia del modello di rete definito dalle funzioni di rete con finalità trasmissive (la rete fisica). I rami, in particolare, sono rappresentativi dei sistemi trasmissivi di linea, mentre i nodi sono in corrispondenza con gli apparecchi terminali di rice-trasmissione.

I due modelli di rete sopra definiti sono in stretta relazione gerarchica, poiché le funzioni di natura logica devono ricorrere a quelle di natura fisica e queste ultime esistono allo scopo di supportare le prime. La relazione tra i due tipi di rete è del tipo client-server, dove il ruolo di client è svolto dalla rete logica, mentre quello di server dalla rete fisica. È importante sottolineare che le due reti in generale non coincidono, poiché la definizione dei cammini logici e fisici seguiti dall'informazione attraverso una rete si basa su diversi criteri tecnici ed economici.

In una rete di telecomunicazioni, ogni volta che degli utenti usufruiscono di uno dei servizi supportati ha luogo uno scambio di informazioni tra punti distinti della rete. Per passare da un punto all'altro l'informazione deve seguire un determinato percorso, costituito da una successione di nodi e archi del grafo. In generale i possibili percorsi su cui far transitare l'informazione sono numerosi e la questione della selezione di quale utilizzare viene generalmente indicata come "problema di instradamento". [30],[31] e[32]

Fine ultimo dello studio presentato in questa sezione è l'ottimizzazione del processo di Constraint Based Routing (CBR) che si presenta nell'ambito di un'architettura di rete DiffServ over MPLS. L'ottimizzazione viene raggiunta attraverso le seguenti fasi:

- modellazione in termini matematici del processo di CBR;
- riformulazione del processo come Problema di Ottimizzazione mediante: individuazione delle soluzioni di base del problema;
- introduzione di una funzione obiettivo relativa alla definizione di un insieme di tunnel che consentano di soddisfare delle richieste di comunicazione generate da un'utenza e che rispettino dei negoziati vincoli di qualità del servizio.

Ipotesi di base fondamentale è che le problematiche associate al routing interdominio siano già state affrontate e risolte da un'entità superiore (si può pensare ad un organo regolatore) in grado di supervisionare globalmente l'intera rete. Resta quindi da affrontare il routing intradominio, ovvero si deve individuare un insieme di Label Switching Path che consentano di collegare un Ingress LSR ad un Egress LSR di una rete e che soddisfino i requisiti di qualità imposti.

4.1.1 Dati in ingresso al problema e assunzioni

Il primo importante dato che deve essere reso disponibile per affrontare il problema di routing è rappresentato dalla topologia della rete di interesse. La topologia può essere

individuata mediante un grafo orientato, i cui nodi sono in corrispondenza con i router presenti nella rete e i cui archi rappresentano collegamenti logici della rete.

L'insieme dei nodi può essere partizionato in modo da distinguere tra router di accesso (*edge router* -LER) e router di transito (*switching router* -LSR). I LER costituiscono il punto di collegamento tra diverse reti e rappresentano anche gli estremi di terminazione dei tunnel la cui costruzione costituisce l'obiettivo di questa trattazione. I LSR sono collocati all'interno della rete e svolgono essenzialmente le funzioni di gestione delle label in transito e di smistamento del traffico che si presenta nelle loro porte di ingresso.

Per quanto riguarda gli archi, si è detto che sono in corrispondenza di connessioni logiche della rete, il che implica che possono non corrispondere a degli elementi fisici di trasmissione dei segnali, ma costituire solo delle connessioni dirette virtuali tra router effettivamente collegati mediante cammini composti di più archi fisici. Ad ogni arco vengono associati due parametri: *bandwidth* e *delay*, misurati rispettivamente in kilobits per second (Kbps) e microsecondi (ms). Successivamente, data l'evoluzione del problema di routing da semplice ricerca di un insieme di paths che soddisfino le richieste, a ricerca di path di minimo costo totale soddisfacenti vincoli di qualità, viene anche introdotto un terzo parametro rappresentante il costo necessario per instradare un'unità informativa sull'arco. Altri parametri che caratterizzano una trasmissione dati su rete, come il jitter e il tasso di perdita di pacchetti, non vengono presi in considerazione dato che interesse dello studio è rimanere ad un certo livello di astrazione che prescindano da questioni spiccatamente riguardanti il livello più fisico delle trasmissioni.

Si assume nel seguito che quando si definiscono parametri di un arco non orientato (i,j), essi siano effettivamente associati a due archi orientati che consentono il collegamento da i a j e da j a i . Le caratteristiche dei router (come ad esempio la capacità di processamento del traffico) sono lasciate da parte, ma potrebbero essere inserite in più complesse e realistiche modellazioni future.

4.1.2 Caratteristiche dei tunnel

Come si è detto in precedenza, il fine ultimo della modellazione condotta è la definizione di un insieme di tunnel che permettano la trasmissione dell'informazione generata dall'utenza rispettando le condizioni specificate con gli SLA ed eventuali vincoli imposti sugli elementi di rete (esempio limitazione della banda utilizzabile su un particolare link).

La costruzione dei tunnel avviene quindi sulla base di richieste espresse dall'utenza, individuabili ognuna mediante una quintupla, indicata con il nome di commodity e presentabile nella forma:

(indirizzo ingress LER, indirizzo egress LER, bandwidth richiesta, massimo delay, classe di servizio)

Si specificano quindi gli indirizzi dei LER inizio e terminazione del tunnel, la banda da mettere a disposizione, il massimo ritardo accumulabile nella trasmissione e la classe di servizio richiesta.

Se si fa cadere il riferimento ai LER e si ragiona in termini di reti di ingresso e di uscita dell'informazione, messe in collegamento per mezzo del tunnel attraversante la rete di interesse. La commodity assume allora la forma:

(network in, network out, bandwidth richiesta, massimo delay, classe di servizio)

dove network in/out denota l'insieme di edge router che consentono il collegamento in ingresso/uscita della rete di interesse con quelle confinanti (sebbene la rappresentazione grafica li ponga spesso vicini, gli edge router comunicanti possono essere collocati fisicamente anche molto lontano).

Si noti che la classe di servizio influenza i due parametri di banda richiesta e massimo delay, in quanto ognuna delle sei classi supportate nell'ambito di una DiffServ in simbiosi con una MPLS (best-effort, quattro classi Assured Forwarding e Expedited-Forwarding) è associata a particolari parametri di banda e ritardo.

La costruzione dei tunnel avviene in uno stato off-line della rete: la fase oggetto di studio è quella precedente alla trasmissione di qualunque informazione di utente e in cui devono essere definiti i tragitti seguiti dal traffico affinché le comunicazioni possano avere luogo. Ricorrendo a concetti propri della Teoria dei Sistemi, la fase di interesse può essere paragonata ad una scatola nera che prende in ingresso le commodity definibili a partire dalle richieste dell'utenza e restituisce in uscita l'insieme di cammini che permettono il trasferimento dell'informazione nel rispetto della qualità voluta. La funzione interna alla scatola nera è rappresentata da un algoritmo di instradamento per la definizione degli LSP secondo opportuni criteri di ottimizzazione dei parametri di qualità. Nel caso di esistenza di una soluzione per il soddisfacimento delle richieste, si avrà quindi associata ad ogni commodity una lista di LSP la cui banda totale è pari alla banda richiesta e il cui delay complessivo rispetta il vincolo corrispondente (il parametro di costo è al momento tralasciato).

Una volta che siano stati impostati i tunnel, le comunicazioni possono avere inizio e si passa ad una situazione dinamica in cui oltre al traffico di base, si ha anche la generazione di nuovo traffico data la presentazione di nuove richieste. La banda disponibile di ogni arco sarà soggetta quindi ad una evoluzione continua, con tunnel creati, distrutti e modificati continuamente, ed è lecito aspettarsi che nascano anche problemi di congestione. Lo stato online della rete costituisce un futuro e molto più complesso scenario di studio che viene al momento tralasciato. In questo paragrafo ci si limita a citare alcune procedure notevoli che potrebbero essere adottate per risolvere problemi di congestione in una situazione online:

- se un LSP congestionato è definito su archi con una porzione di banda ancora non usata, si può procedere ad un incremento della banda dell'LSP, raggiungibile tramite un allargamento del tunnel di una quantità fissata su tutti gli archi che lo compongono;
- si definiscono nuovi LSP in parallelo a quello congestionato, che condividono gli estremi ma attraversano nodi diversi;

- si riduce la larghezza di alcuni tunnel (per esempio quelli associati ad una classe di servizio più infima) e le risorse così disimpegnate possono essere utilizzate per supportare le comunicazioni congestionate.

4.2 Stato dell'arte

La definizione dei percorsi associati agli LSP, e in generale il problema del routing, è un elemento molto importante soprattutto in relazione all'utilizzazione delle risorse di rete e la distribuzione del carico sulla rete. I protocolli di routing usati tradizionalmente nell'architettura IP, come il Minimum Hop Algorithm o l'Open Shortest Path First (OSPF) [33], sono connectionless e non permettono il bilanciamento del carico.

MPLS supera i limiti dell'architettura tradizionale supportando un nuovo tipo di routing, detto Constraint-based Routing (CBR), in cui l'instradamento del traffico è fatto lungo percorsi a costo minimo, rispettando contemporaneamente determinati vincoli (constraint) sullo sfruttamento delle risorse di rete. Un esempio di vincolo potrebbe essere quello di trovare un percorso con una certa minima capacità disponibile in ogni link, oppure evitare o includere un determinato link nel percorso. Soltanto un nodo di ingresso può determinare un percorso secondo il CBR, poiché i nodi interni non sono a conoscenza dei vincoli da rispettare. Siamo quindi in presenza di un meccanismo di explicit routing (o source routing).

Il protocollo più semplice ispirato al routing basato su vincoli è il Constraint Shortest Path First (CSPF) [34] che risolve i problemi relativi al bilanciamento del carico includendo un costo che tiene conto della corrente utilizzazione delle capacità dei link, ma come l'OSPF non possiede sufficienti elementi per supportare il TE in condizioni di traffico pesante [35]. Questa tecnica deve spesso essere supportata da metodologie di admission control per migliorare le prestazioni.

Uno dei primi algoritmi di explicit routing più evoluti in ambito MPLS è il Minimum Interference Routing Algorithm (MIRA) [36] che usando la conoscenza della coppia ingress-egress cerca di prevenire i bottleneck e ridurre il numero di richieste degli LSP rifiutate, selezionando percorsi che massimizzano la minima capacità disponibile tra la coppia di ingress-egress. Questa tecnica risulta, però, molto onerosa computazionalmente, gli autori della [37] ne hanno presentato una variante più performante chiamata Least Interference Optimization Algorithm (LIOA) che ottimizza il routing cercando di minimizzare l'interferenza misurata, in questo caso, come numero di flussi che attraversa quel link. L'algoritmo associa a ogni link una funzione costo che tiene conto del numero di flussi trasportati sul link e della differenza tra il massimo della banda riservabile e il totale della banda riservata dagli LSP attivi su quel link.

Un altro lavoro basato sul concetto di minima interferenza e computazionalmente meno pesante del MIRA è stato presentato nel [38]. L'algoritmo proposto consta di due fasi, nella prima, il preprocessing, si sfruttano le informazioni del traffic profile per risolvere un multicommodity flow problem (MCFP) e preallocare la banda sui link relativamente a ciascuna classe di traffico, mentre, nella seconda, fase di routing online, ogni richiesta di LSP è

instradata usando un shortest path algorithm. Un altro algoritmo che fa uso del MCFP è proposto nel [39], dove si cerca di soddisfare una nuova richiesta per la quale non si hanno risorse libere sufficienti, reinstradando un LSP instaurato precedentemente. È un algoritmo abbastanza semplice che non produce una riottimizzazione globale della configurazione degli LSP sulla rete, ma si limita a ridefinire il path per due LSP, quello della nuova richiesta e uno tra quelli precedentemente instradati, senza fare nessuna distinzione tra gli LSP, che potrebbero avere priorità differenti.

Secondo il [28] si può individuare una funzione di costo che tiene conto dell'evoluzione di un sistema che si trova in uno stato s (caratterizzato da un certo numero di LSP definiti e una certa capacità disponibile), che incorre in un certo evento e (es. richiesta di banda) il quale comporta una decisione da prendere a (setup o ridimensione dell'LSP che provoca una variazione della capacità del sistema, o nessuna azione) tale funzione è data dalla somma di tre contributi:

- $W_{\text{sign}}(S,a)$ costo di segnalazione di setup o di ridimensionamento degli LSP, per aggiornare i router;
- $W_b(S,a)$ costo dovuto alla banda;
- $W_{\text{sw}}(S,a)$ costo dovuto allo switching del traffico;

La **funzione** è: $W(S,a) = W_{\text{sign}}(S,a) + W_b(S,a) + W_{\text{sw}}(S,a)$

in tale funzione $W_{\text{sign}}(S,a)$ dipende linearmente dal numero di salti del percorso fisico su cui l'LSP è reinstradato, inoltre, avrà una componente costante aggiuntiva dovuta alla notifica della variazione della capacità disponibile nella linea. Questo approccio avvalorava ulteriormente la scelta dei parametri che stiamo considerando nel nostro metodo di ottimizzazione.

Come è stato evidenziato precedentemente nell'architettura DS-TE si può far uso della preemption. I meccanismi di preemption costituiscono una delle maggiori peculiarità dell'esplicito routing supportato dal MPLS-Traffic Engineering, essi consentono ai nodi di porre termine alle trasmissioni che avvengono su particolari LSP, forzandone l'abbattimento. La conseguenza è che viene liberata banda preziosa utilizzabile per instaurare un nuovo LSP associato ad una richiesta che altrimenti non sarebbe potuta essere soddisfatta per insufficiente disponibilità di banda.

L'impiego della preemption genera delle interruzioni di servizio se non si provvede in qualche modo al rerouting degli LSP preempted che, a sua volta, produce un aumento del traffico di segnalazione, inoltre, si potrebbero verificare degli sprechi di banda se la scelta degli LSP da abbattere non è oculata. Per ottimizzare la scelta di quali LSP abbattere e ridurre il rerouting, un interessantissimo lavoro è stato presentato in [27]. Gli autori propongono un algoritmo che ottimizza, su un determinato link, la scelta degli LSP da abbattere sulla base di tre criteri: minimizzazione della priorità degli LSP abbattuti, minimizzazione del numero di questi LSP e minimizzare la banda preempted. Nella seconda parte viene proposto un'altra tecnica con la quale cerca di liberare risorse riducendo il rate degli LSP aventi priorità più bassa. La riduzione del rate non comporta dei costi di segnalazioni aggiuntivi in quanto

potrebbe essere usato lo stesso protocollo di segnalazione dei refresh degli LSP, in ogni caso l'onere di tale segnalazione sarebbe notevolmente inferiore rispetto a quello necessario all'abbattimento, il reindirizzamento e il set up per un nuovo LSP. La determinazione del percorso associato agli LSP non è affrontata ma demandata ai protocolli tradizionali, così come non si affronta il problema del reindirizzamento degli LSP preempted.

Questa ultima tecnica fa parte delle tecniche basate su un approccio decentralizzato in cui ogni nodo del percorso è responsabile di applicare l'algoritmo di preemption sugli LSP.

L'approccio opposto è quello centralizzato in cui una entità manager che possiede tutte le informazioni relative agli LSP attivi e lo stato di tutti i link della rete decide quale o quali LSP abbattere per soddisfare la nuova richiesta.

Entrambi gli approcci possiedono i loro vantaggi e svantaggi. L'approccio centralizzato offre un'ottimizzazione globale a scapito, però, di un onere computazionale molto maggiore dovuto soprattutto all'aggiornamento delle informazioni sugli LSP e sui link, infatti, risulta più adatto alla gestione di una situazione statica della rete dove non vi sono repentini cambiamenti. In una rete dinamica, in cui gli LSP vengono stabiliti e demoliti molto frequentemente, l'approccio decentralizzato, nonostante non fornisca soluzioni ottime, offre maggiori benefici e risulta più semplice da implementare.

Il presente lavoro parte dall'analisi dello stato dell'arte e dalle peculiarità dell'architettura di rete per arrivare alla definizione di un framework completo in grado di ottimizzare non solo il processo di routing di un nuovo LSP su una rete carica ma anche il processo di preemption e di rerouting degli LSP abbattuti.

Capitolo 5

Tecniche per l'Instradamento Ottimo

5.1 Introduzione

Il problema di definizione dei tunnel, e quindi del corrispondente insieme di LSP che lo compongono sulla base della topologia della rete e delle richieste presentate dall'utenza, si presta naturalmente ad essere modellato secondo l'approccio di un *Multicommodity Flow Problem (MCFP)*, tipico problema che si presenta in una rete su cui si devono inviare dei flussi e risolvibile tramite le tecniche messe a disposizione dalla Ricerca Operativa.

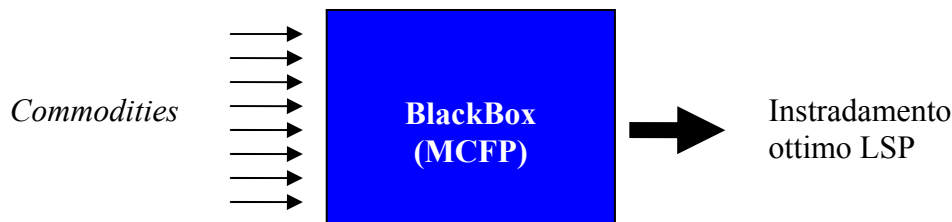


Figura 9 Schema del MCFP

Ovviamente, la ricerca della configurazione ottimale dei cammini per il set up degli LSP deve essere effettuata in due fasi distinte del ciclo di vita di una rete. La prima fase riguarda lo stato di *off-line* in cui su una rete scarica si ricerca la configurazione ottima iniziale. Successivamente, all'avvio della rete si passerà in uno stato *on-line*, dinamico e altamente variabile in cui perverranno le richieste per il set up di nuovi LSP che potrebbero non essere soddisfatte. Si tratterà di determinare la configurazione sub-ottima che mira a fornire risorse ai traffici con priorità superiore e bilanciare il carico sulla rete.

Per la prima fase si propone l'utilizzo di un MCFP risolto con la metodica del *column generation* che offre notevoli vantaggi computazionali. La seconda fase è più complicata e si

propongono due tecniche che sfruttano la *preemption*, la prima basata su un approccio di ottimizzazione locale e incrementale, mentre la seconda basata sulla riottimizzazione globale dei percorsi. Inoltre, è proposta anche una soluzione ibrida che attua le due tecniche su una stessa rete.

5.2 Il Multicommodity Flow Problem

Il classico problema di inviare un flusso massimo di risorse tra due nodi di una rete (Maximum Flow Problem MFP) può essere al tempo stesso generalizzato e reso più complesso con il passaggio ad un MCFP: la necessità di individuare un cammino che massimizzi il flusso inviato tra un nodo source r e un nodo sink s può infatti modificarsi in quella di dover soddisfare delle richieste di invio di più flussi di una certa entità tra una molteplicità di coppie di nodi sorgente-pozzo, creando quindi uno scenario più articolato in cui nascono nuove esigenze. Questo scenario più complesso può essere utilizzato per modellare una molteplicità di situazioni reali che coinvolgono reti di vario tipo: oltre al caso delle telecomunicazioni, si può pensare ad una rete stradale dove devono essere stabiliti i flussi di automezzi pubblici, a reti di distribuzione dell'energia o anche a reti di interconnessione di circuiti VLSI.

Un MCFP può essere schematicamente indicato come:

Input:

1. un grafo orientato $G(V,E)$;
2. coppie $(r_1, s_1), \dots, (r_k, s_k)$ di nodi di V ;
3. capacità di arco non negative $u \in R^{|E|}$;
4. domande non negative $d_1, \dots, d_k \in R$ da soddisfare;

Objective:

per ogni $i = 1, \dots, k$ si deve trovare un (r_i, s_i) -flusso x^i di valore d_i tale che per ogni arco siano soddisfatti i vincoli di capacità ($\forall e \in E$ si deve avere $\sum_{i=1}^k x_e^i \leq u_e$)

Le coppie (r_i, s_i) prendono il nome di commodity e in questa trattazione si ipotizza che si abbia sempre $r_i \neq s_i$. Si può anche introdurre una versione del problema per grafi non orientati, che prevede la sostituzione di ogni arco non orientato $e = (v, w)$ con due archi orientati (v, w) , (w, v) e la successiva individuazione di flussi x^1, \dots, x^k tali che per ogni arco $e = (v, w)$ si abbia

$$\sum_{i=1}^k x_{vw}^i + x_{wv}^i \leq u_e$$

Se viene aggiunto il vincolo che i flussi siano di valore intero, il problema risultante viene detto integer multicommodity flow problem e costituisce un caso particolare del MCFP che si incontra spesso nella realtà. Esso si presenta però di difficile soluzione, anche nel semplice caso di due commodities da soddisfare e capacità degli archi unitarie. Nonostante ciò esiste una fiorente letteratura sull'argomento che si concentra in particolare sulla soluzione di casi trattabili o sulla definizione di algoritmi approssimanti.

La trattazione condotta si pone l'obiettivo di affrontare un MCFP di natura frazionaria sulla base del classico algoritmo di Ford e Fulkerson, cercando quindi di individuare una soluzione al seguente sistema di disequazioni lineari nelle $k \cdot |E|$ variabili x_e^i ($i = 1, \dots, k$ $e \in E$):

$$f_{x^i}(v) = 0 \quad \forall i = 1, \dots, k, v \in V \setminus \{r_i, s_i\} \quad (5.1)$$

$$f_{x^i}(s_i) = d_i \quad \forall i = 1, \dots, k, v \in V \quad (5.2)$$

$$\sum_{i=1}^k x_e^i \leq u_e \quad e \in E \quad (5.3)$$

$$x_e^i \geq 0 \quad \forall i = 1, \dots, k \quad e \in E \quad (5.4)$$

dove $f_{x^i}(\cdot)$ denota il flusso di rete relativo ad un nodo. Il problema è risolvibile in tempo polinomiale e può essere usato un semplice metodo per la programmazione lineare.

Se k assume grande valore, la risoluzione del problema secondo l'approccio visto può non essere conveniente e per questo motivo si procede ad introdurre una formulazione alternativa, che risulta caratterizzata da maggiori dimensioni, ma consente una maggiore efficienza.

Se esiste un (r_i, s_i) -flusso non negativo di valore d_i allora esiste un flusso y_i tale che $x_i \geq y_i$ e y_i è combinazione di flussi non negativi di (r_i, s_i) cammini. Per ogni fissato $i = 1, \dots, k$ possiamo quindi assumere che esistano dei cammini p_{i1}, \dots, p_{in} e $z_{i1}, \dots, z_{in} \in R^+$ tali da soddisfare le relazioni:

$$\sum_{i=1}^{n_i} z_{ij} X_e^{p_{ij}} \leq x_e^i, \quad \forall e \in E \quad (5.5)$$

$$\sum_{i=1}^{n_i} z_{ij} \leq d^i, \quad \forall e \in E \quad (5.6)$$

in cui $X^p \in R^E$ denota il vettore di incidenza del cammino p rispetto all'insieme di archi E ($X^p(e) = 1$ se e appartiene a p , zero altrimenti). Il MCFP di natura frazionaria corrisponde quindi ad individuare un insieme di cammini p_{ij} e dei reali non negativi z_{ij} che soddisfino le relazioni:

$$\sum_{i=1}^{n_i} z_{ij} = d^i, \quad \forall i = 1, \dots, k \quad (5.7)$$

$$\sum_{i=1}^k \sum_{j=1}^{n_i} z_{ij} X_e^{p_{ij}} \leq u_e, \quad \forall e \in E \quad (5.8)$$

Si tratta di nuovo di un sistema lineare, però bisogna fare molta attenzione al fatto che si è verificato un enorme aumento delle variabili coinvolte. Effettivamente, per problemi reali, si potrebbe scrivere a stento l'intero sistema, quindi bisogna ricorrere ad un espediente che consenta di rendere la formulazione trattabile in forma estesa: non vengono considerate tutte le

variabili p_{ij} , ma solo un sottoinsieme di esse secondo un approccio risolutivo diffusamente impiegato nell'ottimizzazione combinatoria che prende il nome di *column generation* [40] (vengono generati i soli cammini corrispondenti a colonne di una matrice di interesse che risultano necessari).

Il primo passo da compiere consiste nel convertire il problema originale di risoluzione del sistema lineare in uno di massimizzazione. A questo scopo devono essere aggiunti per ogni $i = 1, \dots, k$ un nodo r'_i ed un arco (r'_i, r_i) di capacità d_i . Si procede quindi ad eliminare i vincoli e a massimizzare la funzione $\sum_{i=1}^k \sum_{j=1}^{n_i} z_{ij}$ secondo i vincoli rimanenti (sostituendo r_i con r'_i).

Se il valore massimo individuato risulta pari a $\sum_{i=1}^k d_i$, si può affermare di aver individuato una soluzione per il sistema (5.7), (5.8), altrimenti si può concludere che non ammette soluzione.

Secondo questo approccio, la forma del problema di massimizzazione, nel caso che si denoti con P l'insieme di tutti i cammini possibili tra le coppie (r_i, s_i) , risulta:

$$\max \sum_{p \in P} x_p \quad (5.9)$$

$$s.t. \quad \sum_{p \in P} x_p X_e^p \leq u_e \quad \forall e \in E \quad (5.10)$$

$$x_p \geq 0 \quad \forall p \in P \quad (5.11)$$

Si cerca quindi di massimizzare i flussi x_p , di natura non negativa, con il vincolo che non venga superata la capacità di flusso di ogni arco. La forma finale del problema di massimizzazione che si procede a risolvere può essere ottenuta tramite tre passi finali:

1. introduce una variabile di slack x_e per ogni arco $e \in E$;
2. definisce una matrice $[\mathbf{AI}]$ ottenuta affiancando la matrice di incidenza archi-percorsi relativa a tutti i cammini dell'insieme P alla matrice identica \mathbf{I} di ordine $|E|$;
3. si introduce un vettore $w \in R^{|P||E|}$ i cui elementi hanno valore unitario quando corrispondenti ad un cammino $p \in P$ e valore nullo se corrispondenti ad un arco $e \in E$. Il problema in forma vettoriale è quindi in definitiva:

$$\max \quad w^T x \quad (5.12)$$

$$s.t. \quad [\mathbf{AI}]x = u \quad (5.13)$$

$$x \geq 0 \quad (5.14)$$

Il sistema di disequazioni lineari viene quindi ricondotto ad uno di equazioni sempre lineari, che viene a costituire l'insieme di vincoli fondamentale di un problema di

programmazione lineare, non considerato nella sua interezza, ma solo in parte, che viene passo dopo passo ingrandito nelle variabili che lo caratterizzano, al fine di individuare una soluzione ottima con minor onere possibile. Supponendo di avere sottoinsiemi P' di P e E' di E , tali che $|P'| + |E'| = E$, la coppia P', E' costituisce una base se la matrice $[AI]'$, che è formata dalle colonne di $[AI]$ corrispondenti a P', E' a rango pieno. Se si denotano con $x_{[AI]'}$ le variabili associate a P', E' e con $w_{[AI]'}$ la corrispondente parte del vettore w , la soluzione base relativa a $[AI]'$, è definita come l'unica soluzione del sistema $[AI]'$ $x_{[AI]}' = u$, con le rimanenti variabili settate tutte a zero. L'algoritmo del simplesso può essere avviato semplicemente ponendo $\mathbf{B} = \mathbf{I}$, il che implica $P' = 0$ e $E' = E$ si ha allora $x_p = 0$ per ogni $p \in P$ e $x_e = u_e$ per ogni $e \in E$.

La verifica dell'ottimalità della base può eseguirsi calcolando la soluzione y dell'associato problema duale e corrisponde a vedere se sono rispettati i vincoli duali. I vincoli duali ricadono in due tipologie: quella corrispondente a variabili di arco e quella corrispondente a variabili di cammino. I vincoli corrispondenti ad archi sono facili da verificare, in quanto richiedono che per ogni $e \in E$ si abbia y_e non negativo. Se per qualche e si ha quindi $y_e < 0$, lo si inserisce nella base. Si supponga ora di avere variabili di arco tutte non negative. La verifica dei vincoli di cammino è molto più complicata, soprattutto perchè il numero di essi può essere molto grande. Il vincolo duale di cammino è esprimibile per ogni $p \in P$ come:

$$y^T X^P \geq 1 \quad (5.15)$$

Ciò si può tradurre nella condizione che ogni cammino p abbia un costo duale associato di valore almeno unitario. Per ogni coppia (r_i, s_i) si può quindi verificare se i cammini definiti risultano ottimali tramite applicazione dell'algoritmo di Dijkstra, sfruttando la non negatività dei costi y_e l'individuazione di un cammino minimo di costo inferiore a 1 implica che l'insieme di cammini individuati per la coppia non sia sufficiente a risolvere il problema e se ne devono aggiungere quindi altri.

Se l'ottimalità non è verificata si presenta ora la necessità di selezionare un indice j il cui corrispondente elemento lasci la base. Se l'indice dell'elemento che viene inserito nella base è quello di un arco e , si procede a risolvere il sistema $\mathbf{B}z = 1_e$, altrimenti si ha ingresso di un cammino e si risolve allora il sistema $\mathbf{B}x = X^p$. Una volta calcolato z si può individuare l'indice da togliere, secondo una procedura che prevede la selezione dell'elemento j di $P' \cup E'$. per cui la quantità $\frac{x_j}{x_i}$ ha denominatore positivo ed è la più piccola possibile.

Per concludere, si può notare che la terminazione dell'algoritmo visto non è assicurata, dato che l'ingresso e l'uscita di elementi dalla base può assumere le connotazioni di un ciclo, che

può comunque essere evitato adottando opportuni accorgimenti (si può ad esempio adottare un approccio lessicografico nella selezione dell'elemento da far uscire).

5.3 Adattamento dell'algoritmo al caso di studio

La procedura algoritmica della *column generation* per la soluzione di un MCFP deve essere naturalmente adattata alla particolare situazione studiata, soprattutto al fine di guadagnare in termini di efficienza computazionale. Come si è visto, l'operazione fondamentale da compiere ad ogni iterazione consiste nella verifica dell'ottimalità dei cammini generati. Se la condizione non è verificata, si espande il problema con l'aggiunta di nuovi cammini e l'inserimento di nuove colonne e si procede.

Nello scenario oggetto di studio, che, si ricorda, prevede la definizione di tunnels in una rete DiffServ over MPLS, le colonne corrispondono a Label Switching Paths individuati per trasferire l'informazione tra gli edge routers indicati in una lista di commodity.

L'avvio dell'algoritmo richiede la definizione di una soluzione iniziale che corrisponde ad un problema ristretto: le possibilità sono tante e numerosi sono anche i modi per attuarle e, dopo alcune valutazioni, si è scelto di partire definendo un numero, variabile tra 1 e 10, di LSP ammissibili iniziali per ogni commodities, secondo una filosofia del lavorare di più all'inizio con la speranza di poter alleggerire il lavoro successivo [41]. In questo modo già alla prima iterazione si hanno un buon numero di variabili e vincoli con cui trattare, cosa che può incidere sensibilmente in termini di riduzione dei tempi di individuazione della soluzione.

La procedura di scelta dei cammini iniziali di ogni commodity è a carattere euristico e ha permesso di ottenere soluzioni iniziali di buona qualità: per ogni commodity si utilizza l'algoritmo di Dijkstra per individuare il cammino a delay minimo, quindi, per definire un nuovo cammino, si individua l'arco a banda minima (il bottle-neck) del cammino attuale, lo si elimina dal grafo e si ripete poi l'individuazione del cammino a delay minimo tra gli stessi nodi estremi. Quest'operazione può essere ripetuta il numero desiderato di volte o finché si riesce a trovare un cammino tra i nodi di interesse. Naturalmente, se il costo-delay di un cammino è superiore a quello massimo consentito dalla commodity, il cammino non viene accettato. Se per una commodity il cammino a delay minimo individuato nel grafo originale viola il vincolo di delay consentito, il MCFP trattato non ha soluzione ed è previsto che l'algoritmo non prenda nemmeno avvio.

Dopo aver trattato la procedura di individuazione dei cammini iniziali si può finalmente presentare l'algoritmo sviluppato. Base per la sua definizione è stato il lavoro condotto da A. Cerrone in [42]: lo scenario risultava più semplice, data la necessità di individuare un insieme di LSP che permettessero di stabilire comunicazioni nel rispetto dei parametri di banda e delay richiesti dall'utenza.

La novità in questo studio è rappresentata dall'introduzione di un nuovo parametro di arco, coincidente con il costo sostenuto per ogni unità di banda utilizzata. Nel corso del tempo sono state definite le più disparate funzioni di costo di arco, spazianti da forme molto semplici e

costanti a espressioni complesse che coinvolgono una gran quantità di fattori. Una ragionevole soluzione adottabile sarebbe stata frutto della combinazione di due differenti approcci: uno in cui il costo viene fatto coincidere con il delay associato all'arco e uno in cui il costo dipende dall'occupazione di banda di arco, secondo una funzione linearmente crescente che assume crescita esponenziale nel momento in cui si supera una certa percentuale di occupazione (l'idea alla base è quella di scoraggiare l'uso di collegamenti già molto sfruttati così da evitare fenomeni di congestione). Non è però possibile ricorrere a tale soluzione in quanto si avrebbe un'evoluzione dei costi nel tempo caratteristica di uno stato online della rete, che esula dai nostri interessi. La scelta finale è quindi stata quella di una semplice ma diffusa funzione che associa ad un arco un costo pari al suo delay, incrementato di una quantità proporzionale ai suoi costi di instaurazione e di gestione. Si noti che il parametro di costo non presenta in generale una natura univocamente determinata, ma, a seconda delle trattazioni e degli autori, può avere carattere monetario e non monetario. Non esiste un unico orientamento a riguardo.

L'obiettivo in questo caso diventa quello di individuare un insieme di LSP in grado di soddisfare le commodity e che complessivamente abbia costo minimo. Per ogni arco $e \in E$ del grafo $G(V, E)$ considerato si hanno tre parametri:

- delay d_e
- bandwidth b_e
- costo c_e

L'insieme di commodity da soddisfare è denotato con C e ogni commodity $k \in C$ viene individuata mediante una quadrupla (s_k, t_k, B_k, D_k) , dove s_k e t_k rappresentano origine e destinazione della comunicazione, B_k la banda richiesta e D_k il massimo delay consentito. Per ogni commodity $k \in C$ si indica con P_k l'insieme dei corrispondenti cammini ammissibili, ovvero cammini che rispettano il vincolo sul delay massimo ($p \in P_k$ ammissibile se $\sum_{e \in p} d_e \leq D_k$). Ad ogni cammino si associa un valore f_p rappresentante il flusso di informazione che trasporta e un costo $c_p = \sum_{e \in p} c_e$ ottenuto come somma dei costi di arco che lo compongono.

La formulazione matematica che si ottiene è la seguente:

$$\min \sum_{p \in P_k} c_p f_p \quad (5.16)$$

$$s.t. \quad \sum_{p \in P_k} f_p \leq B_k \quad \forall k \in C \quad (5.17)$$

$$\sum_{p \in P_k: e \in p} f_p \leq b_e \quad \forall e \in E \quad (5.18)$$

$$f_p \geq 0 \quad \forall p \in P \quad (5.19)$$

Il vincolo (5.17) esprime il fatto che per ogni commodity si devono definire dei cammini che consentano il soddisfacimento della richiesta di banda, mentre il (5.18) esprime il fatto che il traffico su ogni arco non può eccedere la banda disponibile.

Ad ogni iterazione della procedura di soluzione del problema si ha a disposizione un sottoinsieme $Q \subseteq P$ dell'insieme di tutti i cammini ammissibili, che viene utilizzato per definire un problema ristretto la cui soluzione ottima f^* viene passata ad un oracolo di separazione che ne verifica l'ottimalità per il problema completo. Se f^* è ottima anche per il completo, il problema è risolto, altrimenti l'oracolo fornisce un insieme di cammini $Q' \subseteq P \setminus Q$ che espandono il problema ridotto e dovrebbero consentire un miglioramento del valore della funzione obiettivo e un avvicinamento all'ottimo. Il nuovo problema ristretto viene risolto e si procede come prima. L'oracolo di separazione ha il compito di risolvere il problema ristretto:

$$\min \sum_{p \in Q} c_p f_p \quad (5.20)$$

$$(z_k) \quad \sum_{p \in k} f_p \leq B_k \quad \forall k \in C \quad (5.21)$$

$$(y_e) \quad \sum_{p \in Q: e \in p} f_p \leq b_e \quad \forall e \in E \quad (5.22)$$

$$f_p \geq 0 \quad \forall p \in Q \quad (5.23)$$

L'ottimalità di una soluzione f^* si verifica passando attraverso la soluzione del corrispondente problema duale

$$\max \sum_{k \in C} B_k z_k - \sum_{e \in E} b_e y_e \quad (5.24)$$

$$(f_p) \quad z_k - \sum_{e \in p} y_e \leq c_p \quad \forall p \in Q \quad (5.25)$$

$$y_e \geq 0 \quad \forall e \in E \quad (5.26)$$

Una volta calcolata una soluzione primale f^* e la corrispondente soluzione duale (z^*, y^*) , individuare una colonna da aggiungere al problema primale corrisponde a trovare un vincolo duale violato e quindi un cammino $p' \in P \setminus Q$ tale che si abbia:

$$z_k^* - \sum_{e \in p'} y_e^* \leq c_{p'} \quad (5.27)$$

Poiché il costo di cammino può essere espresso come somma dei costi degli archi che lo compongono ($c_p = \sum_{e \in p} c_e$), la disequazione appena vista può essere riscritta come:

$$\sum_{e \in p'} (c_e + y_e^*) \leq z_k^* \quad (5.28)$$

Le operazioni effettuate dall'oracolo di separazione sono quindi:

1. scelta di una commodity k ;
2. calcolo del cammino minimo da s_k a t_k sul grafo G con pesi rappresentati dalle quantità $(c_e + y_e^*)$;
3. sia p^* il cammino individuato $l(p^*) = \sum_{e \in p} (c_e + y_e^*)$ e il suo peso. Se $l(p^*) < z_k^*$, il vincolo relativo a p^* risulta violato e di conseguenza si aggiunge la colonna corrispondente a p^* . Se invece la violazione non si verifica per alcuna commodity, il problema non deve essere espanso. Ciò che si è effettivamente realizzato nella pratica, ha previsto che ad ogni iterazione si verifichi per ogni commodity se l'associato insieme di cammini rispetti i vincoli duali. In caso affermativo, si è individuata una soluzione e ci si può fermare, altrimenti si procede alla definizione di un nuovo cammino per la commodity per la quale si ha violazione di un vincolo duale.

Evidentemente, la cosa più ragionevole da fare nell'individuazione di un nuovo cammino consiste nel risolvere un problema di cammino minimo con pesi rappresentati dai costi duali $(c_e + y_e^*)$. Ciò però non risulta sufficiente, perchè il nostro obiettivo è di individuare un cammino di peso minimo che al tempo stesso risulti però ammissibile, rispettando il vincolo sul delay massimo.

Si è quindi chiamati a risolvere un problema di cammino minimo vincolato (CSPP - Constrained shortest Path), per il quale esistono numerose procedure risolutive. In [42] si fa ricorso ad una tecnica basata sul Rilassamento Lagrangiano mentre in [41] la scelta è ricaduta sull'algoritmo A*Prune, data la sua relativa semplicità di struttura e le notevoli prestazioni che consente di ottenere.

A*Prune: un algoritmo per il CSPP

A*Prune nasce come algoritmo per l'individuazione, in ordine di peso crescente, dei primi k multiple constrained shortest paths -(kMCSP) tra una coppia di nodi di un grafo orientato in cui gli archi risultano associati ad una molteplicità di parametri. Esso consente quindi l'individuazione di cammini di peso minimo che soddisfino dei vincoli di massimo valore su parametri a loro associati. Il cammino ottimo viene costruito a partire dal nodo iniziale di interesse, cercando una strada ammissibile che conduca al nodo destinazione.

Ad ogni iterazione vengono eliminati tutti i cammini che implicano una violazione dei vincoli e si mantengono solo quelli che, almeno potenzialmente, possono rimanere ammissibili e da cui, alla fine, si estrarrà l'eventualmente esistente cammino ottimo.

L'appellativo Prune deriva dal fatto che la struttura dati usata per l'immagazzinamento dei cammini costruiti è un heap, un albero binario quasi completo, che viene potato (pruned)-sfoltito di cammini che contiene man mano che l'algoritmo procede ed ottiene migliori informazioni circa la natura dei cammini costruiti. Nell'implementazione dell'algoritmo, la

struttura dati effettivamente utilizzata risulta comunque essere una lista data la sua maggiore semplicità di gestione (il passaggio da heap a lista non è un'operazione impropria dato che un heap può comunque essere ricondotto ad una lista).

La scelta di quali cammini immagazzinati nell'heap vanno ampliati e quali invece vanno scartati si compie sulla base di una funzione di costo proiettata, ottenibile aggiungendo al costo sostenuto per raggiungere un certo nodo il costo che si prevede dover sostenere al minimo per raggiungere la destinazione (si può a questo proposito ricorrere all'algoritmo di Dijkstra).

Risultati sperimentali dimostrano che A*Prune si comporta molto bene, individuando soluzioni esatte con un comportamento comparabile ai migliori algoritmi approssimanti per grafi generati in modo random.

Per quanto riguarda la formulazione del problema e le definizioni, sia $G(V, E)$ un grafo in cui ogni arco (i, j) . E risulta associato ad un numero R di parametri a valori non negativi $w_r(i, j)$ con $r = 1, \dots, R$ e sia:

$$w_0(i, j) = \sum_{r=1}^R a_r w_r(i, j) \quad (5.29)$$

una length function con coefficienti $a_r \in \mathfrak{R}$. Dati un nodo origine s e una destinazione t e valori $C_r(s, t)$; $r = 1, \dots, R$ rappresentanti i massimi valori assumibili dai parametri per un cammino da s a t , il problema dell'individuazione di kMCSP ha il seguente vincolo fondamentale:

$$W_r(p(s, t)) \equiv \sum_{(i, j) \in p(s, t)} w_r(i, j) \leq C_r(s, t), \forall r \in \{1, 2, \dots, R\} \quad (5.30)$$

esso esprime il fatto che la somma dei pesi degli archi componenti un cammino non deve eccedere il massimo valore consentito. Nel caso particolare $k = 1$, se si utilizza una variabile booleana $x(i, j)$ per indicare se un arco (i, j) è presente all'interno del cammino minimo, la formulazione completa del problema risulta:

$$\begin{aligned} & \min W_0 \\ & s.t. \quad x(i, j) \in \{0, 1\} \quad \forall (i, j) \in E \\ & \quad \sum_{(i, j) \in E} w_r(i, j) x(i, j) \leq C_r \quad \forall r \in \{1, 2, \dots, R\} \\ & \quad \sum_{(i, j) \in E} x(i, j) - \sum_{(j, i) \in E} x(j, i) = \begin{cases} 1 & \text{se } i = s \\ -1 & \text{se } i = t \\ 0 & \text{altrimenti} \end{cases} \end{aligned} \quad (5.31)$$

5.4 Ottimizzazione del routing online-dinamico

Nel momento in cui si passa da uno *stato offline* ad uno online della rete, lo scenario di studio cambia radicalmente e le difficoltà che si devono affrontare aumentano in modo considerevole. Lo *stato offline* implica una situazione statica che può essere trattata con relativa facilità: le richieste di trasferimento dati generate dall'utenza sono tutte note e definite nei parametri che le caratterizzano (nel nostro caso: router origine, router destinazione, banda richiesta, classe di servizio) e possono essere soddisfatte tramite l'individuazione ottima di un insieme di LSP nella rete. Se si fa riferimento ad un'evoluzione temporale della rete MPLS, lo *stato offline* viene generalmente identificato con l'istante $t = 0$, in cui non sono ancora transitati dati sulla rete e si hanno a disposizione senza limitazioni tutte le risorse di banda. Come visto nella precedente sezione, il problema dell'individuazione dell'insieme di LSP ottimi si presta ad essere modellato come un MCFP e la *column generation* costituisce un'efficiente metodologia di soluzione, che consente di aggirare le difficoltà derivanti dalle grandi dimensioni del problema.

Nel caso dello *stato online*, la rete si trova in una situazione dinamica ed è soggetta ad una continua evoluzione: ciò che caratterizza lo scenario è la presenza di un processo di arrivo di nuove richieste dell'utenza, che devono essere soddisfatte utilizzando le risorse al momento disponibili. Ogni nuova richiesta si presenta infatti in un istante $t > 0$, in cui un insieme di LSP sono già definiti sulla rete al fine di soddisfare richieste presentatesi precedentemente. Parte delle risorse della rete sono quindi già impiegate e non possono essere utilizzate. Le possibilità di definizione di nuovi LSP risultano allora fortemente limitate e, fatto importantissimo, una condizione di ottimalità complessiva degli LSP definiti sulla rete, come quella ottenuta *offline* in $t = 0$, non viene più raggiunta: i nuovi LSP definiti per soddisfare una nuova richiesta risultano infatti ottimizzati rispetto alla corrente situazione della rete e ai LSP correntemente presenti. Al primo istante di tempo $t > 0$ in cui avviene l'instaurazione di un nuovo LSP, l'ottimalità complessiva raggiunta in $t = 0$ è compromessa. Un'ulteriore complicazione dello scenario online deriva dal fatto che una rete MPLS over DiffServ implementa dei meccanismi per cui le commodity di classi di servizio più alte non solo possono accedere alle risorse di rete disponibili al momento della loro presentazione, ma possono anche sfruttare le risorse assegnate a LSP definiti per commodity di classi di servizio più basse: la politica di Traffic Engineering considera il traffico generato nell'ambito di una certa classe di servizio più prezioso di quello di classi inferiori e, di conseguenza, si consente di abbattere LSP che trasmettono dati di classi inferiori, al fine di liberare risorse e permettere la trasmissione di dati di classi superiori. Il meccanismo di abbattimento viene indicato con il nome di *preemption*, un termine molto adatto considerando che in inglese è usato per indicare la confisca di beni di un neutrale da parte di un belligerante.

L'abbattimento di un insieme di LSP e la successiva instaurazione di un nuovo LSP associato ad una più alta classe di servizio non sono le uniche operazioni espletate quando si presenta una nuova richiesta e le risorse disponibili non sono sufficienti: un ulteriore passo necessario è rappresentato dal processo di rerouting, tramite cui il traffico che veniva fatto

transitare su LSP abbattuti viene dirottato su nuovi LSP. Il rerouting costituisce naturalmente un fattore aggiuntivo di complicazione della dinamica online. [43]

5.4.1 Uno schema generale per il routing dinamico

In uno stato online della rete la definizione degli LSP è eseguita in modo incrementale: gli LSP vengono costruiti consecutivamente uno alla volta, per soddisfare le richieste dell'utenza. Durante la definizione del cammino di un LSP non si ha coscienza degli altri LSP presenti nella rete, ma si hanno a disposizione solo informazioni sullo stato dei link (nel caso studiato, l'informazione a disposizione è la banda disponibile in ognuno degli archi al netto di quella impiegata).

In uno stato offline, al contrario, si ha piena coscienza di tutti gli LSP presenti sulla rete ed è quindi possibile definirne la migliore riorganizzazione complessiva ad un dato istante (un compito comunque assai oneroso dal punto di vista computazionale).

Entrambi gli stati possono essere trattati secondo un *approccio centralizzato*, in cui la gestione degli LSP è totalmente affidata ad un server centrale, o un *approccio decentralizzato*, in cui invece si ha una gestione delegata localmente agli ingress router. Si può comunque notare che l'adozione di un approccio centralizzato è più opportuna in uno stato offline, dove si dispone di una visione globale della rete, mentre l'approccio decentralizzato è più opportuno in un contesto online, in cui il delegare localmente ai router le decisioni di instradamento comporta notevoli riduzioni della complessità di gestione e risparmi in termini computazionali.

Date queste premesse, la soluzione più razionale nel caso di studio è naturalmente quella di adottare un approccio decentralizzato. Il passo successivo consiste nell'individuazione di un'opportuna procedura che individui i percorsi di instradamento dell'informazione associata a nuove richieste dell'utenza: si tratta quindi di definire uno schema specifico per il particolare problema di instradamento dinamico. Prima di giungere alla definizione di uno schema specifico è però utile individuare gli elementi fondamentali che caratterizzano uno schema e che devono essere adattati alle peculiarità dello scenario di studio.

In uno schema generale per realizzare il routing dinamico possono essere individuate tre componenti principali:

1. una score function
2. un constraint predicate
3. un algoritmo di definizione del nuovo LSP

La score function ricopre il ruolo di guida dell'individuazione del nuovo LSP e può essere vista come il corrispettivo della funzione obiettivo in un problema di ottimizzazione: per ogni LSP in grado di soddisfare la nuova richiesta dell'utenza, essa definisce un valore numerico che indica la "bontà" dell'LSP nel soddisfare la particolare richiesta. La sua forma non è univocamente individuata, ma può assumere le strutture più diverse, coinvolgendo i più disparati parametri che rispecchiano i particolari interessi del gestore di rete. Se l'obiettivo è, ad esempio, contenere la probabilità di blocco della rete, nella funzione si può inserire un

termine che tiene conto del bilanciamento del carico sulla rete e quindi di come si distribuisce il traffico scegliendo il nuovo LSP. Nel caso in cui si cerchi invece di minimizzare l'utilizzo di risorse, si può inserire un termine che valuti più vantaggiosi gli LSP associati ad un minor numero di hop.

Il constraint predicate è una condizione che deve essere verificata al fine di discriminare tra link utilizzabili e link non utilizzabili per soddisfare la nuova richiesta. Si può così individuare l'insieme composto dai LSP che possono soddisfare la nuova richiesta e tra i quali viene selezionato il nuovo percorso di instradamento. una possibile condizione per un generico arco e è:

$$\sum_{p \in P: e \in p} f_p + B_k \leq b_e \quad (5.32)$$

corrispondente allo stabilire che un arco risulta utilizzabile se la somma $\sum_{p \in P: e \in p} f_p$ delle bande utilizzate dall'insieme P degli LSP già presenti nella rete e di quella B_k associata alla nuova richiesta k è minore o uguale della capacità di banda b_e dell'arco. Si tratta di una condizione alquanto restrittiva che non contempla la possibilità di abbattere LSP già presenti, limitando fortemente le prospettive di instaurazione di nuovi LSP.

La terza ed ultima componente dello schema è l'algoritmo di instradamento vero e proprio, che specifica la sequenza di operazioni da effettuare real fine di individuare il nuovo LSP, utilizzando la score function adottata e rispettando i constraint predicate imposti. Anche nel caso dell'algoritmo di routing, non esiste una forma univocamente determinata, ma si possono rintracciare le strutture più diverse, che si adattano ai particolari constraint predicate e score function adottati. In ogni caso, gli algoritmi di routing possono essere ricondotti a due classi principali: quelli di ottimizzazione e quelli euristici. Nel primo caso, si ha una procedura di soluzione di un problema di ottimizzazione, in cui si cerca l'LSP che minimizza la score function e soddisfa tutti i constraint predicate. Implementare un algoritmo di ottimizzazione è il metodo ideale in quanto vengono così individuate le soluzioni migliori del problema di routing. D'altra parte ci si scontra con il grosso ostacolo della complessità computazionale, in quanto il problema di minimizzare la score function rispettando i constraint predicate è NP-Hard (per comprendere il significato di questa classificazione si veda l'Appendice A relativa alla complessità). Per aggirare l'ostacolo della complessità, giungono in soccorso le Euristiche, metodi di soluzione per i quali non è possibile dimostrare proprietà di convergenza verso la soluzione ottima del problema, ma che permettono comunque di individuare soluzioni di buona qualità con costi computazionali contenuti.

5.2.4.1 Meccanismi di preemption

I meccanismi di *preemption* costituiscono una delle maggiori peculiarità dell'*explicit routing* supportato dal MPLS-Traffic Engineering: essi consentono ai nodi di porre termine alle trasmissioni che avvengono su particolari LSP, forzando nell'abbattimento. La conseguenza è che viene liberata banda preziosa utilizzabile per instaurare un nuovo LSP associato ad una

richiesta che altrimenti non sarebbe potuta essere soddisfatta per insufficiente disponibilità di banda.[44]

La preemption si basa sull'utilizzo di due scale di priorità comprendenti ciascuna otto livelli, dallo 0 al 7, con 0 valore di massima priorità. Ad ogni LSP viene associata una coppia di valori (s,h) : s rappresenta la *setup priority*, cioè la priorità dell'LSP nell'acquisire risorse a scapito di altri LSP, mentre h rappresenta la *holding priority*, cioè l'importanza di conservare intatto l'LSP.

Il meccanismo base su cui è fondata la preemption è molto semplice: ogni richiesta di definizione di un nuovo LSP ha il potere di provocare l'abbattimento di un qualunque LSP già presente sulla rete e caratterizzato da una holding priority inferiore alla setup priority del nuovo LSP. Si ricorda che, rappresentando priorità maggiori con valori di classe minori, ciò corrisponde a chiedere che il valore di setup sia minore di quello di holding:

$$LSP_{new} \text{ può abbattere } LSP_{old} \Leftrightarrow s_{LSP_{new}} < h_{LSP_{old}}$$

Sebbene un LSP possa essere associato ad una coppia (s,h) con $s \neq h$ viene comunque richiesto che si abbia $s \geq h$, cioè che la holding priority sia superiore e quindi numericamente inferiore a quella di setup. Ciò viene imposto per evitare che un LSP instaurato con successo sia abbattuto da una richiesta immediatamente successiva di uguale setup priority. All'interno di questa trattazione si assumerà comunque, senza perdita di generalità, che ogni LSP sia caratterizzato da setup e holding priority di uguale valore e si potrà quindi utilizzare un singolo parametro di priorità s per caratterizzare ogni LSP.

L'introduzione dei meccanismi di preemption ha effetti anche sulla definizione dei constraint predicate, poiché, per verificare se un link ha sufficiente banda per soddisfare una richiesta, non si dovrà considerare più la sola banda non utilizzata, ma anche quella che può eventualmente essere liberata abbattendo LSP. Si rende quindi necessaria l'introduzione di un nuovo parametro $preempt_e^\sigma$, che specifica, per ogni arco $e \in E$ del grafo, la banda totale che può essere liberata mediante preemption operata da un LSP di priorità σ . La priorità di un LSP è pari al quarto elemento della quadrupla $(s_k, t_k, B_k, D_k, \sigma_k)$ che si può usare per individuare una *commodity* k (si ricorda che i primi due elementi s_k, t_k individuano i nodi origine e destinazione dell'LSP da definire e B_k è la banda richiesta).

La relazione matematica che definisce la banda preemptable in un arco e per la classe di priorità σ è:

$$preempt_e^\sigma = \sum_{p \in P: e \in p \wedge \sigma_k < \sigma} f_p \quad (5.33)$$

Se si introduce un parametro res_e^σ indicante la banda dell'arco e complessivamente utilizzata da LSP di priorità σ che contengono l'arco, il constraint predicate per una *commodity* $(s_k, t_k, B_k, D_k, \sigma_k)$ diventa:

$$\sum_{\sigma < \sigma_k} res_e^\sigma + B_k \leq b_e \quad (5.34)$$

equivalente a verificare se la somma della banda res_e^σ e utilizzata da LSP non abbattibili e di quella B_k richiesta risulta inferiore o uguale alla capacità di banda dell'arco. Rispetto al constraint predicato utilizzato in uno scenario senza preemption, ciò che cambia è che la banda utilizzata da altri LSP non è più totalmente intoccabile, ma può essere liberata abbattendo LSP di minore importanza rispetto alla corrente richiesta di instaurazione: gli LSP caratterizzati da valori di priorità maggiori e quindi meno importanti è come se non esistessero e non vengono considerati nella verifica della disponibilità di banda.

Conseguenza dell'adozione dei meccanismi di preemption è il sorgere di un nuovo problema: secondo quali criteri e modalità selezionare degli LSP da abbattere al fine di liberare banda? Il successivo paragrafo è dedicato a dare una risposta a questa domanda, mostrando come trattare il problema e risolverlo tramite una semplice ma efficace euristica.

Quando non vi è sufficiente disponibilità di risorse ed arriva ad un ingress router una richiesta per il set up di un nuovo LSP la tecnica della preemption potrebbe consentire il soddisfacimento della richiesta, assicurando che i flussi di traffico con maggiore priorità siano serviti nel modo più efficiente possibile. Se per eseguire la preemption di un LSP si considerasse solamente la sua la priorità purché la banda offerta sia maggiore o uguale a quella richiesta, si potrebbe verificare un spreco della banda. Per esempio ipotizziamo che in un ingress router arrivi una richiesta da parte di un LSP L_1 con priorità 1 e banda pari a 50 Mbit/s e che vi siano un LSP L_2 con priorità 5 caratterizzato da una banda di 100Mbit/s, e un altro L_3 con priorità 4 ma banda 55 Mbit/s, se scegliessimo in base alla priorità la nostra scelta sarebbe L_2 con uno spreco di banda di 50 Mbit/s. Inoltre un solo LSP potrebbe non essere sufficiente a soddisfare la richiesta di banda, in questo caso la preemption potrebbe essere applicata a più LSP. Il numero di questi LSP dovrebbe essere il minimo possibile per limitare al massimo il rerouting ed i costi di segnalazione.

Nell'ottica dell'ottimizzazione della rete che si intende realizzare, risulta particolarmente importante la scelta, non solo, di quali LSP rimpiazzare, ma anche, di dove questi LSP rimpiazzati verranno reinstradati.

La Figura 10 mostra un esempio di LSP preemption con rerouting: l'LSP BLU viene abbattuto per lasciare risorse al LSP ROSSO e viene reinstradato lungo un percorso subottimo.

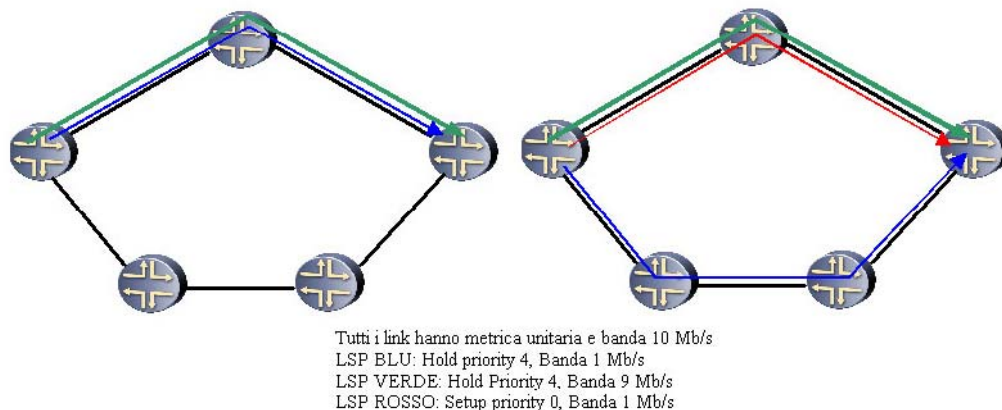


Figura 10 Esempio di preemption con rerouting di LSP

Il rerouting degli LSP in seguito alla preemption risulta in definitiva applicabile alle esigenze del servizio, ed è in grado di garantire l'affidabilità, fintanto che sulla rete è presente un path sul quale è possibile instradare il LSP a banda garantita.

5.2.4.2 Definizione di una procedura di routing dinamico preemption based

Come si è visto nelle precedenti sezioni, allo scopo di definire una procedura di routing in uno stato online della rete è fondamentale definire anche una procedura di preemption che specifica come deve avvenire la selezione degli LSP da abbattere. A questo fine, nella presente sezione, fatte le debite premesse e introdotti i necessari elementi, si procederà prima all'introduzione di un *Problema di Programmazione Lineare Binaria (PL01)* per l'individuazione ottima dei cammini da abbattere e successivamente alla definizione di un'euristica che consente l'individuazione di soluzioni di buona qualità del problema di preemption, affrontando costi computazionali più contenuti.

Si consideri l'arrivo di una richiesta di trasmissione dati individuata da una *commodity* $(s_k, t_k, B_k, D_k, \sigma_k)$. Se il soddisfacimento della richiesta di banda B_k rende necessario operare la preemption, gli LSP da abbattere vengono scelti tra quelli con priorità $\sigma > \sigma_k$ in modo da raggiungere la quantità:

$$bwToFree_e^k = B_k - freeBW_e \quad (5.35)$$

rappresentante la banda complessiva da liberare sull'arco e per soddisfare la *commodity* k (tale valore può essere ottenuto sottraendo alla banda richiesta B_k la banda libera

$$freeBW_e = b_e - \sum_{p \in P_k: e \in E} f_p \quad (5.36)$$

dell'arco e , pari alla differenza tra la capacità di banda dell'arco b_e e la banda complessivamente occupata da tutti gli LSP già presenti sulla rete che transitano sull'arco).

Nella successiva trattazione, il parametro può anche essere presente nella forma $freeBW_{ij}$ dove l'arco e a cui è associato viene denotato specificando i suoi nodi coda e testa (i, j) .

La formulazione del nuovo **MCFP** viene definita rispetto ad un grafo ottenuto modificando il grafo originale della rete. Si ricorre in particolare ad un *multigrafo* $M(V, F)$ ovvero un grafo in cui è ammissibile che una molteplicità di archi colleghino una stessa coppia di nodi. La necessità di ricorrere a questo speciale tipo di grafo deriva dal fatto che, per ogni arco, l'invio di informazioni può avvenire utilizzando o la banda libera non usata da LSP o quella preemptable: l'utilizzo dei due tipi di banda comporta costi diversi, in quanto la preemption implica oneri di gestione aggiuntivi, e di conseguenza, per ogni arco $e \in E$ del grafo originale, nel multigrafo si definiscono due archi paralleli f e f' dei quali, uno ha capacità pari alla banda libera di e e uno ha capacità pari alla banda preemptable di e .



Passaggio dal grafo al multigrafo

Per modellare il fatto che in ogni arco del grafo originale prima si utilizza la banda libera e poi eventualmente si ricorre a quella ottenibile tramite preemption, il costo dell'arco associato alla banda preemptable può essere definito come un multiplo del costo dell'arco associato alla banda libera: il fattore di moltiplicazione, che persegue anche lo scopo di scoraggiare il ricorso alla preemption, può essere fissato uguale per tutti gli archi o, nel caso sia ritenuto opportuno e conveniente, anche essere differenziato in relazione agli oneri di preemption che potenzialmente si potrebbero essere costretti ad affrontare.

Il multigrafo $M(V, F)$ si costruisce nel seguente modo:

- l'insieme dei nodi V coincide con quello del grafo originale $G(V, E)$
- l'insieme degli archi F include due archi (i, j) e $(i, j)'$ per ogni arco $(i, j) \in E$
- i parametri dell'arco $(i, j) \in F$ sono posti pari a:
 - $d_{ij} = d_{ij}$
 - $b_{ij} = freeBW_{ij}$
 - $c_{ij} = c_{ij}$
- i parametri dell'arco $(i, j)' \in F$ sono posti pari a:
 - $d_{ij'} = 0$
 - $b_{ij'} = preempt_e^\sigma$
 - $c_{ij'} = \theta c_{ij}$

Si noti che i parametri presenti al membro destro delle uguaglianze sono quelli dell'arco $(i, j) \in E$. Il fattore θ che moltiplica il costo di invio di un'unità di flusso su archi associati a banda preemptable è un valore positivo sufficientemente grande che ha l'importante effetto di far usare per ogni arco prima la banda libera e poi eventualmente quella preemptable. Il *delay*

dell'arco (i, j) può di conseguenza essere posto a zero, poiché la banda preemptable viene usata solo dopo che tutta la banda libera disponibile sia stata utilizzata e quindi solo dopo che sia stata inviata informazione sull'arco $(i, j) \in E$, sperimentando il relativo delay d_{ij} che non deve essere ulteriormente considerato.

Il nuovo **MCFP** viene risolto sul *multigrafo* $M(V, F)$ rispetto alla sola *commodity* $(s_k, t_k, B_k, D_k, \sigma_k)$, associata alla nuova richiesta. Nel caso che il problema non ammetta soluzione, si può concludere che la nuova richiesta non può essere soddisfatta nemmeno ricorrendo alla preemption. Se invece esiste una soluzione, si ottiene l'insieme di cammini di costo minimo che consentono di inviare il flusso B_k dal nodo s_k al nodo t_k . Come procedura soluzione si può nuovamente adottare la column generation, tenendo presente che tra i due **MCFP** sussistono delle differenze che impattano solo sulla dimensione del problema: si considera un multigrafo invece di un grafo, con numero di archi raddoppiato, e una singola commodity invece di una molteplicità di commodity.

La soluzione del MCFP individua quanta banda di ogni arco del multigrafo $M(V, F)$ deve essere impiegata per soddisfare la nuova richiesta e permette quindi di stabilire quali sono gli archi su cui deve essere effettuata la preemption e la quantità di banda che deve essere liberata. Una volta individuati questi archi, si tornano a considerare i loro corrispettivi sul grafo $G(V, E)$ e per ognuno di essi viene portata avanti la procedura di preemption secondo le modalità introdotte nella successiva parte della trattazione.

5.4.2 Individuazione degli LSP da abbattere

Da questo punto in poi l'attenzione verrà focalizzata sulla procedura di preemption operata su un generico arco $e \in E$ del grafo (ottenuto mediante la risoluzione del MCFP sopra introdotto) e tutte le considerazioni si riferiranno ad un singolo arco preso come esempio. Per facilitare la leggibilità delle formulazioni e alleggerire la notazione, l'indice e denotante il generico arco verrà sottinteso in tutta la successiva trattazione, a meno di esigenze e casi particolari opportunamente evidenziati.

Si introduce l'insieme Ψ degli LSP transitanti su un generico arco e aventi priorità minore (numericamente superiore) rispetto a quella σ_k dell'attuale commodity che richiede la definizione di un nuovo LSP. Per ogni $p \in \Psi$ si possono individuare la banda f_p da esso utilizzata e la priorità associata σ_p , deducibile dalla quadrupla individuante la commodity per cui l'LSP è stato definito. Si definisce inoltre per ogni $p \in \Psi$ un parametro di costo t_p che viene utilizzato come addendo nella score function e che assume la semplice forma di una funzione lineare nella priorità dell'LSP:

$$t_p = 8 - c_p \tag{5.37}$$

Tramite questa forma, il costo di un LSP $p \in \Psi$ cresce linearmente all'aumentare dell'importanza dell'LSP considerato. Inserito all'interno della score function, il parametro t_p

modella l'interesse a limitare l'abbattimento di LSP che risultano più importanti e a liberare risorse mediante abbattimento di LSP poco importanti.

La variabile di decisione alla base del problema di ottimizzazione deve individuare l'insieme ottimo di LSP da abbattere ed è rappresentata da un vettore booleano $x \in \{0,1\}^{|\Psi|}$ che possiede tante componenti quanti sono gli LSP che possono essere abbattuti e il cui generico elemento è definito nel seguente modo:

$$x_p = \begin{cases} 1 & \text{se l'LSP } p \in \Psi \text{ viene abbattuto} \\ 0 & \text{altrimenti} \end{cases} \quad (5.38)$$

L'insieme ammissibile risulta descritto da un singolo vincolo lineare che modella la necessità di abbattere un insieme di LSP $\Psi^e \subseteq \Psi$ tale da liberare una banda uguale o superiore a quella $bwToFree_e^k$ (riportata di seguito senza indice e) necessaria per soddisfare la nuova commodity k :

$$\sum_{p \in \Psi^e} f_p \cdot x_p \geq bwToFree_k \quad (5.39)$$

Per quanto riguarda la funzione obiettivo, si perseguono complessivamente tre obiettivi: minimizzare la priorità degli LSP abbattuti; minimizzare il numero degli LSP abbattuti; minimizzare la banda liberata.

I tre corrispondenti elementi presenti nella funzione risultano linearmente combinati secondo coefficienti reali α, β e γ che devono essere opportunamente dimensionati al fine di individuare il loro peso relativo nel guidare la scelta ottima delle soluzioni ammissibili. La funzione obiettivo assume quindi la forma:

$$\sum_{p \in \Psi} [\alpha(t_p \cdot x_p) + \beta x_p + \gamma(f_p \cdot x_p)] \quad (5.40)$$

Il primo addendo della sommatoria penalizza l'abbattimento di LSP importanti, caratterizzati da bassi valori di priorità e quindi alti valori della funzione di costo. Il secondo è semplicemente la somma del numero di LSP abbattuti, moltiplicata per il corrispondente coefficiente di costo. Il terzo addendo, infine, è pari alla somma delle bande liberate tramite abbattimento, moltiplicata per il coefficiente di penalizzazione γ .

Se si adotta una notazione vettoriale, utilizzando i vettori $t, f \in R^{|\Psi|}$ per rappresentare rispettivamente i costi e la banda occupata degli LSP preemptable e il vettore 1 a $|\Psi|$ componenti unitarie, il problema di ottimizzazione della procedura di preemption può essere scritto come:

$$\min \alpha(t^T \cdot x) + \beta(1^T \cdot x) + \gamma(f^T \cdot x) \quad (5.41)$$

$$s.t. \quad f^T \cdot x \geq bwToFree_k \quad (5.42)$$

$$x \in \{0,1\}^{|\Psi|} \quad (5.43)$$

Il problema introdotto deve essere risolto per ogni arco del grafo interessato da preemption. La sua risoluzione si scontra però con un notevole ostacolo: il problema è NP-Hard (come dimostrato in [45]) quindi può diventare improponibile risolverlo in reti di grandi dimensioni su cui continuamente si instaurano, modificano e abbattono un numero vastissimo di LSP. Per ovviare alle difficoltà, si introduce una semplice euristica che consente di individuare delle soluzioni di qualità per il problema di ottimizzazione, utilizzando una procedura di facile implementazione. La scelta degli LSP da abbattere è in particolare guidata da una score function $S(\cdot)$, che associa un valore ad ogni LSP $p \in \Psi$ in base ai tre obiettivi precedentemente introdotti, assumendo la forma:

$$S(p) = \alpha \cdot t_p + \beta \frac{1}{f_p} + \gamma (f_p - bwToFree_k)^2 \quad (5.44)$$

Il primo addendo rappresenta il costo associato ad ogni LSP. Il secondo addendo persegue la minimizzazione del numero di LSP abbattuti, includendo un fattore di penalità $\frac{1}{f_p}$ inversamente proporzionale alla banda occupata dal LSP considerato: si favorisce così l'abbattimento degli LSP che occupano la maggiore banda e che in numero minore riescono a liberare la banda richiesta. Il terzo addendo orienta la scelta di abbattimento verso LSP che implicano il minor spreco di banda liberata, includendo un fattore $(f_p - bwToFree_k)^2$ dove è presente la differenza tra la banda liberata e quella richiesta (l'elevazione al quadrato ha il compito di rendere ancora più evidente la differenza). Per il dimensionamento dei pesi α, β e γ si può fare riferimento a [27].

Le operazioni fondamentali eseguite dall'euristica ad ogni iterazione sono:

1. verifica della disuguaglianza $preempt_k \geq bwToFree_k$ - si testa se la banda totale liberata tramite preemption $preempt_k$ per soddisfare la commodity k è pari o superiore alla banda $bwToFree_k$ che deve essere liberata. In caso di soddisfacimento, l'euristica si arresta e si abbattono gli LSP individuati, altrimenti l'euristica prosegue per individuare altri LSP da abbattere e raggiungere il volume di banda che si richiede di liberare;
2. selezione del $\tilde{p} \in \Psi : S(\tilde{p}) \leq S(p), \forall p \in \Psi$ - viene selezionato per l'abbattimento l'LSP a cui è associato il valore minimo della score function;
3. $preempt_k = preempt_k + f_{\tilde{p}}$ -aggiornamento del parametro che indica la banda totale liberata tramite preemption;
4. $\Psi^r = \Psi^r \cup \{\tilde{p}\}$ - aggiornamento dell'insieme di LSP da abbattere con unione di quello selezionato
5. $\Psi = \Psi - \{\tilde{p}\}$ - aggiornamento dell'insieme di LSP candidati all'abbattimento con eliminazione di quello selezionato.

L'output dell'euristica è quindi rappresentato da un insieme di LSP $\Psi^\tau \subseteq \Psi$ che sono caratterizzati dai più bassi valori della score function $S(\cdot)$ e che una volta abbattuti liberano sufficienti risorse di banda per il soddisfacimento della nuova commodity k .

La soluzione del problema di ottimizzazione corrispondente all'output generato dall'euristica si ottiene ponendo pari ad uno le componenti del vettore di decisione $x \in \{0,1\}^{|\Psi|}$ che corrispondono ai LSP contenuti nell'insieme $\Psi^\tau \subseteq \Psi$ e lasciando a zero le restanti.

Si noti che, al fine di cercare di contenere l'utilizzo di banda, l'ordine di selezione degli LSP non tiene conto solo del corrispondente valore della score function: a parità di tale valore, gli LSP vengono selezionati per crescente banda occupata f_p .

Pseudocodice dell'Euristica di Preemption

Dopo aver visionato i fondamenti della procedura, si può ora passare a definire in dettaglio tutta la sequenza ordinata di operazioni messe in atto dall'euristica. Di seguito si riporta il suo pseudocodice:

INPUT:

- insieme Ψ degli LSP candidati all'abbattimento
- vettore f delle occupazioni di banda degli LSP in Ψ
- vettore t dei costi degli LSP in Ψ
- larghezza di banda da liberare $bwToFree_k$
- coefficienti della score function α, β e γ

OUTPUT:

- vettore booleano x degli LSP da abbattere

Inizializzazione: $preempt_k = 0$

- 1 - **IF** $\sum_{i \in \Psi} f_i \geq bwToFree_k$ {
 - 1.1 Calcola $S(i) = \alpha \cdot t_i + \beta \frac{1}{f_i} + \gamma(f_i - bwToFree_k)^2, \forall i \in \Psi$
 - 1.2 Ordina i vettori (S, f) per valore crescente di $S(i)$
 - 1.3 Se S presenta componenti di valore uguale, ordina per valore crescente di f
- 2 - **WHILE** $preempt_k < bwToFree_k$ {
 - 2.1 Seleziona il primo LSP p non analizzato del vettore S
- 3 - **IF** $f_p < bwToFree_k$ {
 - 4 - **IF** S non presenta componenti di valore uguale {
 - 4.1 $preempt_k = preempt_k + f_p$
 - 4.2 $x_p = 1$
 - 4.3 $p = p + 1$
- ELSE (rif. IF4) {
 - 5 - **IF** $\exists i: (S_i = S_p) \wedge (f_i \geq bwToFree_k)$

```

5.1  $preempt_k = f_i$ 
5.2  $x_i = 1$ 
5.3  $x_j = 0, \forall j \neq i$ 
5.4 BREAK
}
ELSE (rif.IF5) {
6 - IF  $\exists i : (S_i = S_p) \wedge (preempt_k + f_i > bwToFree_k)$ 
6.1  $preempt_k = preempt_k + f_i$ 
6.2  $x_i = 1$ 
6.3 BREAK
}
ELSE (rif.IF6) {
 $j = \max\{i : S_i = S_p\}$ 
 $k = j + 1$ 
7 - WHILE  $(preempt_k > bwToFree_k) \text{ OR } (j \geq p)$  {
7.1  $preempt_k = preempt_k + f_i$ 
7.2  $x_j = 1$ 
7.3  $j = j - 1$ 
7.4  $p = k$ 
}
}
}
}
ELSE (rif.IF3) {
 $preempt_k = f_p$ 
 $x_p = 1$ 
 $x_j = 0, \forall j \neq i$ 
BREAK
}
}
}
}

```

ELSE (rif.IF1) la commodity k non può essere soddisfatta

L'inizializzazione ha il compito di porre a zero il parametro $preempt_k$, che individua la banda liberata tramite abbattimento degli LSP.

La prima struttura condizionale (**1-IF**) verifica se vi è sufficiente banda che può essere liberata per soddisfare la richiesta k : la somma $\sum_{i \in \Psi} f_i$ dei volumi di banda occupati da tutti gli LSP candidati per l'abbattimento viene confrontata con quella $bwToFree_k$ da liberare. Se la somma è minore del volume da liberare, la richiesta non potrà essere soddisfatta nemmeno

abbattendo tutti gli LSP e di conseguenza l'euristica si arresta senza poter trovare una soluzione (il problema di ottimizzazione è inammissibile - l'insieme delle soluzioni è vuoto).

Le tre operazioni **(1.1)**, **(1.2)**, **(1.3)** sono finalizzate allo stabilire un ordinamento degli LSP: in primo luogo viene associato ad ogni LSP di Ψ il valore stabilito dalla score function $S(\cdot)$, poi i valori vengono immagazzinati in un vettore $S \in R^{|\Psi|}$ secondo un ordine crescente. Il vettore f delle bande occupate viene riordinato secondo l'ordine assunto dai LSP in S . Se degli LSP possiedono lo stesso valore S_i vengono ordinati per valore crescente della banda occupata f_i . Il ciclo **(2 - WHILE)** viene inserito per ripetere la procedura di abbattimento di un LSP finché non è stata liberata sufficiente banda e la sua prima operazione **(2.1)** consiste nella selezione del primo LSP (indicato come p) non ancora analizzato del vettore S . Si passa quindi ad una nuova struttura condizionale, **(3-IF)**, in cui si verifica se l'LSP p selezionato riesce da solo a liberare la banda necessaria. Se sì, l'euristica si arresta fornendo in uscita un vettore con un'unica componente unitaria corrispondente all'LSP p da abbattere (le particolari operazioni sono specificate nel blocco **ELSE** corrispondente).

Il blocco condizionale **(4-IF)** verifica la natura delle componenti di S e incanala verso il corretto blocco di operazioni successive da seguire: se le componenti sono tutte distinte, si devono semplicemente abbattere gli LSP, uno alla volta e secondo l'ordine di S , finché non è liberata sufficiente banda. Le operazioni eseguite sono l'aggiornamento del parametro di banda abbattuta **(4.1)**, il settaggio ad uno della corrispondente variabile nel vettore x **(4.2)** e l'aggiornamento dell'indice che punta l'LSP da analizzare p così da puntare il successivo **(4.3)**.

Il blocco **ELSE** corrispondente al **(4-IF)** include tre blocchi **IF** che esauriscono le eventualità che si possono presentare quando nel vettore S sono presenti valori duplicati. In **(5-IF)** si verifica se tra gli LSP che hanno lo stesso valore S_p di quello attualmente analizzato, ve ne sia uno che, una volta abbattuto, possa da solo liberare la banda necessaria. Se sì, l'LSP individuato è il solo ad essere abbattuto e l'euristica termina con fornita in uscita la soluzione in cui è settato ad uno solo il corrispondente elemento j . Nel blocco **(6-IF)** si verifica se tra gli LSP che hanno lo stesso valore S_p di quello attualmente analizzato, ve ne sia uno che, in combinazione agli altri già abbattuti, consenta di liberare la banda necessaria. Se sì, l'LSP individuato viene aggiunto agli altri abbattuti aggiornando il vettore x e l'euristica termina. Il blocco **ELSE** più annidato (corrispondente di **(6-IF)**) viene eseguito solo quando S ha delle componenti di uguale valore e l'LSP al momento analizzato non è in grado, da solo o in combinazione agli altri LSP già abbattuti, di liberare sufficiente banda. Si rende allora necessario scandire i sottoinsiemi di LSP caratterizzati da stesso valore S , liberando risorse di banda iterazione dopo iterazione. La prima operazione consiste nel individuare l'indice maggiore j del sottoinsieme di LSP di identico valore. Si determina poi l'indice minore k del successivo blocco di LSP di uguale valore che vanno eventualmente scanditi. Si entra quindi nel ciclo **(7 - WHILE)**, dove vengono abbattuti gli LSP in ordine decrescente di banda occupata f fino a quando gli LSP del blocco da scandire non finiscono o si è liberato il volume

di banda richiesto. Se gli LSP del blocco sono finiti e non si ha ancora la quantità di banda desiderata, si passa al successivo blocco di LSP di valore S superiore.

5.4.3 Tecnica per il processo di rerouting degli LSP abbattuti

L'implementazione dei meccanismi di preemption all'interno di una rete pone una questione assai importante: cosa fare dell'informazione trasmessa su LSP che vengono selezionati per l'abbattimento? Si potrebbe adottare la soluzione più semplice, consistente nel non fare assolutamente nulla, ma la conseguenza sarebbe di perdere completamente l'informazione non ancora trasmessa all'istante di abbattimento. Ciò andrebbe però contro l'intento fondamentale che si persegue in questo studio: definire delle procedure di routing che migliorino complessivamente la capacità di gestione del traffico della rete [46], [47], [48].

La semplice preemption senza ricovero degli LSP abbattuti ha il solo effetto di sostituire traffico di una certa priorità con traffico di priorità maggiore: LSP vengono abbattuti per liberare risorse e consentire l'instaurazione di più prioritari LSP, a scapito di un'utenza che vede improvvisamente terminato il proprio processo di comunicazione. Anche se i processi terminati possono essere spesso reinstradati poco dopo l'abbattimento, utenti di classi di priorità più basse si trovano a sperimentare una discontinuità del servizio che risulta deleteria per la loro soddisfazione. Il gestore di rete ha quindi tutto l'interesse a minimizzare i fenomeni di “*service destruction*”, adottando opportune procedure di rerouting degli LSP abbattuti, che risultino il più possibile trasparenti per l'utenza.

L'intento di migliorare complessivamente la capacità di gestione della rete può essere tradotto nell'obiettivo di massimizzare il volume di traffico totale transitante sulla rete e massimizzare, quindi, il throughput associato ai LSP instaurati. Il processo di preemption deve di conseguenza essere seguito da un processo di rerouting, il cui scopo fondamentale è preservare il traffico associato a LSP abbattuti, deviandolo su percorsi alternativi dove la trasmissione può continuare senza perdite. A questo fine, nel momento in cui si presenta una nuova commodity che richiede abbattimenti, si può adottare una particolare procedura, la soft preemption [49], che si articola nei seguenti passi:

1. selezione, tramite un algoritmo di preemption, dell'LSP p da abbattere;
2. individuazione dei percorsi alternativi per p ;
3. instaurazione dei percorsi su cui deviare il traffico di p e deviazione del traffico su di essi;
4. abbattimento del percorso p originario e rilascio delle corrispondenti risorse;
5. instaurazione dell'LSP corrispondente alla nuova commodity.

La procedura sopra introdotta viene eseguita per ogni LSP di cui si è deciso l'abbattimento e consente di deviare su un percorso alternativo il traffico di un LSP da abbattere prima che l'abbattimento abbia luogo. Il suo nodo centrale è rappresentato dal concedere un “periodo di grazia” ad ogni LSP da abbattere, durante il quale si tenta di individuare dei percorsi alternativi

su cui continuare la comunicazione associata. Il vantaggio che si ottiene è così duplice: commodity di alta priorità possono acquisire risorse ed essere soddisfatte e il traffico già in transito sulla rete viene preservato contenendo o eliminando del tutto i casi di service disruption.

L'individuazione dei percorsi alternativi (operazione numero 2) può essere messa in atto secondo due modalità principali, distinte in base al criterio di individuazione adottato e indicate con i nomi di *global rerouting* e *local rerouting*: nel caso del global rerouting, i percorsi alternativi vengono ricercati tra i nodi origine e destinazione dell'LSP da abbattere, mentre nel caso del local rerouting, i percorsi sono ricercati tra il nodo coda e il nodo testa dell'arco su cui si sta operando la preemption. Naturalmente l'approccio globale comporta una complessità molto maggiore della procedura di individuazione rispetto a quello locale, essendo necessario valutare generalmente alternative molto più numerose ed articolate.

Problemi di Massimo Flusso nel processo di rerouting

Il rerouting a carattere locale si presta naturalmente ad essere utilizzato in un contesto dove la gestione dell'instradamento è delegata localmente ai singoli router di transito, consentendo di ottenere, rispetto ad un approccio accentrato e globale, notevoli semplificazioni della complessità di gestione e di elaborazione e una velocizzazione dello stesso processo di rerouting. L'individuazione dei percorsi alternativi può essere attuata basandosi sulla risoluzione di un problema di **Massimo Flusso**, in cui i nodi sorgente e pozzo sono rispettivamente rappresentati dal nodo coda e dal nodo testa dell'arco su cui viene operata la preemption. La risoluzione del problema consente di individuare la banda totale disponibile e i cammini che possono essere utilizzati per il rerouting, evitando l'arco su cui gli LSP vengono abbattuti. Più precisamente, la formulazione del problema per il grafo in cui non si considera l'arco (\tilde{s}, \tilde{t}) , è:

$$\max F \tag{5.45}$$

$$s.t. \quad \sum_{j \in V} y_{ij} - \sum_{j \in V} y_{ji} = \begin{cases} F & \text{per } i = \tilde{s} \\ 0 & \forall i \in V - \{\tilde{s}, \tilde{t}\} \\ -F & \text{per } i = \tilde{t} \end{cases} \tag{5.46}$$

$$0 \leq y_{ij} \leq freeBW_e \tag{5.47}$$

$$F \leq \sum_{p \in \Psi^r} f_p \tag{5.48}$$

L'obiettivo è massimizzare il flusso F , corrispondente alla banda che può essere complessivamente utilizzata per trasmissioni dal nodo \tilde{s} al nodo \tilde{t} (rispettivamente coda e testa dell'arco su cui si sta operando la preemption), ricorrendo a cammini alternativi che non includono l'arco (\tilde{s}, \tilde{t}) . Il vincolo (5.46) assicura la conservazione del flusso: la somma dei flussi entranti in un nodo j deve essere pari alla somma dei flussi uscenti da esso (fanno eccezione il nodo sorgente \tilde{s} , che ha solo flusso uscente, e il nodo pozzo, che ha solo flusso entrante). Il vincolo (5.47) specifica che il flusso y_{ij} , cioè la banda di ogni arco utilizzata per il

rerouting, deve essere non negativo e non superiore alla banda libera dell'arco $freeBW_{(i,j)}$. Il vincolo (5.48), infine, stabilisce un limite superiore al valore del flusso, pari alla somma delle bande $\sum_{p \in \Psi^\tau} f_p$ degli LSP abbattuti che devono essere reinstradati.

La soluzione ottima del problema di massimo flusso è costituita da un numero reale F che rappresenta il volume di banda che può essere utilizzato nel processo di rerouting, escludendo l'arco su cui viene operata la preemption. Ad esso è associato un vettore $y \in \mathfrak{R}^{|E|}$, le cui componenti, ognuna corrispondente ad un arco del grafo, specificano la banda libera che viene utilizzata per il reinstradamento.

Dal confronto di F con il volume di banda $\sum_{p \in \Psi^\tau} f_p$ liberato tramite preemption, si può capire se parte del volume di traffico è destinato ad andare perduto: se F risulta maggiore della banda liberata vi saranno infatti uno o più LSP per cui non è possibile operare il rerouting conducendo a fenomeni di *service disruption*.

È importante sottolineare che la banda utilizzabile per il rerouting non viene sempre completamente utilizzata. Tale concetto può essere chiarito da un esempio: si supponga che l'insieme degli LSP abbattuti sia $\Psi^\tau = \{p_1, p_2, p_3, p_4, p_5\}$ e che il corrispondente vettore delle occupazioni di banda (espresso in generiche unità di occupazione) sia $f = (6, 8, 4, 5)$. Si supponga inoltre che la banda disponibile per il rerouting sia $F = 26$. Se gli LSP vengono reinstradati secondo l'ordine con cui sono indicati nell'insieme Ψ^τ , sarà possibile reinstradare solo i primi quattro LSP: la somma delle bande occupate da essi è pari a 24 e ci sarebbe quindi possibilità di reinstradare ancora 2 unità di banda. Il successivo LSP da reinstradare ha però un'occupazione pari a 5, superiore a quella disponibile, e di conseguenza è destinato ad andare perduto. La banda effettivamente utilizzata è quindi pari a 24 e minore di quella complessivamente disponibile per il rerouting.

Dall'esempio precedente, si evince che, data in generale l'impossibilità di reinstradare tutti gli LSP abbattuti, si devono definire dei criteri secondo cui stabilire un ordine di reinstradamento degli LSP $\in \Psi^\tau$. Data la grande attenzione e cura che si suppone il gestore presti ai processi di comunicazione di più alta priorità, un primo fondamentale criterio di ordinamento appare costituito dalla classe di priorità: il rerouting sarà operato in ordine decrescente di priorità degli LSP e quindi di valore crescente del parametro σ_p associato ad ogni $p \in \Psi^\tau$. Nel caso siano presenti LSP abbattuti caratterizzati dallo stesso valore di priorità, l'ordinamento è guidato da una funzione di costo di rerouting (indicata con l'acronimo RRCF - ReRouting Cost Function):

$$RRCF(p) = C_{hop} \cdot H_p + FC \quad (5.49)$$

Il primo addendo è una componente variabile che tiene conto dei costi delle operazioni di segnalazione dell'abbattimento, operate in ogni nodo attraversato dall'LSP p : esso risulta

proporzionale al numero di router attraversati e quindi al numero di hop H_p eseguiti da p , secondo un coefficiente di costo C_{hop} del singolo hop. Il secondo addendo, FC , modella invece il costo fisso di notifica dell'abbattimento. L'obiettivo è naturalmente la minimizzazione dei costi, di conseguenza l'ordinamento degli LSP di pari priorità avviene per costo $RRCF(p)$ crescente.

Fatte queste premesse, si può giungere finalmente alla definizione di un algoritmo per il rerouting, le cui operazioni chiave sono due: individuare iterazione dopo iterazione l'insieme di LSP abbattuti per cui è possibile attuare il rerouting e definire, per ogni LSP abbattuto e reinstradabile, l'insieme di cammini su cui viene deviato il traffico.

Si richiama l'attenzione sul fatto che il traffico di un LSP può venire reinstradato su una molteplicità di cammini diversi che complessivamente consentono di trasferire l'informazione dal nodo origine al nodo destinazione: avere a disposizione sufficiente banda per il reinstradamento di un LSP non implica infatti che la banda sia disponibile su unico cammino e di conseguenza da un LSP abbattuto possono essere generati un numero $m \geq 1$ di nuovi LSP di reinstradamento.

Di seguito si riporta lo pseudocodice dell'algoritmo:

INPUT:

- insieme Ψ^τ degli LSP che verranno abbattuti
- vettore f delle occupazioni di banda degli LSP in Ψ^τ
- algoritmo A per l'individuazione di un cammino tra due nodi di un grafo

OUTPUT:

- insieme $\Psi^{re} \subseteq \Psi^\tau$ degli LSP reinstradabili
- insieme Ω dei nuovi LSP su cui verrà deviato il traffico

Inizializzazione: $\Psi^{re} = \emptyset$, $\Omega = \emptyset$

1 - Risolvi il Problema di Massimo Flusso ottenendo (F, y)

2 - IF $F = \sum_{p \in \Psi^\tau} f_p$ {

2.1 $\Psi^{re} = \Psi^\tau$

3 - WHILE $\Psi^{re} \neq \emptyset$ {

3.1 Seleziona un LSP $p \in \Psi^\tau$

3.2 Individua un insieme di cammini Ω_p che consentono di reinstradare p , usando il vettore y e l'algoritmo A

3.3 $\Psi^\tau = \Psi^\tau - \{p\}$

3.4 $\Omega = \Omega \cup \Omega_p$

}

}

4 - ELSE (rif. IF2) {

4.1 Calcola $RRCF(p)$, $\forall p \in \Psi^\tau$

4.2 Immagazzina gli LSP $p \in \Psi^\tau$ in un vettore $\psi \in \mathfrak{R}^{|\Psi^\tau|}$

4.3 Ordina il vettore ψ per valore crescente di priorità σ_p

4.4 Se esistono LSP di stessa priorità σ_p , ordinali in ψ per valore crescente di $RRCF(p)$

4.5 $i = 0$

4.6 $RRBW = 0$

5 - WHILE $(i \neq |\Psi^\tau| \text{ AND } F - RRBW \geq f_{\psi_i})$ {

5.1 Individua un insieme di cammini Ω_{ψ_i} che consentono di reinstradare ψ_i usando il vettore y e l'algoritmo A

5.2 $\Psi^{\mathfrak{R}} = \Psi^{\mathfrak{R}} \cup \{\psi_i\}$

5.3 $\Psi^\tau = \Psi^\tau - \{\psi_i\}$

5.4 $RRBW = RRBW + f_{\psi_i}$

5.5 $\Omega = \Omega \cup \Omega_{\psi_i}$

5.6 $i = i + 1$

}

}

L'algoritmo A, indicato come input nello pseudocodice, è una procedura non meglio specificata che consente di individuare dei cammini tra due nodi di un grafo. Nel caso del routing, può essere in particolare utile fare riferimento ad uno dei tanti algoritmi per l'individuazione dei primi k cammini minimi aciclici tra due nodi: se si utilizza il numero di hop come misura della lunghezza di un cammino, instradare su cammini minimi consente di limitare l'utilizzo delle risorse di banda ad un numero minore di archi rispetto a qualunque altro algoritmo, riducendo quindi la probabilità di blocco della rete per congestione.

Il primo passo dell'algoritmo consiste nell'inizializzare come vuoti $\Psi^{\mathfrak{R}}$ e Ω , che rappresentano rispettivamente l'insieme degli LSP reinstradati e quello degli LSP di reinstradamento e che vengono popolati durante l'esecuzione dell'algoritmo per poi essere forniti come output. La successiva operazione (1) prevede di risolvere il Problema di Massimo Flusso, tra il nodo coda e il nodo testa dell'arco su cui è attuata la preemption, al fine di individuare la banda F , complessivamente utilizzabile per il rerouting, e l'associato vettore y , che specifica la banda utilizzabile su ogni arco del grafo. La verifica della condizione effettuata in (2-IF) conduce ai due possibili rami di esecuzione dell'algoritmo: si verifica se la banda F risulta pari o inferiore a quella $\sum_{p \in \Psi^\tau} f_p$ di tutti gli LSP abbattuti che si devono reinstradare. Si ricorda che la relazione $F \leq \sum_{p \in \Psi^\tau} f_p$ deve sussistere, in quanto imposta come vincolo nella formulazione del corrispondente Problema di Massimo Flusso.

Se la banda F è pari a quella da reinstradare, la procedura da eseguire è relativamente poco complicata: tutti gli LSP abbattuti Ψ^τ possono essere reinstradati e l'insieme $\Psi^{\mathfrak{R}}$ può

direttamente essere posto uguale a Ψ^τ (operazione (2.1)). Il ciclo (3-WHILE) è inserito per definire i percorsi di reinstradamento di ogni LSP abbattuto: finché l'insieme degli LSP abbattuti da reinstradare non è vuoto (condizione di esecuzione del ciclo), si provvede ad estrarre un LSP p da esso (operazione (3.1)) e ad individuare l'insieme di cammini alternativi Ω_p su cui far transitare il traffico, facendo riferimento al vettore p delle disponibilità di banda e usando un algoritmo A di individuazione dei cammini tra due nodi di un grafo (operazione (3.2)). Per ultimo si provvede ad aggiornare la composizione di Ψ^τ , eliminando l'LSP p reinstradato (operazione (3.3)), e la composizione di Ω , unendo l'insieme Ω_p degli LSP di reinstradamento.

Nel caso in cui la banda F sia inferiore a quella da reinstradare, si procede al blocco **ELSE**: non tutti gli LSP abbattuti possono essere reinstradati e di conseguenza deve essere stabilito un ordine di precedenza di rerouting secondo i criteri precedentemente esposti. Ciò richiede, in primo luogo, di calcolare il costo di reinstradamento $RRCP$ di ogni LSP (operazione (4.1)). Successivamente gli LSP vengono immagazzinati in un vettore, nel quale sono poi riordinati secondo la classe di priorità e, a parità di essa, secondo il costo ((4.2), (4.3), (4.4)). Le operazioni seguenti introducono due parametri di servizio del successivo ciclo (5-WHILE): il contatore i è definito per scandire il vettore ψ , puntando iterazione dopo iterazione la componente-LSP ψ_i a maggiore priorità di reinstradamento, mentre il parametro $RRBW$ (ReRoutingBandWidth) tiene traccia della banda totale sfruttata per il reinstradamento di tutti gli LSP.

La condizione di esecuzione del ciclo specifica che il corpo viene eseguito o fino a quando non sono stati reinstradati tutti gli LSP abbattuti (condizione corrispondente all'aver raggiunto il valore $|\Psi^\tau|$ con il contatore i) o fino a quando rimane sufficiente banda per reinstradare l'LSP ψ_i attualmente puntato nel vettore ψ (condizione corrispondente al fatto che la banda residua per l'instradamento $F - RRBW$ sia maggiore o uguale di quella f_{ψ_i} utilizzata dal LSP da reinstradare). La prima operazione del corpo del ciclo, (5.1), prevede la ricerca dell'insieme di cammini Ω_{ψ_i} che consentono il reinstradamento dell'LSP ψ_i attualmente puntato. Le operazioni seguenti sono tutte operazioni di aggiornamento delle entità coinvolte: (5.2) aggiorna l'insieme degli LSP abbattuti e reinstradati unendovi ψ_i , (5.3) elimina ψ_i dall'insieme degli LSP abbattuti e da reinstradare, (5.4) aggiorna il parametro della banda utilizzata per il reinstradamento sommandovi quella del nuovo LSP reinstradato, (5.5) unisce all'insieme dei cammini di reinstradamento quelli individuati nella corrente operazione e (5.5), infine, aggiorna il contatore di ciclo.

Migliorare la qualità delle soluzioni di rerouting

La presenza del vincolo $F \leq \sum_{p \in \Psi^\tau} f_p$ nella formulazione del Problema di Massimo Flusso ha l'effetto di limitare l'esplorazione delle possibilità di reinstradamento: ci si accontenta di trovare un flusso massimo F al più pari alla banda totale richiesta per il

rerouting, senza preoccuparsi di valutare quale sia quella massima F^{\max} complessivamente disponibile. In questo modo, il vettore y associato ad F specifica solo una delle possibili distribuzioni del flusso sui differenti archi, non consentendo di valutare le diverse alternative disponibili e scegliere quella di migliore qualità. La limitazione può essere superata rimuovendo il vincolo e giungendo alla nuova formulazione, a cui si farà riferimento con il nome di *Problema di Massimo Flusso Non Limitato*:

$$\max F^{\max} \tag{5.50}$$

$$s.t. \sum_{j \in V} y_{ij}^{\max} - \sum_{j \in V} y_{ji}^{\max} = \begin{cases} F^{\max} & \text{per } i = \tilde{s} \\ 0 & \forall i \in V - \{\tilde{s}, \tilde{t}\} \\ -F^{\max} & \text{per } i = \tilde{t} \end{cases} \tag{5.51}$$

$$0 \leq y_{ij}^{\max} \leq freeBW_{(i,j)} \tag{5.52}$$

Si ipotizzi ora che la banda massima disponibile per il reinstradamento sia superiore a quella $\sum_{p \in \Psi^\tau} f_p$ necessaria per reinstradare tutti gli LSP abbattuti. Se si risolve il Problema di Massimo Flusso Limitato, si ottiene un valore $F = \sum_{p \in \Psi^\tau} f_p$ esattamente uguale alla banda necessaria per reinstradare tutti gli LSP abbattuti ed un vettore y che specifica l'utilizzo di banda degli archi strettamente sufficiente per far viaggiare il flusso di reinstradamento. Se si risolve invece il Problema di Massimo Flusso Non Limitato, si ottiene un valore $F^{\max} > F$ ed un vettore y^{\max} che può essere utilizzato come base per individuare tutte le possibili alternative di reinstradamento. In termini di componenti, sussiste la relazione $y_{ij}^{\max} \geq y_{ij}$ e deve esistere almeno un arco (k,l) tale che $y_{kl}^{\max} \geq y_{kl}$ il vettore y^{\max} offre così un insieme di soluzioni di rerouting che oltre a quelle di y ne comprende anche di aggiuntive.

Dato che il vettore y^{\max} consente di individuare tutte le alternative ammissibili di reinstradamento, acquista senso adottare un approccio ottimizzante nella scelta dei nuovi cammini su cui deviare l'informazione degli LSP abbattuti. L'insieme ottimo di cammini è in particolare ottenuto risolvendo un Multicommodity Flow Problem. Prima di arrivare alla risoluzione di questo problema si deve però passare attraverso una serie di fasi preparatorie, di seguito introdotte, necessarie per individuare tutti gli elementi in esso utilizzati. Di seguito si definisce un algoritmo il cui output è rappresentato dalla soluzione ottima del problema di rerouting:

INPUT:

- insieme Ψ^τ degli LSP abbattuti
- vettore f delle occupazioni di banda degli LSP in Ψ^τ

OUTPUT:

- insieme $\Psi^{\text{R}} \subseteq \Psi^{\tau}$ degli LSP abbattuti che possono essere reinstradati
- insieme Ω dei nuovi LSP su cui verrà deviato il traffico

Inizializzazione: $\Psi^{\text{R}} = \emptyset, \Omega = \emptyset$

1 - Risolvi il Problema di Massimo Flusso Non Limitato ottenendo $(F^{\text{max}}, y^{\text{max}})$

2 - **IF** $F^{\text{max}} = \sum_{p \in \Psi^{\tau}} f_p$ {

2.1 Calcola $RRCF(p), \forall p \in \Psi^{\tau}$

2.2 Immagazzina gli LSP $p \in \Psi^{\tau}$ in un vettore $\psi \in \mathfrak{R}^{|\Psi^{\tau}|}$

2.3 Ordina il vettore ψ per valore crescente di priorità σ_p

2.4 Se esistono LSP di stessa priorità σ_p , ordinali in ψ per valore crescente di $RRCF(p)$

2.5 $i = 0$

2.6 $RRBW = 0$

3 - **WHILE** $(F - RRBW \geq f_{\psi_i})$ {

3.1 $\Psi^{\text{R}} = \Psi^{\text{R}} \cup \{\psi_i\}$

3.2 $\Psi^{\tau} = \Psi^{\tau} - \{\psi_i\}$

3.3 $RRBW = RRBW + f_{\psi_i}$

3.4 $i = i + 1$

}

}

ELSE (rif. IF2)

$\Psi^{\text{R}} = \Psi^{\tau}$

4 - Definisci un grafo $G_{RR}(V, E)$ in cui $b_{ij} \geq y_{ij}^{\text{max}}$

5 - Elimina da $G_{RR}(V, E)$ l'arco (\tilde{s}, \tilde{t})

6 - Elimina da $G_{RR}(V, E)$ gli archi $\dots \dots \dots$

7 - Elimina da $G_{RR}(V, E)$ i nodi $v \in V : v$ è isolato

8 - Risolvi l'MCFP su $G_{RR}(V, E)$, in cui si ha una commodity $\forall p \in \Psi^{\text{R}}$ associata ad una domanda di flusso f_p tra i nodi (\tilde{s}, \tilde{t}) , e ottieni l'insieme Ω

Il primo passo nel nuovo processo di rerouting consiste nell'individuare la banda totale disponibile per il reinstradamento: a questo scopo viene risolto il *Problema di Massimo Flusso Non Limitato*, ottenendo il valore F^{max} e il vettore y^{max} .

Dal confronto tra le quantità F^{max} e $\sum_{p \in \Psi^{\tau}} f_p$ discende come procedere ulteriormente: se F^{max} è minore, non si potranno reinstradare tutti gli LSP abbattuti per mancanza di banda e sarà quindi necessario selezionarne un sottoinsieme che viene poi reinstradato. Si esegue così il blocco **(2-IF)**, in cui la selezione degli LSP viene operata secondo una procedura pressochè identica a quella utilizzata nell'altro algoritmo di rerouting (a cui si rimanda per chiarimenti sul significato delle singole operazioni). Nel caso in cui $F^{\text{max}} \geq \sum_{p \in \Psi^{\tau}} f_p$, tutti gli LSP possono essere reinstradati e si passa al blocco **ELSE**, dove l'unica operazione consiste nel uguagliare

l'insieme degli LSP reinstradabili a quello degli LSP abbattuti. Le operazioni **3**, **4**, **5** e **6** sono finalizzate a definire un nuovo grafo $G_{RR}(V,E)$ su cui viene risolto il MCFP: come punto di partenza si utilizza la topologia del grafo completo della rete, ai cui archi, come parametro b_{ij} , si associa non la capacità di banda come nel grafo originale $G_{RR}(V,E)$, ma la banda y_{ij}^{\max} disponibile per il reinstradamento. Si ottiene così un nuovo grafo in cui alcuni archi possono avere banda nulla, o perchè sono completamente occupati o perchè non utilizzati nel processo di rerouting: tali archi vengono eliminati in quanto risultano inutili e anzi appesantiscono la risoluzione del MCFP. Si elimina anche l'arco su cui avviene la preemption e che non può quindi essere utilizzato. Sempre al fine di alleggerire la formulazione vengono cancellati anche i nodi che diventano isolati in seguito all'eliminazione degli archi.

L'operazione **6** corrisponde alla risoluzione di un MCFP in cui:

- la rete è rappresentata dal grafo $G_{RR}(V,E)$
- si definisce una commodity k_p per ogni LSP $p \in \Psi^{\text{ri}}$ abbattuto e da reinstradare
- ogni commodity k_p richiede di inviare un flusso pari alla banda f_p dell'LSP associato da reinstradare
- tutte le commodity hanno come origine e destinazione del flusso rispettivamente i nodi \tilde{s} , \tilde{t} estremi dell'arco su cui si effettua la preemption

Definire la formulazione del MCFP richiede di inserire in $G_{RR}(V,E)$ un nodo \tilde{s} ed un arco \tilde{s}, \tilde{t} , fittizi di capacità: $b_{\tilde{s}, \tilde{t}} = \sum_{p \in \Psi^{\text{ri}}} f_p$.

Secondo questo approccio, la forma del problema di massimizzazione del flusso multicommodity, nel caso che si denoti con R l'insieme di tutti i cammini ammissibili tra (\tilde{s}, \tilde{t}) risulta:

$$\min \sum_{r \in R} c_r \phi_r \quad (5.53)$$

$$s.t. \quad \sum_{r \in R_{k_p}} \phi_r = f_p \quad \forall k_p \quad (5.54)$$

$$\sum_{r \in R_{k_p}, e \in r} \phi_r \leq b_e \quad \forall e \in E \quad (5.55)$$

$$\phi_r \geq 0 \quad \forall r \in R \quad (5.56)$$

La variabile di decisione ϕ_r individua il flusso non negativo inviato su ogni cammino ammissibile $r \in R$. L'obiettivo è individuare per ogni commodity k_p il corrispondente insieme di cammini R_{k_p} che consente il reinstradamento e che risulta a minimo costo. A questo scopo la funzione obiettivo (5.53) è definita come la somma dei costi c_r associati ai cammini

R e il vincolo (5.54) impone che la somma dei flussi ϕ_r dei cammini definiti per una commodity (flussi che, si ricorda, corrispondono a banda utilizzata su ogni arco del cammino) sia pari alla banda dell'LSP $p \in \Psi^{\text{st}}$ da reinstradare. Il vincolo (5.55) esprime il limite di capacità di un arco $e \in E$: la somma delle occupazioni di banda ϕ_r di cammini che vi transitano non può eccedere la sua capacità b_e .

Per quanto riguarda la definizione del costo c_r , una possibile e semplice alternativa è data dall'adozione di una funzione che associa ad ogni cammino r il suo numero di hop: la selezione viene così orientata verso i cammini che occupano il numero minore di archi, definendo una soluzione ottima del rerouting che sfrutta il minor numero di risorse di banda della rete. In alternativa, si può porre

$$c_r = \sum_{e \in E: e \in r} c_e$$

definendo il costo di cammino come somma dei parametri di costo associati agli archi che lo compongono. Un'ulteriore alternativa, più complessa, è costituita dall'adottare una funzione di costo definita come combinazione pesata delle due precedenti:

$$c_r = \alpha \sum_{e \in E: e \in r} c_e + \beta H_r$$

Essa ha il pregio di tenere conto sia del costo intrinseco sia di quello legato alla lunghezza misurata in hop, ma d'altra parte fa scaturire il problema del dimensionamento dei pesi α e β , per i quali non esistono dei criteri univoci di definizione.

5.5 Seconda tecnica: approccio globale

In alternativa all'approccio incrementale, in base al quale si individuano per le nuove richieste i percorsi di instradamento migliori relativamente alla corrente condizione della rete, nel momento in cui si presenta una nuova richiesta, si può adottare un approccio di revisione globale di tutti gli LSP definiti: invece di utilizzare gli shortest path rispetto ad un criterio di lunghezza fissato, liberandoli eventualmente tramite preemption, si risolve un Multicommodity Flow Problem adattato in cui, oltre alle commodity relative a richieste in fase di soddisfacimento, viene aggiunta la commodity che individua la nuova richiesta. A fronte di maggiori oneri di gestione, derivanti dalla necessità di dover considerare e rivalutare i percorsi di instradamento dell'intera rete, si ottiene un allestimento ottimale degli LSP rispetto alle correnti richieste, senza dover affrontare gli oneri del processo di preemption e del conseguente processo di rerouting..

Il MCFP introdotto all'inizio del documento come strumento per la definizione di un allestimento iniziale ottimale degli LSP della rete può essere utilizzato come base per la definizione di un nuovo problema di ottimizzazione. Una prima differenza fondamentale tra i due problemi è che in questa nuova situazione può avvenire che tutte le richieste non possano essere contemporaneamente soddisfatte per mancanza di risorse e sia quindi necessario porre

termine a dei processi di comunicazione. Ciò si presenta analogamente anche nell'approccio alternativo, in cui la nuova richiesta può risultare non soddisfacibile oppure soddisfacibile ricorrendo alla preemption senza però che sia possibile reinstradare tutti gli LSP abbattuti.

A causa di questa differenza, si modifica il grafo $G(V, E)$ per giungere alla definizione di un nuovo grafo $G(V, \vec{E})$, in cui l'insieme dei nodi V non varia e all'insieme degli archi E si aggiunge, per ogni commodity $(s_k, t_k, B_k, D_k, c_k)$, un arco fittizio (s_k, t_k) , di parametri:

$$\begin{aligned} b_{s_k, t_k} &= B_k \\ d_{s_k, t_k} &= 0 \\ c_{s_k, t_k} &= M_\sigma \end{aligned} \quad \begin{aligned} M_\sigma &\gg \max\{c_p : p \in P\} \quad \text{per } \sigma = 0, 1, \dots, 7 \\ M_0 &\gg M_1 \gg \dots \gg M_7 \end{aligned}$$

si noti che l'introduzione di questo arco ha in particolare l'effetto di aggiungere all'insieme P un cammino fittizio $\pi_k = (s_k, (s_k, t_k), t_k)$ di costo $c_{\pi_k} = c_{s_k, t_k}$.

Scopo dell'introduzione di questi elementi è fare sì che il nuovo problema possieda almeno una soluzione ammissibile, anche nel caso che non vi siano sufficienti risorse di banda per il soddisfacimento di tutte le commodity: la capacità di banda dell'arco fittizio introdotto per ogni commodity $k \in C$ è posta pari alla corrispondente richiesta di banda B_k e, nel caso non esistano altre alternative, tutto il flusso di informazione può essere inviato sul LSP fittizio π_k costituito dal solo arco fittizio (s_k, t_k) . Esiste quindi sempre un cammino su cui inviare tutto il flusso informativo di una richiesta. Al fine di impedire l'uso dei π_k quando vi sono sufficienti risorse di banda sugli archi reali, il costo degli archi fittizi viene posto pari a valori M_σ molto più grandi di quelli di un qualunque altro LSP $p \in P$ realmente presente nella rete: tali costi risultano crescenti, con sensibili differenze di valore, al crescere della priorità della commodity. La funzione obiettivo orienta quindi la scelta verso cammini reali e seleziona cammini fittizi, enormemente costosi, solo se costretta dalla mancanza di risorse di banda. La forte differenza di costo tra cammini fittizi di commodity appartenenti a differenti classi di priorità è introdotta al fine di evitare che richieste a priorità maggiore non siano soddisfatte.

Come conseguenza dell'introduzione degli archi fittizi, l'insieme P dei LSP ammissibili per il MCFP può essere partizionato in:

- P^{R} l'insieme dei LSP realmente esistenti
- P^{F} l'insieme dei LSP fittizi

e per costruzione si ha: $P = P^{\text{R}} \cup P^{\text{F}}$ e $P^{\text{R}} \cap P^{\text{F}} = \emptyset$.

Se si introduce l'insieme $P_{\text{old}}^{\text{R}} \subseteq P$ dei LSP ammissibili utilizzati per soddisfare le richieste prima che si presentasse la nuova, la formulazione del nuovo MCFP, denotato d'ora in poi come $MCFP^{\text{F}}$, sul grafo $G(V, \vec{E})$ per tutte le commodity compresa la nuova, risulta:

$$\min \sum_{p \in P_{old}^{SR}} \mu c_p f_p + \sum_{p \in P_{old}^{SR}} c_p f_p + \sum_{\pi \in P^S} c_\pi f_\pi \quad (5.57)$$

$$s.t. \sum_{p \in P_k} f_p \leq B_k \quad \forall k \in C \quad (5.58)$$

$$\sum_{p \in P: e \in p} f_p \leq b_e \quad \forall e \in \vec{E} \quad (5.59)$$

$$f_p \geq 0 \quad \forall p \in P \quad (5.60)$$

L'unico elemento di differenziazione della formulazione è la funzione obiettivo (5.57), che risulta ora formata da tre addendi:

- il primo individua il costo totale di flussi inviati sui LSP $p \in P_{old}^{SR}$, utilizzati per soddisfare le richieste prima che si presentasse la nuova. Si noti che è stato introdotto un parametro $0 \leq \mu \leq 1$ che, scelto opportunamente piccolo, ha lo scopo di diminuire notevolmente il costo di LSP già instaurati ed attualmente impiegati, così da favorirne nuovamente la selezione e limitare le operazioni di abbattimento e rerouting;
- il secondo individua il costo totale di flussi inviati su LSP reali non utilizzati per soddisfare l'insieme di richieste presenti prima dell'arrivo della nuova. Si tratta della semplice moltiplicazione del flusso inviato sul LSP per il relativo costo di invio di un'unità di flusso;
- il terzo individua il contributo di costo dei LSP fittizi. Anche in questo caso si tratta della moltiplicazione del flusso inviato per il costo di invio di un'unità di flusso.

La soluzione ottima del problema individua il sottoinsieme di LSP $P^{new} \subseteq P$ e i flussi f_p inviati su di essi, che consentono di soddisfare a minimo costo il nuovo insieme di richieste. Le richieste corrispondenti a commodity che vengono soddisfatte inviando flusso sui LSP fittizi sono quelle che, nella soluzione ottima individuata, non risultano soddisfacibili e devono quindi essere rifiutate. Occorre però fare attenzione al fatto che la formulazione non è in grado di distinguere l'invio di flusso tra LSP fittizi di uguale costo: se π_1 e π_2 sono LSP fittizi di identico costo e si può risolvere il MCFP inviando, comunque distribuito, un flusso di entità F su di essi, il solutore potrebbe fornire come soluzione ottima una qualunque soluzione tale che $f_{\pi_1} = x$, $f_{\pi_2} = y$ e $f_{\pi_1} + f_{\pi_2} = x + y = F$. Può allora avvenire che un flusso inviabile su un unico LSP fittizio venga frazionato su due LSP fittizi, passando l'informazione inesatta che le richieste non soddisfacibili siano due quando in realtà è una sola.

Al fine di ovviare a questo problema, si propone una procedura di postprocessamento della soluzione, fondata su un'euristica che risolve una successione di MCFP sul grafo della rete con opportunamente ridefinite le capacità degli archi. Base per la definizione del nuovo grafo è il

sottoinsieme di commodity $C^{sat} \subseteq C$ che risultano completamente soddisfatte nella loro richiesta di banda, secondo la soluzione fornita dal $MCFP^{\mathfrak{S}}$ e per le quali, quindi, la somma dei flussi f_p inviati sui cammini reali $p \in P_k$ è tale che: $\sum_{p \in P_k} f_p \leq B_k$.

Questa condizione sussiste naturalmente se e solo se il flusso inviato sul cammino fittizio π_k della commodity risulta nullo. Il nuovo grafo introdotto, indicato con $\Gamma(V,E)$, differisce rispetto al grafo originale $G(V,E)$ solo per il parametro di capacità di banda di arco b_e , che viene posto pari a: $b_e = b_e - \sum_{p \in P_k: k \in C^{sat}} f_p$ ovvero pari alla banda che rimane disponibile dopo

l'instaurazione dei LSP reali $p \in P_k$ definiti per le commodity $k \in C^{sat}$ completamente soddisfatte. I cammini fittizi $P^{\mathfrak{S}}$ possono ora essere tralasciati in quanto non si è più interessati a forzare l'ammissibilità del problema, facendo in modo che esista sempre una soluzione, ma solo a verificare se esistono o meno ulteriori commodity che possono essere soddisfatte.

Dopo aver definito il nuovo grafo si procede a stabilire un ordinamento delle commodity non soddisfacibili $\overline{C^{sat}} = C - C^{sat}$, utilizzato come base per la definizione dei parametri di costo di ogni MCFP della successione da risolvere. L'ordinamento è definito secondo i seguenti criteri:

1. le commodity $\overline{C^{sat}}$ sono ordinate per classe di priorità σ_k crescente
2. a parità di σ_k , l'ordinamento segue l'ordine di presentazione delle richieste secondo una filosofia **FIFO**

Si ottiene così un ordinamento $(\tilde{k}_1, \tilde{k}_2, \dots, \tilde{k}_{|\overline{C^{sat}}|})$. Lo scopo di ogni MCFP della successione

è verificare se un sottoinsieme delle commodity che non risultano soddisfacibili secondo la soluzione di $MCFP^{\mathfrak{S}}$, possono essere effettivamente soddisfatte con le risorse rimaste disponibili. Si ricorda infatti che l'insoddisfacibilità può in realtà essere solo apparente, come mostrato con l'esempio introdotto precedentemente.

Si introduce la notazione $MCFP^m$ per denotare l'm-esimo MCFP della successione. In ogni $MCFP^m$ si tenta di soddisfare la richiesta di banda di un sottoinsieme $C^{\nabla \overline{sat}} \subseteq \overline{C^{sat}}$ delle commodity nominalmente non soddisfacibili: se il problema ammette soluzione (esistono quindi LSP che permettono di soddisfare la richiesta di banda di ogni commodity $k \in \overline{C^{sat}}$) si è scoperto che in realtà sono soddisfacibili e si passa a risolvere il $MCFP^{m+1}$, in cui all'insieme $C^{\nabla \overline{sat}}$ si unisce la prima commodity non ancora considerata dell'ordinamento $(\tilde{k}_1, \tilde{k}_2, \dots, \tilde{k}_{|\overline{C^{sat}}|})$. Nel caso che il problema non ammetta soluzione, si deduce invece che il particolare sottoinsieme non è soddisfacibile con le risorse di banda residue.

Di seguito si riporta formalmente la procedura di definizione dei sottoinsiemi candidati:

1 Definizione dell'ordinamento $(\tilde{k}_1, \tilde{k}_2, \dots, \tilde{k}_{|C^{\overline{sat}}|})$ delle commodity $k \in C^{\overline{sat}}$

2 Inizializzazione: $C^{\nabla \overline{sat}} = 0$

3 $counter = 1$

4 WHILE ($counter \neq |C^{\nabla \overline{sat}}| + 1$) {

4.1 $C^{\nabla \overline{sat}} = C^{\nabla \overline{sat}} \cup \{\tilde{k}_{counter}\}$

4.2 Risoluzione del $MCFP^{counter}$ **su** $\Gamma(V, E)$ **per** $C^{\nabla \overline{sat}}$

4.3 IF $MCFP^{counter}$ **ammette soluzione ottima** {

4.3.1 $counter = counter + 1$

4.3.2 CONTINUE

4.4 $C^{\nabla \overline{sat}} = C^{\nabla \overline{sat}} - \{\tilde{k}_{counter}\}$

4.5 $counter = counter + 1$

}

}

RETURN $C^{\nabla \overline{sat}}$

RETURN $f_p^* \forall p \in P_k^*, k \in$

L'insieme $C^{\overline{sat}}$, disponibile al termine dell'esecuzione, è quello definitivo individuato come soddisfacibile: in unione all'insieme delle commodity che possono essere soddisfatte secondo la soluzione di $MCFP^{\overline{sat}}$, esso definisce l'insieme delle commodity complessivamente soddisfacibili secondo l'euristica. Il $MCFP^{\overline{sat}}$ fornisce gli LSP e i corrispondenti flussi ottimi definiti per le richieste riconosciute come soddisfacibili nella fase precedente il postprocessamento. La soluzione ottima di $MCFP^{|C^{\nabla \overline{sat}}|+1}$ rende invece disponibili LSP e flussi delle commodity che sono riconosciute come soddisfacibili al termine dell'euristica. Ad ogni iterazione del ciclo **WHILE**, si verifica se il corrente sottoinsieme di commodity $C^{\overline{sat}}$ risulta soddisfacibile: nel caso che la soddisfacibilità non sia verificata, l'ultima commodity inserita nell'insieme viene rimossa e si procede a risolvere il problema con inserita la successiva commodity dell'ordinamento. Il parametro $counter$ svolge il ruolo di contatore e viene utilizzato per scandire l'ordinamento \tilde{k} delle commodity. Il comando **CONTINUE** del blocco **IF** ha il compito di far proseguire l'esecuzione del ciclo, dopo l'aggiornamento del contatore, nel caso che il sottoinsieme corrente risulti soddisfacibile.

La formulazione del MCFP risolto ad ogni iterazione della procedura euristica è la seguente:

$$\min \sum_{p \in P} c_p f_p \quad (5.61)$$

$$s.t. \quad \sum_{p \in P_k} f_p = B_k \quad \forall k \in C^{\nabla sat} \quad (5.62)$$

$$\sum_{p \in P: e \in p} f_p \leq b_e \quad \forall e \in E \quad (5.63)$$

$$f_p \geq 0 \quad \forall p \in P \quad (5.64)$$

Si tratta della formulazione più volte trattata di un MCFP, risolto questa volta sul grafo $\Gamma(V,E)$ per il sottoinsieme di commodity $C^{\nabla sat}$, considerando tutti i cammini ammissibili P compresi quelli fittizi di ogni commodity. L'esistenza di una soluzione ottima, costituita dal flusso di minimo costo f_p^* inviato su ogni cammino $p \in P$, permette di dedurre che l'insieme corrente di commodity è soddisfacibile.

In alternativa a questa procedura, dove ogni $MCFP^m$ della successione viene risolto per un sottoinsieme di commodity $C^{\nabla sat}$, se ne può considerare una in cui l'insieme di commodity è costituito dall'unione del corrente $C^{\nabla sat}$ con l'insieme di commodity C^{sat} soddisfacibili secondo la soluzione del $MCFP^S$. In questo modo, mediante una procedura computazionalmente più onerosa, dato che si risolve un MCFP per un insieme di commodity più ampio, si può giungere a soluzioni di migliore qualità che utilizzano più efficientemente le risorse di banda disponibili, rimaneggiando eventualmente la configurazione di rete fornita da $MCFP^S$. Poiché si consente di revisionare i percorsi di instradamento delle commodity C^{sat} , il grafo su cui si risolve la successione degli $MCFP^m$ è quello originale $G(V,E)$ e non più quello delle capacità residue $\Gamma(V,E)$. La formulazione del MCFP risolto ad ogni iterazione diventa in questo caso:

$$\min \sum_{p \in P} c_p f_p \quad (5.65)$$

$$s.t. \quad \sum_{p \in P_k} f_p = B_k \quad \forall k \in C^{\nabla sat} \cup C^{sat} \quad (5.66)$$

$$\sum_{p \in P: e \in p} f_p \leq b_e \quad \forall e \in E \quad (5.67)$$

$$f_p \geq 0 \quad \forall p \in P \quad (5.68)$$

e l'ultimo problema della successione fornisce direttamente individua l'insieme di tutte le commodity che è possibile soddisfare insieme ai relativi LSP e flussi inviati su di essi.

5.6 Tecnica Ibrida

Le due tecniche proposte, potrebbero essere applicate entrambe in una stessa rete. Infatti, la prima risulta essere più efficiente dal punto di vista dei costi relativi, per esempio, al reinstradamento degli LSP, che dovrebbero essere minimizzati al massimo, mentre la seconda, eseguendo una riottimizzazione globale della rete, è più performante relativamente ai *service disruption* e load balancing. Si suggerisce di utilizzare la prima tecnica ogni volta che arriva una nuova richiesta, salvo poi eseguire una riottimizzazione globale, quindi usare la seconda tecnica, periodicamente nel tempo o all'occorrenza del superamento di una certa soglia relativa al numero di abbattimenti degli LSP attivi sulla rete. In questo modo si dovrebbe ottenere un giusto compromesso tra abbattimenti e costi.

Parte II- Tecniche di Ottimizzazione: Video streaming

Capitolo 6

Problematiche Servizio Video

Streaming su Wireless LAN

6.1 Introduzione

I recenti sviluppi nel campo delle tecnologie di accesso wireless e della codifica video come per esempio MPEG-4 e H-264, il servizio di video streaming ha guadagnato un ruolo chiave nel nuovo scenario delle telecomunicazioni mobili. Dal momento che la domanda di contenuti multimediali cresce di giorno in giorno sono state sviluppate e promosse una serie di applicazioni e i servizi video on-demand e real-time sono diventati fruibili anche dall'ultima generazione dei dispositivi mobili. Nonostante questi progressi esistono ancora molti problemi da risolvere relativamente al video streaming su canali wireless. Il motivo principale di questi problemi è insito nella natura stessa dell'ambiente wireless in cui le comunicazioni subiscono gli effetti di un'alta variabilità delle condizioni del canale, quali le fluttuazioni di alcuni ordini di grandezza in meno di un secondo del bit error rate (BER). Inoltre, a causa della natura contention-based delle comuni tecniche di accesso wireless, l'interferenza radio e la collisione dei pacchetti decresce rapidamente con l'aumento del throughput di canale. Ne consegue che per garantire l'affidabilità e la qualità della comunicazione video, è necessaria un'attenta analisi delle caratteristiche del video da trasmettere e di quelle relative al canale wireless per poi applicare un sistema di controllo del rate adattivo che modifica dinamicamente i parametri del sistema seguendo le fluttuazioni del canale.

Il lavoro presentato in questa tesi è basato su un approccio *end-system centric* che presenta il vantaggio di richiedere cambiamenti minimi sulla core network. La sfida principale in questo contesto riguarda come ottimizzare la rete, ovvero, in che modo progettare un algoritmo di controllo del rate che realizzi la massimizzazione della qualità del video e dell'utilizzazione del canale.

Per prima cosa allo scopo di ottenere una trasmissione affidabile è stato adottato schema ibrido tra perward error correction (FEC) e automatic repeat request (ARQ). Successivamente

è stato proposto un metodo per il controllo del rate mirato alla fluidità del playback del video che si basa sul controllo dell'occupazione del playback buffer in modo che il rate delle occorrenze di starvation nel buffer non superi una soglia desiderata. Mentre viene garantito questo vincolo l'ancoder massimizza il bit rate sorgente. Il controllo del rate è realizzato adattivamente sulla base di un approccio a finestre temporali che offre il vantaggio di ridurre le fluttuazioni repentine del bit rate di sorgente assicurando dei cambiamenti più dolci della qualità del video ed evitare il così detto "saw" effect che si osserva tipicamente nel controllo del rate di tipo frame-based. Il sistema proposto lavora lato server e richiede al client solo un feedback sull'occupazione del buffer di playback. Al fine di ottimizzare il sistema si assume che questa informazione venga passata al codificatore attraverso il canale di feedback ARQ disponibile dalla comunicazione di tipo two-way. In questo modo, si minimizza l'overhead della trasmissione e il carico computazionale, principalmente lato server, risulta molto basso così come è richiesto per le applicazioni di tipo real-time in ambiente wireless.

6.2 Stato dell'arte

Il problema della trasmissione real-time di video su canali wireless è stata largamente studiata negli anni passati. Una comune classe di soluzione è relative all'ottimizzazione dell'affidabilità del livello link (link-layer) come proposto da Kallel and Deng in [50] e [51] rispettivamente, gli schemi ARQ sono spesso usati per realizzare l'affidabilità nelle comunicazioni. Generalmente nelle configurazioni ARQ, il throughput di canale è regolato sulla base delle condizioni di canale, quindi: quando le condizioni sono ottimali allora il throughput cresce e per la trasmissione viene usata tutta la banda; quando, invece, le condizioni del canale peggiorano sono necessarie molte ritrasmissioni e, di conseguenza, il throughput decresce. In molte soluzioni avanzate, gli schema ARQ viene accoppiato con un altro approccio il quale regola il rate sulla base dei parametri relativi alla qualità del video end-to-end.

Il lavoro di Zhang *et al.* nel [52] fa una distinzione tra due classi di approccio il network-centric l'end-system centric. Nel primo caso si fornisce la Quality of Service (QoS) per configurare i nodi della rete in modo tale che i requisiti di rate dei dati, delay bound, e packet siano soddisfatti [52],[53]. lo scopo chiave di questo tipo di approccio è il progettamento di un'architettura cross-layer architecture, che permetta ai differenti livelli MAC/physical di comunicare tra loro, fornendo una mappatura efficace della QoS sul cross-layer e l'adattamento della QoS. Nonostante queste soluzioni realizzino l'ottimizzazione soprattutto relativamente alle performance del sistema, esse presentano un grande svantaggio, ovvero richiedono degli upgrade computazionalmente significativi sui componenti della core network. D'altro canto, gli approcci end-system centric consistono in un insieme di tecniche di controllo, come il controllo della congestione, il controllo dell'error, il controllo della potenza, che lavorano a livello applicazione senza richiedere nessun supporto di QoS a livello di rete. In questo modo, possono essere realizzate delle performance elevate con dei cambiamenti minimi nella core network e con un overhead inferiore rispetto a quello del caso network-centric. Per questa ragione, le ultime ricerche nel campo dell'ottimizzazione del video streaming sono focalizzate sulla seconda classe di soluzioni.

L'obiettivo di un approccio *end-system centric* è quello di rendere le applicazioni video adattive alle variazioni delle condizioni della wireless network. Alcune soluzioni consistono nell'estendere all'ambiente wireless il popolare TCP Friendly Rate Control (TFRC) [54], che è stato sviluppato per applicazioni su wired network. TFRC calcola il rate come funzione del rate di perdita dei pacchetti, del round trip time e della dimensione dei pacchetti e dipende dall'assunzione che la perdita di pacchetti sia segno di una congestione in atto. Quest'assunzione non è valida nel caso della rete wireless, dove la perdita di pacchetti è principalmente dovuta agli errori del canale fisico che non richiede nessun cambiamento nel rate di trasmissione. In [55] è proposto uno schema TCP Friendly wireless, in cui il TFRC è accoppiato ad un sistema di controllo della congestione che lavora lato client. Un altro schema, chiamato MULTFRC, è stato proposto da Chen *et al.* in [56]. Tale schema dipende dalle connessioni TFRC multiple monitorando il round trip time per decidere se cambiare il numero di connessioni. Lo svantaggio principale della procedura è che richiede molte risorse per gestire connessioni multiple ed eredita rilevanti fluttuazioni del rate a causa dei frequenti cambiamenti nel numero di connessioni simultanee.

Altre soluzioni realizzano il controllo del rate presso l'end-system sorgente, adattando la codifica video e i parametri del buffer dell'encoder alle variazioni della trasmissione di canale. Nell'articolo [57] è proposto un controllo del rate di tipo region-based, nel quale la codifica è basata su un metodo di segmentazione a blocchi che estrae le regioni di interesse dal video sorgente e riduce la quantità di informazione da trasmettere. Gli autori del [58] analizzano i problemi della qualità del video e propongono uno schema di controllo del rate che evita la degradazione in termini di peak signal to noise ratio (PSNR) il quale potrebbe essere causa di una riduzione del bit rate. Questo schema perisce una ritrasmissione condizionata e una strategia *low-delay interleaving*, in cui il buffer dell'encoder è usato come parte della memoria *interleaving*. Nell'articolo [59], gli autori introducono un meccanismo di controllo del rate basato su un modello stocastico a priori della sorgente e del canale prioritario. Mentre nel [60] è proposto uno schema di controllo del rate per l'allocazione dei bit dell'H.264, nel quale per calcolare la stima del throughput di canale sia a livello frame che di unità base si tiene in conto delle condizioni del canale e dello stato del buffer dell'encoder. Questi studi spesso non tengono conto delle dinamiche del buffer di playback che, invece, risulta particolarmente importante al fine di mantenere la continuità della riproduzione del video. Inoltre, l'intensità computazionale dei framework proposti li rende poco attrattivi per l'impiego in applicazioni real-time.

Capitolo 7

Metodo di ottimizzazione proposto

7.1 Framework e obiettivi

Nel lavoro presentato l'obiettivo prefisso è la realizzazione del controllo del rate per le applicazioni video streaming considerando un tipico scenario wireless dove le risorse di canale sono condivise tra un numero variabile di stazioni che usano un meccanismo di controllo degli accessi di tipo contention-based. Questa caratteristica, insieme all'alta variabilità del carico di traffico e le fluttuazioni del bit error rate portano ad un throughput di canale di tipo bursty come percepito da ogni end-user che si contende il canale con le altre stazioni [61]. Allo scopo di ridurre il numero dei pacchetti corrotti e ottenere una trasmissione affidabile, è stato adottato un ibrido tra lo schema di perward error correction (FEC) e quello di automatic repeat request (ARQ). La codifica del FEC aggiunge ridondanza ai pacchetti in modo da poter correggere gli eventuali errori dovuti al canale, mentre lo schema ARQ ritrasmette quei pacchetti che sono stati ricevuti con errore. In particolare, si assume che venga applicata una politica di tipo stop-and-wait ARQ. Questa assunzione è giustificata quanto il ritardo round trip di propagazione è inferiore al tempo di trasmissione del pacchetto, come succede nei tipi ambienti wireless LAN.

7.1.1 Architettura e assunzioni base

La Figura 11 rappresenta l'architettura del sistema di video straming considerato. Esso consiste in una stazione client mobile che comunica mediante un link wireless con il video server; il server potrebbe essere anch'esso una stazione mobile, oppure un sistema fisso collegato attraverso una wired network all'Access Point (AP) del canale wireless. In quest'ultimo caso si assume che la wired network abbia un canale ad alto throughput. Per entrambi i sistemi client e server sono mostrati tutti i sottocomponenti.

Il tipico scenario relativo al servio di video streaming include:

- la sorgente video;
- il display;
- il canale transceiver;

- il codificatore (encoder);
- il decodificatore (decoder);
- i loro rispettivi buffer;

inoltre si aggiunge a questi elementi un modulo di controllo del rate di sorgente che lavora lato server. Questo è il componente chiave dell'algoritmo proposto, che si occupa di impostare il rate di sorgente al valore ottimo sulla base del periodico feedback relativo al throughput di canale e l'occupazione del buffer di playback. L'intera procedura verrà descritta nello specifico di seguito.

Si assume che il codificatore di base sia in grado di regolare i parametri del codificatore in modo da raggiungere il rate richiesto, secondo il valore calcolato dall'algoritmo del controllo del rate. Nella figura, R_f è il rate di codifica e di playback espresso in frames al secondo, e che assumiamo resti fisso durante la sessione di streaming, mentre R_{in} è il rate dei frame in ingresso al client, che invece varia in accordo alle condizioni del canale.

La sequenza video può essere sia generata in real-time che recuperate da un archivio video. Quando un video è stato codificato, risulta segmentato in uno o più pacchetti che verranno poi consegnati al MAC layer e trasmessi sul link wireless. Ciascun pacchetto contiene un numero di bit totali di $T = L+H$, dove L è il numero dei bit di informazione e H è il numero di bit a correzione d'errore. Si assume che il valore di L rimanga invariato per tutti i pacchetti durante la sessione di video streaming, mentre H è fisso per tutti i pacchetti appartenenti allo stesso frame video.

Per un dato schema di codifica FEC come per esempio il codice BCH e una data coppia (L,T) , può essere facilmente determinato il numero massimo di bit error che possono essere corretti in un pacchetto ($E_{max} = E_{max}(L,T)$). In aggiunta, la politica ARQ permette al ricevitore di spedire attraverso un canale di feedback una risposta positiva (ACK) o negativa (NAK) a seconda il pacchetto ricevuto sia corretto o no. Quando viene ricevuto un messaggio di NAK il trasmettitore provvederà alla ritrasmissione del pacchetto. Il tempo richiesto per spedire un singolo frame attraverso il canale e il numero di ritrasmissioni richieste per trasferire con successo al ricevitore un pacchetto varia sulla base del tempo dovuto al contendimento della linea e al BER variabile.

Nell'architettura considerate, le ritrasmissioni ARQ sono gestate dall'ARQ controller, che raccoglie i messaggi ACK/NAK dal ricevitore e inverte l'encoder buffer quando è necessaria una ritrasmissione. Manipolando gli acknowledge ricevuti, il controller ARQ è in grado anche di calcolare il tempo richiesto per trasmettere con successo un pacchetto. In particolare, inizia un primo tentativo di trasmissione dei pacchetti, il buffer del codificatore spedisce un messaggio al controller ARQ; aspetta poi l'ACK corrispondente e calcola l'intervallo di tempo totale richiesto per consegnare con successo il pacchetto. In questo modo, alla ricezione di ogni ACK, viene spedito un feedback al modulo di controllo che indica l'intervallo di tempo totale che è stato speso per spedire l'ultimo pacchetto. È stato, inoltre, assunto che il messaggio di ARQ trasmette un feedback dal decoder pervenendo l'occupazione del playback buffer. Essendo i pacchetti di controllo relativamente piccoli, circa poche decine di byte, possono essere adeguatamente protetti con il solo FEC, assicurando che il canale di feedback sia privo d'errore. Di conseguenza, il controller ARQ è in grado di calcolare l'intervallo di tempo necessario perché ogni pacchetto venga trasmesso con successo, quest'informazione, insieme all'occupazione del playback buffer viene spedita al modulo di controllo del rate di sorgente.

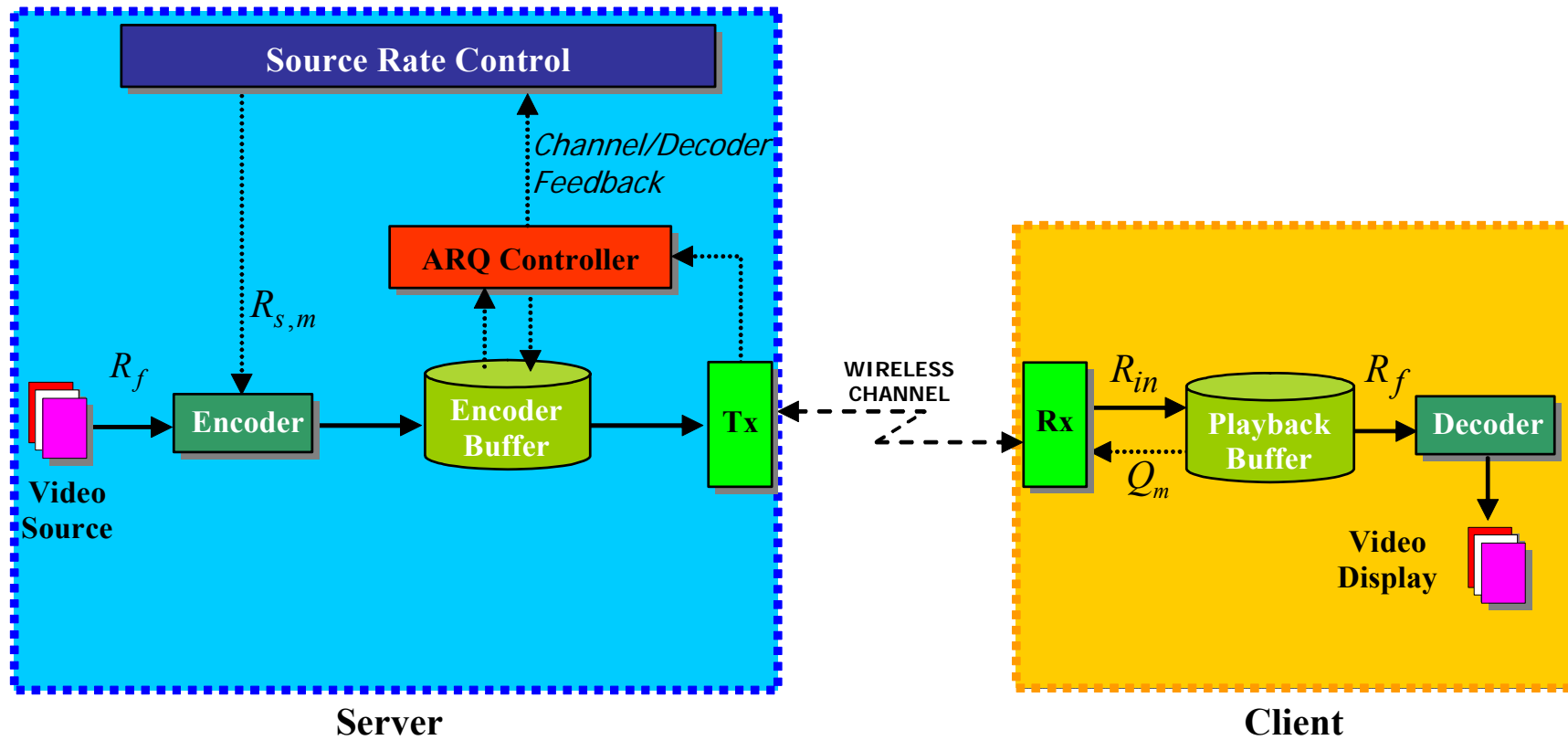


Figura 11 Architettura di riferimento per il sistema di video streaming.

7.1.2 Scopo

L'obiettivo principale che si vuole perseguire è assicurare la continuità della riproduzione del video limitando il numero di eventi di starvation nel buffer di playback. Per arrivare a tale scopo è stato adottato un approccio basato su finestre temporali. Il rate di codifica di sorgente è regolato considerando il suo effetto sull'intera finestra di dimensioni fissate in modo tale da permettere al playback buffer di "accumulare" pacchetti durante i burst dei periodi di alto throughput e "scaricarsi", senza svuotarsi completamente (starvation), durante i periodi di basso throughput. In questo modo, la produzione di burst da parte del throughput di canale è compensata senza introdurre il noto "saw" effect nel rate di sorgente che tipicamente si osserva quando si ricorre ad un controllo del rate di tipo frame-based. Di conseguenza, l'asse dei tempi è diviso in finestre di dimensione fissa pari a T , ognuna delle finestre sarà indata con m ($m=0,1,2,\dots$). All'inizio di ogni finestra temporale m , sulla base dell'occupazione del buffer e delle inpermazioni relative al comportamento del canale, il modulo di controllo del rate calcola la probabilità che si verifichi un evento di starvation nel playback buffer durante la finestra successiva. A questo punto imposterà il rate di codifica di sorgente al valore $R_{s,m}$, che applicato durante la finestra corrente fa in modo di ottenere una probabilità di starvation inferiore ad una soglia desiderata.

7.2 Controllo della Probabilità di Starvation di Playback

Si consideri l'occupazione del buffer durante m -esima finestra $Q_m(\tau)$ (per $0 < \tau \leq T$), che varia in accordo con la seguente equazione:

$$Q_m(\tau) = Q_m + \frac{R_f}{R_{s,m}} Y_m(\tau) - R_f \tau, \quad 0 < \tau \leq T, \quad (7.1)$$

dove $Y_m(\tau) = X(\tau + mT) - X(mT)$, con $X(t)$ che rappresenta l'arrival process che conta l'arrivo di traffico cumulativo che arriva al playback buffer nell'intervallo $(0,t)$.

L'equazione (7.1) dice che l'occupazione del buffer è data dal numero di frame presenti nel buffer di playback all'inizio dell' m -esima finestra (Q_m) più il numero di frame spediti nell'intervallo τ , meno il numero di frame che vengono riprodotti nello stesso intervallo, in altri termini la differenza tra numero di frame che entrano e numero di frame che escono nell'intervallo considerato. Si noti che $R_{s,m}$ rappresenta il bit rate di codifica di sorgente per l' m -esima finestra, il cui valore è imposto all'encoder dal modulo di controllo. Come si è appena affermato, l'obiettivo del modulo di controllo è quello di calcolare il valore di $R_{s,m}$ che assicuri una probabilità di starvation più bassa della soglia desiderata e quando questo vincolo risulta soddisfatto viene selezionato il massimo valore del rate. quindi si selezionerà il valore massimo del rate che, comunque, soddisfi tale vincolo.

Se $\Phi_m(\tau)$ rappresenta la probabilità di starvation durante l' m -esima finestra si richiede che:

$$\Phi_m(\tau) = \Pr \left[Q_m + \frac{R_f}{R_{s,m}} Y_m(\tau) \leq R_f \tau \right] \leq \varepsilon, \quad 0 < \tau \leq T, \quad (7.2)$$

dove $0 < \varepsilon \ll 1$ è un valore predefinito. Questa condizione è imposta singolarmente per ogni intervallo di tempo di streaming pari a T . per soddisfare il vincolo espresso nella (7.2), il modulo di controllo necessita delle informazioni relative all'occupazione iniziale del buffer di playback e al traffico di rete durante la prossima finestra. La prima informazione è fornita in maniera diretta dal loop di feedback stabilito attraverso I messaggi ARQ, come è stato descritto nella sezione precedente. Si noti che per la prima finestra ($m=0$) To fulfill the constraint in (7.2), the control module needs the informazione on the initial playback buffer occupancy and the network traffic during next window. The first informazione is directly provided by the feedback loop established through ARQ messages, as described in the previous section. Note that per the first window ($m=0$), Q_0 è uguale a N frame che sono precaricati prima che il playback cominci. La seconda informazione relativa al processo $Y_m(\tau)$ è, invece, sconosciuta, ma può essere predetta sulla base dei ritardi di trasmissione osservati nei pacchetti spediti precedentemente, questi ritardi sono forniti attraverso il canale di ritorno. Avendo raccolto tutte le informazioni necessarie si può procedere come verrà qui di seguito riportato.

Si associ a $P_\tau(n)$ la probabilità che nell'intervallo τ esattamente n frames vengano trasmessi con successo. Per calcolare il valore di $P_\tau(n)$ è stata considerata la trasmissione con successo di $n+1$ frame, ognuno dei quali utilizza un intervallo di tempo pari a t_i ($i=1, \dots, n+1$), in modo tale che si verificano le seguenti relazioni $\sum_{i=1}^n t_i \leq \tau$ e $t_{n+1} \geq \tau - \sum_{i=1}^n t_i$. t_i allora rappresenta l'intervallo di tempo richiesto per trasmettere con successo il pacchetto i : è il tempo tra il primo tentativo di trasmettere il pacchetto i l'istante di ricezione dell'ACK; si noti che lo stesso frame potrebbe essere trasmesso più di una volta nell'intervallo t_i , prima che di avere una ricezione con successo al ricevitore. Il numero di ritrasmissioni dipende dallo schema FEC applicato e dallo stato del canale. Ora supponiamo che t_i siano i.i.d.; si omette pertanto il pedice i e si definisce $f_d(t)$ che corrisponde alla funzione di densità di probabilità (pdf) relativa a t . Si considera, inoltre, $f_d(t)$ nota e si rimanda ad una sezione apposita la discussione in merito a tale funzione. Fatte le dbite premesse la $P_\tau(n)$ potrà essere così calcolata:

$$P_\tau(n) = \int_0^\tau \int_0^{\tau-t_1} \dots \int_0^{\tau-t_1 \dots t_n} \int_{\tau-t_1 \dots t_n}^\infty \prod_{i=1}^{n+1} f_d(t_i) dt_1 \dots dt_{n+1}. \quad (7.3)$$

Da questa probabilità si può calcolare la pdf $f_{Y_{\tau,m}}(\cdot)$ per il processo $Y_m(\tau)$ con la seguente relazione:

$$f_{Y_{\tau,m}}(y) = \sum_{n=0}^{\infty} P_{\tau}(n) \pi_{L,n}(y), \quad (7.4)$$

Dove $\pi_{L,n}(\cdot)$ rappresenta una funzione rettangolare uguale a uno nel range $(n-1)L \div nL$ e uguale a zero altrove (si ricorda che L è la dimensione di ogni frame). Si noti che $f_{Y_{\tau,m}}(\cdot)$ è una funzione continua a pezzi così come mostrato nella Figura 12.

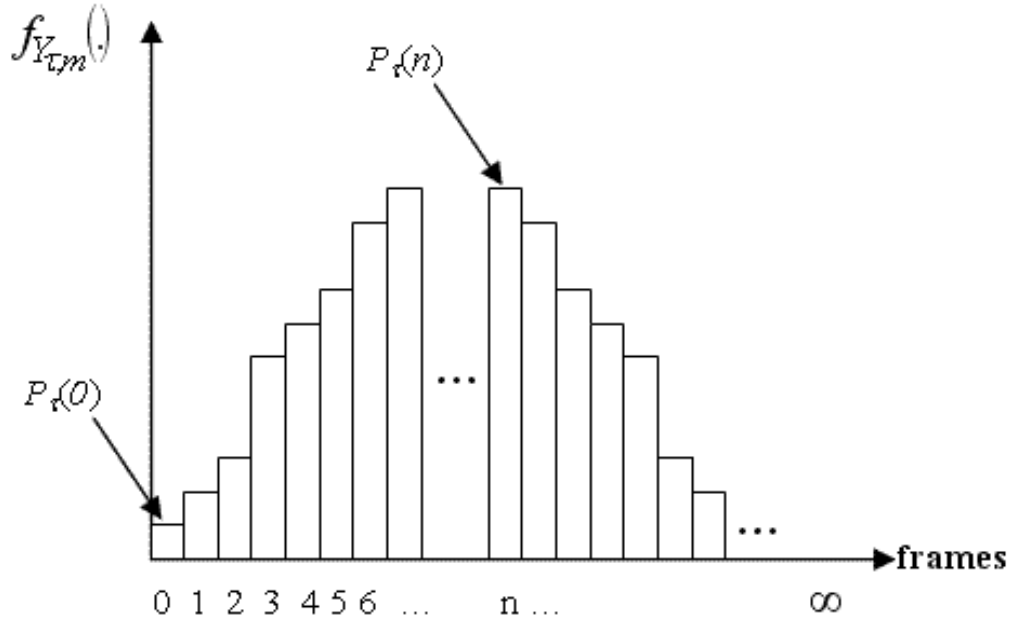


Figura 12 Pdf per il processo $Y_m(\tau)$.

Ora, si supponga che, per la finestra corrente m , sia stato fissato il valore di $R_{s,m}$. Dalla (7.1) si evince che le occorrenze di starvation si verificheranno ogni qualvolta si ha:

$$Y_m(\tau) \leq Y_{\tau}^* = (R_f \tau - Q_m) \frac{R_{s,m}}{R_f} \quad (7.5)$$

E la probabilità di starvation sarà uguale a :

$$\Phi_m(\tau) = \int_0^{Y_{\tau}^*} f_{Y_{\tau,m}}(y) dy. \quad (7.6)$$

Nel framework proposto in modo inverso, ovvero, si impone un valore della probabilità di starvation e da essa si determina il valore appropriato di $R_{s,m}$. Ne consegue che la prima ad essere calcolata è la Y_{τ}^* che si ottiene dalla relazione (7.6) facendo in modo che $\Phi_m(\tau)$ sia

uguale a ε per $0 < \tau \leq T$. Successivamente, ottenuta Y_τ^* siamo in grado di calcolare $R_{s,m}$ dalla relazione (7.5).

7.3 Processamento dinamico dei Delay Statistical Trend

L'algoritmo proposto si basa sulla disponibilità della pdf $f_d(t)$ e l'efficacia dello schema di controllo del rate necessita di un semplice metodo statistico per fare la predizione della pdf per le trasmissioni future sulla base delle misure osservate. Inoltre, le caratteristiche della rete variano nel tempo e, conseguentemente, lo fa anche il trend statistico dei ritardi; è quindi necessario un aggiornamento continuo di queste informazioni per tenere conto delle evoluzioni relative alla rete. Per valutare i trend statistici dei ritardi della rete in relazione all'analisi dello storico e delle informazioni correnti possono essere impiegati diversi approcci.

7.3.1 Tecniche di Aging

Il metodo *full aggregation* accumula i dati in una curva di distribuzione delle probabilità, associando a ciascun campione, sia esso vecchio o recente, lo stesso peso; tale approccio è molto semplice ma non consente al sistema di reagire prontamente alle variazioni del traffico di rete. Differentemente, il metodo *flush and refresh* costruisce la curva di distribuzione delle probabilità sulla base degli ultimi M pacchetti osservati tagliando fuori lo storico delle informazioni, e generando un overhead molto alto.

Una tecnica intermedia che è stata adottata anche nel presente lavoro è il metodo così detto *store and track* ([62]). In questo caso i vecchi dati non vengono scartati interamente, ma viene gradualmente ridotto il loro effetto nell'istogramma che approssima la distribuzione statistica. Ogni valore dell'istogramma è determinato contando il numero delle occorrenze di pacchetti spediti entro un certo intervallo di ritardo (bin). Si noti che tanto minore è il bin, ovvero l'unità base dell'intervallo di tempo, tanto maggiore è l'accuratezza della predizione di $f_d(t)$. D'altro canto però, minore è il bin tanto più aumenta la complessità computazionale dell'algoritmo (7.3) necessario a calcolarlo numericamente. Per ridurre il peso dei campioni più vecchi, ogni bin dell'istogramma viene periodicamente scalato, diminuendo il suo apporto, attraverso un fattore di aging F . Questa operazione nota con il nome di *aging* è applicata con una frequenza f .

Sia $H_j(x)$ la funzione dell'istogramma dopo aging j -esimo e nd sia il network delay di un nuovo pacchetto, si ha:

$$H_j(x) = \begin{cases} F \times H_{j-1}(x), & 0 \leq x < \infty, x \neq nd \\ F \times H_{j-1}(x) + 1, & x = nd \end{cases} \quad (7.7)$$

Sono stati considerati tre algoritmi di aging. Nel primo di essi il valore del fattore F è definito come:

$$F = c \quad (7.8)$$

Dove c è il coefficiente di aging scelto dall'utente o da un'applicazione e il suo range di variazione è tra 0 e 1. Nel secondo algoritmo, invece il valore di F è ottenuto come segue:

$$F = \frac{c}{(1-c) \times \int_0^{\infty} H_{j-1}(x) dx} \quad (7.9)$$

In fine, la terza funzione di aging è la seguente:

$$F = \frac{c \times f}{(1-c) \times \int_0^{\infty} H_{j-1}(x) dx} \quad (7.10)$$

dove f è la frequenza di aging, che definisce quanto spesso dovrebbe essere invocato il fattore di aging. Eseguire azioni di aging più frequentemente, corrispondente ad un più basso valore di f , significa che i dati vengono aggiornati più spesso e i risultati generati saranno più accurati. Si noti che l'istogramma $H_j(x)$ viene aggiornato continuamente durante lo streaming, ma i suoi valori sono usati per determinare la $f_d(t)$ solamente all'inizio di ogni finestra di canale.

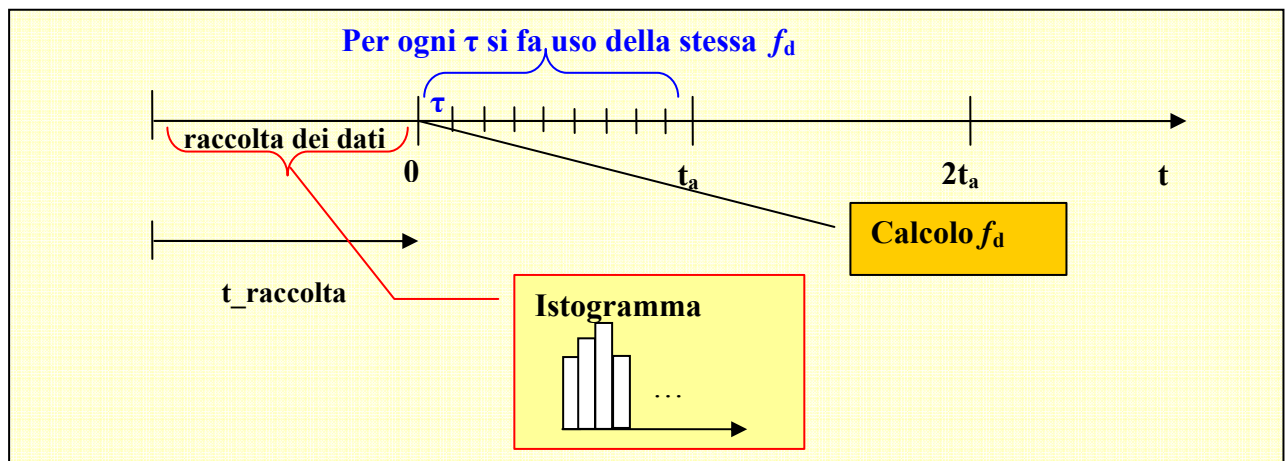


Figura 13 Calcolo della f_d

7.4 Workflow dell'algoritmo proposto

La Figura 14 mostra il diagramma di flusso che descrive le operazioni realizzate dal modulo di controllo del rate. Durante streaming video, vengono collezionati i valori dei ritardi di trasmissione t_i per ogni pacchetto spedito. Questa informazione è usata per costruire la pdf $f_d(t)$ secondo uno dei tre metodi precedentemente descritti. Allo scopo di seguire le variazioni del traffico di rete, la funzione sarà aggiornata con una frequenza pari a $f = 1/t_a$ il cui valore dipende dall'accuratezza e dalla complessità del sistema che si desiderano raggiungere. L'azione di aging sarà, quindi, messa in atto negli istanti $t = k \cdot t_a$ con $k = 1, 2, \dots, n$.

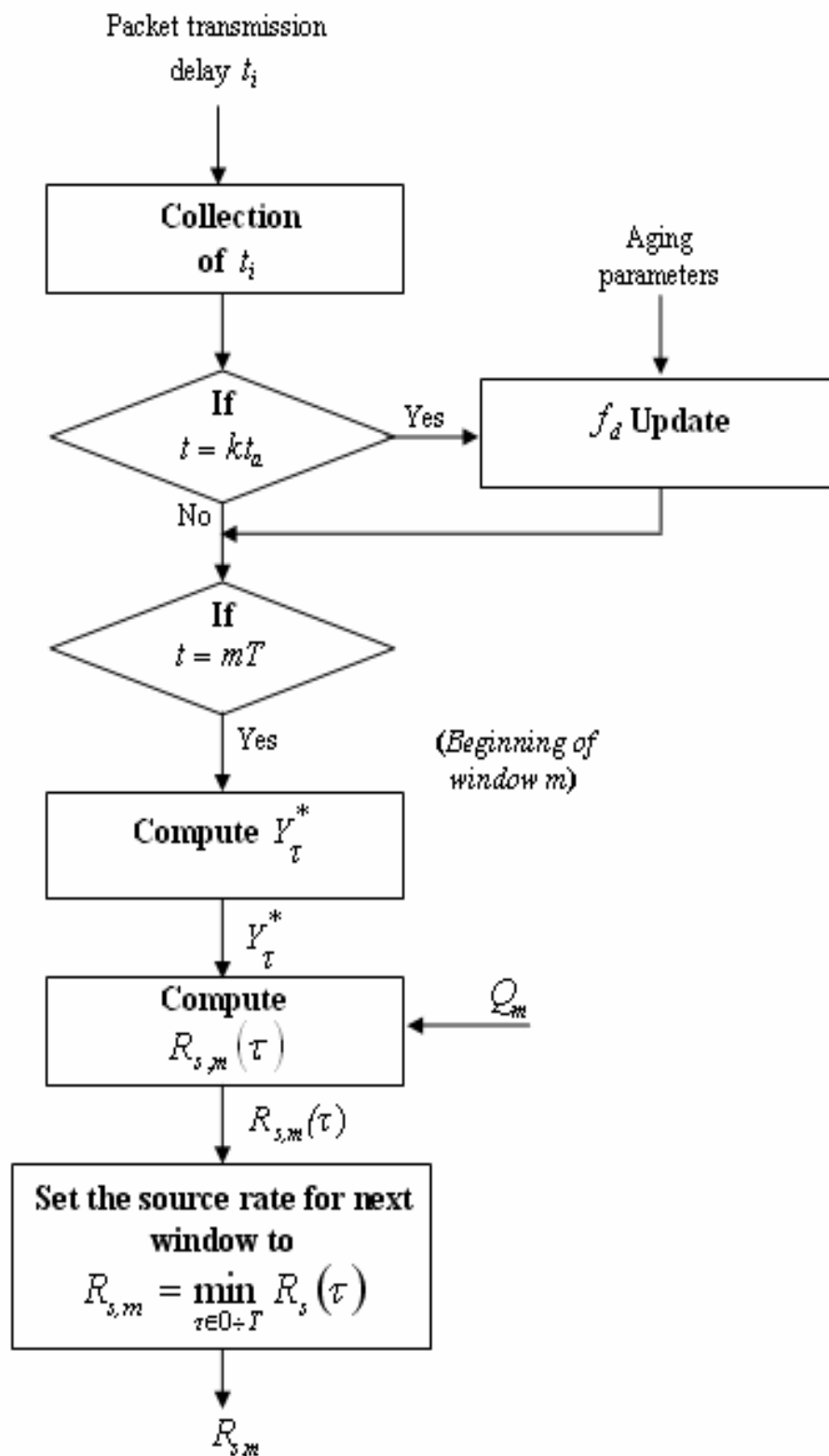


Figura 14 Diagramma di flusso dello schema proposto

All'inizio di ogni finestra, il modulo di controllo esegue il computo di Y_τ^* che verifichi l'uguaglianza tra $\Phi_m(t)$ e ε secondo quanto espresso nella (7.6) per ogni intervallo τ ($0 < \tau \leq T$). È importante osservare che l'operazione in questione richiede il calcolo dell'integrale, il

quale, inoltre è la somma delle probabilità $P_\tau(n)$ per $n=0,1,2,\dots$ finché la somma non è uguale a ε . Si tenga conto che ε deve essere un valore molto piccolo affinché l'evento di starvation nel playback buffer si verifichi di rado durante la riproduzione dello stream video.

Un valore tipico è quello di considerare una soglia di 10^{-3} . Ne consegue che $P_\tau(n)$ verrà calcolata solo per pochi valori che consentono il raggiungimento della soglia, così come si evince dalla Figura 15. Ovviamente, questo accorgimento costituisce un notevole vantaggio in termini computazionali, dal momento che $P_\tau(n)$ si ottiene da una serie di convoluzioni successive, come mostrato nella (7.3).

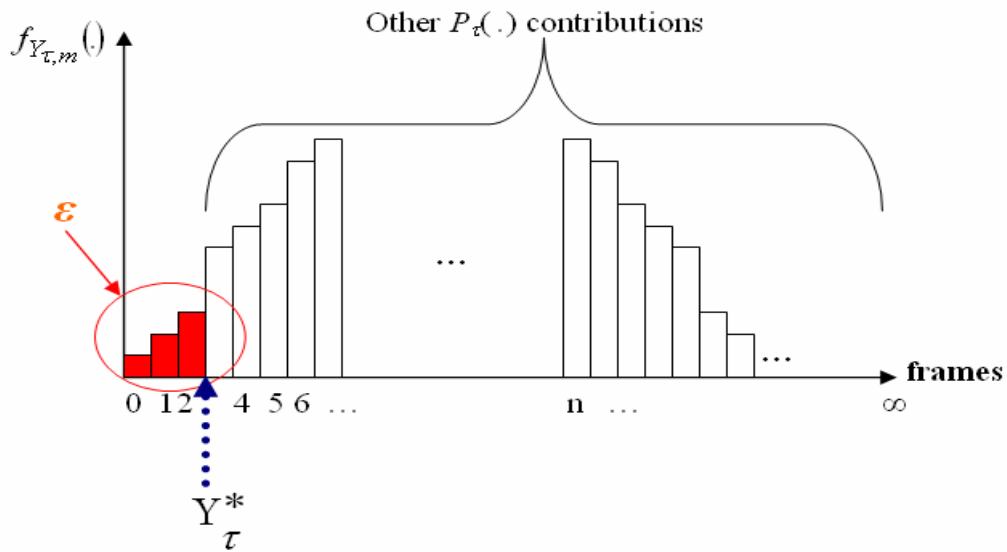


Figura 15 Calcolo di Y_τ^* .

Y_τ^* permette il calcolo del rate di codifica della sorgente in accordo con la relazione inversa $R_s(\tau) = Y_\tau^* R_f / (R_f \tau - Q_m)$. Il passo successivo riguarda, quindi, la selezione del valore ottimale del rate $R_{s,m}$ tra tutti quelli calcolati secondo la seguente espressione:

$$R_{s,m} = \min_{\tau \in 0 \div T} R_s(\tau). \quad (7.11)$$

La complessità computazionale dell'algoritmo è ridotta ulteriormente dal momento che lo step appena descritto non viene calcolato per tutti i valori di τ nel range $(0 < \tau \leq T)$ ma soltanto per $\tau=T$. Tale semplificazione è giustificata dal fatto che $R_s(\tau)$ decresce all'aumentare di τ , quindi il valore calcolato per τ alla fine dell'intervallo, risulterà essere il minore tra tutti i valori ottenibili nella finestra. La discussione in merito all'argomento è rimandata alla sezione successiva.

Capitolo 8

Risultati Sperimentali

8.1 Configurazioni Generali

La valutazione delle performance del sistema di controllo delle starvation è stato condotto per mezzo di simulazioni estensive che replicavano le tipiche comunicazioni in una LAN di tipo wireless in condizioni di traffico altamente variabili. Il traffico di background è rappresentato da comunicazioni ftp client-server di tipo on-off i cui periodi di on e off seguono le distribuzioni di Pareto e Esponenziale, rispettivamente.

L'algoritmo di controllo del rate proposto è implementato su un video server che condivide le risorse di canale con le comunicazioni ftp. Il canale wireless è modellato mediante un processo continuo nel tempo a due stati (good/bad) con un andamento dei periodi good e bad che segue una distribuzione di tipo gamma. Durante i periodi good il canale è caratterizzato da bassi valori di BER al contrario di quanto accade nei periodi bad in cui il valore del BER è alto.

La tabella 1 sintetizza i valori dei parametri chiave usati nelle simulazioni per la configurazione del sistema.

Il significato delle nuove variabili che compaiono in tabella è il seguente: C indica la capacità della network, P_g e P_b sono, rispettivamente, le probabilità di errore ottenuta sperimentalmente durante i periodi good e bad, e R è il throughput finale medio calcolato nella stazione server.

Tutte le simulazioni sono state realizzate mediante il simulatore Opnet Modeler insieme al modulo wireless.

TABELLA 2 VALORI DEI PARAMETRI USATI NELLA SIMULAZIONE

Parametri	Valori
Numero utenti FTP	15
T	1 sec
C	11 Mbps
P_g	10^{-4}
P_b	10^{-2}
R	107,2 kbps
Bin size	10^{-4} sec
R_f	25 fps
E	10^{-4}

8.2 Valutazioni sulla stima del ritardo di trasmissione dei pacchetti

La Figura 16 mostra il tipico andamento della pdf dei ritardi rispetto al tempo necessario a spedire un pacchetto con successo e considerando la dimensione del bin uguale a 10^{-4} sec. La funzione si distende su un range di valori abbastanza largo a causa dell'alta variabilità del throughput del canale. La media dei ritardi si attesta intorno a 10 msec (100-esimo bin), perciò il server è in grado di spedire, durante una finestra temporale di un secondo, mediamente 100 pacchetti.

Per verificare l'assunzione riguardante l'indipendenza tra due trasmissioni di pacchetto successive che ha permesso il calcolo probabilità $P_f(n)$, è stata valutata l'autocorrelazione tra i ritardi dei pacchetti trasmessi. Il risultato della verifica è mostrato in Figura 17. Sono stati presi in considerazione i ritardi osservati per 100,000 pacchetti, con un risultante valor medio pari a $\eta=9.7558*10^{-3}$.

La funzione di autocorrelazione ottenuta assume il valore di picco $1.8959*10^{-4}$ per $t = 0$ e diventa uguale $\eta^2=9.5176*10^{-5}$ abbastanza presto ($t \geq 3$). Da questi risultati si evince che il processo considerato è praticamente privo di memoria, memoryless, dunque le trasmissioni di due pacchetti consecutivi possono essere considerati indipendenti, e in ultimo le assunzioni che hanno consentito di scrivere la (7.3) sono giustificate.

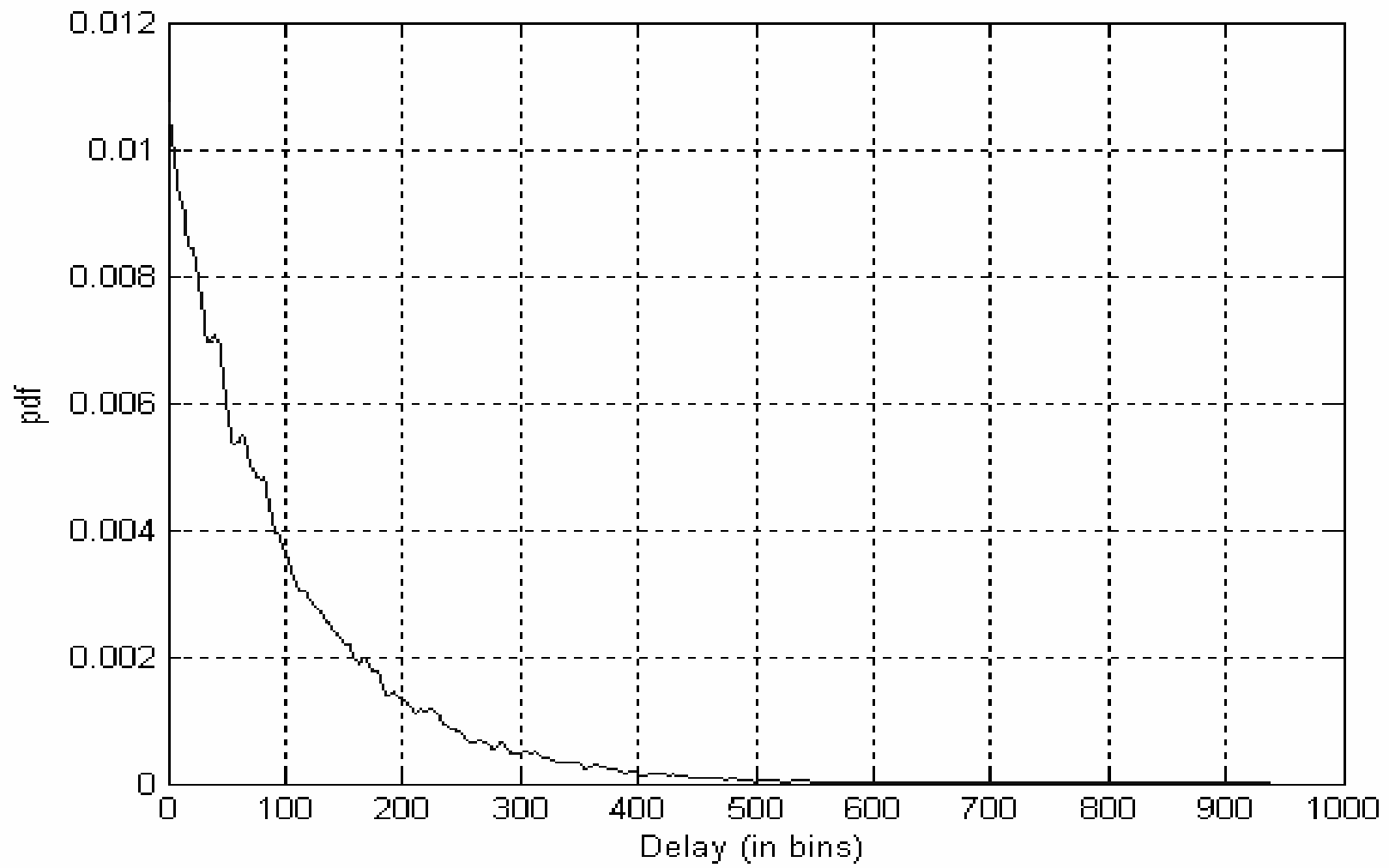


Figura 16 Pdf del delay di rete.

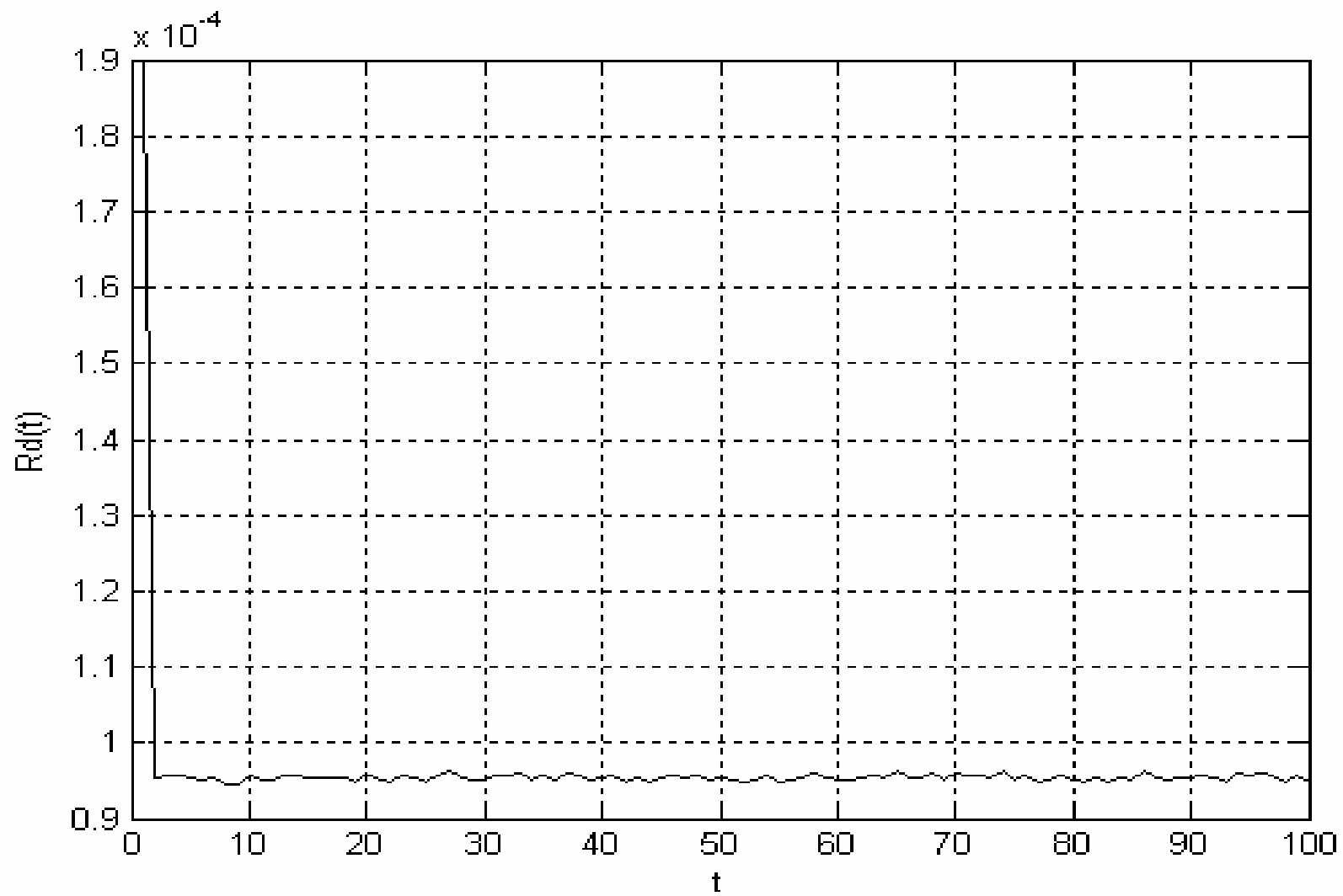


Figura 17 Autocorrelazione del delay di rete.

8.2.1 Chi-quadro test

L'operazione chiave nell'algoritmo proposto è il calcolo della funzione di densità della probabilità $f_d(t)$, che si determina sulla base delle informazioni sulle misure passate. Risulta indispensabile che la predizione fatta sia quanto più realistica possibile per ottenere una procedura di ottimizzazione affidabile. Per analizzare la corrispondenza tra valori predetti e valori reali, si è fatto ricorso al test del Chi-Quadro, che viene frequentemente impiegato per verificare se i campioni di un certo dato appartengano o meno ad una popolazione caratterizzata da una certa distribuzione. Con questo test siamo in grado di valutare la bontà della predizione verificando se il valore di chi quadro che si ottiene dal confronto dei dati risulti inferiore o no rispetto ad un valore critico per un dato livello di significatività, o di confidenza.

Gli esperimenti consistono nel calcolo del valore medio del Chi-Quadro per un numero significativo di cicli, si ricordi che la dimensione di ciascuna finestra è pari a T , e successivamente comparando questa quantità con il valore critico. Per ogni ciclo, si esegue il computo dell'istogramma dei ritardi osservati durante il ciclo sotto analisi e si determina la frequenza attesa dalla pdf precedentemente predetta. Per eseguire queste stime, sono state considerate tutte e tre le tecniche di aging descritte nel capitolo precedente.

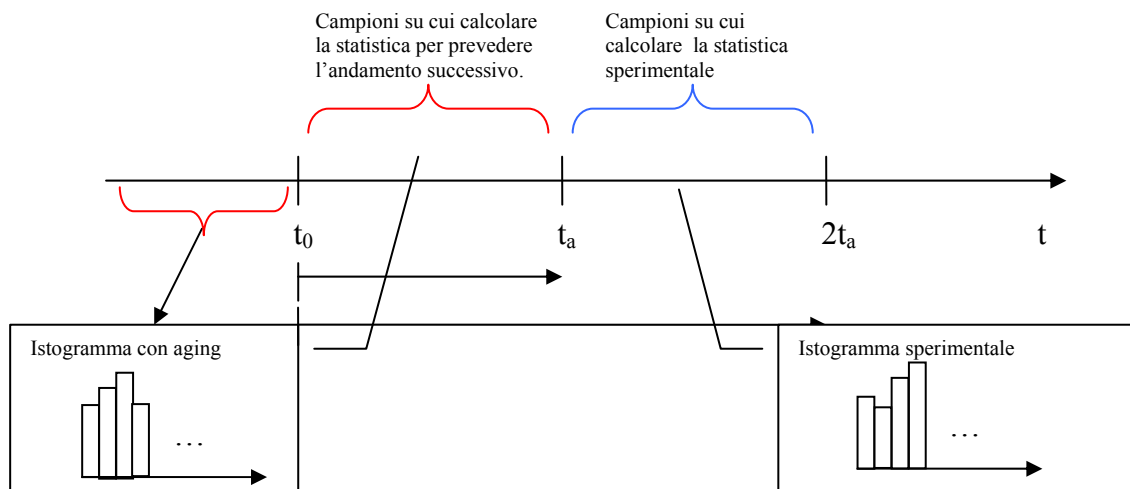


Figura 18 Determinazione dei campioni da confrontare nel Chi-Test.

Un inconveniente del Chi-Quadro Test è dovuto alla necessità di avere una dimensione dei campioni sufficiente affinché l'approssimazione del chi-quadro possa essere ritenuta valida. Per questa ragione la dimensione dei bin dell'istogramma relativo ai dati osservati è settato ad un valore di 10^{-3} sec. Inoltre, perché l'approssimazione del test sia valida è necessario che la frequenza attesa sia al minimo pari a 5; quindi, l'algoritmo sviluppato per il test automaticamente provvede all'accorpamento dei bin adiacenti ciclo-per-ciclo finché non si raggiunge la soglia desiderata. Si noti che in questo modo il numero dei gradi di libertà ν non

risulta costante tra un ciclo e l'altro. L'espressione che consente il calcolo del chi-quadro è la seguente:

$$\chi^2 = \frac{\sum_{l=1}^k (O_l - E_l)^2 E_l}{E_l}, \quad (8.1)$$

dove O_l indica la frequenza osservata per il bin l e E_l indica, invece, la frequenza attesa per lo stesso bin.

Sono stati calcolati per i cicli presi in considerazione (10,000) i valori medi dei gradi di libertà ($\bar{\nu}$) e del chi quadro ($\bar{\chi}^2$) è l'ipotesi assunta è stata rigettata se $\bar{\chi}^2 > \chi^2_{(\alpha, \bar{\nu})}$, dove $\chi^2_{(\alpha, \bar{\nu})}$ è la funzione di punto percentuale per $\bar{\nu}$ gradi di libertà e una soglia di confidenza pari ad α , il cui valore, normalmente, è scelto pari all'1% o 5%. Inoltre, più basso è il valore di $\bar{\chi}^2$ maggiore sarà l'accuratezza della stima.

Nei test effettuati il grado di libertà risulta attorno a 33, ne deriva un valore del chi-quadro all'1% e al 5%, rispettivamente di $\chi^2_{(0.01, 33)} = 54.8$ and $\chi^2_{(0.05, 33)} = 47.4$. Nelle Figura 19, Figura 20 e Figura 21 sono riportati i valori delle sperimentazioni relativi alle tre tecniche di aging al variare del coefficiente di aging c e dell'intervallo di aging t_a nei range $0.1 \div 0.9$ e $1 \div 30$ sec, rispettivamente. Per migliorare la presentazione, le curve sono state tagliate per i valori superiori a 200, che risulta molto più alto rispetto alle soglie di interesse.

Da tutte e tre le figure risulta che entrambi i parametri risultano fondamentali nella predizione. Relativamente all'intervallo di aging, la funzione chi-quadro è caratterizzata da una curva monotona decrescente fino a circa 20 sec, dopo del quale la curva devia da questo andamento. Questo fenomeno ci dice che aggiornamenti troppo frequenti della pdf conducono a delle predizioni non stabili, le quali risulteranno troppo influenzate dalle ultime statistiche osservate decrementando la frequenza si osserva che la predizione migliora fino al punto in cui i dati più vecchi hanno un peso troppo alto rispetto alle misurazioni più recenti. Questo fenomeno caratterizza tutte e tre le tecniche.

Relativamente al coefficiente di aging, i risultati mostrano che riducendo questo coefficiente il valore del chi quadro cresce raggiungendo valori più alti della soglia critica, per i quali l'ipotesi iniziale deve essere rigettata. Si è osservato che le migliori performance si registrano per un valore del coefficiente di aging entro l'intervallo $0.7 \div 0.9$.

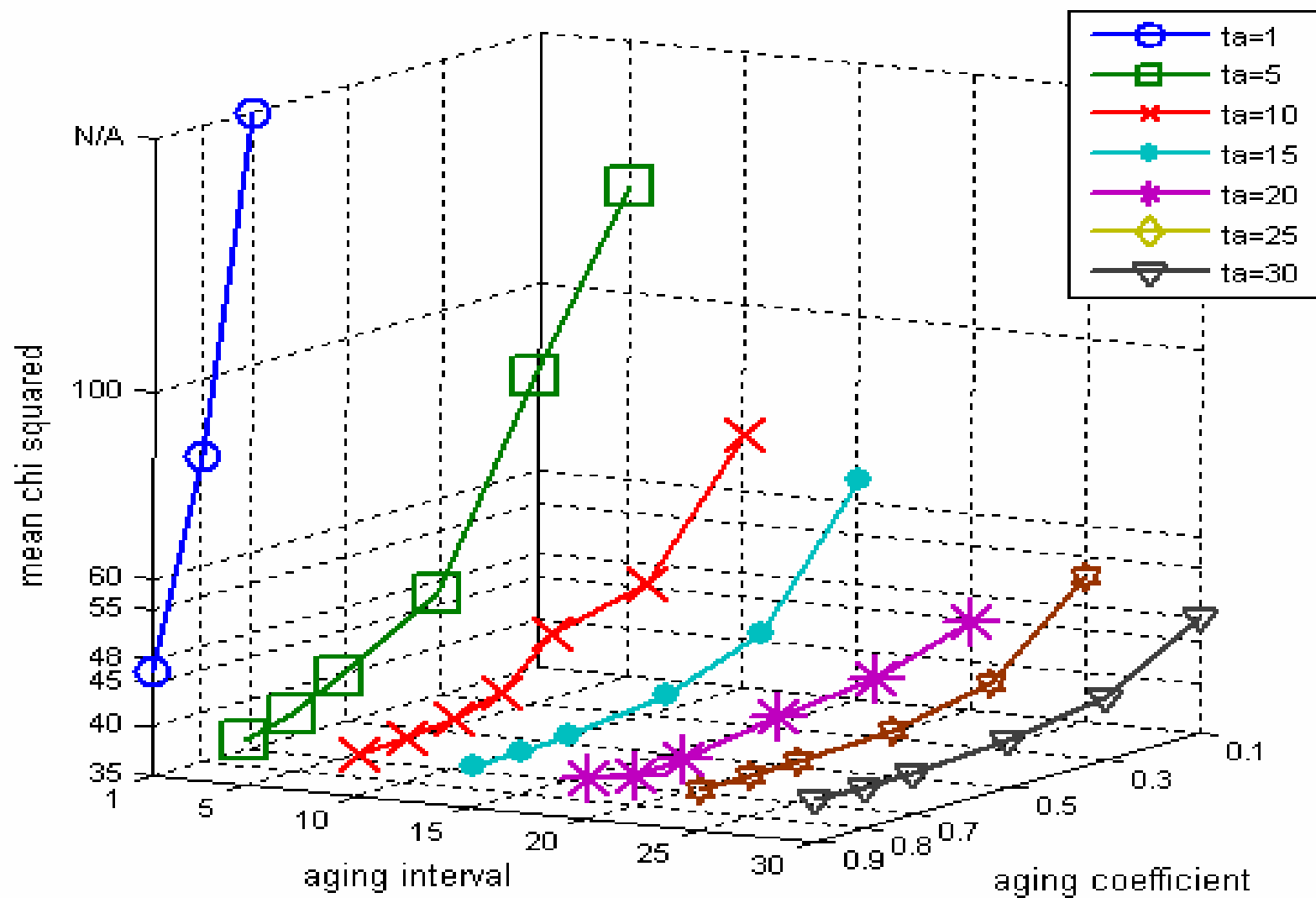


Figura 19 Valore medio del chiquadro contro c ed f (in secondi). Prima tecnica di aging.

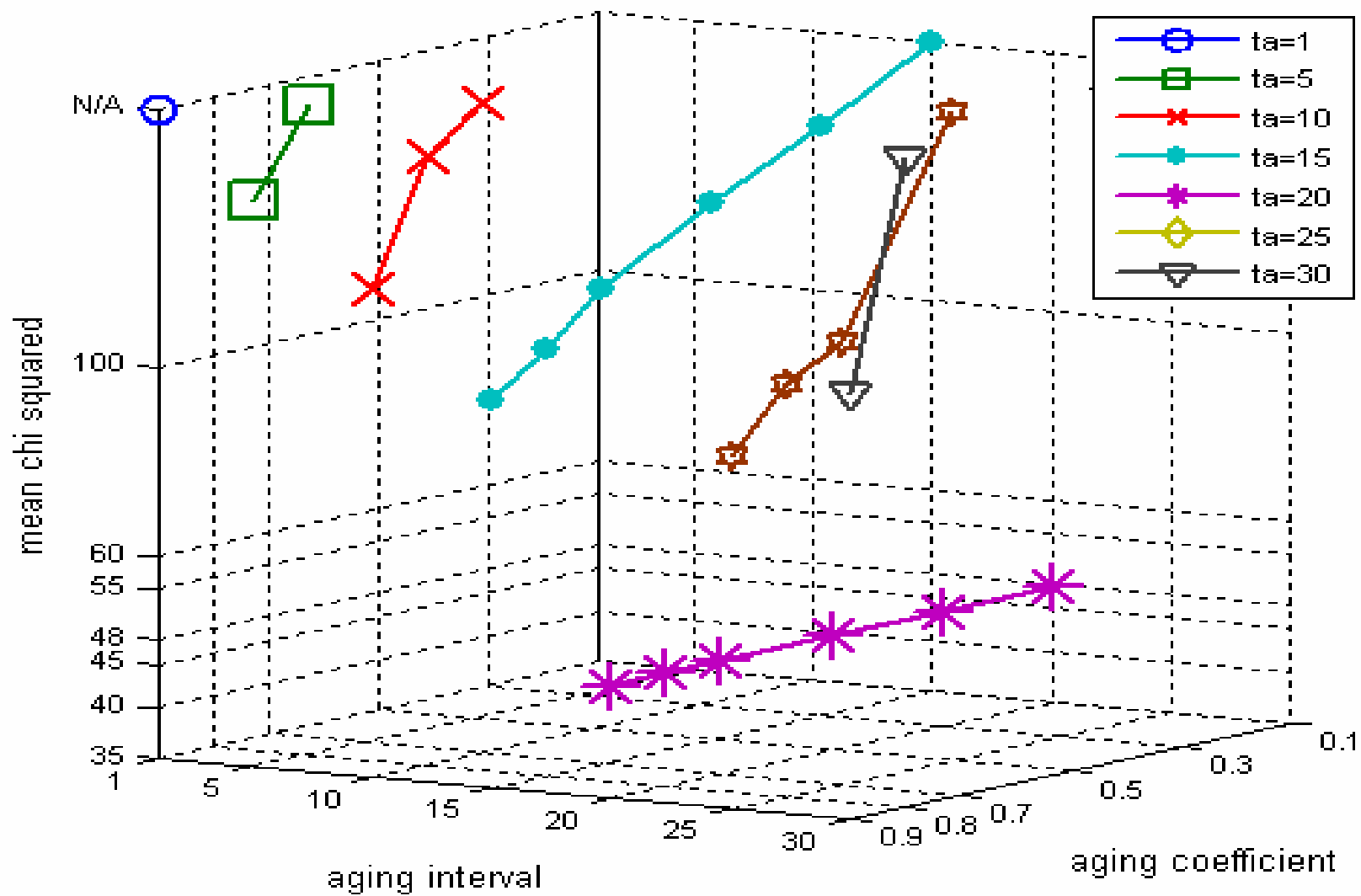


Figura 20 Valore medio del chiquadro contro c ed f (in secondi). Seconda tecnica di aging.

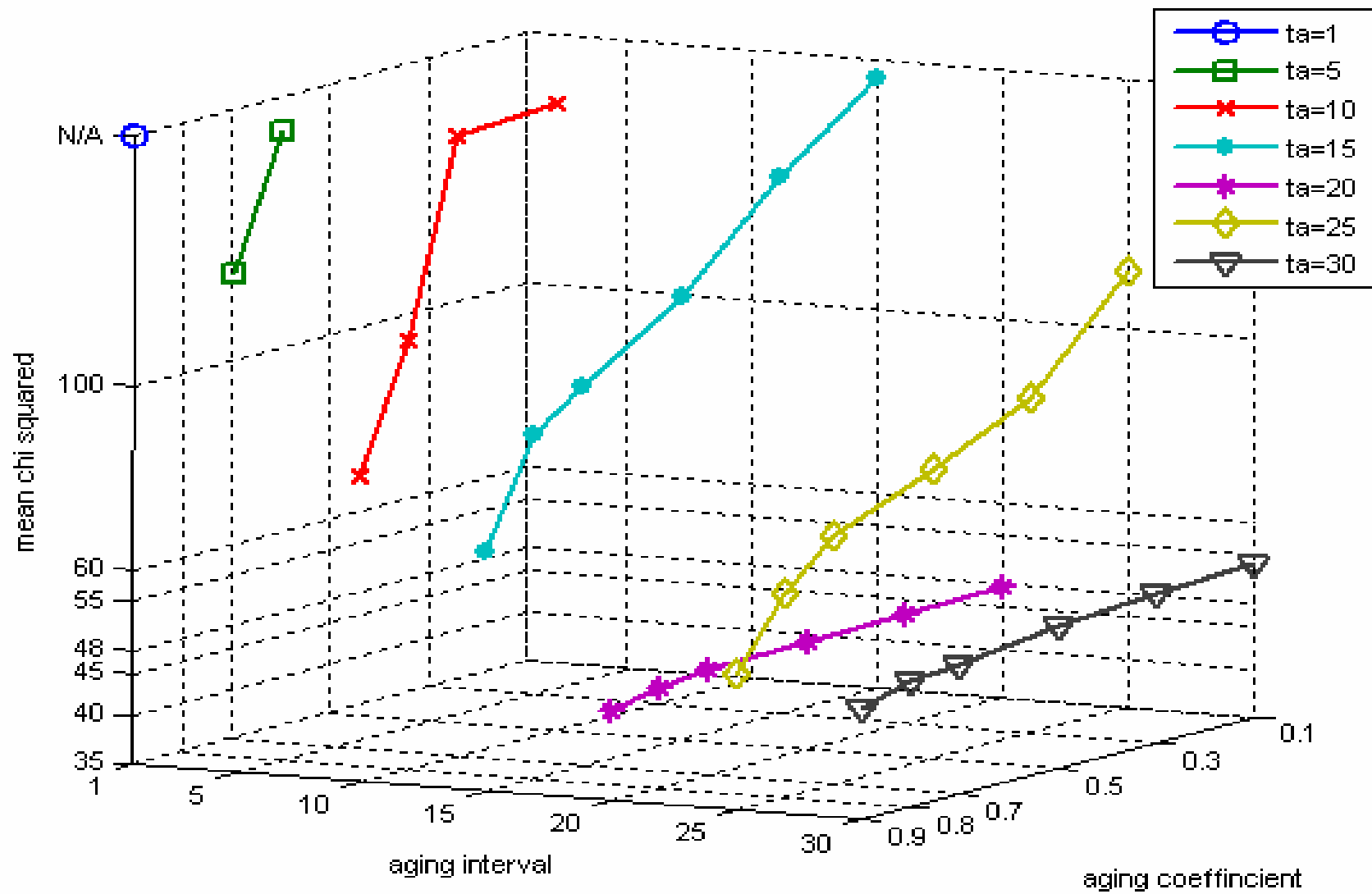


Figura 21 Valore medio del chiquadro contro c ed f (in secondi). Terza tecnica di aging.

Comparando i risultati delle tre tecniche, si evince che tra le tre la prima tecnica risulta più stabile al variare dei parametri di aging. Infatti, per i valori di t_a uguali a 20, 25 e 30 secondi, la maggior parte dei valori dei chi-quadro determinati sono sotto la soglia critica per $\alpha = 5\%$, mentre sono sempre sotto la soglia di $\chi^2_{(0.01,33)}$. Quando t_a è uguale a 15 o 10 sec solo alcuni punti relativi a $c = 0.1$ sono sotto la soglia di $\chi^2_{(0.05,33)}$. In fine, per f uguale a 5, il valore di chi-quadro è maggiore di $\chi^2_{(0.01,33)}$ solo per $c < 0.5$. Le altre due tecniche sono caratterizzate da un comportamento che risulta più legato alle impostazioni dei parametri, comunque, ci sono sempre anche in questo caso delle impostazioni per la coppia di parametri che determinano un valore del chi-quadro corrispondente inferiore alla soglia di $\chi^2_{(0.01,33)}$. Le prove sul chi test sono state fatte anche quando non viene applicato nessun fattore di aging. Il valore medio di chi-quadro $\overline{\chi^2}$ ottenuto è uguale a 39.4, più basso della soglia critica per un livello di significatività pari al 5% e ancor più sotto il valore critico corrispondente a $\alpha = 0.01$.

La disponibilità al modulo di controllo dei dati attuali e della predizione permette di adattare in real-time le impostazioni dei parametri in modo che vengano sempre utilizzati i valori ottimali durante lo streaming. Questo setting adattivo può essere guidato dal test del chi-quadro così come è stato presentato in questa sezione, o impiegando un approccio più semplice che confronta dati predetti e dati osservati attraverso la misura del mean square error (MSE).

8.3 Validazione dell'ipotesi semplificativa per il calcolo di R_s

Per rispettare il target relativo alla probabilità di starvation la (7.11) dovrebbe essere analizzata per tutto l'intervallo $0 < \tau < T$, tuttavia ci possiamo aspettare che maggiore è il valore di τ minore sarà il bit rate di codifica sorgente. Tale affermazione è supportata dalle osservazioni sulla varianza, da cui si evince che: il numero dei bit ricevuti nell'intervallo di tempo cresce all'aumentare della lunghezza dell'intervallo di tempo considerato. Questo impone un bit rate di codifica di sorgente più conservativo per una finestra temporale più lunga. Il fenomeno appena descritto è stato dimostrato sperimentalmente. La Figura 22 mostra l'evoluzione del bit rate di sorgente al variare dell'intervallo τ per differenti valori dell'occupazione del buffer all'inizio di ogni finestra. Si noti che più è alto il valore di τ minore risulta il valore di R_s . L'andamento decrescente di R_s dimostra che i risultati ottenuti per $\tau = T$ sono sufficienti a garantire la condizione imposta sulla starvation per ogni $\tau < T$. Tutto ciò si traduce in un enorme vantaggio relativamente alla complessità computazionale dell'algoritmo di controllo del rate proposto.

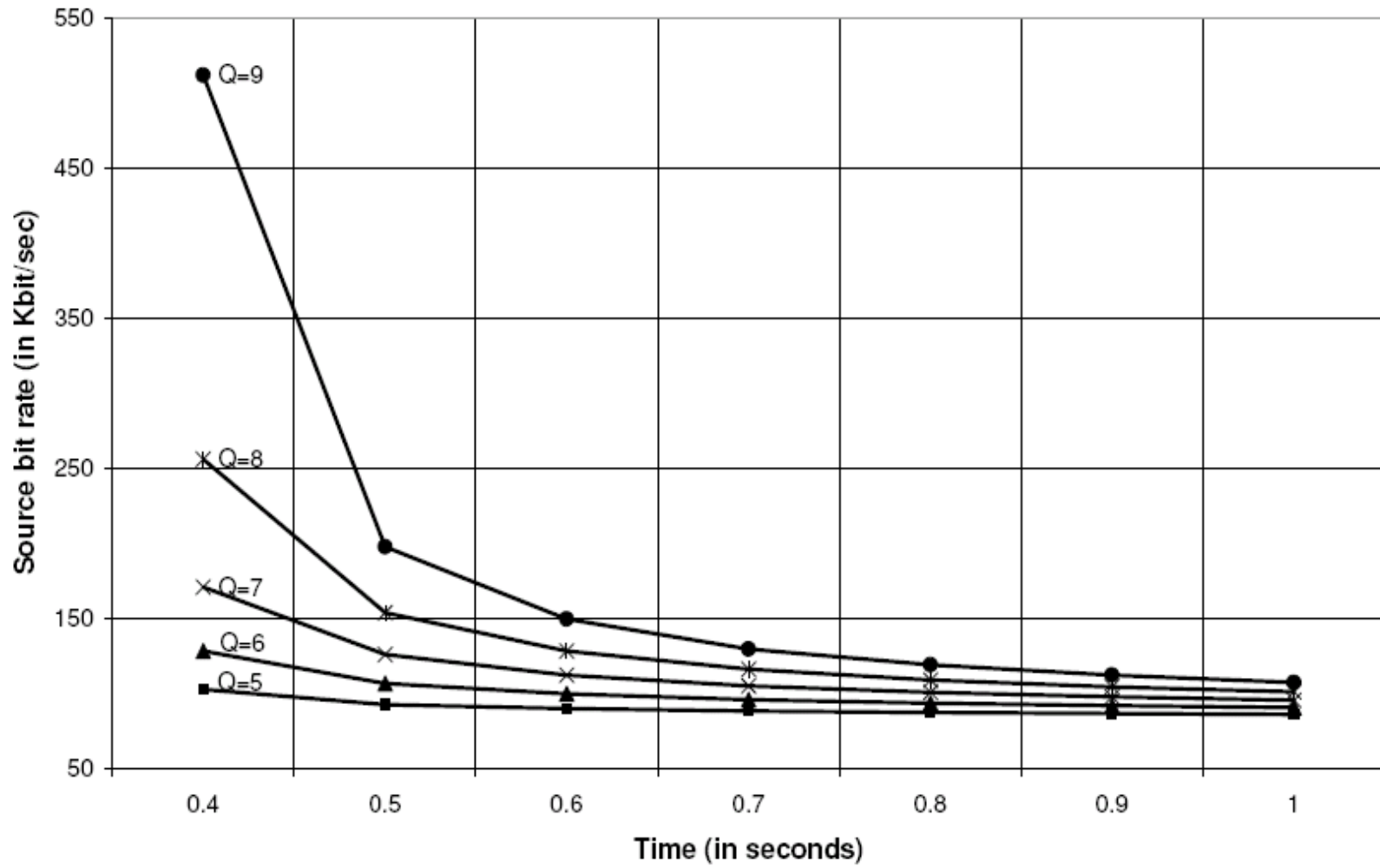


Figura 22 Source bit rate versus T per differenti valori iniziali dell'occupazione del buffer.

8.4 Valutazioni finali sul metodo proposto

La tabella Table II mostra la media dei valori, ottenuti per diverse simulazioni, degli indicatori di performance chiave relativamente all' algoritmo proposto. Le tre tecniche di aging vengono raffrontate al caso in cui non si usi l'aging, corrispondente all'approccio full aggregation. Per ogni funzione di aging sono riportati in tabella i risultati ottenuti impostando i valori migliori dei parametri che risultavano nell'analisi precedentemente descritta, che dovrebbero, quindi, portare alla migliore qualità del video. Come si evince dalla tabella, la frequenza di starvation ottenuta è sempre solamente di poco superiore alla soglia ε , il che è un buonissimo risultato.

TABELLA 3 RESULTATI DELL'ALGORITMO PROPOSTO

	<i>PROBABILITÀ DI STARVATION</i>	<i>OCCUP. BUFFER MEDIA</i>	<i>SOURCE RATE MEDIO</i>	<i>DEVIAZIONE STANDARD DI R_s</i>	Y_τ^*
No-aging	3.3×10^{-4}	8.3601	119kbps	65407bps	67
Aging 1	3.3×10^{-4}	8.3308	118kbps	65365bps	67.2
Aging 2	3.3×10^{-4}	8.3029	118kbps	63424bps	66.5
Aging 3	3.4×10^{-4}	8.0302	117kbps	62818bps	68.3

8.4.1 Valutazione della qualità video

Per valutare la qualità del video, sono stati calcolati anche:

- il valore medio del rate di codifica della sorgente R_s , il quale essendo uguale a 107Kbps si attesta intorno al valore medio del throughput di canale;
- la deviazione standard, che risulta significativamente bassa.

La deviazione standard R_s , è indice della qualità in questi termini: più basso è il suo valore, minore è la variazione della qualità del video tra finestre adiacenti. Il valore più basso registrato per questo indicatore è quello ottenuto con il terzo metodo di aging.

8.4.2 Valutazioni sul rate sorgente e occupazione del playback buffer

Per quanto concerne l'occupazione del buffer, essa decresce al migliorare della qualità del video, così come ci si aspetta. Il valore medio di Q è più basso nel caso della terza tecnica di aging che, comunque, fornisce il valore più alto delle probabilità di starvation osservate. In tutti e quattro i casi il valore medio dell'occupazione del buffer è intorno a 8 frame

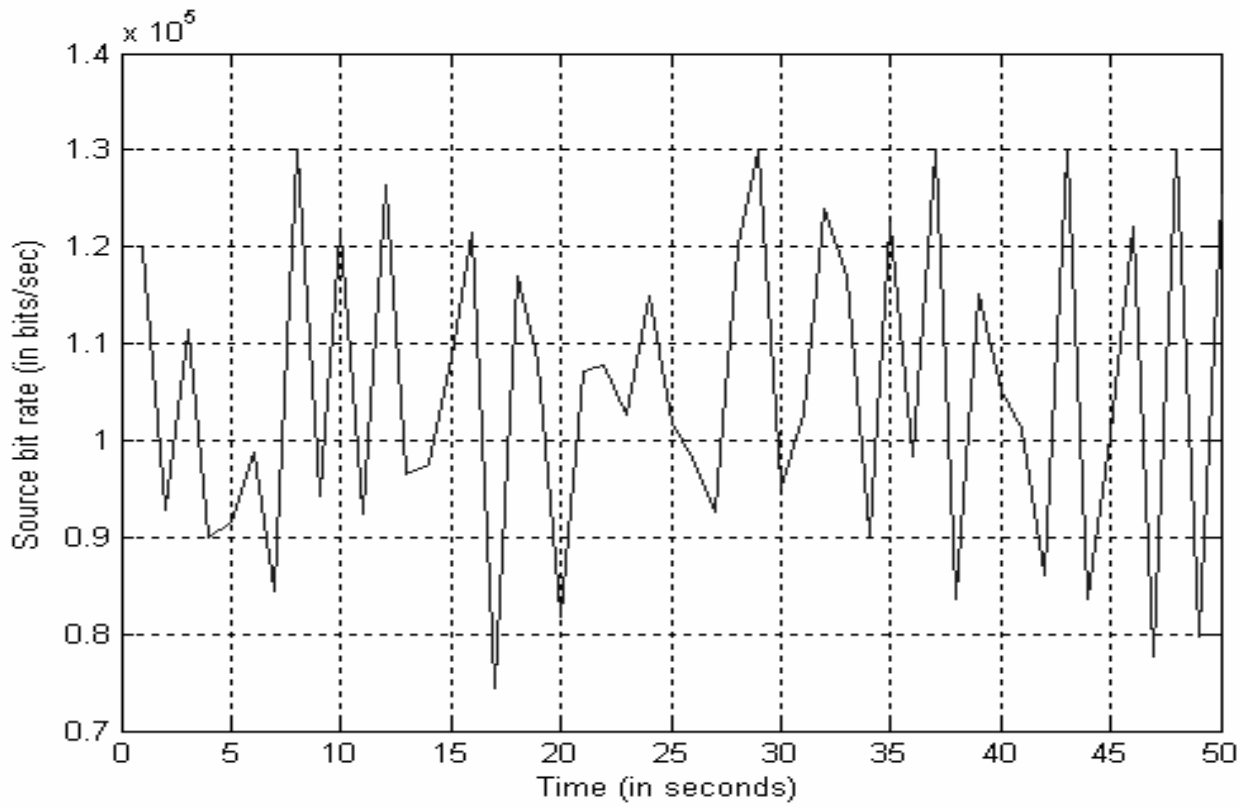


Figura 23 Source bit rate versus tempo.

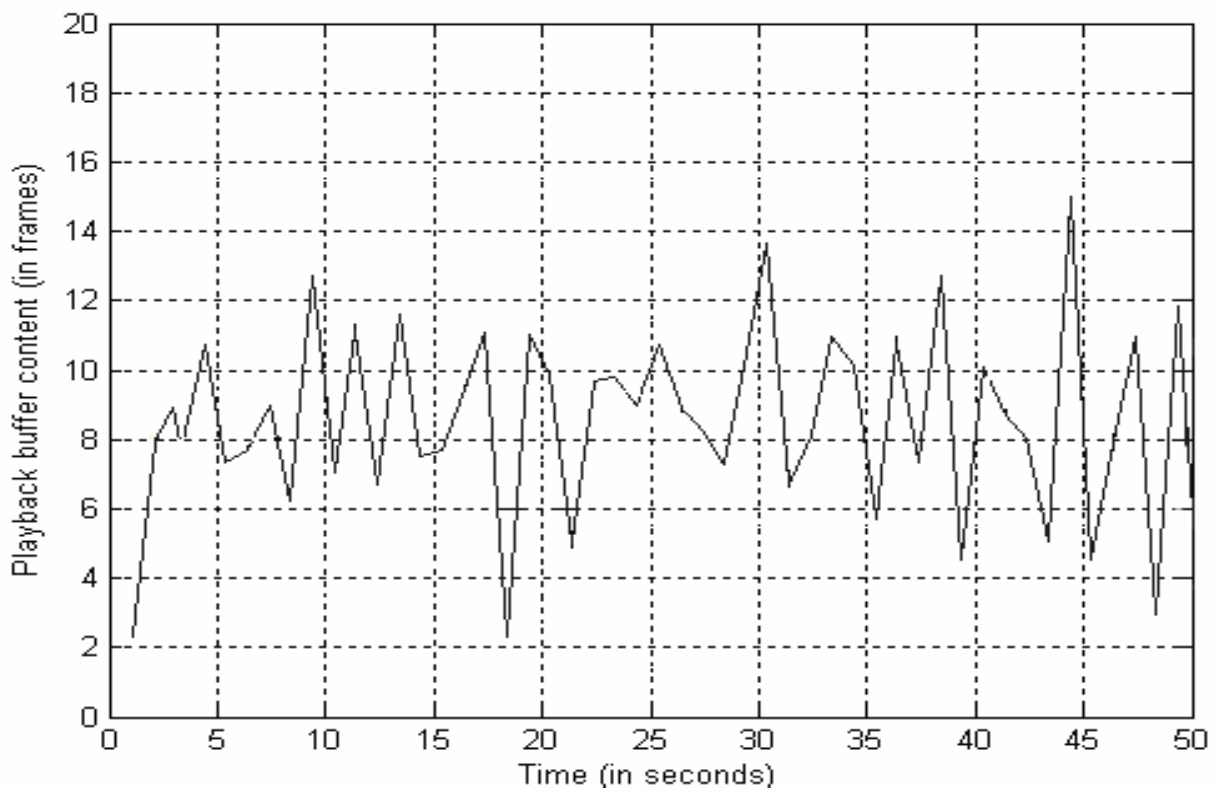


Figura 24 Occupazione del Playback buffer versus tempo.

. Questo valore rappresenta la quantità di frame che mediamente è necessario precaricare nel buffer, al fine di garantire il controllo della probabilità di starvation.

La Figura 23 mostra l'andamento temporale del bit rate di sorgente, mentre la Figura 24 riporta l'evoluzione dell'occupazione del buffer, entrambi sono relativi agli stessi 50 cicli successivi, considerando la prima tecnica di aging con $c = 7$ e $f = 100$. Come ci si aspetta, i comportamenti di R_s e Q sono in proporzione inversa: un evento di aumento dell'occupazione del buffer corrisponde ad un decremento del bit rate di codifica sorgente e viceversa.

8.4.3 Valutazioni sulla complessità computazionale

Inoltre, è stato riportato in tabella il valore di Y_τ^* , il quale dà indicazione riguardo alla complessità computazionale dell'algoritmo. Infatti, si può osservare che non è necessario il calcolo di tutta la funzione $f_{Y_{\tau,m}}(y)$ ma solo per i valori di y più bassi, nello specifico, quelli per cui si raggiunge la soglia ε . Di conseguenza anche la probabilità $P_\tau(n)$ deve essere calcolata per pochi valori di n sulla base all'entità di Y_τ^* . Per tutti e quattro i casi risulta che il valore di Y_τ^* è molto simile e si attesta nell'intorno di 67.5 Kbit.

8.5 Effetti delle tecniche di aging sul sistema di controllo

Allo scopo di stimare gli effetti delle tecniche di aging sullo schema di controllo del rate proposto sono state valutate le performance al variare dei parametri di aging. In particolare si è analizzato il comportamento di: deviazione standard di R_s , la complessità computazionale e occupazione media del buffer in termini di aging factor F . È stato ottenuto che, decrementando F la qualità del video riprodotto migliora a scapito della complessità computazionale e dell'occupazione del buffer. Infatti, quando F diventa inferiore a 1, la pdf relativa ai ritardi di rete si appiattisce e il calcolo della probabilità $P_\tau(n)$ necessita di un numero superiore di n , e quindi vi sarà un aumento del valore di Y_τ^* . Ne consegue che R_s cresce al crescere della complessità computazionale, mentre l'occupazione del buffer decresce in accordo con la relazione (7.1).

Gli esperimenti hanno dimostrato che per $Y_\tau^* > 69$, il valore medio di Q diventa negativo e il buffer di plaback si svuota molto frequentemente, vanificando la validità del metodo proposto.

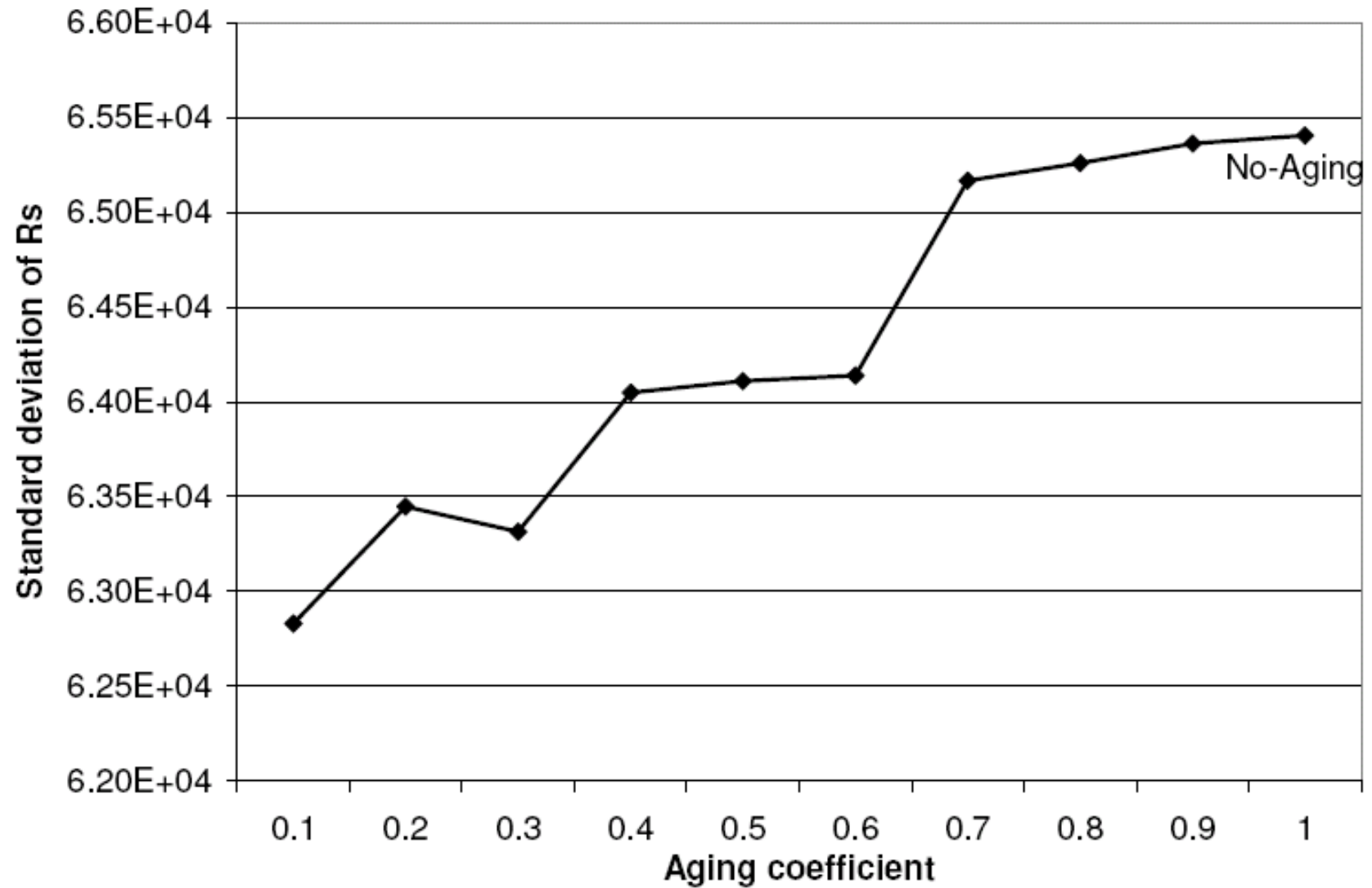


Figura 25 Andamento della deviazione standard del rate sorgente per Aging 1.

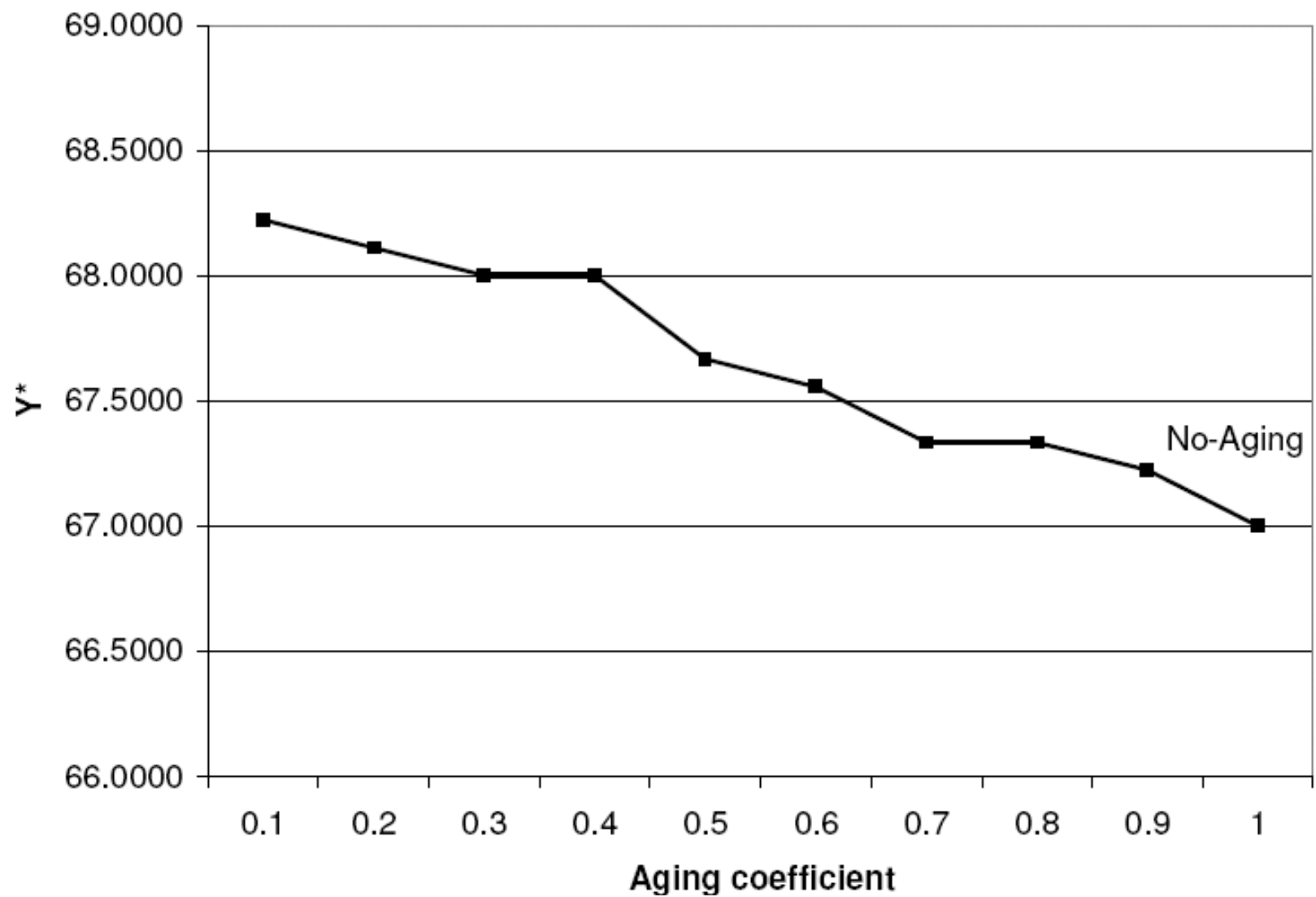


Figura 26 Andamento della Y_{τ}^* per Aging 1.

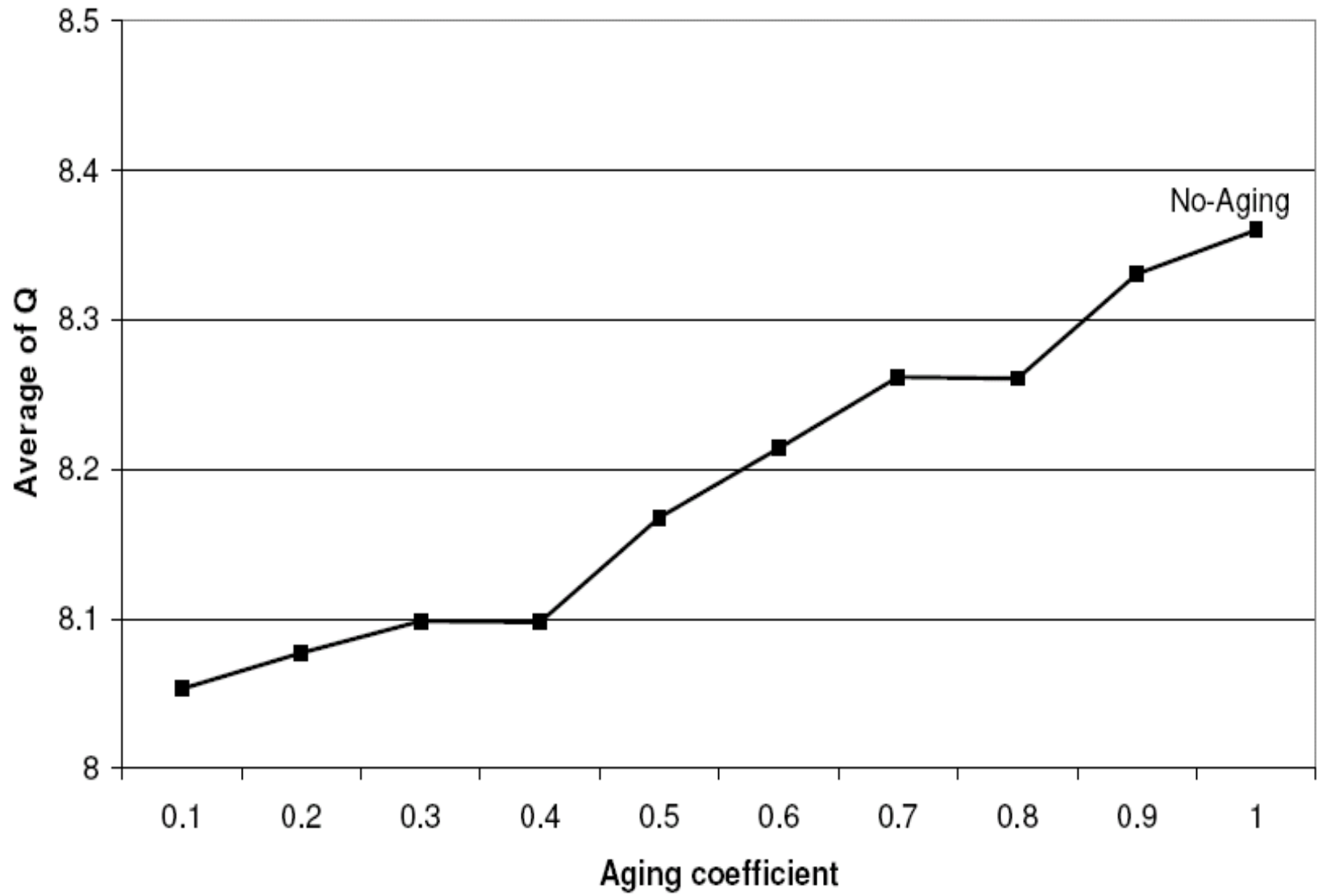


Figura 27 Andamento della occupazione media del buffer per Aging 1.

Le figure precedenti mostrano i risultati relativi alla prima tecnica di aging in cui F è uguale al coefficiente di aging c . In questo specifico caso, il valore più basso di c corrisponde a $Y_{\tau}^* = 68.2222$, implicando buoni risultati i valori di aging. Nel caso del secondo metodo di aging impostando lo stesso valore del coefficiente c , il valore del fattore F risulta più basso che nel primo caso a causa della sua relazione inversa con l'area dell'istogramma (7.9). Questo produce un incremento della probabilità di stovation osservata, la quale supera la soglia ε per $c < 7$ corrispondente a $Y_{\tau}^* > 69.2222$. La terza tecnica di aging definisce un fattore di aging molto simile a quello della seconda tecnica ma in più aggiunge una proporzionalità relativa alla frequenza di aging f . In questo caso, si ottengono dei buoni risultati per $0.3 < c < 0.9$ e $f = 100$. Decrementando f , cioè, incrementando la frequenza di aggiornamento dei ritardi dei pacchetti si osserva lo stesso effetto sui valori dei parametri del sistema che si osservano quando si decrementa il coefficiente di aging.

Conclusioni

L'introduzione delle reti NGN, basate sul paradigma IP, consente di incrementare la produttività della rete ottimizzando la gestione delle risorse ed aumentando l'appetibilità dei servizi offerti all'utente. Tuttavia l'introduzione delle nuove piattaforme NGN deve avvenire in simbiosi con l'uso di appropriate procedure per la gestione della qualità del servizio (QoS). Infatti, una soluzione NGN, a fronte dei vantaggi descritti, porta ad un aumento della complessità di gestione degli strumenti per il controllo della qualità del servizio rispetto alle soluzioni di rete tradizionali.

Nel lavoro proposto è stata eseguita una classificazione e caratterizzazione dei servizi tramite la quale si gettano le fondamenta per l'implementazione ottimale di un sistema di gestione della qualità. Si sono poi analizzate varie soluzioni architetturali e si è scelta per la rete di supporto una combinazione di esse, la DS-TE, che sembrerebbe rispondere meglio alle esigenze di differenziazione del traffico, tramite DiffServ, e di qualità end-to-end, mediante MPLS, inoltre, introducendo il Traffic Engineering, oltre ad una gestione ottimale del traffico di rete si ottiene una migliore gestione anche delle risorse settando a seconda delle esigenze del provider una serie di attributi.

Una volta definita l'architettura di riferimento si è passati allo studio di alcune metodologie di ottimizzazione relativamente ad alcuni argomenti specifici: il primo inerente l'instradamento degli LSP nella rete di backone, mentre il secondo riguarda il servizio video-streaming su una wireless LAN che potrebbe essere intesa come una delle comuni reti di accesso della rete multidominio.

Elemento di base delle tecniche di ottimizzazione del routing è il Multicommodity Flow Problem che viene impiegato sia per la mappatura iniziale dei percorsi, nello stato off-line della rete, che nella fase dinamica e altamente variabile, di gestione delle nuove richieste nello stato di on-line. Relativamente a quest'ultima fase, sono stati proposti tre approcci per la risoluzione del problema d'instradamento. Il primo è un approccio che localmente e in maniera incrementale cerca, sulla base della priorità della richiesta, il miglior percorso per il set up del nuovo LSP. Se non vi sono risorse sufficienti al soddisfacimento della richiesta si utilizza la preemption di LSP già esistenti sulla rete. Il percorso ottimale è ottenuto minimizzando una funzione di costo i cui obiettivi sono non solo la il coinvolgimento degli LSP con priorità inferiore, ma anche la minimizzazione del numero di LSP abbattuti e la riduzione della banda preempted. La tecnica si fa carico anche della ricerca di percorsi di reinstradamento ottimi degli LSP coinvolti nella preemption che evita o limita il numero dei service

disruption, in accordo con le necessità del provider. La configurazione che viene ottenuta attraverso l'impiego di questa metodologia con approccio decentralizzato produce un'ottimizzazione nell'uso delle risorse di tipo locale. La seconda tecnica proposta, invece, a fronte di una maggiore complessità affronta il problema del routing on-line secondo un approccio centralizzato, basato anch'esso sull'MCFP, che garantisce una riottimizzazione globale dell'utilizzazione delle risorse di rete e, quindi, un miglior load balancing con una forte riduzione dei service disruption. Per ottenere il miglior compromesso tra complessità e prestazioni, si è pensato ad una metodologia ibrida che impiega la prima tecnica di default salvo usare la seconda ogni qualvolta si registri un calo eccessivo delle prestazioni, misurate per esempio attraverso il raggiungimento di una certa soglia da parte del numero di interruzioni di servizio, in questo modo, periodicamente, la seconda tecnica provvede ad una riottimizzazione globale della rete che induce un incremento delle risorse utilizzabili.

Nella seconda parte della tesi è stato presentato uno schema di controllo del rate per il video streaming su canali wireless che realizza un approccio basato su finestre che permette di ridurre le fluttuazioni del bit rate di sorgente. Il rate di codifica è calcolato imponendo al buffer di playback che le occorrenze di starvation restino sotto una soglia desiderata. La parte chiave dell'algoritmo è la predizione dei ritardi futuri sulla base dei dati che sono stati precedentemente raccolti. I risultati sperimentali dimostrano l'efficacia dell'approccio proposto nel controllo della frequenza di underflow del buffer. Una leggera discrepanza tra la probabilità di starvation desiderata e il rate relativo agli eventi di starvation osservato è principalmente dovuto all'assunzione dell'indipendenza tra due frame trasmessi consecutivamente. I risultati sperimentali mostrano che la predizione dei ritardi per i futuri pacchetti è influenzata cospicuamente dai parametri che controllano l'aggiornamento e i pesi delle osservazioni sul tempo. La disponibilità dei dati predetti e attuali al modulo di controllo permette un adattamento real-time del setting dei parametri così come fare uso dei parametri ottimali durante lo streaming. Questa operazione di setting adattivo può essere guidata dal chi-quadro test così come è stato presentato nella sezione degli esperimenti.

Inoltre, sono state valutate le performance del sistema di controllo del rate considerando le tre tecniche di aging al fine di determinare il miglior trade-off tra la probabilità di starvation, qualità video e complessità computazionale.

Lavori futuri

Per quanto riguarda le metodiche di ottimizzazione del routing e rerouting della rete di supporto si sta completando la fase di sperimentazione delle tecniche in ambiente di simulazione e si prevede una fase successiva di implementazione sulla rete reale sviluppata nei laboratori IKNOS. Le sperimentazioni sono tese a verificare le prestazioni delle varie soluzioni proposte e confrontare i risultati con quelli delle tecniche attualmente utilizzate sulle reti.

Per valutare meglio l'efficacia della tecnica proposta relativamente al servizio di video streaming è stata avviata la fase di implementazione dell'algoritmo in una rete reale con WiFi e Hiperlan2 network.

Aknowledgments

Il lavoro proposto relativamente alle tecniche di ottimizzazione del routing e rerouting è stato svolto in compartecipazione con il Dipartimento di Informatica e Sistemistica “A.Ruberti” Università degli Studi di Roma “La Sapienza”, nelle persone de: il Prof. Ing. Carlo Mannino e, soprattutto, il PhD. Student Ing. Fabio D’Andreagiovanni, con i quali si sta ancora collaborando per la produzione di articoli inerenti al lavoro svolto, e che si ringraziano per la professionalità e lo spirito collaborativo dimostrato.

Per quanto riguarda la tecnica di ottimizzazione relativa al video streaming, il lavoro presentato fa uso dei risultati prodotti dal Cybersar Project gestito dal Consorzio COSMOLAB. Tale progetto è stato finanziato dal Ministero Italiano dell’Università e Ricerca (MUR) nell’ambito del Programma Operativo Nazionale 2000-2006 "Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione" per le Regioni Italiane dell’Obiettivo 1 (Campania, Calabria, Puglia, Basilicata, Sicilia, Sardegna) – Asse II, Misura II.2 “Società dell’Informazione”, Azione a “ Sistemi di calcolo e simulazione ad alte prestazioni”. Altre informazioni sul progetto sono disponibili nel sito <http://www.cybersar.it>.

Report e Articoli prodotti

Report del progetto IKNOS

I report del primo Obiettivo Realizzativo *OR1 - Analisi e definizione dei servizi e necessità* relativi alle attività:

- *ATT1.1 - Architettura di rete di riferimento*
- *ATT 1.2 - Servizi di riferimento.*

I report del secondo Obiettivo Realizzativo *OR2 – Sviluppo di Metodologie* relativi alle attività:

- *ATT2.1 - Definizione dei KPI*
- *ATT 2.4-Ottimizzazione.*

Conferenze

M. T. Carta, T. Onali and L. Atzori

“Window-based rate control approach for video streaming over wireless networks”, *IEEE ICC*, Glasgow, UK, 24-28, June 2007

M. T. Carta, T. Onali, N. Aste and L. Atzori

“Video Transmission Over Wireless Networks: Rate Control For Bursty Throughput channels”, *MobiMedia*, Alghero, Italy, 18 - 20, September 2006

Journal

M.T. Carta, T. Onali and L. Atzori

“Video Streaming over Wireless Channels: Rate Control to Restrain Buffer Starvation Occurrences” *Hindawi Advances in Multimedia Journal* - Sottomesso

Riferimenti

- [1] T. KWOK: "Residential broadband Internet services and application requirements", IEEE Communications Magazine, June 1997
- [2] MINISTERO DELLE TELECOMUNICAZIONI: La qualità dei servizi nella rete ICT, <http://www.comunicazioni.it/it/index.php?IdNews=88>
- [3] R. Braden, D. Clark, and S. Shenker "RFC 1633 - Integrated Services in the Internet Architecture: An Overview. Informational" RFC, June 1994
- [4] G. Huston, "Next Steps for the IP QoS Architecture", IETF RFC 2990, Nov. 2000
- [5] ITU-T Rec. Y.1241, "Support of IP-based Services Using IP Transfer Capabilities," Mar. 2001
- [6] ISO 8402, "Quality Management and Quality Assurance— Vocabulary," 1994
- [7] W.C. Hardy, QoS Measurement and Evaluation of Telecommunications Quality of Service, Wiley, 2001
- [8] ETSI, "Network Aspects (NA); General Aspects of Quality of Service (QoS) and Network Performance (NP)," Tech. rep. ETR003, 2nd ed., Oct. 1994
- [9] ITU-T Rec. E.800, "Terms and Definitions Related to Quality of Service and Network Performance Including Dependability," Aug. 1993
- [10] ITU-T Rec. I.350, "General Aspects of Quality of Service and Network Performance in Digital Networks, Including ISDNs," Mar. 1993.
- [11] ITU-T Rec. I.380, "Internet Protocol Data Communication Service — IP Packet Transfer and Availability Performance Parameters," Feb. 1999
- [12] ITU-T Rec. G.1000, "Communications Quality of Service: A Framework and Definitions," Nov. 2001
- [13] E. Crawley et al., "A Framework for QoS-Based Routing in the Internet," IETF RFC 2386, Aug. 1998
- [14] R. Braden, D. Clark, and S. Shenker, "Integrated Services in the Internet Architecture: an Overview," IETF RFC 1633, June 1994
- [15] S. Blake et al., "An Architecture for Differentiated Services," IETF RFC 2475, Dec. 1998
- [16] D.O. Awduche, "MPLS and Traffic Engineering in IP Networks", IEEE Communication Magazine vol. 37, N. 12, Dec. 1999
- [17] ITU-T Rec. G.1540, "IP Packet transfer and Availability Performance Parameters," Dec. 2002
- [18] ITU-T Rec. G.1541, "Network Performance Objectives for IP-Based Services," May. 2002
- [19] F. Le Faucheur and W. Lai, "Requirements for support of differentiated services-aware MPLS traffic engineering," IETF, RFC 3564, July 2003
- [20] A framework for integrated service operation over Diffserv networks, RFC 2998

- [21] M. Listanti “Multi Protocol Label Switching (MPLS): Architettura e *Ingegneria del Traffico*” Dipartimento INFOCOM, 2003
- [22] Multiprotocol Label Switching, support of differentiated services, RFC 3270
- [23] Multi-Protocol Label Switching Architecture, RFC 3031
- [24] MPLS Traffic Engineering, Cisco Systems Inc., 2000
- [25] P. Vaananen and R. Ravikanth, Framework for Traffic Management in MPLS Networks, IETF Internet Draft, March 1998
- [26] F. Le Faucheur, “Considerations on bandwidth constraint models for DS-TE,” IETF, Internet Draft, work in progress, June 2002
- [27] Jaudelice C. de Oliveira, Caterina Scoglio, Ian F. Akyildiz, and George Uhl “New Preemption Policies for DiffServ-Aware Traffic Engineering to Minimize Rerouting in MPLS Networks”, IEEE/ACM TRANSACTIONS ON NETWORKING, VOL. 12, NO. 4, AUGUST 2004
- [28] Jaudelice C. de Oliveira, Caterina Scoglio, Ian F. Akyildiz, and George Uhl “Optimal policy for label switched path setup in MPLS networks” Computer Networks 39 (2002) 165–183
- [29] A. Roveri “Reti di Telecomunicazioni - principi generali” Scuola Superiore G. Reiss Romoli, 1995
- [30] O. Younis, S. Fahmi Constraint-based routing in the Internet: basic principles and recent research, Department of Computer Sciences, Purdue University
- [31] A. Bley, T. Koch, Integer programming approaches to access and backbone IP-network planning, ZIB-Report 02-41, 2002
- [32] E. Atteo, S. Avallone, S. P. Romano, A link weight assignment algorithm for traffic engineered networks, Functional Programming 1, 2005
- [33] Cisco Systems. <http://www.cisco.com>
- [34] E. Crawley, R. Nair, B. Jajagopalan, H. Sandick. “A Framework for QoS-based Routing in the Internet”, RFC, <http://www.ietf.org/rfc/rfc2386.txt>, August 1998
- [35] I. W. Widjaja, I. Saniee, A. Elwalid, D. Mitra. “Online Traffic Engineering with Design-Based Routing”, 15th ITC Specialist Seminar on Internet Traffic Engineering and Traffic Management, Würzburg, Germany, July 2002
- [36] K. Kar, M. Kodialam, T. V. Lakshman: “Minimum Interference Routing with Application to MPLS Traffic Engineering”, Proc. IEEE INFOCOM, Vol 2, pp 884–893, Tel-Aviv, Israel, March 2000
- [37] A. B. Bagula, M. Botha, and A. E. Krzesinski “Online Traffic Engineering: The Least Interference Optimization Algorithm” Communications, 2004 IEEE International Conference
- [38] S. Suri, M. Waldvogel e P. Ramesh Warkhede “Profile-Based Routing: A New Framework for MPLS Traffic Engineering” Quality of Future Internet Services: Second COST 263 International Workshop, Qofis 2001, Proceedings, Coimbra, Portugal, September 24-26, 2001
- [39] A. Juttner, B. Szviatovszki, A. Szentesi, D. Orincsay, J. Harmatos “On-demand optimization of label switched paths in MPLS network” Computer Communications and Networks, Proceedings. Ninth International Conference, 2000
- [40] K. Holmberg, D. Yuan A multicommodity network-flow problem with side constraints on paths solved by column generation, Journal on Computing, INFORMS 2003
- [41] F. D’Andreagiovanni, Instradamento ottimo con vincoli di QoS in reti di telecomunicazioni - Tesi di Laurea Specialistica, 2006
- [42] A. Cerrone, Progetto di algoritmi di ottimizzazione per il QoS routing in reti con architettura DiffServ/MPLS, 2005
- [43] F. Blanchy, L. Melon, G. Leduc Routing in a MPLS network featuring preemption mechanisms, 10th International Conference on Telecommunications, 2003

- [44] P.Dharwadkar, H.J.Siegel, E.K.P.Chong A heuristic for dynamic bandwidth allocation with preemption and degradation for prioritized requests, Proc. 21st Int. Conf. Distributed Computing Systems, 2001
- [45] J.A.Garay, I.S.Gopal Call preemption in communication networks, IEEE INFOCOM 1992
- [46] C.H.Lau, B.H.Soong, S.K.Bose Preemption algorithm with re-routing to minimize service disruption, Ninth International Conference on Communications Systems, 2004
- [47] R.Boutaba, W.Szeto, Y.Iraqi Dynamic Online Routing Algorithm for MPLS Traffic Engineering, The International Journal of Network and Systems Management - Volume 10, Issue 3, 2002
- [48] B.Szviatovszki, 'A.Szentesi, A.J"uttner "Minimizing re-routing in MPLS networks with preemption-constraint-based routing,"Computer Communications Journal on Advances in Performance Evaluation of Computer and Telecommunications Networking, 25(11-12), May 2003
- [49] M.R.Meyer, D.Maddux, J.P.Vasseur, C.Villamizar, A.Birjandi MPLS Traffic Engineering Soft Preemption, IETF Internet Draft, October 2003
- [50] S. Kallel. Analysis of memory and incremental redundancy ARQ schemes over a nonstationary channel. IEEE Trans. on Commun, 40:1474–1480, 1992
- [51] R. Deng. Hybrid ARQ schemes employing coded modulation and sequence combining. IEEE Trans. on Commun., pages 2239–2245, Feb.-Apr. 1994
- [52] Q. Zhang, W. Zhu, and Y.-Q. Zhang, "End-to-end QoS for video delivery over wireless Internet," Proceedings of the IEEE, 93(1), pp. 123-134, Jan. 2005
- [53] M. van der Schaar and D. Shankar. "Cross-layer wireless multimedia transmission: Challenges, principles, and new paradigms," IEEE Wireless Communications, pages 50–58, Aug. 2005
- [54] S. Floyd et al., "Equation-Based Congestion Control per Unicast Applications," Proc. ACM SIGCOMM 2000, pp. 43–56, Aug. 2000
- [55] F. Yang, Q.Zhang, W. Zhu, and Y.-Q. Zhang, "End-to-End TCP-Friendly Streaming Protocol and Bit Allocation per Scalable Video Over Wireless Internet," IEEE Journal on Selected Areas in Communications, vol. 22, no. 4, May 2004
- [56] M. Chen, and A. Zakhor, "Multiple TFRC Connections Based Rate Control per Wireless Networks," IEEE Trans. on Multimedia, vol. 8, no. 5, Oct 2006
- [57] Y. Sun, I. Ahmad, D. Li, and Y.-Q. Zhang, "Region-Based Rate Control and Bit Allocation per Wireless Video Transmission," IEEE Trans. on Multimedia, vol. 8, no. 1, Feb 2006
- [58] S. Aramvith, C. Lin, S. Roy, and M. Sun, "Wireless video transport using conditional retransmission and low-delay interleaving," IEEE Trans. on Circuits and Systems per Video Tech, pages 558–565, June 2002
- [59] J. Cabrera, A. Ortega, and J. Ronda, "Stochastic rate-control of video coders per wireless channels," IEEE Trans. on Circuits and Systems per Video Tech, 12(6):496–510, June 2002
- [60] N. Srisawaivilai, and S. Aramvith, "Improved frame and basic unit layers bit allocation scheme per H.264 video transmission over ARQ-based wireless channels," International Conference on Intelligent Informazione Hiding and Multimedia Signal Processing, IHH-MSP 2006, pages 205 – 210, Dec. 2006
- [61] O. Tickoo, B. Sikdar "On the Impact of IEEE 802.11 MAC on Traffic Characteristics", IEEE Journal on Selected Areas In Communications, vol. 21, no. 2, February 2003
- [62] C.J. Sreenan, J.-C. Chen, P. Agrawal, and B. Narendran, "Delay Reduction Techniques per Playout Buffering," IEEE Trans. On Multimedia, pages 88–100, June 2000

Appendice A

Classi di complessità dei problemi di ottimizzazione

Quando si ha a che fare con problemi di ottimizzazione è sempre importante avere chiaro cosa formalmente si intenda per “problema facile” e “problema difficile” e quali sono le caratteristiche che rendono un problema tale. A tal fine, in questa sezione, vengono richiamati i concetti fondamentali alla base della classificazione di un problema.

Le definizioni formali saranno raggiunte gradualmente passando attraverso una trattazione più intuitiva che si poggia sui quattro elementi seguenti:

- una classe di problemi C di cui si esegue la classificazione;
- una sottoclasse non vuota C_f di problemi “facili”;
- una sottoclasse non vuota C_d di problemi “difficili”;
- una relazione “non più difficile di” definita tra coppie di problemi di C .

Introdotti questi elementi si può evidenziare un primo risultato notevole:

Proposizione 1 Siano P_1, P_2 due problemi di C .

Se P_2 è “facile” ($P_2 \in C_f$) e P_1 è “non più difficile di” P_2 allora P_1 è “facile” ($P_1 \in C_f$).

Se P_1 è “difficile” ($P_1 \in C_d$) e P_1 è “non più difficile di” P_2 allora P_2 è “difficile” ($P_2 \in C_d$).

Quanto detto risulta molto intuitivo e l'obiettivo è ora riformularlo in termini più rigorosi e formali. Come primo passo si introduce una forma dei problemi che risulta molto più trattabile in un processo di classificazione: il classico problema di ottimizzazione

$$\min\{c^T x : x \in F\}$$

una cui generica istanza è individuata da una coppia (c, F) , viene sostituito dal *problema di decisione*

$$(Esiste x \in F : c^T x \leq k)?$$

una cui generica istanza è individuata da una tripla $(c, F, k \in Z)$. Si passa quindi dalla prospettiva di ricerca di una soluzione ottima a quella di risposta ad una domanda con un semplice “SI” o “NO”.

Un ultimo passo necessario per giungere ad una classificazione dei problemi di decisione è rappresentato dall'introduzione dei due concetti di *lunghezza di un'istanza* e di *tempo di esecuzione di un algoritmo*. Considerata un'istanza I di un generico problema P , si può definire una rappresentazione “standard” dell'istanza e ricondurla ad una forma binaria. Il numero di bit necessari per individuare la rappresentazione binaria standard viene indicato come lunghezza dell'istanza e si denota con $L(I)$. Risolvere l'istanza I di un problema richiede di applicare un particolare algoritmo A ed eseguire quindi una serie ordinata di operazioni che includono delle operazioni elementari di calcolo. Se si indica con $f_A(I)$ il numero di operazioni elementari necessarie per risolvere l'istanza I mediante l'algoritmo A , si può definire il *tempo di esecuzione di un algoritmo* come:

$$f_A^*(l) = \sup_I \{f_A(I) : L(I) = l\}$$

Si prende quindi a riferimento come tempo di esecuzione quello peggiore, relativo all'istanza la cui soluzione tramite l'algoritmo richiede maggiore tempo.

Un algoritmo A si dice *polinomiale* per un problema P se esiste un intero p tale che si abbia:

$$f_A^*(l) = O(l^p)$$

La funzione *O-grande* $O(x)$ viene utilizzata per caratterizzare il comportamento asintotico di una funzione quando il suo argomento diventa molto grande. La definizione formale è:

Definizione 1 (Funzione O-grande) Considerate due funzioni reali a valori positivi, $f, g \in \mathfrak{R}_+$, $f(x)$ è una funzione $O(g(x))$ se esistono due numeri positivi c, \tilde{x} tali che

$$f(x) \leq c \cdot g(x), \forall x \geq \tilde{x}$$

In altre parole, la funzione $g(x)$ costituisce un limite superiore ai valori assunti da $f(x)$ e per valori x sufficientemente grandi cresce al più tanto velocemente quanto $g(x)$.

A questo punto si hanno a disposizione tutti gli elementi di base per portare avanti la classificazione dei problemi e introdurre una serie di definizioni che caratterizzano le classi dei problemi “facili” e “difficili”.

Definizione 2 Si indica con **NP** la classe dei problemi di decisione con la proprietà che per ogni istanza per cui la risposta è “SI” esiste un algoritmo polinomiale per dimostrarlo.

Definizione 3 Si indica con **P** la sottoclasse dei problemi di decisione di **NP** per i quali esiste un algoritmo polinomiale di soluzione.

Grazie a tutti gli elementi introdotti, la relazione “non più difficile di” può essere riformulata in modo molto più rigoroso sotto altro nome:

Definizione 4 Siano P_1, P_2 due problemi di NP.

Se ogni istanza di P_1 può essere convertita in tempo polinomiale in un'istanza di P_2 allora P_1 si dice polinomialmente riducibile a P_2 .

È questo un risultato davvero notevole, la cui importanza risiede nel fatto che se si dispone di un algoritmo per la soluzione di P_2 , esso può essere utilizzato per risolvere anche P_1 con un aumento del costo computazionale, dovuto al processo di riduzione, che è polinomiale nella dimensione dell'istanza.

Per completare la classificazione dei problemi, viene individuata un'ultima classe:

Definizione 5 Si indica con NPC la classe dei problemi NP-completi, costituita dal sottoinsieme di problemi $\tilde{P} \in NP$ ai quali ogni generico problema $Q \in NP$ è polinomialmente riducibile:

$$\tilde{P} \in NPC \Leftrightarrow \forall Q \in NP, Q \text{ è polinomialmente riducibile a } P$$

Tramite quest'ultima definizione si può ora riformulare anche la Proposizione 1, giungendo al cosiddetto *Reduction Lemma*:

Proposizione 1 (Reduction Lemma) Siano P_1, P_2 due problemi di NP.

Se $P_2 \in P$ e P_1 è polinomialmente riducibile a P_2 allora $P_1 \in P$.

Se $P_1 \in NPC$ e P_1 è polinomialmente riducibile a P_2 allora $P_2 \in NPC$.

Da cui si può derivare un corollario capitale:

Corollario 1 Se $P \cap NPC \neq \emptyset$ allora $P = NP$

La sua importanza risiede tutta nella tesi, uno dei grandi punti aperti della teoria della complessità: se si riuscisse a dimostrare che $P \cap NPC \neq \emptyset$; allora esisterebbe un problema risolvibile in tempo polinomiale a cui potrebbero essere polinomialmente ridotti tutti gli altri problemi. Tutti i problemi sarebbero quindi polinomiali e “facili” da risolvere!

Si sa con certezza che $NPC \neq \emptyset$, ma ancora nulla si sa riguardo a quale sia

l'alternativa vera: $P = NP$ o $P \neq NPC$? Grande gloria attende il teorico che riesca a dimostrare una delle due alternative.